Хмельницький національний університет

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

ДИПЛОМНИЙ ПРОЕКТ

Інформаційна система моніторингу діяльності салону з продажу мобільних телефонів

Назва теми

Рівень вищої освіти Перший (бакалаврський)

Галузь знань 12 «Інформаційні технології»

Спеціальність 121 «Інженерія програмного забезпечення»

Освітня програма Освітньо-професійна програма «Інженерія програмного забезпечення»

Шифр ДПІПЗ.180105.01.03.ПЗ

Виконав студент IV курсу група ІПЗ-18-1 _____ В. В. Король
Підпис Ініціали, прізвище

Керівник канд. техн. наук, доцент _____ Ю.В. Форкун
Науковий ступінь, звання Підпис Ініціали, прізвище

Нормоконтролер канд. техн. наук, доцент _____ І.В. Гурман
Підпис Ініціали, прізвище

До захисту допускаю:
Завідувач кафедри інженерії програмного забезпечення _____ Л. П. Бедратюк
Підпис Ініціали, прізвище

13 червня 2022 р.

Хмельницький 2022

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

акультет Програмування та комп'ютерних і телекомунікаційних систем
афедра Інженерії програмного забезпечення
івень вищої освіти Перший (бакалаврський)
алузь знань 12 «Інформаційні технології»
пеціальність 121 «Інженерія програмного забезпечення»
світня програма Освітньо-професійна програма «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ
Завідувач кафедри
Л. П. Бедратюк
05 02 2022 р.

ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ)

Королю Володимиру Володимировичу
Прізвище, ім'я, по батькові студента

Тема проєкту (роботи) Інформаційна система моніторингу діяльності салону з одажу мобільних телефонів

ерівник проєкту (роботи) Форкун Юрій Вікторович, канд. техн. наук, доцент
Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

тверджена наказом ректора університету від 01.03.2022 р. № 18
Строк подання студентом проєкту (роботи) на кафедру 01.06.2021 р.
Вихідні дані до проєкту (роботи) Матеріали переддипломної практики

Зміст пояснювальної записки (перелік питань, які потрібно розробити)
слідження предметної області, проектування програмного забезпечення, програмна алізація та тестування програмного забезпечення

Перелік графічного матеріалу (із зазначенням обов'язкових креслень)
Презентаційні матеріали (слайди)

6. Консультанти розділів дипломного проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прий
Нормоконтроль	Гурман І.В., доцент кафедри ІПЗ		
Антиплагіат	Гурман І.В., доцент кафедри ІПЗ		

7. Дата видачі завдання « 05 » лютого 2022р.

КАЛЕНДАРНИЙ ПЛАН

Назва етапів (розділів) дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Прим
Ознайомлення з тематикою дипломного проектування (ДП), визначення та узгодження індивідуальних там ДП	01.12 – 30.12.2021	
2 Дослідження предметної області, в якій планується використання програмного засобу (ПЗ), визначення задач та вимог, розробка технічного завдання	02.01 – 31.01.2022	
3 Проектування програмного забезпечення	01.02 – 28.02.2022	
4 Програмна реалізація	01.03 – 10.04.2022	
5 Тестування програмного забезпечення	11.04 – 30.04.2021	
6 Написання вступу, загальних висновків, оформлення джерел посилання та додатків. Оформлення пояснювальної записки ДП згідно вимог стандартів	01.05 – 25.05.2022	
Попередній захист ДП	Травень 2022 (згідно графіка)	
8 Перевірка ДП на плагіат, нормконтроль, отримання відгуків та рецензій. Брошурування (зшиття) пояснювальної записки)	26.05 – 30.05.2022	
9 Підготовка до захисту та захист ДП	з 01.06.2022	

Студент _____ В.В. Король
Підпис Ініціали, прізвище

Керівник проекту (роботи) _____ Ю.В. Форкун
Підпис Ініціали, прізвищ

АНОТАЦІЯ

Тема дипломного проекту: «Інформаційна система моніторингу діяльності салону з продажу мобільних телефонів».

Автор проекту: Король Володимир Володимирович.

Керівник проекту: Форкун Юрій Вікторович.

Пояснювальна записка: 65 с., 25 рис., 7 табл., 3 дод., 12 джерела.

Графічна частина: 12 презентаційних слайдів.

Інформаційна система, API, TTS, C#, Visual Studio, IIS, Web, CLR

Метою проекту є розробка програмного продукту моніторингу діяльності салону з продажу мобільних телефонів.

У дипломному проекті проведено аналіз інформаційного забезпечення предметної області та інформаційних потоків, розроблена структура застосування. Для розробки програмного продукту використано мову C# та середовище розробки Microsoft Visual Studio 2019. За допомогою цих засобів розроблено програмне забезпечення моніторингу діяльності салону з продажу мобільних телефонів.

Впровадження розробленого програмного продукту дозволяє організувати та автоматизувати процес моніторингу роботи менеджерів, та салону з продажу мобільних телефонів загалом. А також, здійснювати облік і продаж товарів та здійснює полегшення керування даними.

ВІДОМІСТЬ ДОКУМЕНТІВ

№ рядка	Формат	Позначення документа	Найменування документа	К-сть аркушів	№ екз.	Примітка
			Текстові документи			
1	А	ДППІЗ.180105.01.03.ПЗ	Пояснювальна записка	65		
2	А		Завдання на дипломний	1		
			проект			
3	А		Анотація	1		
			Графічні документи			
4	А		Презентаційні матеріали	12		

Змн.	Арк.	№ докум.	Підпис	Дата	ДППІЗ.180105.01.03.ПЗ			
Виконав		Король В.В.		6.06	Інформаційна система моніторингу діяльності салону з продажу мобільних телефонів Відомість документів	Літ.	Арк.	Аркушів
Керівник		Форкун Ю.В.		6.06			2	122
Н. контр.		Гурман І.В.		13.06		ХНУ, ІПЗ-18-1		
Зав. каф.		Бедратюк Л.П.		17.06				

ЗМІСТ

Вступ 6
1 Дослідження предметної області та постановка задачі 8
1.1 Змістовний аналіз предметної області, її структурних та функціональних особливостей 8
1.2 Аналіз останніх публікацій та наявного програмно-технічного забезпечення предметної області 9
1.3 Визначення вимог до програмного забезпечення та розробка технічного завдання 14
2 Проектування програмного забезпечення 18
2.2 Архітектура та функціональна структура додатка 18
2.2 Структура та функціональне призначення модулів додатку та їх зв'язок. 19
2.3 Вибір мови програмування та засобів 28
3 Програмна реалізація .....
3.1 Розробка програмних модулів 34
3.2 Розробка інтерфейсу користувача 37
3.3 Вимоги до технічних засобів 43
4 Тестування програмного забезпечення 47
4.1 Вибір та обґрунтування методів тестування додатку 47
4.2 Тестування інтеграції 50
4.3 Створення проекту тестування інтеграції 51
4.4 Створення тесту на інтеграцію 53
Висновки 58
Перелік джерел посилання 61
Додаток А Технічне завдання 63

ПЕРЕЛІК СКОРОЧЕНЬ

ВВ	–	варіанти використання
ПЗ	–	програмне забезпечення
API	–	Application programming interface
CLR	–	Common Language Runtime
GUI	–	The graphical user interface
IDE	–	Integrated development environment
.NET	–	Технологія Платформа від Microsoft, яка дозволяє створювати програмне хабезпечення
TCP	–	The Transmission Control Protocol
VS		Microsoft Visual Studio
UML	--	Unified Modeling Language
WWW	–	World Wide Web

ВСТУП

В наш час поняття смартфон і мобільний телефон так швидко увійшли до навколишнього світу, що уявити процес надання послуг без застосування смартфонів не можливо. Тим паче, в такій висококонкурентній галузі як салону продажу. Також на звичний порядок надання послуг вплинула пандемія коронавірусу, і це ще одна причина з якої власники та менеджери салонів з продажу мобільних телефонів мають можливість вдосконалювати процес надання послуг в торговельній галузі, з бажанням не тільки зберегти свій бізнес, а й розвивати його. В даному випадку на допомогу приходять інформаційні технології, за допомогою яких здійснюється втримання та залучення існуючих клієнтів.

Торговельна галузь це галузь, яка найбільше постраждала під час пандемії коронавірусу. Тому актуальним є питання вдосконалення та розробки програмного забезпечення, яке б відповідало викликам часу та надавало конкурентну перевагу за рахунок зручності використання та потрібної функціональності.

Метою дипломного проектування та переддипломної практики є розробка системи моніторингу діяльності салону з продажу мобільних телефонів. Для досягнення поставленої мети були сформовані завдання на дипломне проектування та переддипломну практику:

- дослідити предметну області торговельного бізнесу та виявити потреби потенційних користувачів програмного продукту;
- проаналізувати наявне програмне забезпечення в цій галузі та сформулювати вимоги на розробку нового програмного забезпечення, що відповідає потребам часу;
- спроектувати структуру інформаційно-довідкової системи та розробити базу даних;
- вибрати технології для розробки інформаційно-довідкової системи;
- розробити та протестувати інформаційну-довідкову систему торгової фірми.

Об’єктом дослідження роботи виступають веб-сервіси, що спеціалізуються на моніторингу продажу та обслуговування мобільних телефонів менеджерами магазинів.

Предметом дослідження виступають алгоритми та методи моніторингу діяльності, які використовуються для прогнозування діяльності роботи менеджерів та роботи магазину загалом.

Результатом дипломного проектування є розроблена інформаційна система моніторингу діяльності салону з продажу мобільних телефонів.

1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Змістовний аналіз предметної області, її структурних та функціональних особливостей

Система (system) - сукупність об`єктів nf відношень між ними, zrs утворюють єдине ціле в процесі системогенезу.[1]

Моніторинг - комплекс наукових, технічних, технологічних, організаційних та інших засобів, які забезпечують систематичний контроль (стеження) за станом та тенденціями розвитку досліджуваних процесів. [1]

Методологічно моніторинг це проведення ряду однотипних замірів при цьому головна інформація полягає навіть не в самих значеннях результатів, а в їх зміні, динаміці від одного заміру до іншого.

Система автоматизації моніторингу – це система призначена для збору, аналізу та зберігання інформації по фінансових операціях.

Функції моніторингу:

- інтегративна - розробка комплексної характеристики процесів, що відбуваються в області дослідження;
- діагностична - сканування області дослідження та сканування змін, що відбуваються в ній;
- компаративістська - відстеження області дослідження в цілому або різних її елементів як у часі, так і просторі;
- експертна - здійснення експертизи стану, концепції, форм, методів розвитку області дослідження, її компонентів і підсистем;
- інформаційна - отримання і розповсюдження інформації, що дозволяє аналізувати і прогнозувати область дослідження;
- прагматична - використання моніторингової інформації при прийнятті максимально обґрунтованих та адекватних вимогам ситуації рішень, насамперед управлінських.

Основним завданням моніторингу є збір та аналіз інформації по діяльності певного суб’єкта, для прийняття подальших рішень що - до підвищення конкретних показників, зі зростанням яких зросте загальна ефективність діяльності підприємства в цілому.

1.2 Аналіз останніх публікацій та наявного програмно-технічного забезпечення предметної області

На даний момент створено велика кількість програмних продуктів для моніторингу діяльності об’єктів торгівельної діяльності. Дані програмні продукти відрізняються між собою своєю функціональністю, зручністю у користуванні. Розглянемо деякі програмні продукти для здійснення моніторингу діяльності об’єктів торгівельної діяльності.

Платформа Supasoft — універсальна програма для обліку даних. Таких як клієнти, продажі, замовлення, партнери, постачальники, договори, відвантаження, оплати, доходи, витрати, товари, фінанси, продукти, техніка, матеріали, комплектуючі та ін.. Supasoft забезпечує налаштування обліку та моніторингу діяльності, здійснення пошук, обмежувати доступ користувачам, будувати звіти і багато іншого. При багатокористувацькій роботі, різним користувачам є можливість приховування чи відображення певних даних. Конструктор звітів Supasoft дозволяє створювати різноманітні звіти за будь - якими даними.

Платформа Supasoft розповсюджується у двох варіантах локальна (безкоштовна) та мережева (платна).

Програмна система компанії Парус. Група компаній "Парус" (Засновані в Україні 1990)— українська компанія, розробник програмного забезпечення для підприємств малого та середнього бізнесу, великих корпорацій і холдингів, а також установ та організацій державного сектору економіки. (Рисунок 1.1)

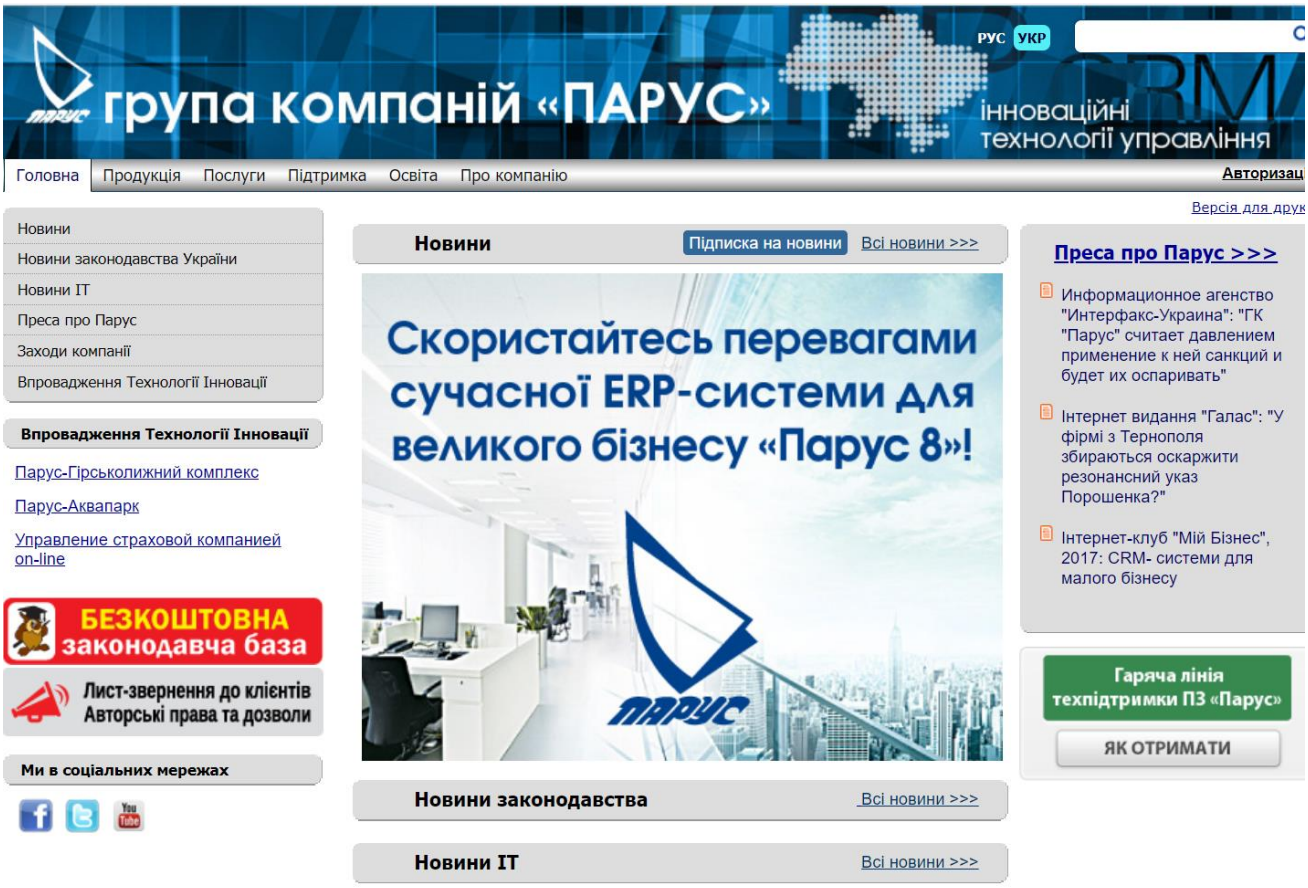


Рисунок 1.1 – Сайт платформи «Парус»

Дана система є великим громіздким комплексом підсистем, які призначені для роботи у різних сферах діяльності.

Сьогодні розвиваються та підтримуються декілька ліній програмних продуктів:

- JПарус (працює з використанням «хмарних технологій»);
- Парус-Підприємство 8 (ERP-система);
- Парус-Підприємство 8 (система для автоматизації управління підприємствами середнього та малого бізнесу);
- Парус-Бюджет (система автоматизації управління установами державного сектору економіки);
- ІАС Парус — Інтернет Консультант (довідкова інформаційно-аналітична система по законодавству України з on-line оновленням).

Для здійснення моніторингу діяльності об’єктів торгівельної діяльності використовуються дві підсистеми: Парус – Торгівля та Склад та Парус – Персонал.

Програмний продукт Парус - Торгівля і Склад - інформаційна система, що дозволяє забезпечувати управління запасами складської логістикою, реалізацією товарів і послуг, що володіє наступним набором функцій:

- підготовка та облік первинних документів;
- облік товарно-матеріальних цінностей;
- взаємовідносини з контрагентами.

Парус – Персонал - автоматизована система управління персоналом призначена для комплексної автоматизації бізнес-процесів HR-департаменту та відділу кадрів. Використання системи значно підвищує ефективність управління персоналом і полегшує HRM-процеси (Human Resource Management), в т.ч. завдання рекрутингу, мотивації, підвищення кваліфікації. Функціонал системи дозволяє автоматизувати:

- рекрутинг;
- кадровий облік;
- HRM;
- звітність.

Система Парус-Персонал знаходиться в єдиному інформаційному середовищі з іншими модулями Парус-Підприємства, зокрема, вона може використовуватися спільно з Парус-Заробітна плата для формування більш повного звітування та для полегшення розрахунку заробітної плати.

Одним з найвідоміших систем здійснення моніторингу об’єктів торгівельної діяльності на українському ринку є BJet - комплексне хмарне рішення для розвитку та управління бізнесом (Рисунок 1.2).

BJet – програмний продукт, який призначений для автоматизації діяльності на підприємстві.

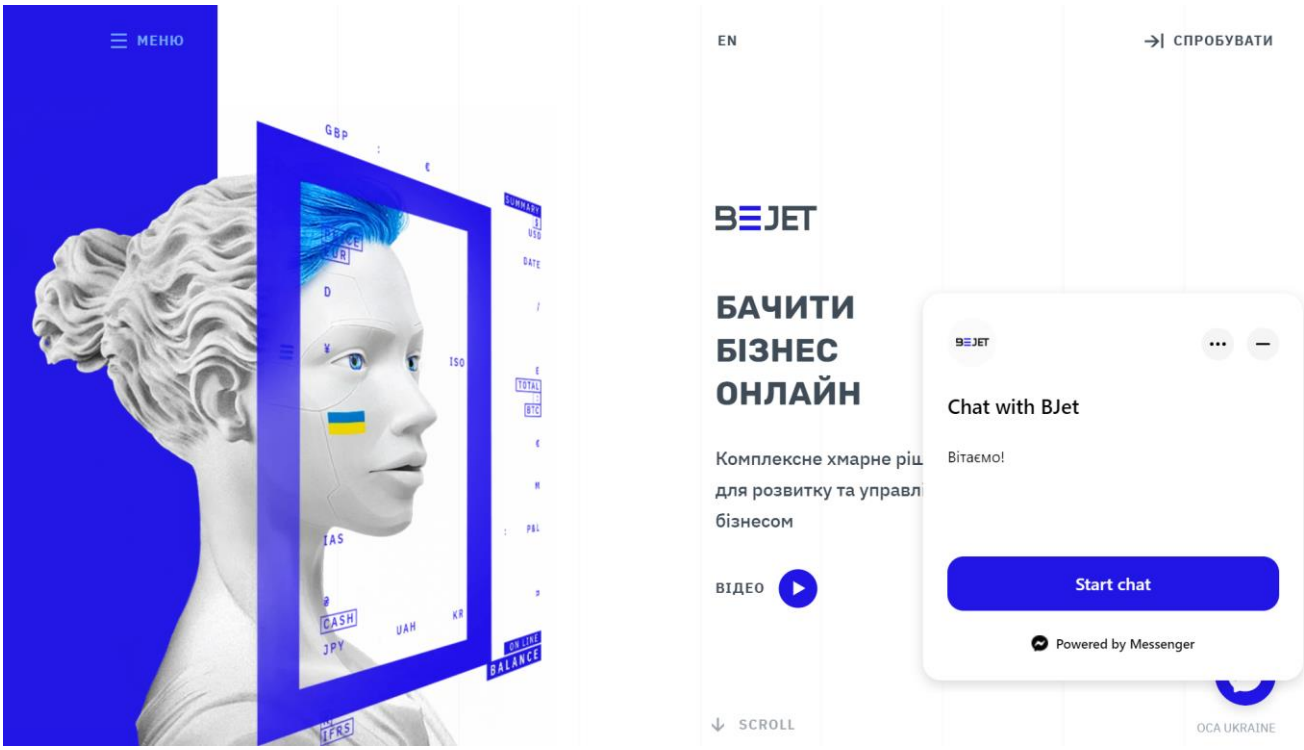


Рисунок 1.2 – Програмний продукт BJet

Історія платформи BJet та компанії BusinessJetPro беруть свій початок у 2016 році. Тоді на майданчику Львівського IT-кластеру зібралася група ентузіастів, заряджених бажанням допомогти українському бізнесу та знайти альтернативне рішення існуючим системам для управління підприємством.

Ця ідея знайшла підтримку серед топ-менеджерів провідних українських IT-компаній, фінансистів та експертів з менеджменту. Технологічний продукт, над яким у 2018 році почалася робота, мав два амбітні завдання: не лише стати українською альтернативою існуючим рішенням, але й вирішити низку проблем, які стоять перед бізнесом та принципово змінити ринок.

На різних етапах роботи до формування і створення продукту долучились представники декількох сфер бізнесу, експерти компанії SoftServe та команда BusinessJetPro, що втілила у життя цей проект в тісній співпраці з асоціацією Одоо в Україні.

Основна місія - забути про монополію та закриті системи, позбутися залежності від провайдерів і супроводу, змінити парадигму використання систем

з управління підприємством. Основне прагнення дати поштовх до розвитку компаній за рахунок продукту, а не навпаки.

BJet для приватних підприємців призначена для автоматизації обліку фізичних осіб – підприємців (ФОП), які здійснюють торговельну діяльність (оптову, роздрібну) і займаються наданням послуг. Конфігурація розрахована на ФОП, які працюють на єдиному або фіксованому податку без виділення ПДВ і забезпечує вирішення основних завдань обліку, які стоять перед ФОП при веденні торгової діяльності, а також дозволяє вести регламентований облік.

Основні функціональні можливості системи:

- облік взаєморозрахунків з постачальниками та покупцями;
- рух готівкових та безготівкових грошових коштів в розрізі банківських рахунків, кас і статей руху;
- облік запасів в розрізі місць зберігання, запасів і партій;
- облік прямих і непрямих витрат;
- ціноутворення;
- розрахунок заробітної плати, податків і відрахувань до фондів;
- контроль перевищення лімітів;
- формування необхідного пакету регламентованої звітності тощо.

BJet пропонує комплексне хмарне рішення для розвитку та управління бізнесом. Компанія пропонує бібліотека з 400 модулів. Система легко пристосовується до специфіки бізнесу. Потрібно лише обрати необхідні модулі з великої бібліотеки, яка безперервно поповнюється, або скористатися готовим рішенням від BJet.

Після детального аналізу існуючих систем моніторингу діяльності об’єктів торгівельної діяльності, зведемо одержані дані до таблиці (Таблиця 1.1), в якій відобразимо список основних функцій і їх підтримку існуючими системами та їх характеристиками, що дозволить не тільки встановити, які характеристики доцільно включити в наш програмний продукт, але визначити доцільність їх застосування та використання.

Таблиця 1.1 – Функціональність існуючих систем моніторингу діяльності об’єктів торгівельної діяльності

Критерій\Система	Supasoft	Парус	BJet
Безкоштовний продукт	+ -	-	-
Підтримка української мови	-	-	+
Критерій\Система	Supasoft	Парус	BJet
Підтримка кількох основних ОС	+	+	+
Експорт/імпорт інформації	-	+	+
Клієнт-серверний принцип роботи	+	-	-
Зрозумілий інтерфейс	+	+	-
Потребує інсталяції на ПК	+ -	+	+
Громіздкість системи	-	+	+

В таблиці подано основні характеристики трьох існуючих систем моніторингу діяльності об’єктів торгівельної діяльності: Supasoft, Парус, BJet:. Навпроти кожної з характеристик системи стоїть позначка “плюс” або “мінус”, яка й показує, чи підтримує дану функцію кожна з систем. Як бачимо з таблиці у всіх систем є спільні недоліки зокрема: продукти платні, або частково безкоштовні, що зменшує їхню доступність, відсутній клієнт - серверний принцип роботи, що значно уповільнює роботу, системи є громіздкими, часто для збільшення функціональності необхідно витрачати кошти на додаткове програмне забезпечення.

1.3 Визначення вимог до програмного забезпечення та розробка технічного завдання

Метою даного проекту є розробка інформаційної системи моніторингу діяльності салону з продажу мобільних телефонів, яка б могла полегшити та

спростити роботу персоналу салону з продажу мобільних телефонів та надати більше інформації його клієнтам. Система повинна виконувати основні задачі поставлені перед нею:

- зберігання і надання доступу до даних;
- забезпечення безпеки доступу до інформаційної системи з розподілом прав доступу для різного типу користувачів;
- вести облік товарів;
- формувати базу даних товарів;
- проводити аналіз бази даних товарів;
- формувати фінансовий та статистичний звіти про надходження, продажу, повернення та гарантійний ремонт товарів;
- формувати базу співробітників підприємства;
- проводити аналіз роботи менеджерів;
- формувати фінансовий та статистичний звіти про діяльність менеджера за будь-який період.

В результаті роботи спроектованої системи має спроститись процес отримання інформації користувачу про проведені операції продажу товарів, спрощення обліку даних співробітників підприємства. Система має полегшити роботу персоналу при оформленні продажів, автоматизувати пошук інформації.

Спроектована система має відповідати функціонально своїм аналогам та усувати їхні недоліки.

В формалізованому вигляді дана задача загалом може бути представлена у загальному вигляді діаграми варіантів використання, оскільки API є не лінійною структурою, де кожна з його частин використовується незалежно она від одної. Тобто, сам проект, загалом, подається як CRUD-функціонал відповідно для кожної з моделей, що повинні бути реалізовані.

Тобто, побудова основного базису нашого проекту зводиться до реалізації кожної з чотирьох дій (додавання, видалення, оновлення та читання) - для кожної з моделей. Для побудови даних дій потрібно розкрити сутність цих моделей, так,

наприклад, модель користувача не може бути доступна для читання. Оновлення повинно бути доступне лише для самого користувача. Видалення - тільки для користувачів з ролями адміністратор, а створення - тільки через процес реєстрації.

Підсумовуючи, можемо стверджувати, що дуже важливу роль в розробці проекту відіграє реалізація моделі управління заснованої на ролях користувачів. Тому, важливою надбудовою до цього базису, потрібне введення захисту вищеописаних CRUD дій, як від не авторизованих користувачів так і користувачів які не мають достатньо привілей для виконання певних дій. Реалізація даної логіку найкраще можна здійснити за допомогою атрибутів авторизації та відповідних ролей, що в свою чергу відносяться до адмінстраторів, які потребують захисту.

Функціональне призначення спроектованої системи (рисунок 1.3)

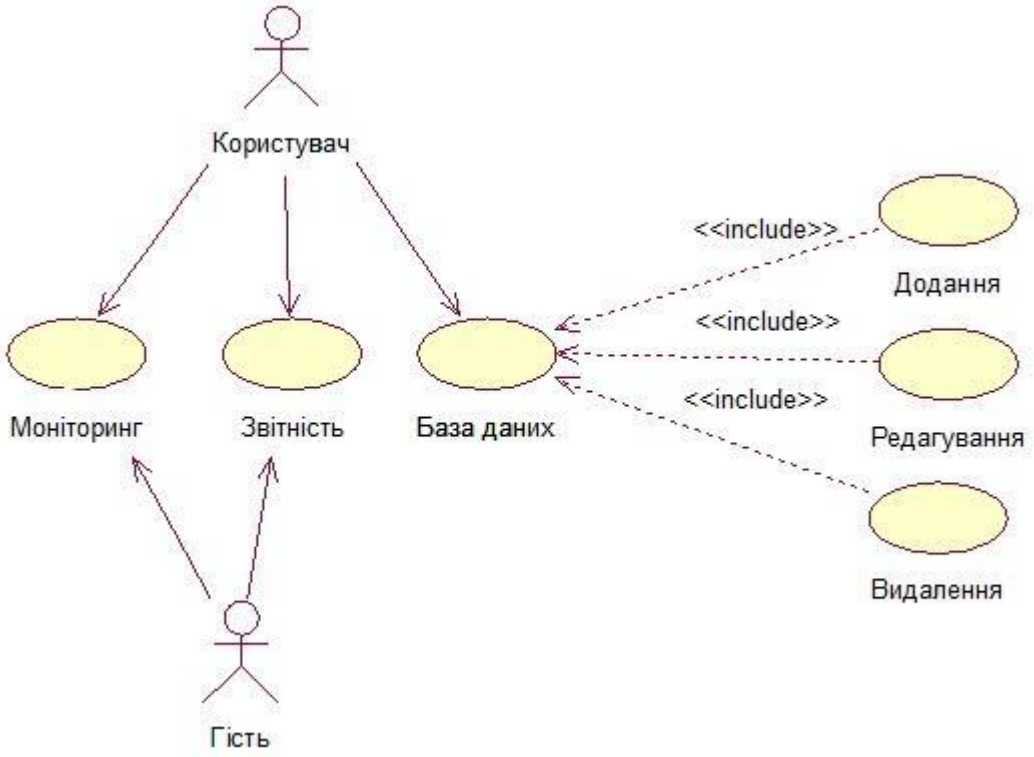


Рисунок 1.2 – Діаграма варіантів використання

Вимоги до програмного продукту:

– інтерфейс користувача програмного продукту повинен бути інтуїтивно зрозумілий та максимально простий та зручний для використання;

- мати вбудовані механізми захисту інформації від несанкціонованого доступу для забезпечення ідентифікації та автентифікації користувачів;
- забезпечувати надання необхідної інформації у зручному для сприйняття вигляді;
- забезпечувати оперативний пошук інформації , яка створюється програмним продуктом;
- можливість роботи в всесвітній мережі Інтернет.

Отже, в даному розділі здійснено змістовний аналіз предметної області. Визначено поняття системи, моніторингу, його функції та методологічних засад його проведення. Здійснено аналіз останніх публікацій та наявного програмного забезпечення, з визначенням задач необхідних для реалізації нашого проекту. Також визначено функціональне призначення та вимоги до розроблюваного програмного забезпечення.

2 ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Архітектура та функціональна структура додатка

Для побудови будь-якої складної інформаційної системи обов’язковим є проведення якісного системного аналізу, що дозволить чітко визначити мету і призначення системи, яка розробляється, зрозуміти структуру і механізм її функціонування, сформувати конкретні критерії відбору засобів для досягнення цілей системи.

Для проведення системного аналізу, необхідно наявність слабко структурованих проблем у системі. Опишемо проблему, яка повинна розв’язуватися в межах створюваної системи. Необхідно створити систему моніторингу діяльності салону с продажу мобільних телефонів – систему автоматизації послуг оформлення замовлення, видачі певної звітності, яка буде забезпечувати функціональність уже існуючих систем із додатковими можливостями видачі статистичної інформації на основі моніторингу.

Розроблювальна нами система повинна відповідати певним вимогам, основним положенням, принципам системного підходу, які є основними вимогами до подібних ситем:

- принцип остаточної мети: створення системи моніторингу діяльності салону з продажу мобільних телефонів;
- принцип єдності: системи моніторингу діяльності салону з продажу мобільних телефонів повинна бути цілісною і функціонувати як єдина система, але одночасно повинна складатися з окремих компонентів, підсистем для: обробки і систематизації інформації, обслуговування клієнтів та ведення обліку продажу товарів, обліку товарів .
- принцип зв’язності: довільний елемент системи моніторингу діяльності салону з продажу мобільних телефонів повинен бути розглянутий сумісно з іншими елементами та зв’язками з ними.

- принцип модульності: в системі необхідно реалізувати декомпозицію на складові з різною функціональністю, кожна з яких має виконувати певні, призначені їй дії: робота з базою даних, моніторинг роботи, створення звітності; необхідно розглянути всі модулі в сукупності та реалізувати зв'язки між ними.
- принцип ієрархії: система повинна мати ієрархічну будову, кожен функціональний елемент системи, в залежності від його важливості, повинен підпорядковуватися іншим елементам, які являються більш загальними і включають в себе необхідні функції.
- принцип функціональності: структура системи та її функції повинні розглядатися сумісно з пріоритетом функції над структурою; структура системи повинна бути побудована таким чином, щоб кожен елемент системи міг коректно працювати, не заважаючи роботі інших елементі системи.
- принцип розвитку: система повинна мати можливість для розвитку, враховуючи змінність системи, її розширення, заміну складових частин у майбутньому, при необхідності; систему необхідно реалізувати таким чином, щоб її розвиток не завдавав труднощів при її подальшій експлуатації, удосконаленні, заміні деяких її складових частин.
- принцип децентралізації: співвідношення між централізацією та децентралізацією, в управлінні системою, визначається призначенням та метою системи.

принцип невизначеності: при побудові системи, необхідно взяти до уваги невизначеності та випадковості; потрібно врахувати можливі подальші шляхи розвитку діяльності салону з продажу мобільних телефонів у майбутньому, та врахувати їх у стратегію та тактику розвитку системи.

Система моніторингу діяльності салону з продажу мобільних телефонів є множиною об'єктів разом з відношеннями між об'єктами та їхніми властивостями, що утворюють цілісність. Під даною системою розуміється сукупність функціональних частин, кожна з яких виконує поставлені перед нею задачі:

- зберігання і надання доступу до даних;
- вести облік товарів;
- формувати базу співробітників підприємства;
- проводити аналіз роботи менеджерів;
- формувати фінансовий та статистичний звіти про діяльність менеджера за будь-який період.

Всі елементи системи повинні бути взаємопов'язані, мати доступ до функціоналу інших елементів системи, та даних, які вони використовують.

Мета системи конкретизується за допомогою цілей. Одним зі способів розкриття внутрішньої структури мети є побудова дерева цілей.

Дерево цілей системи (Рисунок 2.1) складається з трьох рівнів. Верхній рівень дерева демонструє основну, невизначену ціль системи – реалізація обліку та моніторингу салону з продажу мобільних телефонів. На середньому рівні містяться конкретні цілі, які повинна виконувати система, але не вказано яким чином вона повинна це робити. На нижньому рівні містяться кроки, які необхідно виконати для досягнення цілей нашої системи, визначених на середньому рівні дерева цілей.

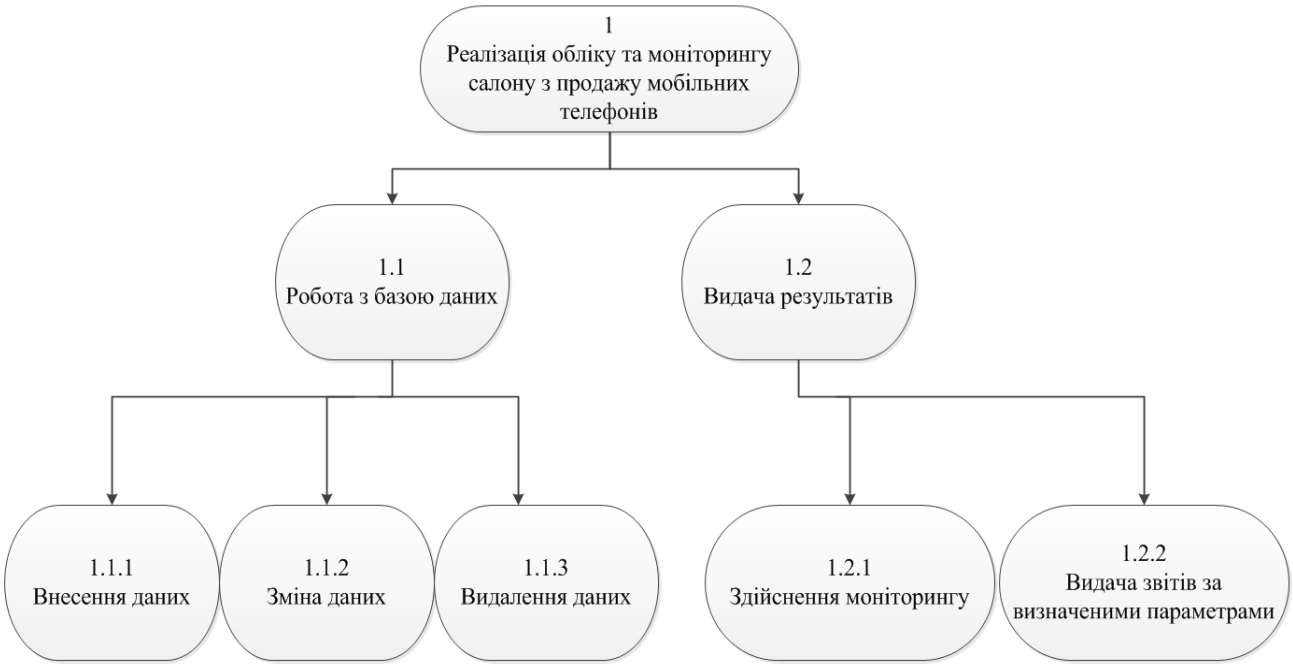


Рисунок 2.1 – Дерево цілей системи

Системний аналіз було здійснено з метою чіткого визначення мети і призначення системи, що розробляється, розуміння структури і механізму її функціонування, сформовано конкретні критерії відбору засобів для досягнення цілей системи. Було визначено яким вимогам і принципам системного аналізу відповідає дана система. Чітко сформовані функціональні частини системи дозволили краще уявити систему в загальному, та спростити її розробку. Було визначено середовище, загальну структуру та принцип роботи системи зсередини.

Загалом, проаналізувавши вищесказане можна зробити висновок, що дана система буде мати змішану архітектуру, тобто поєднувати у собі елементи архітектури незалежних компонент (клієнт – серверна система), репозиторної архітектури (наявність бази даних) та архітектури потоків даних.

2.2 Структура та функціональне призначення модулів додатку їх взаємозв'язок

Система моніторингу діяльності салону з продажу мобільних телефонів являє собою інформаційну систему, яка складається з окремих структурних компонентів та модулів. Кількість модулів та їх внутрішня реалізація може варіюватися в процесі розробки або модернізації програмного продукту, однак це не вплине на надійність інформаційної системи в цілому. Така архітектура інформаційної системи відкриває широкі можливості щодо нарощування функціональних можливостей за рахунок додавання нових модулів, удосконалення алгоритмів функціонування окремих модулів без необхідності внесення змін до інших модулів системи.

За рахунок використання подібної архітектури зростає надійність програмного забезпечення інформаційної системи, оскільки помилки в одному з модулів не призведе до припинення роботи всієї інформаційної системи та сайту заглом, а призведе лиш до тимчасової втрати користувачами однієї з багатьох функціональних можливостей .

При цьому інформаційна система в цілому продовжуватиме функціонувати в той час, як паралельно будуть здійснюватися заходи щодо виявлення причин аварійного завершення непрацюючого модуля, усунення помилок та відновлення роботи модуля.

Охарактеризуємо основні модулі нашої інформаційної системи моніторингу діяльності салону мобльних телефонів.

Модуль гостя – забезпечує видачу звітності, перегляду інформації, відображення результатів моніторингу.

Модуль адміністрування – який забезпечує інформативне наповнення та керування сайтом моніторингу діяльності салону з продажу мобільних телефонів.

Модуль обробки даних – даний модуль забезпечує логіку роботи усіє системи моніторингу діяльності салону продаж мобільної техніки, як менеджерів, так продаж товарів загалом. Зокрема визначені модулі обробки аналізі, та обробки даних з їх наступною обробкою, виведенням результатів в базу даних та на екранні форми, як для користувачів системи, керівників так і адміністраторів даного програмного засобу.

Для проектування даної системи нами було використано об’єктно-орієнтований підхід, та відповідно розроблено ряд діаграм послідовностей для варіантів використання. Діаграма послідовності призначена для відображення взаємодії об'єктів, які впорядковані за часом. Так, зокрема, такий тип діаграм відображає усі задіяні об'єкти та відповідно послідовність відправлення повідомлень в системі. Тобто діаграми послідовностей відображають саме часові особливості передачі-прийому повідомлень певними об'єктами.

Наведені діаграми послідовностей використаємо для уточнення діаграм прецедентів, а також більш детального опису логіки роботи побудованих сценаріїв використання.. На наших діаграмах послідовностей відобразимо тільки ті об'єкти, які безпосередньо будуть брати участь в їх взаємодії та повідомлення, для обміну між ними, та повертають результати пов'язані з відповідними повідомленнями.

Зокрема, на рисунку 2.2 подано діаграму послідовності модуля «Аналіз запитів-файлів»

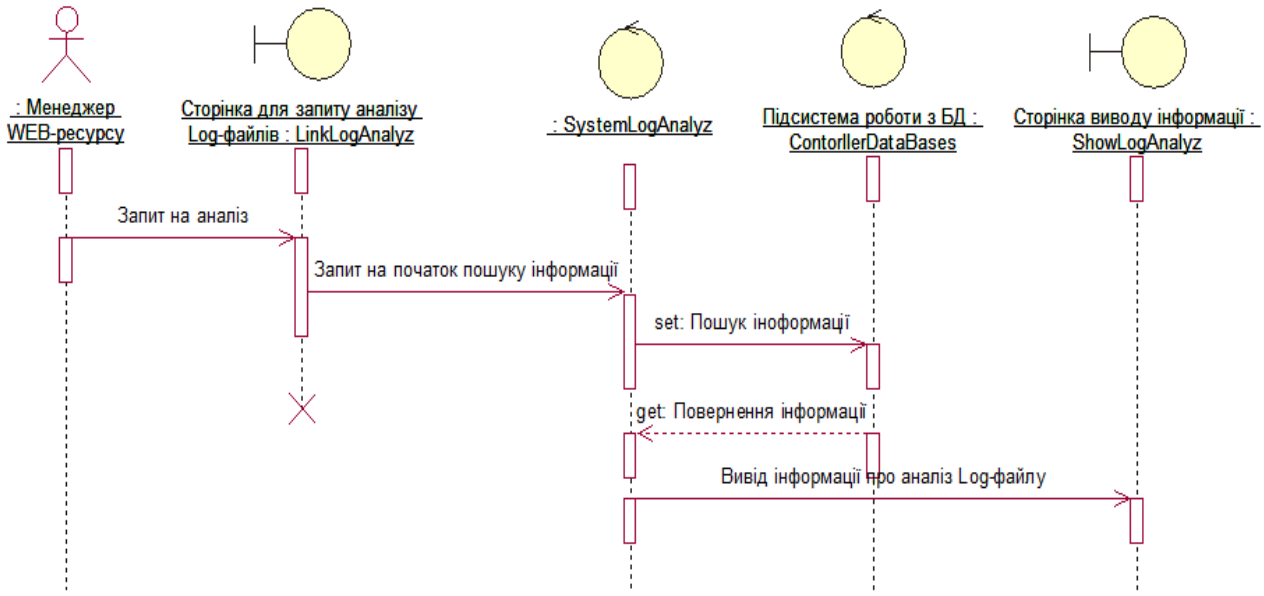


Рисунок 2.2 – Діаграма послідовності модуля аналізу запитів

Наступним кроком є побудова діаграми кооперацій, яка надає представлення про відношення класів, підсистем та інтерфейсів. На рисунку 2.3 подано діаграму кооперацій модуля аналізу запитів.

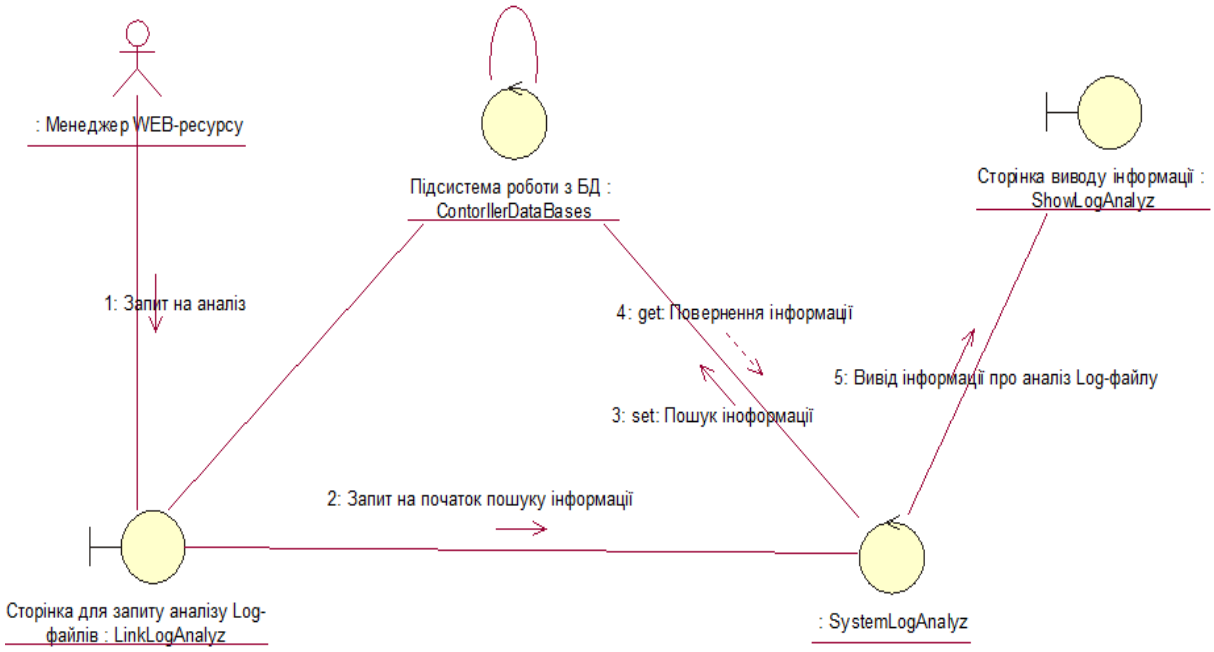


Рисунок 2.3 – Діаграма кооперацій модуля аналізу запитів

Аналогічним чином створюємо діаграму модуля для статистичних сервісів (рисунок 2.4).

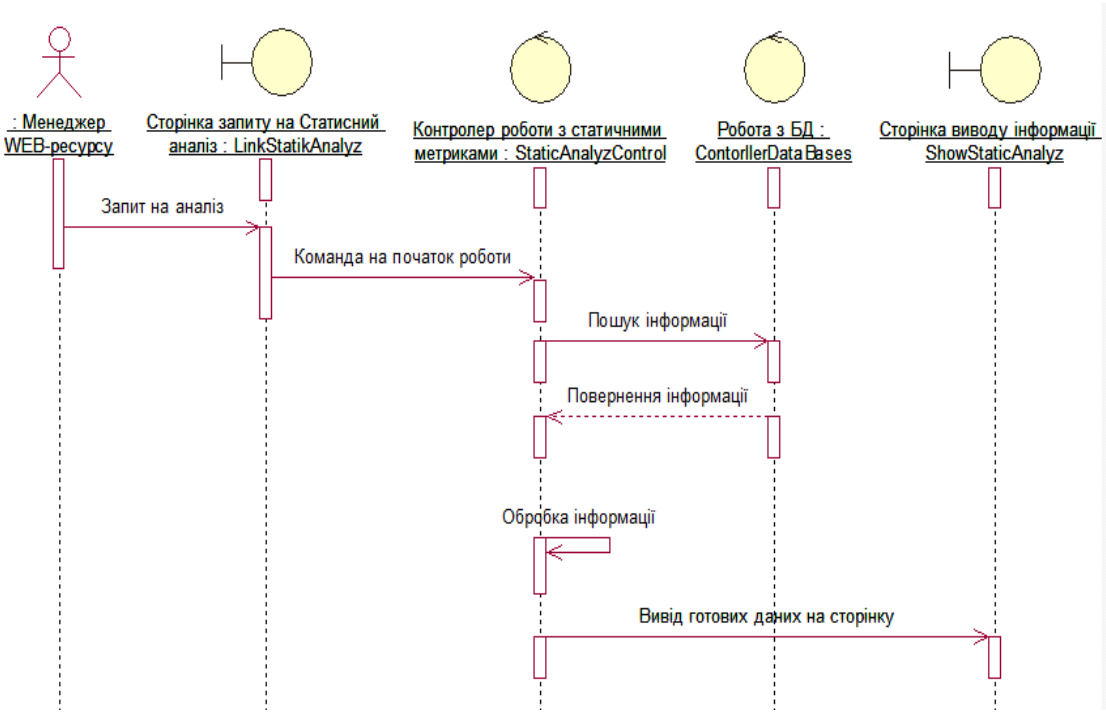


Рисунок 2.4 – Діаграма послідовності модуля «Статистичні сервіси»

На рисунку 2.5 подано діаграму кооперацій для модуля статистичних сервісів.

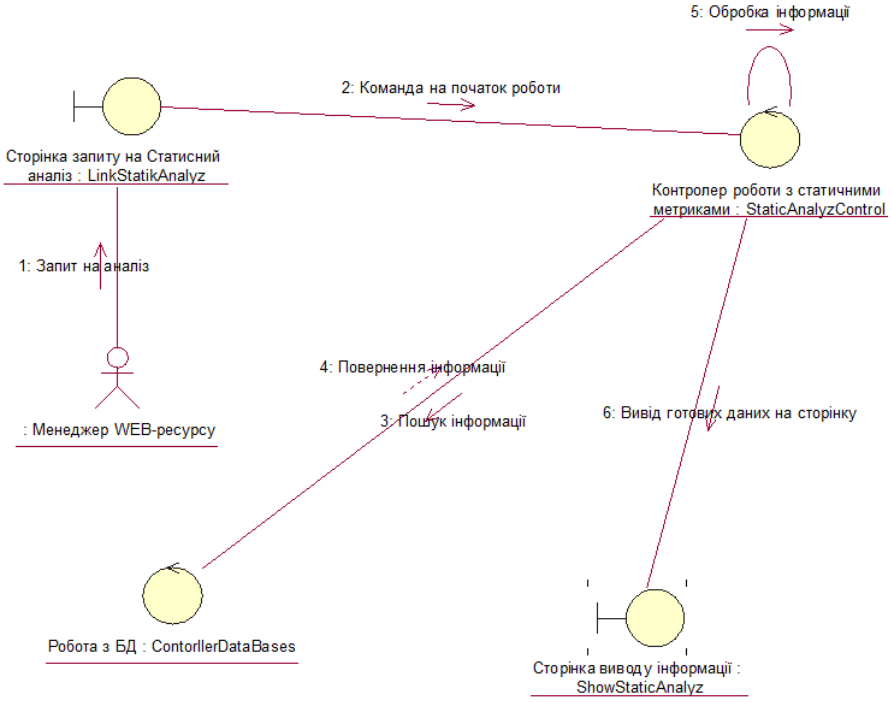


Рисунок 2.5 – Діаграма кооперацій модуля «Статистичні сервіси»

Наступник кроком опишемо відношення спроектованих класів для модуля наших сервісів (рисунок 2.6).

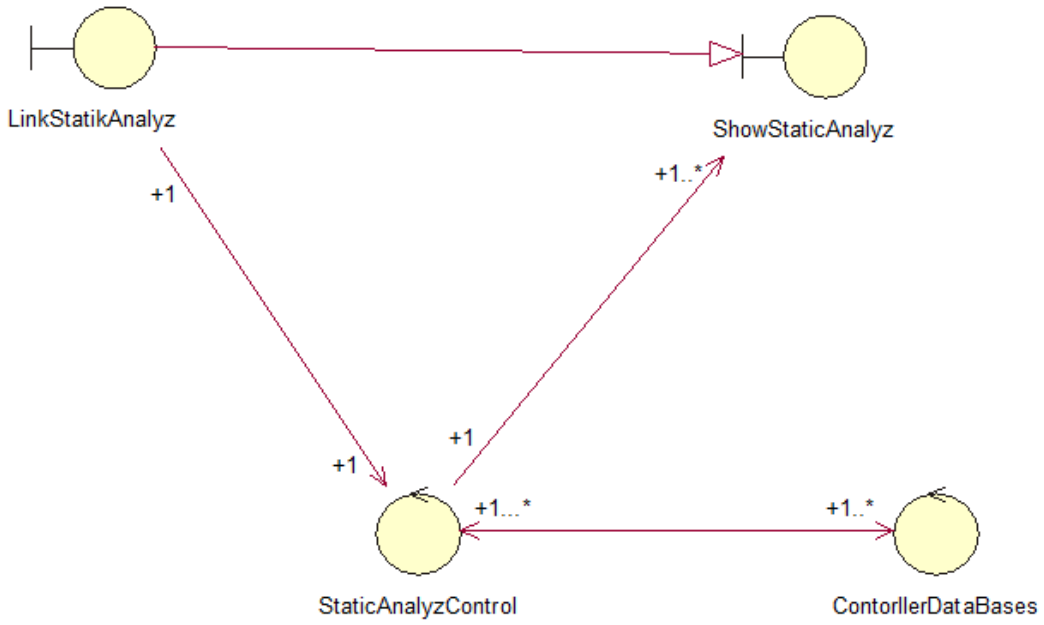


Рисунок 2.6 – Діаграма класів для модуля статистичних сервісів

Далі нам необхідно визначити структуру даних нашої системи та побудувати інфологічну модель (Рисунок 2.7)

Отже, для коректної роботи системи необхідно забезпечити зберігання даних у п’яти таблицях: Виробники та Товари (містять дані про товари салону), Менеджери (містять дані про персонал салону), ОбсягПродажів(містять дані про операції купівлі - продажу), ПродажТоварів(об’єднує дані попередніх таблиць та визначає дату операції купівлі - продажу). Структура таблиць наведена нижче, подана у скороченій формі, для можливості її реалізації у будь-якій СКБД реляціного типу з представленням її у фізичній моделі. У нашому проекті дана модель реалізована в СКБД Microsoft SQL Server, з подальшим її використанням у додатку за допомогою програмного засобу Microsoft SQL Server Express та технологій доступу до баз даних, реалізованих за допомогою .NET Core -

універсальною, модульною, кросплатформною технологією з відкритим вихідним кодом (Таблиця 2.1- 2.5)

Таблиця 2.1 – Структура таблиці Виробники

Назва поля	Тип поля
КодВиробника	цілочисельний
НазваВиробника	текстовий

Таблиця 2.2 – Структура таблиці Товари

Назва поля	Тип поля
КодТовару	цілочисельний
НазваТовару	текстовий
КодВиробника	цілочисельний
Тип	текстовий
Ціна	дійсний
НаСкладі	цілочисельний

Таблиця 2.3 – Структура таблиці Менеджери

Назва поля	Тип поля
КодМенеджера	цілочисельний
Прізвище	текстовий
Імя	текстовий
ПоБатькові	текстовий
Адреса	текстовий
РікПрийому	цілочисельний
Оклад	дійсний

Таблиця 2.4 – Структура таблиці ПродажТоварів

Назва поля	Тип поля
КодПродажу	цілочисельний

КодМенеджера	цілочисельний
Дата	дата

Таблиця 2.5 – Структура таблиці ОбсягПродажів

Назва поля	Тип поля
№ продажу	цілочисельний
КодПродажу	цілочисельний
КодТовару	цілочисельний
Кількість	цілочисельний

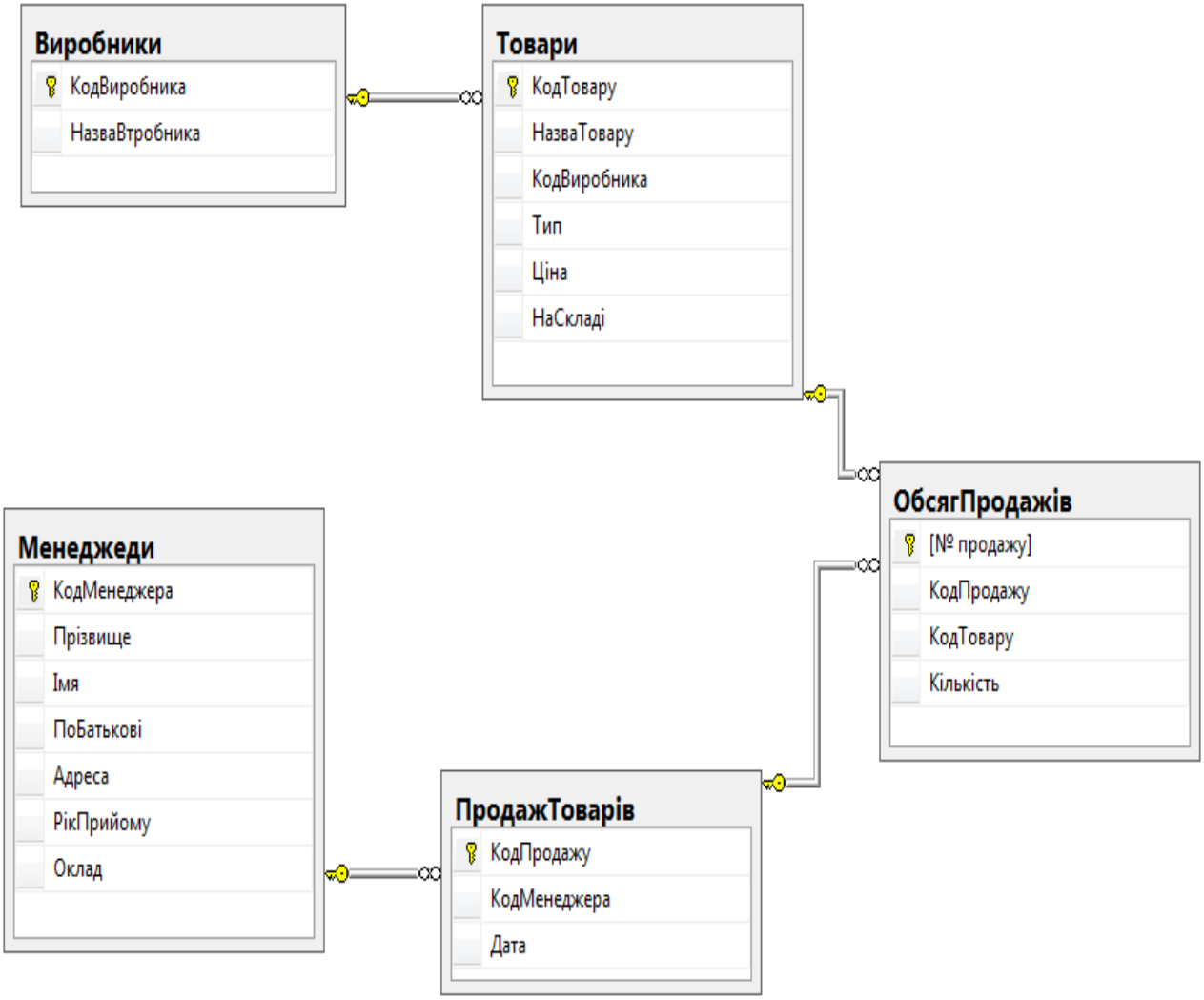


Рисунок 2.7 – Інфологічна модель

2.3 Вибір мови програмування та засобів для розробки для розробки

Сьогодні існує чимало засобів, які дозволяють створювати як невеличкі сайти-візитки, так і великі потужні портали, корпоративні сайти, інформаційні проекти, але з множини засобів чітко можна виділити два основних напрямки:

– Linux Apache MySQL;

– MS Windows IIS ASP MSSQL/MS Access.

Для розробки системи було обрано другий напрям, цей вибір ґрунтується на ряді переваг ASP.Net.

ASP.Net - це найсучасніша технологія розробки веб - сайтів та веб - додатків на даний момент. Велика частина веб - проектів у світі на зараз реалізується саме за допомогою цієї технології.

Основні переваги asp. Net:

– сучасна компільована мова програмування (C#, VB.NET) з підтримкою об'єктно-орієнтованого програмування. C# є найпоширенішою у світі мовою програмування як для розробки Windows-додатків, так для розробки веб - додатків. Відповідно існує велика кількість фахівців знаючих дану мову;

– зручне середовище розробки програм, наявність відладчика, прискорює процес розробки;

– наявність засобів багаторазового використання коду - для користувача елементи управління і веб – контролери;

– наявність великої кількості компонентів для вирішення стандартних завдань - роботи з даними, авторизації, навігації тощо;

– наявність засобів кешування, що дозволяють збільшити продуктивність додатків;

– зручні засоби розробки дизайну - майстер-сторінки, скіни;

– вбудовані засоби зберігання даних сесії та додатків на сервері;

– підтримка багатомовності;

– ефективна технологія доступу до даних ADO. NET;

– велика кількість стандартних об'єктів, що надається платформою Microsoft .Net Framework;

– строга типізація, зменшуюча ймовірність створити код з помилками та підвищує швидкодію програм;

– модель програмування, заснована на подіях, аналогічна використовуваної при розробці програм для Windows;

– відділення коду від візуальної частини;

– вбудована підтримка AJAX;

– підтримка веб - сервісів;

– ефективна робота з усіма основними серверами баз даних, найчастіше використовується MS SQL - високопродуктивний сервер, який є найпоширенішим в світі.

Це найбільш очевидні, але далеко не всі переваги платформи asp. Net. Найголовніше - це сучасність, і перш за все саме тому потрібно вибрати технологію asp. Net, щоб не жити вчорашнім днем. Існує кілька хибних тверджень про asp. Net. По-перше - те, що це дуже дорого через необхідність використання платного програмного забезпечення як на хостингу, так і при розробці програм. В дійсності вартість Windows - хостингу вже майже зрівнялася з хостингом Linux. Розробники також не обов'язково повинні використовувати платні інструменти, існує, наприклад, безкоштовний Visual Web Developer, обмеження якого - неможливість використання для написання програм під Windows (для веб-проектів це і не потрібно) і неможливість компіляції програм (сайт буде автоматично відкомпільований на хостингу, при першому завантаженні сторінок). По-друге - те, що asp. Net сайти повільно працюють. Вони працюють достатньо швидко, якщо кваліфіковано розроблені, повільно сторінки завантажуються тільки в перший раз, якщо для них вимагається компіляція.

Також на вибір другого шляху вплинуло те, що всі програмні засоби створені однією компанією Microsoft, що забезпечує абсолютну їхню сумісність.

IIS (Internet Information Server) — це набір серверів для декількох служб Інтернету від компанії Майкрософт. IIS поширюється з операційними системами родини Windows NT.

Основний компонент IIS — веб-сервер, який дозволяє розміщувати в Інтернеті сайти. IIS підтримує протоколи HTTP, HTTPS, FTP, POP3, SMTP, SSH, TCP, NNTP.

Microsoft SQL Server — комерційна система керування базами даних, що розповсюджується корпорацією Microsoft. Мова, що використовується для запитів — Transact-SQL, створена спільно Microsoft та Sybase. Transact-SQL є реалізацією стандарту ANSI/ISO щодо структурованої мови запитів (SQL) із розширеннями. Використовується як для невеликих і середніх за розміром баз даних, так і для великих баз даних масштабу підприємства. Багато років вдало конкурує з іншими системами керування базами даних.

2.4 Структура і функціональне призначення модулів системи, їх взаємозв’язок

Розробка сайту складається з aspx – сторінок дизайну та aspx.cs – сторінок коду мови C#.

До основних модулів системи відносяться компоненти з яких вона складається, а саме сторінки системи, засоби визначення зовнішнього вигляду (файли стилів, шрифти, зображення, скрипти та інше) , файли конфігурацій,а також база даних з якою вони зв'язані.

Сукупність сторінок до яких є вільний доступ є клієнтською частиною, за допомогою якої можна проглядати інформацію на головній сторінці сайту, здійснювати процес авторизації, пошуку даних, надсилання запитів на видачу звітів чи моніторингу.

Сукупність сторінок до яких закритий вільний доступ є адміністративною частиною, через яку здійснюється опрацювання (внесення, зміна, вилучення) інформації бази даних.

Загальна структура розроблювального додатку в середовищі Microsoft Visual Studio (вікно Solution Explorer) наведена на рисунку 2.8. На даному рисунку зображено основні модулі проекту, сторінки сайту aspx, їх властивості, модулі підключення до бази даних тощо.

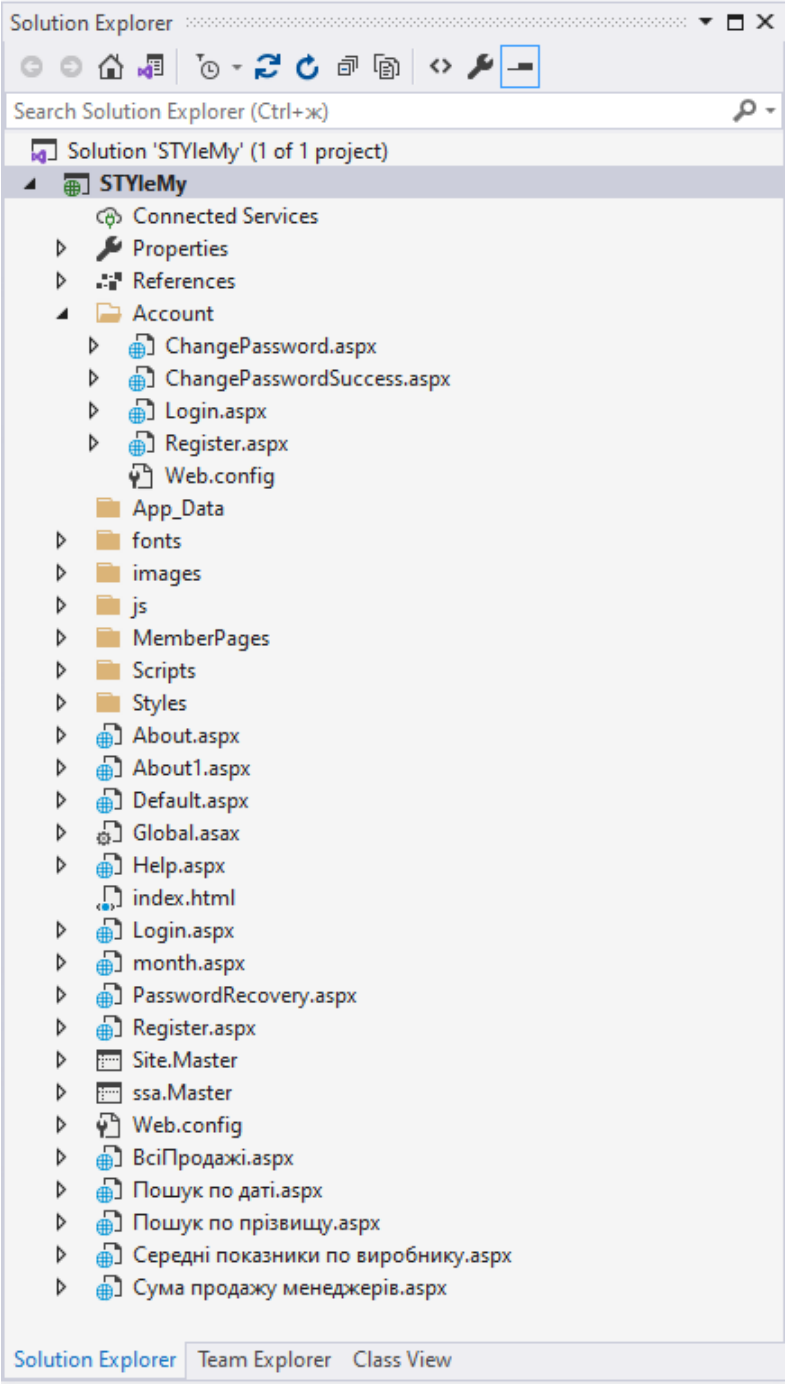


Рисунок 2.8 – Вікно Solution Explorer

Розглянемо детальніше призначення основних модулів розроблювальної інформаційної системи:

About.aspx – aspx-сторінка з інформаціє про розробників інформаційної системи та їх контактні дані;

Default.aspx – aspx-сторінка яка використовується на сайті по замовчуванню при запуску сайту в браузері;

Global.asax – призначений для запуску коду при запуску додатку, при завершені роботи додатку, при виникненні помилки, при запуску нового сеансу та при запуску додатку для режиму sessionstate;

Help.aspx – aspx-сторінка, яка містить коротку інструкцію користувача;

Login.aspx – aspx-сторінка авторизації користувача на сайті;

month.aspx – aspx-сторінка обробки результатів моніторингу діяльності салону з продажу мобільних телефонів;

PasswordRecovery.aspx – aspx-сторінка відновлення паролю користувача;

Register.aspx – aspx-сторінка реєстрації користувачів системи;

Web.config – модуль налаштування конфігурації сайту;

ВсіПродажі.aspx – aspx-сторінка обробки продажу мобільних телефонів менеджерами салону;

Пошук по даті.aspx – aspx-сторінка обробки пошукових запитів повязаних з датою продаж та закупівель;

Пошук по прізвищу.aspx – aspx-сторінка обробки запитів по роботі менеджерів салону;

Середні показники по виробнику.aspx – aspx-сторінка обробки інформації продажу телефонів певного виробника;

Сума продажу менеджерів.aspx – aspx-сторінка обробки запитів об’ємів продаж певного менеджера та загалом;

Site.Master – карта сайту та меню.

Крім наведених вище модулів системи також присутні моду ,що містять інформацію та їх обробку, зокрема fonts, images, js-скрипти, інші скрипти, таблиці

стилів та самі стилі, інформація про додаткові сторінки сайту, пукти меню та їх обробку тощо.

Отже, в результаті проектування програмного забезпечення моніторингу діяльності салону з продажу мобільних телефонів, було розроблено архітектуру та функціональну структуру додатку. Побудовано дерево цілей системи. Розроблено структуру та функціональне призначення системи з визначенням основних модулів та їх функціональним призначенням. Спроектовано інфологічну модель та побудовано базу даних. Далі здійснено вибір та обґрунтування мови та засобів розробки системи. Результатом проектування розділу було наведено структуру та функціональне призначення розроблювального додатку в середовищі Microsoft Visual Studio (вікно Solution Explorer).

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Розробка програмних модулів

Модуль роботи з базою даних – забезпечує внесення, редагування та вилучення інформації бази даних.

Уся робота з базою даних відбувається за допомогою запитів мови SQL, для передачі тексту запиту нам необхідний посередник, в asp.NET роль посередника виконує елемент управління SqlDataSource, якому задаються такі властивості: ConnectionString (зв'язок з базою даних), а також тексти різних запитів Select (відображення даних), Insetr(внесення даних), Update(зміна даних), Delete(вилучення даних). Також необхідно визначити елементи (контроли) з яких буде передаватись інформація запиту, до таких елементів відносяться DropDownList, TextBox, Calendar, та елементи візуального відображення результатів запиту GridView, Chart.

```
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%$ ConnectionStrings:ConnectionString %>"
            SelectCommand="SELECT [КодВиробника],
[НазваВтробника] FROM [Виробники]"
            DeleteCommand="DELETE FROM Виробники WHERE
(КодВиробника = @Kvur3)"
            InsertCommand="INSERT INTO
Виробники(НазваВтробника) VALUES (@nvur1)"
            UpdateCommand="UPDATE Виробники SET НазваВтробника
= @nvur2 WHERE (КодВиробника = @kvur2)">
            <DeleteParameters>
                <asp:ControlParameter
ControlID="DropDownList3" Name="Kvur3"
                    PropertyName="SelectedValue" />
            </DeleteParameters>
            <InsertParameters>
                <asp:ControlParameter ControlID="TextBox1"
Name="nvur1" PropertyName="Text" />
            </InsertParameters>
            <UpdateParameters>
```

```
                <asp:ControlParameter ControlID="TextBox2"
Name="nvur2" PropertyName="Text" />
                <asp:ControlParameter
ControlID="DropDownList2" Name="kvur2"
                    PropertyName="SelectedValue" />
            </UpdateParameters>
        </asp:SqlDataSource>
```

Для виконання запиту при здійсненні певної події необхідно у коді C# викликати відповідні методи елемента SqlDataSource

```
   protected void Button2_Click(object sender, EventArgs e)
           {
               SqlDataSource1.Insert();
           }

           protected void Button3_Click(object sender,
EventArgs e)
           {
               SqlDataSource1.Update();
           }

           protected void Button4_Click(object sender,
EventArgs e)
           {
               SqlDataSource1.Delete();
               GridView1.DataBind();
               GridView2.DataBind();
               GridView3.DataBind();
               GridView4.DataBind();
           }
```

Метод DataBind(); використовується для елементів відображення, щоб оновити отриману інформацію на екрані.

Модуль моніторингу та видачі звітності –схожий до попереднього, але в ньому використовуються більш складні запити з вибіркою по декількох таблицях за певними параметрами з агрегуючими функціями, для яких здійснюється попереднє групування.

```
SELECT Менеджеди.Прізвище, Менеджеди.Імя, Менеджеди.ПоБатькові, SUM(ОбсягПродажів.Кількість * Товари.Ціна) AS Сума FROM ПродажТоварів INNER JOIN Менеджеди ON ПродажТоварів.КодМенеджера = Менеджеди.КодМенеджера INNER JOIN ОбсягПродажів ON ПродажТоварів.КодПродажу = ОбсягПродажів.КодПродажу INNER JOIN Виробники INNER JOIN Товари ON Виробники.КодВиробника = Товари.КодВиробника ON ОбсягПродажів.КодТовару = Товари.КодТовару GROUP BY Менеджеди.Прізвище, Менеджеди.Імя, Менеджеди.ПоБатькові
```

```
SELECT Менеджеди.Прізвище, Менеджеди.Імя, Менеджеди.ПоБатькові, Виробники.НазваВтробника, Товари.НазваТовару, Товари.Ціна * ОбсягПродажів.Кількість AS Сума, ПродажТоварів.Дата FROM ПродажТоварів INNER JOIN Менеджеди ON ПродажТоварів.КодМенеджера = Менеджеди.КодМенеджера INNER JOIN ОбсягПродажів ON ПродажТоварів.КодПродажу = ОбсягПродажів.КодПродажу INNER JOIN Товари ON ОбсягПродажів.КодТовару = Товари.КодТовару INNER JOIN Виробники ON Товари.КодВиробника = Виробники.КодВиробника WHERE (MONTH(ПродажТоварів.Дата) = @321)
```

Також необхідно встановити для певної частини інформаційної системи доступ, який можливий тільки після проходження авторизації в системі.

Даний доступ задається у вікні конфігурації asp.NET на вкладці Безпека.

Зовнішній вигляд сайту детально описується у файлі стилів з розширенням *.css.

Так, на рисунку 3.1 зображено вікно покрокового налаштування прав доступу asp.NET.

Даний майстер налаштування доступу є вбудованим майстром середовища Microsoft Visual Studio.

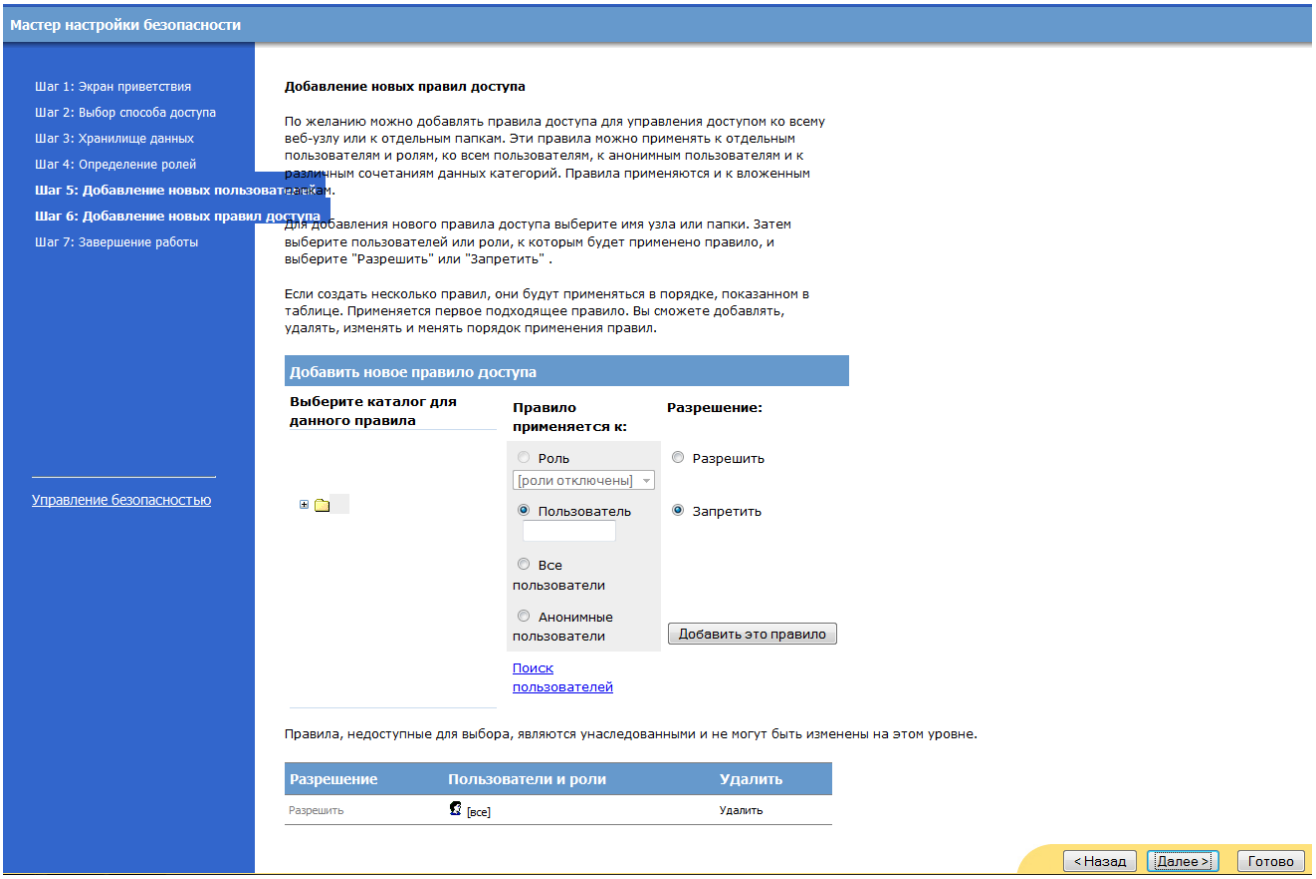


Рисунок 3.1 – Вікно покрокового налаштування прав доступу asp.NET

3.2 Розробка інтерфейсу користувача

Щоб розпочати роботу з системою моніторингу діяльності салону з продажу мобільних телефонів потрібно запусти інтернет браузер та ввівши адресу сайту, яка може бути різною в залежності від розміщення. Ввівши вірну адресу користувач потрапляє на головну сторінку (рисунок 3.2).

Користувач може ознайомитись з вмістом сайту, та його можливостями за допомогою верхнього меню. Головне меню сайту моніторингу діяльності салону з продажу мобільних телефонів дає можливість доступу до розділів сайту:

Головна, Бази даних, Моніторинг роботи, звіти, довідка та інформацію про програму, її розробників і координати для зв’язку.

У правій частині сайту знаходяться блоки з відомостями про останні новини, а також перелік корисних посилань, які можуть знадобиться користувачеві при роботі з сайтом.

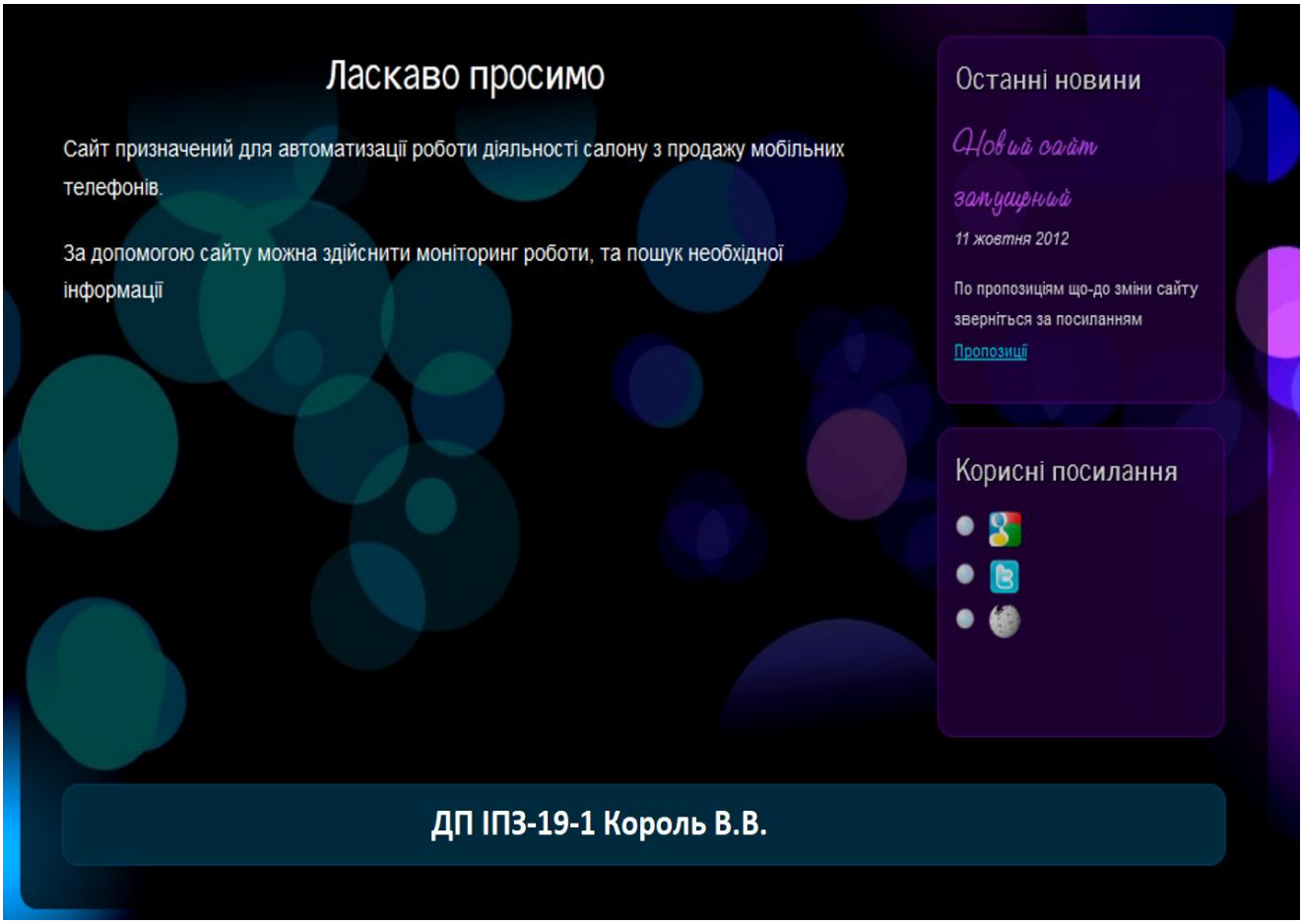


Рисунок 3.2 – Головна сторінка системи моніторингу діяльності салону з продажу мобільних телефонів

Для використання повної функціональності сайту користувачу необхідно зареєструватися. Форма реєстрації (Рисунок 3.3) з’являється після вибору команди Реєстрація. Дана форма містить чотири обов’язкових поля, які користувачу необхідно коректно та без помилок заповнити. Після закінчення процесу реєстрації у браузері з’являється інформація про успішну реєстрацію та продовження роботи (рисунок 3.4).

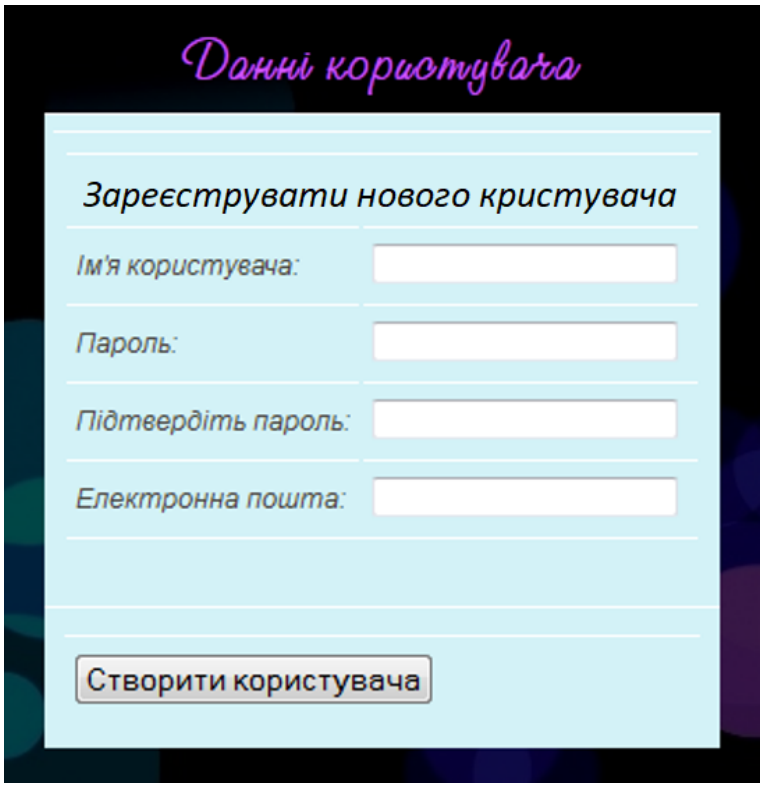


Рисунок 3.3 – Реєстрація користувача

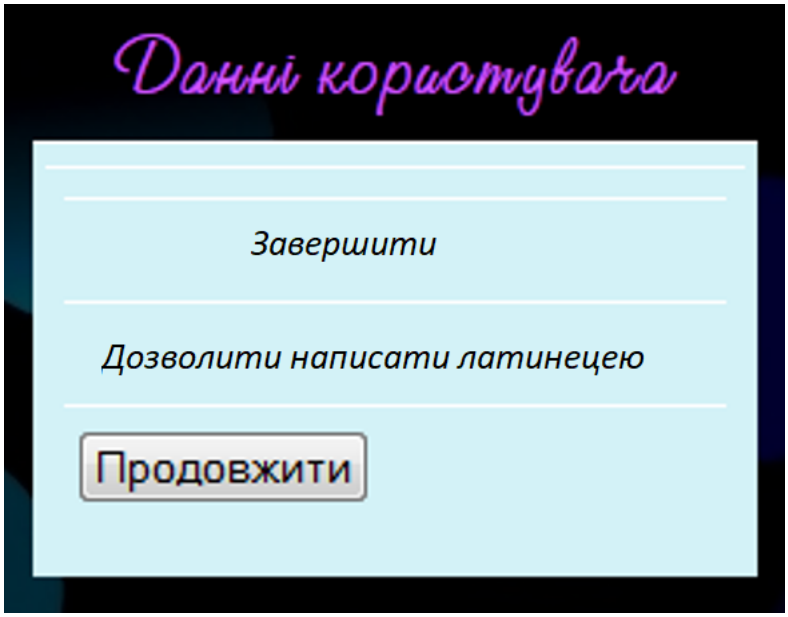


Рисунок 3.4 – Реєстрація користувача

Подальша робота з сайтом відбувається через головне меню, яке описано вище. Загалом принцип роботи сайту побудований таким чином, щоб користувач легко міг ним користуватись. Тут застосовані як звичайні технології роботи з веб-сторінками в поданні з роботою в звичайних інформаційних системах за

принципом роботи діалогових вікон, та в режимі роботи користувача «запитання-відповідь».

Крім того на сайті доступна звітність по роботі інформаційної систем и моніторингу діяльності роботи салону з продажу мобільних телефонів, як для менеджерів ккомпанії так і для її власників і керівників.

Робота з інформацією бази даних доступна лише зареєстрованим користувачам у пункті меню «База даних». Для цього необхідно обрати потрібний пункт, після чого завантажитися сторінка з відображеними даними таблиці (рисунок 3.5), та вибором дії з випадаючого списку, після вибору сторінка оновиться залежно від вибору відобразиться форма внесення даних (рисунок 3.6), зміни даних (рисунок 3.7), вилучення даних (рисунок 3.8), в яких необхідно заповнити усі поля та натиснути на відповідну кнопку.

КодТовару	НазваТовару	КодВиробника	Тип	Ціна	НаСкладі
1	Galaxy S Duos	3	Моноблок	2739,00	125
2	Lumia 900	1	Моноблок	3377,00	200
3	C2-05	1	Слайдер	574,00	250
4	S720e One X	2	Моноблок	5376,00	100
5	LG Optimus L9	4	Моноблок	3288,00	150
7	Asha 200	1	QWERTY	677,00	55
8	Lumia 800	1	Моноблок	2655,00	200
9	1280	1	Моноблок	210,00	300
10	Sensation XL	2	Моноблок	4472,00	175
11	Desire V	2	Моноблок	2828,00	150
12	Titan	2	Моноблок	3700,00	75
13	Galaxy S III	3	Моноблок	5329,00	400
14	Note II	3	Моноблок	6097,00	150
15	Wave M	3	Моноблок	1874,00	50
16	T370	4	Моноблок	715,00	150
17	Optimus P970	4	Моноблок	2030,00	200
18	Xperia Mini	8	Моноблок	1586,00	75
19	IQ441	9	Моноблок	1227,00	250
20	E141	9	Моноблок	469,00	125
21	Ezzy Trendy	9	Розкладний	625,00	200

Рисунок 3.5 – Сторінка перегляду даних таблиці

КодТовару	НазваТовару	КодВиробника	Тип	Ціна	НаСкладі
1	Galaxy S Duos	3	Моноблок	2739,00	125
2	Lumia 900	1	Моноблок	3377,00	200
3	C2-05	1	Слайдер	574,00	250
4	S720e One X	2	Моноблок	5376,00	100
5	LG Optimus L9	4	Моноблок	3288,00	150
7	Asha 200	1	QWERTY	677,00	55
8	Lumia 800	1	Моноблок	2655,00	200
9	1280	1	Моноблок	210,00	300
10	Sensation XL	2	Моноблок	4472,00	175
11	Desire V	2	Моноблок	2828,00	150
12	Titan	2	Моноблок	3700,00	75
13	Galaxy S III	3	Моноблок	5329,00	400
14	Note II	3	Моноблок	6097,00	150
15	Wave M	3	Моноблок	1874,00	50
16	T370	4	Моноблок	715,00	150
17	Optimus P970	4	Моноблок	2030,00	200
18	Xperia Mini	8	Моноблок	1586,00	75
19	IQ441	9	Моноблок	1227,00	250
20	E141	9	Моноблок	469,00	125
21	Ezzy Trendy	9	Розкладний	625,00	200

Рисунок 3.6 – Сторінка внесення даних у таблицю

КодТовару	НазваТовару	КодВиробника	Тип	Ціна	НаСкладі
1	Galaxy S Duos	3	Моноблок	2739,00	125
2	Lumia 900	1	Моноблок	3377,00	200
3	C2-05	1	Слайдер	574,00	250
4	S720e One X	2	Моноблок	5376,00	100
5	LG Optimus L9	4	Моноблок	3288,00	150
7	Asha 200	1	QWERTY	677,00	55
8	Lumia 800	1	Моноблок	2655,00	200
9	1280	1	Моноблок	210,00	300
10	Sensation XL	2	Моноблок	4472,00	175
11	Desire V	2	Моноблок	2828,00	150
12	Titan	2	Моноблок	3700,00	75
13	Galaxy S III	3	Моноблок	5329,00	400
14	Note II	3	Моноблок	6097,00	150
15	Wave M	3	Моноблок	1874,00	50
16	T370	4	Моноблок	715,00	150
17	Optimus P970	4	Моноблок	2030,00	200
18	Xperia Mini	8	Моноблок	1586,00	75
19	IQ441	9	Моноблок	1227,00	250
20	E141	9	Моноблок	469,00	125
21	Ezzy Trendy	9	Розкладний	625,00	200

Рисунок 3.7 – Сторінка редагування даних таблиці

КодТовару	НазваТовару	КодВиробника	Тип	Ціна	НаСкладі
1	Galaxy S Duos	3	Моноблок	2739,00	125
2	Lumia 900	1	Моноблок	3377,00	200
3	C2-05	1	Слайдер	574,00	250
4	S720e One X	2	Моноблок	5376,00	100
5	LG Optimus L9	4	Моноблок	3288,00	150
7	Asha 200	1	QWERTY	677,00	55
8	Lumia 800	1	Моноблок	2655,00	200
9	1280	1	Моноблок	210,00	300
10	Sensation XL	2	Моноблок	4472,00	175
11	Desire V	2	Моноблок	2828,00	150
12	Titan	2	Моноблок	3700,00	75
13	Galaxy S III	3	Моноблок	5329,00	400
14	Note II	3	Моноблок	6097,00	150
15	Wave M	3	Моноблок	1874,00	50
16	T370	4	Моноблок	715,00	150
17	Optimus P970	4	Моноблок	2030,00	200
18	Xperia Mini	8	Моноблок	1586,00	75
19	IQ441	9	Моноблок	1227,00	250
20	E141	9	Моноблок	469,00	125
21	Ezzy Trendy	9	Розкладний	625,00	200

Рисунок 3.8 – Сторінка видалення даних таблиці

Пункт меню моніторинг роботи використовується для визначення ефективності певних показників за певний період та відображенню графічно результатів моніторингу.

Прізвище	Імя	ПоБатькові	НазваВтробника	НазваТовару	Сума	Дата
Маляр	Олена	Анатоліївна	Apple	12	7400,00	14.11.2012
Линюк	Олег	Ігорович	Sumsung	Galaxy	1874,00	15.01.2022
Маляр	Олена	Анатоліївна	Apple	13	4060,00	14.01.2022
Поліщук	Ірина	Олександрівна	Apple	13	1430,00	20.01.2022
Поліщук	Ірина	Олександрівна	Apple	12	1227,00	20.01.2022
Поліщук	Ірина	Олександрівна	Sumsung	Galaxy	21316,00	20.11.2022
Поліщук	Ірина	Олександрівна	Sumsung	Galaxy	630,00	22.01.2022
Поліщук	Ірина	Олександрівна	Sumsung	Galaxy	715,00	22.01.2022

Діаграма продажів менеджерів за місяць

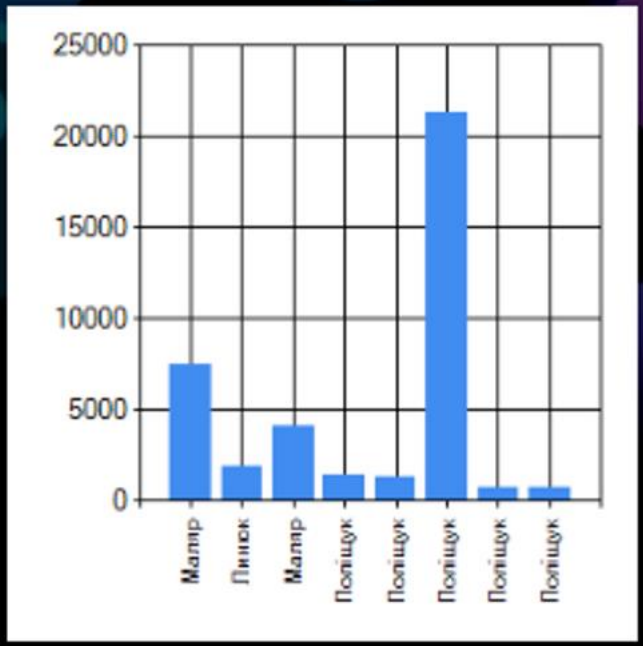


Рисунок 3.9 – Сторінка результатів моніторингу продажів за певний місяць

Пункт меню Звіти призначений для формування звітів за певний вибраний період (Рисунок 3.10)

Прізвище	Імя	ПоБатькові	НазваВтробника	НазваТовару	Кількість	Сума продажу	Дата
Петров	Петро	Петрович	APPLE	12	2	6754,00	06.05.2022
Петров	Петро	Петрович	APPLE	12	1	2655,00	06.05.2022
Петров	Петро	Петрович	APPLE	13	3	1722,00	06.05.2022

Рисунок 3.10 – Сторінка результатів звітності за певну дату

Пункт меню «Довідка» призначений для надання користувачу довідкової інформації по роботі з ситемою.

Пункт меню «Про програму» містить відомості програмного продукту та інформацію його розробника (рисунок 3.11)

Рисунок 3.11 – Сторінка про програму

3.3 Вимоги до технічних засобів

Оскільки, розроблена система моніторингу діяльності салону з продажу мобільних телефонів працює на основі клієнт-серверної технології, окремо розглянемо вимоги до технічних засобів клієнта та сервера.

Програмне забезпечення сервера:

- операційна система Windows Server;
- повинен бути встановлений та налаштований сервер IIS;
- повинен бути встановлений Framework 4.0 та вище;
- MS SQL Server;

Апаратне забезпечення сервера:

- підключення до мережі Internet;
- обсяг оперативної пам'яті від 512 MB;
- мінумум одне ядро багатоядерних процесорів компаній Intel чи AMD;
- вільне місце на жорсткому диску для зберігання веб-додатку, що складає приблизно 20 MB;
- вільне місце на жорсткому диску для зберігання бази даних приблизно 20Мб;

Програмне забезпечення клієнта:

- операційна система Windows 10? Windows 11;
- наявність веб-браузера, для повного функціонування має бути включена можливість виконання JavaScript скриптів;

Апаратне забезпечення клієнта:

- підключення до мережі Internet;
- обсяг оперативної пам'яті від 512 MB.

Дані вимоги є мінімальними вимогами, які забезпечують нормальне функціонування розробленої системи.

Загалом у розділі здійснено та показано процес розробки програмних модулів. Наведено структуру основних запитів в системі. Детально описано процес авторизації та розмежування прав доступу до даних. Наведено процес розробки інтерфейсу інформаційної системи з врахуванням розробки інтерфейсу користувача та наведено технічні характеристики необхідні для серверної та клієнтської частини самої інформаційної системи.

4 ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Вибір та обґрунтування методів тестування додатку

NET Core був випущений в 2016 році. З виходом .NET Core 2.0 у Microsoft з'явилася наступна основна версія платформи - універсальна, модульна, кросплатформна з відкритим вихідним кодом. .NET Core створив багато API, доступних у поточній версії .NET Framework. Спочатку він був побудований для наступного покоління ASP.NET рішень, але тепер став драйвером і основою для багатьох інших сценаріїв, включаючи Інтернет речей, хмарні обчислення та мобільні рішення наступного покоління. Розглянемо деякі можливості та переваги .NET Core в методах тестування додатків і про те, як це може принести користь не тільки традиційним розробникам .NET, але і всім технічним фахівцям, яким необхідно вивести на ринок надійні, ефективні та економічно ефективні рішення.

При використанні ASP.NET Core 2.1 Web API для створення і розробки багатого набору API, необхідно пам'ятати, що це лише перший крок до досягнення стабільного і ефективного рішення. Дуже важливо забезпечити стабільне середовище для нашого рішення. Отримання чудового рішення вимагає не тільки повного дизайну API, але також вимагає ретельного тестування нашого API, щоб переконатися, що користувачі матимуть хороший програмний засіб.

Загалом ASP.NET Core - це нова веб-платформа, створена корпорацією Майкрософт, щоб позбутися застарілих технологій, яка існує з ASP.NET 1.0. ASP.NET Core 2.1 позбавляється від цих застарілих залежностей і розробляє фреймворк з нуля, щоб він міг забезпечити розробникам кращу продуктивність і побудований для кросплатформного виконання.

Для деяких людей тестування програмного забезпечення може бути новою концепцією, але це дуже просто. Почнемо з модульних випробувань. Вікіпедія строго визначає це як «метод тестування програмного забезпечення, який

перевіряє незалежні одиниці вихідного коду, одну або кілька груп модулів комп'ютерних програм і пов'язаних з ними даних управління, використовує програми і робочі програми, щоб з'ясувати, чи підходять вони для використання». Таке пояснення дає досить загальне загальне представлення про тестування програмного забезпечення, тому ми, при проведені тестування нашого додатку будемо орієнтуватись на професійні визначення, та дотримуватись їх вимог. Модульне тестування повинно гарантувати, що після додавання нових функцій або виправлення помилок код у вашому рішенні буде працювати належним чином. Протестуємо простий зразок коду, щоб переконатися, що він відповідає нашим очікуванням:

```
[Fact]public async Task MobTelGetAllAsync(){    // Arrange    // Act    var mobtel = await _repo.GetAllAsync();    // Assert    Assert.Single(mobtel);}
```

Загалом модульний тест складається з трьох частин: Перша частина - організація. Ця частина використовується для налаштування будь-яких ресурсів, які можуть знадобитися в тесті. У наведеному коді не внесено жодних коригувань, тому частина «Упорядкувати» порожня (але зарезервовано коментар до неї). Друга частина - це деталь, яка використовується для виконання тесту. У нашому коді – це тип сутності мобільний салон бази даних, який повертає повну сутність джерела даних у поточній бібліотеці. Остання частина тесту на одиницю вимірювання частини, використовується для перевірки правильності операції, яка тестується. Для цього тесту я перевірив, чи було взято альбом на редагування з бази даних.

Ми використаємо інструмент xUnit для модульного тестування. xUnit — це пакунок .NET Framework з відкритим вихідним кодом, а тепер і .NET Core. Нам потрібна версія xUnit для .NET Core, яка виходить безпосередньо шляхом встановлення .NET Core 2.1 SDK. Створимо новий блок тестового проекту за

допомогою команди перевірки dotnet в cli .NET Core, а саме за допомогою шаблону проекту в IDE Microsoft Visual Studio 2019 (Рисунок 4.1).

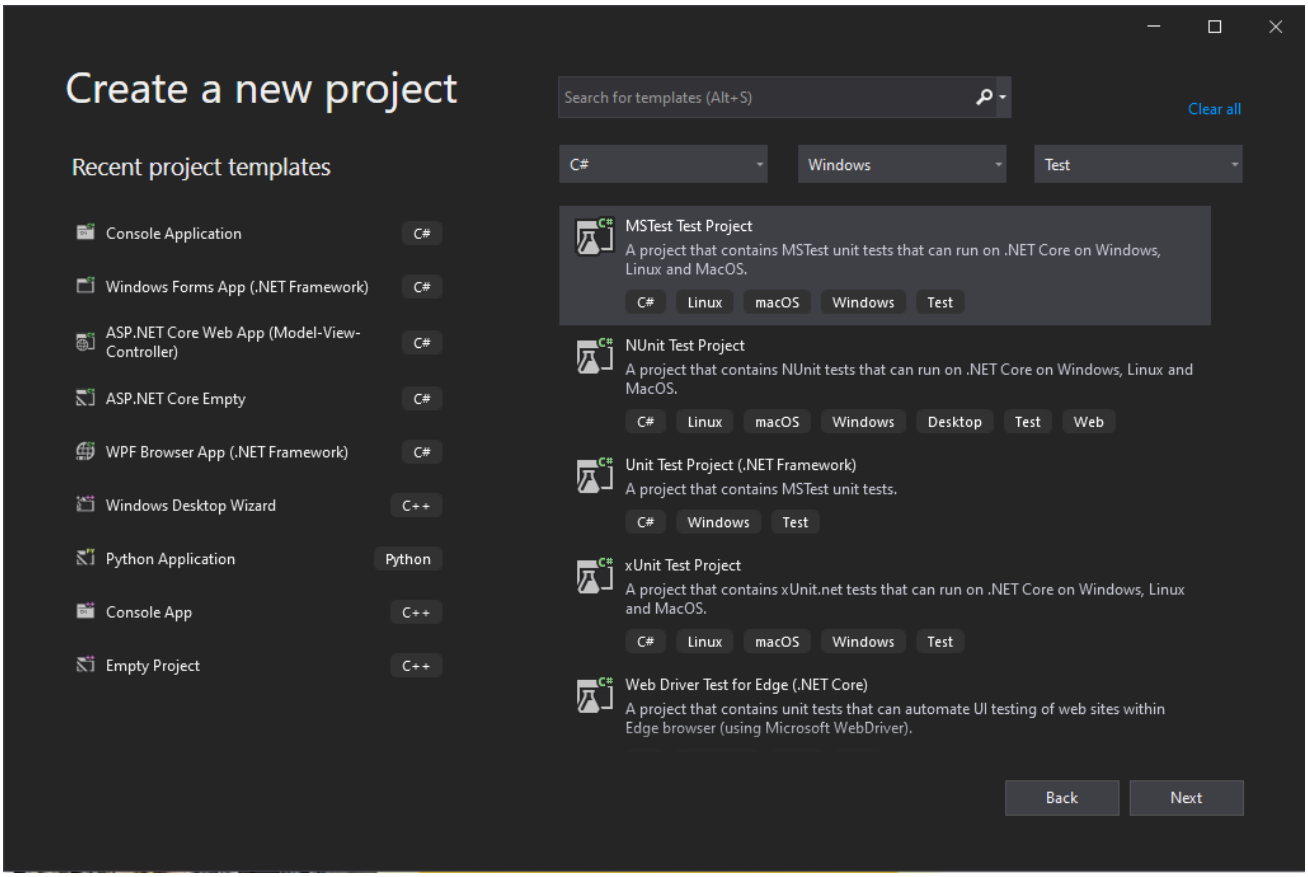


Рисунок 4.1 – Створення нового тесту в середовищі Microsoft Visual Studio 2019

Далі перейдемо до модульного тесту, щоб перевірити наше рішення ASP .NET Core Web API.

Оскільки наш зразок MobTel.API невеликий і може бути протестований за допомогою інтеграційного тестування, було виявлено, що найбільше зосередження на модульному тестуванні в проектах доменів та даних.

Ми обговорили, чому нам потрібно проводити модульні випробування, і що таке модульне тестування. Далі важливо зрозуміти, як правильно перевірити свій ASP.NET рішення для веб-API Core. У центрі уваги тестування API знаходяться дані. Тестування передбачуваних наборів даних дуже важливо. Ось чому не рекомендовано використовувати виробничі дані або інші дані, які можуть змінюватися з плином часу в залежності від знань і сприйняття. Нам потрібен

стабільний набір даних, щоб забезпечити виконання всіх модульних тестів і переконатися, що тести між сегментами коду однакові. Наприклад, для тестування додатку, потрібно отримати запит з ІДЕНТИФІКАТОРОМ 42. Для цього потрібно переконатися, що він дійсно існує, мав такі деталі, як назва телефону, і був пов'язаний з частиною даних про виконавця. Я також переконатися, що коли отримуємо серію телефонів з джерела даних, назва відповідає тесту одиниці, який написаний.

Багато людей в галузі використовують термін «імітовані дані» для позначення цього типу даних. Існує багато способів генерувати фіктивні дані для модульного тестування, і потрібно створити якомога більше «реальних» наборів даних. Чим точнішими будуть дані, які ви створюють для тесту, тим кращими будуть результати тестів. Загальна рекомендація переконатися, що дані не мають проблем з конфіденційністю і не містять особистих або конфіденційних даних компанії або клієнтів.

Щоб задовольнити нашу потребу в чистих і стабільних даних, створено окремий проект для інкапсуляції даних моделювання тестового проекту одиниці. Назвемо його Chinook.MockData (як можна бачити на прикладі). ПроектMobTel майже такий же, як і проект Chinook.Data. Всі вони мають однакову кількість баз даних, і всі вони підключені до одного інтерфейсу. Потрібно, щоб проект MockData зберігався в контейнері для ін'єкції залежностей (DI), щоб проект Chinook.Domain можна було використовувати так, ніби він підключений до джерела даних про виробництво. Ось чому потрібна ін'єкція залежностей. Це може допомогти перемикати проекти даних через конфігурацію без будь-яких змін коду.

4.2 Тестування інтеграції

Після того, як ми запустимо і протестуємо модульні тести для нашогододатку на ASP .NET Core Web API, розглянемо інший тип тесту.

Модульне тестування може підтвердити і задовольнити очікування внутрішніх компонентів рішення. Коли ми задоволені якістю внутрішнього тестування, нам потрібно виконувати тестування Application Program Interface (API) з кінця, яке прийнято називати інтеграційним тестуванням.

Інтеграційні тести повинні бути написані та запущені після завершення всіх компонентів, щоб наш API можна було перевірити правильними відповідями HTTP. Модульний тест перевіряє ізольовані сегменти коду, а тест інтеграції перевіряє загальну логіку кожного API на кінцевій точці HTTP. Тест буде слідувати за повним робочим процесом API, від контролера проекту API до менеджера проектів домену, і, нарешті, до бібліотеки проектів даних (повернення для відповіді).

4.3 Створення проекту тестування інтеграції

Щоб використовувати наявні знання тестування, функція тестування інтеграції ґрунтується на існуючій бібліотеці модульних тестів. В даному дипломнму проекті ми будемо використовувати бібліотеку тестів xUnit для створення тестів інтеграції.

Для цієї біблотеки Щоб використовувати xUnit, до проекту тестів (тобто UnitTestApp.Tests) необхідно додати через NuGet три пакети:

- Microsoft.NET.Test.Sdk;
- xunit - функціонал фреймворку тестування;
- xunit.runner.visualstudio, який забезпечує інтеграцію з Visual Studio.

Після того, як ми створили новий тестовий проект xUnit під назвою Chinook.IntegrationTest, нам потрібно додати відповідний пакет NuGet (диспетчер пакетів для .NET, який дозволяє програмістам створювати та використовувати різні бібліотеки .NET, а також надавати спільний доступ до них загалом. За допомогою клієнтських засобів NuGet надається можливість створювати та

використовувати бібліотеки як пакети). Додаємо пакет Microsoft.AspNetCore.TestHost до проекту MobTel.IntegrationTest. Цей пакет містить ресурси для запуску тестів інтеграції (Рисунок 4.2)

.

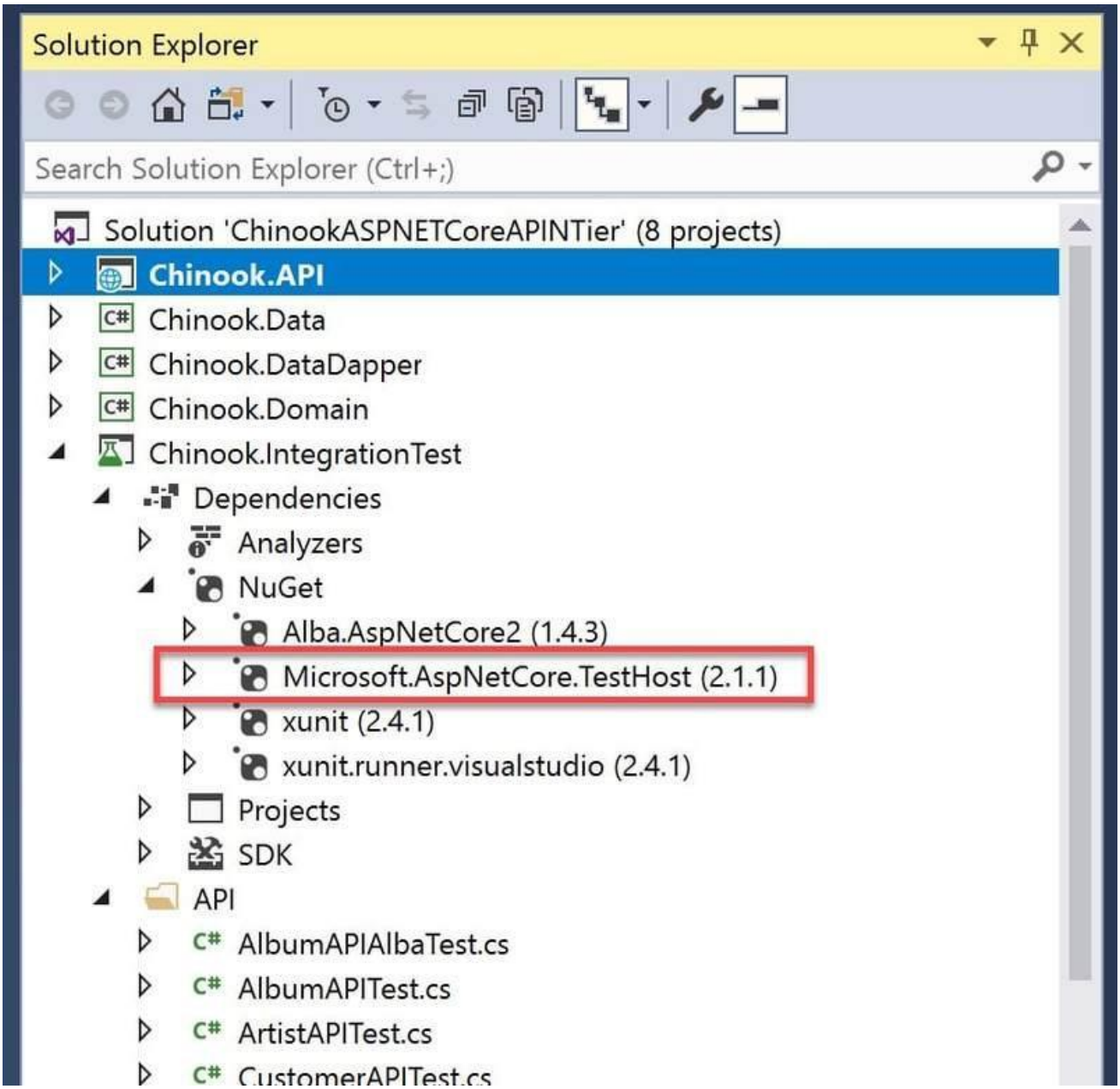


Рисунок 4.2 - Додавання пакета Microsoft.AspNetCore.TestHost NuGet

Далі створимо перший тест на інтеграцію, щоб протестувати API нашої інформаційної системи ззовні.

4.3 Створення тесту на інтеграцію

Щоб запустити всі зовнішні тести API в нашому рішенні, необхідно створити папку під назвою API, яка містить тести. Також потрібно створити новий тестовий клас для кожного типу сутності в домені API. Наш перший тест на інтеграцію буде охоплювати тип сутності телефону.

Створимо новий клас MobTelAPITest.cs в папці API та додамо до файлу наступні простори імен (Рисунок 4.3):

```
using Xunit;
using Chinook.API;
using Microsoft.AspNetCore.TestHost;
using Microsoft.AspNetCore.Hosting;
```

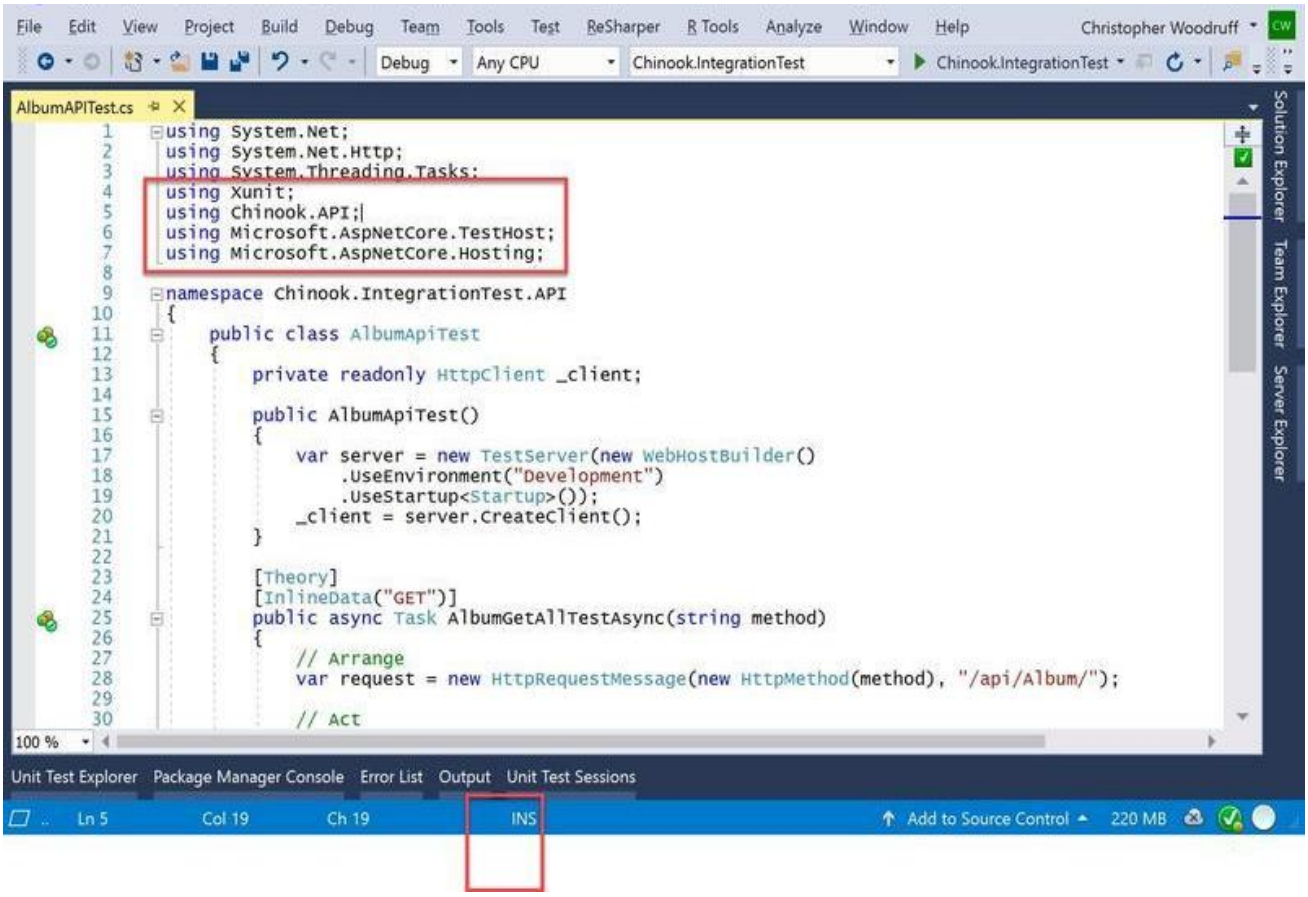


Рисунок 4.3 - Інструкція по застосуванню для інтеграційного тестування

Далі використовуємо TestServer для налаштування класу та HttpClient для запуску нашого тесту. Для цього, необхідна приватна змінна з назвою _client типу HttpClient, яка створюється з TestServer, ініціалізованого в конструкторі класу AlbumAPITest. TestServer - це оболонка для невеликого веб-сервера, створена на базі Chinook.API Startup і необхідного середовища розробки. В даному випадку ми використовую середовище розробки MS Visual Studio. Тепер у нас є веб-сервер, на якому запущено API, і клієнт, який розуміє, як викликати API в TestServer. Запишемо інтеграційний тестовий код (Рисунок 4.4.)

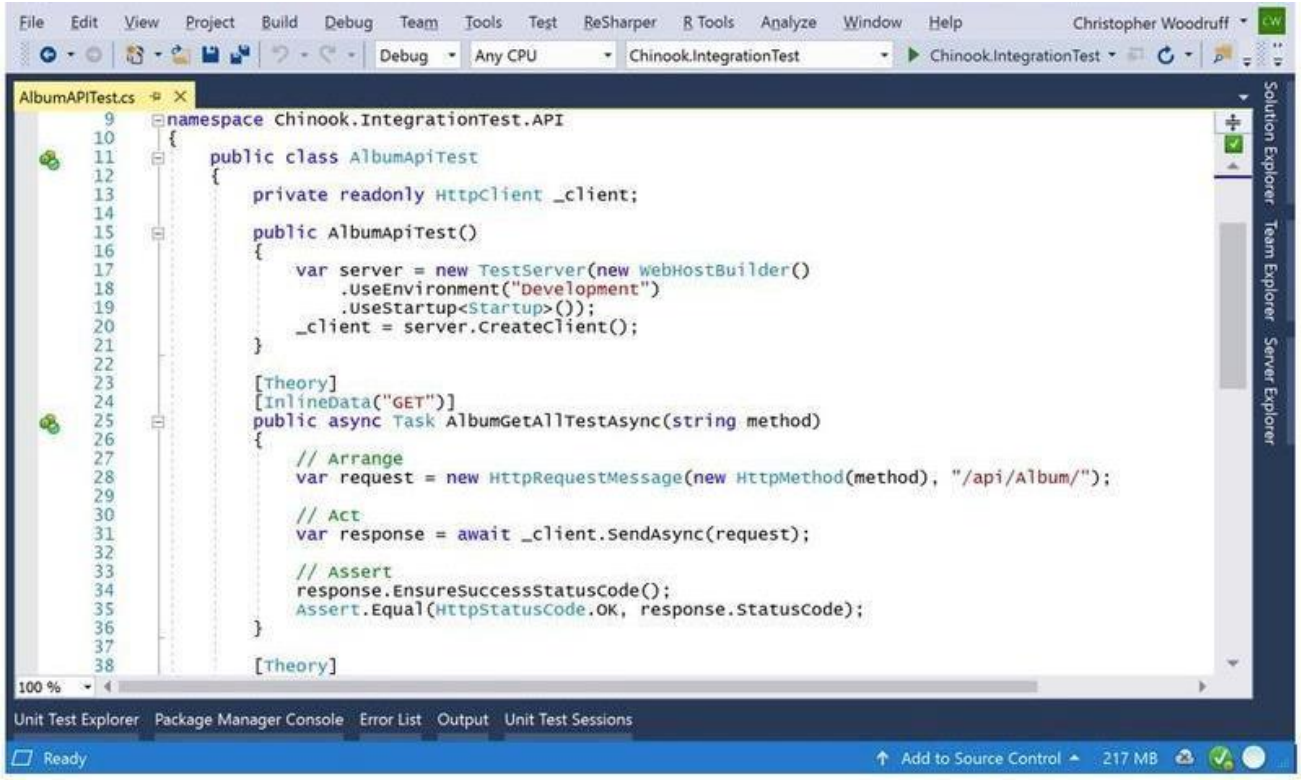


Рисунок 4.4 - на інтеграцію, отримання всіх деевайсів

Крім дизайнерського коду, на малюнку 4 також показаний код для першого інтеграційного тесту. Метод AlbumGetAllTestAsync перевірить і перевірить виклик, щоб отримати всі альбоми з API. Як і в тесті одиниці, обговорюваному раніше, логіка перевірки інтеграції також використовує «Організувати», «Діяти» та «Стверджувати». По-перше, ми створюємо об'єкт HttpRequestMessage, де HTTP

піддається змінній в анотації InlineData, а частина URI являє собою виклик для всіх альбомів ("/api/Album/"). Нарешті, ми _client перевіримо, чи відповідає відповідь HTTP нашим очікуванням, яких у цьому прикладі 200. На малюнку 4 я показую два способи перевірки викликів API. Ви можете використовувати будь-який з них, але я віддаю перевагу другому методу, оскільки він дозволяє мені використовувати той же шаблон для перевірки відповіді на конкретний код відповіді HTTP.

response.EnsureSuccessStatusCode();Assert.Equal(HttpStatusCode.OK, response.StatusCode);

Ми також можемо створювати інтеграційні тести, необхідні для перевірки конкретних ключів сутностей з API. Для цього типу тесту нам потрібно додати додаткові значення в анотацію InlineData, яка буде передана через параметр методу AlbumGetTestAsync. Наш новий тест буде використовувати ту ж логіку і ті ж ресурси, що і попередній тест, але ми передамо ключ сутності на стороні API URI для об'єкта HttpRequestMessage. Ви можете переглянути код на Рисунок 4.5.

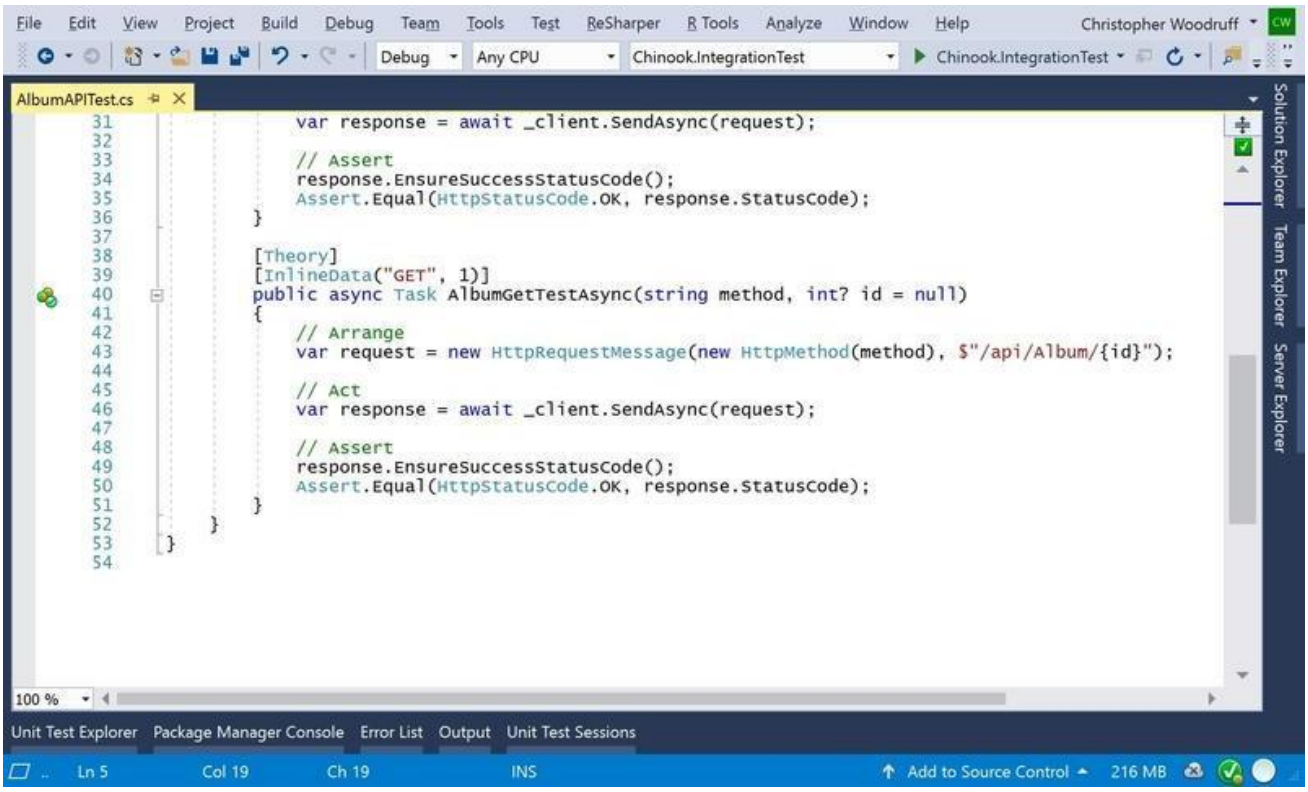


Рисунок 4.5 - Другий тест інтеграції

Після того, як ви створили всі інтеграційні тести для тестового API, вам потрібно запустити їх через Test Runner і переконатися, що всі вони проходять. Будь-які тести, які ви створюєте, також можуть бути запущені через процес безперервної інтеграції DevOps (CI), щоб ви могли протестувати свій API протягом усього процесу розробки та розгортання. Тепер ви повинні мати шлях виконання, щоб переконатися, що ваш API тестується та підтримується на етапах розробки, якості та розгортання, щоб користувачі ВАШОГО API могли безперебійно працювати (Рисунок 4.6).

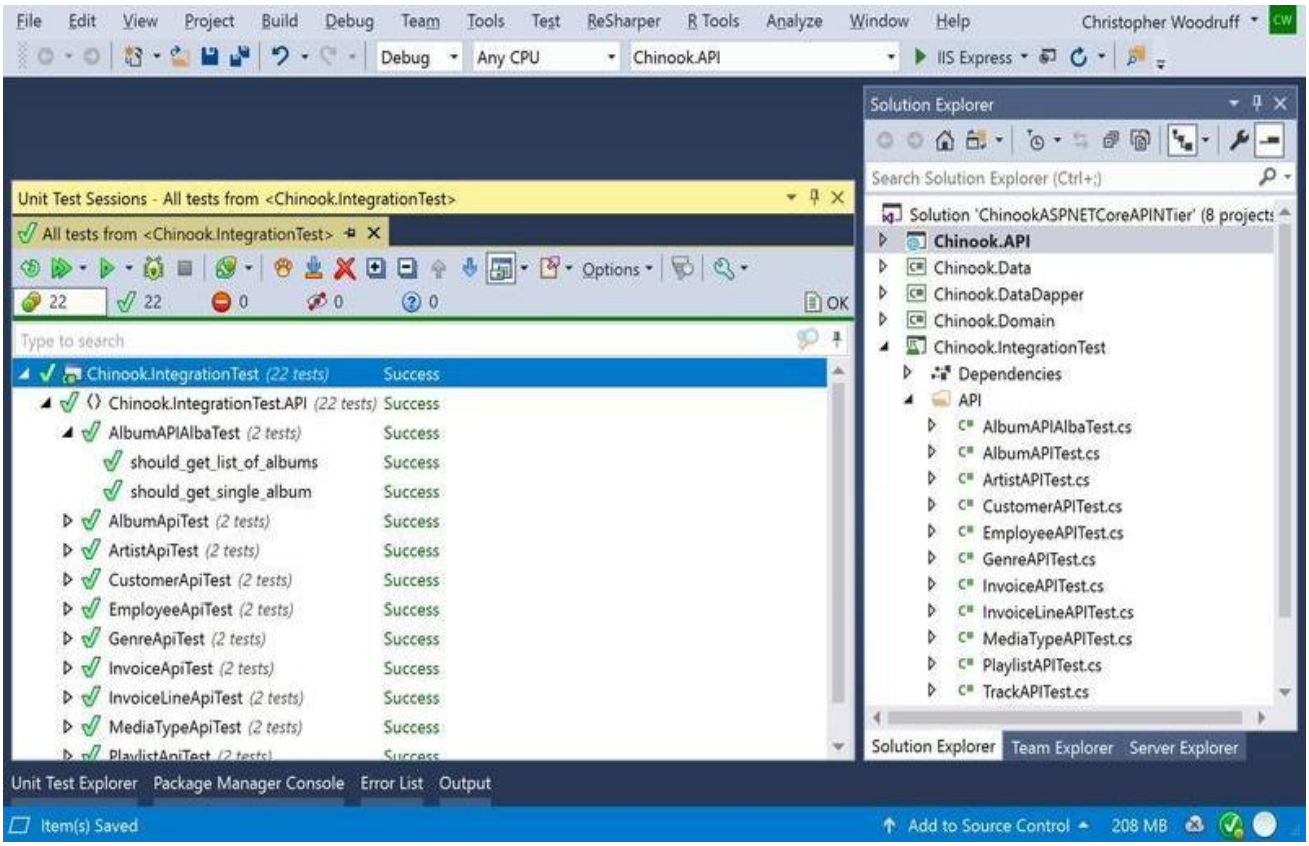


Рисунок 4.6 - Запуск інтеграційних тестів у Visual Studio 2019

Слід відмітити, що крім проведення тестування, можуть бути також виявлені певні складнощі в установці та розгортанні сервера додатків. Це може проявлятись, як в технічній причині так в питаннях, пов’язаних з налаштуванням самого веб-сайту (Рисунок 4.7).

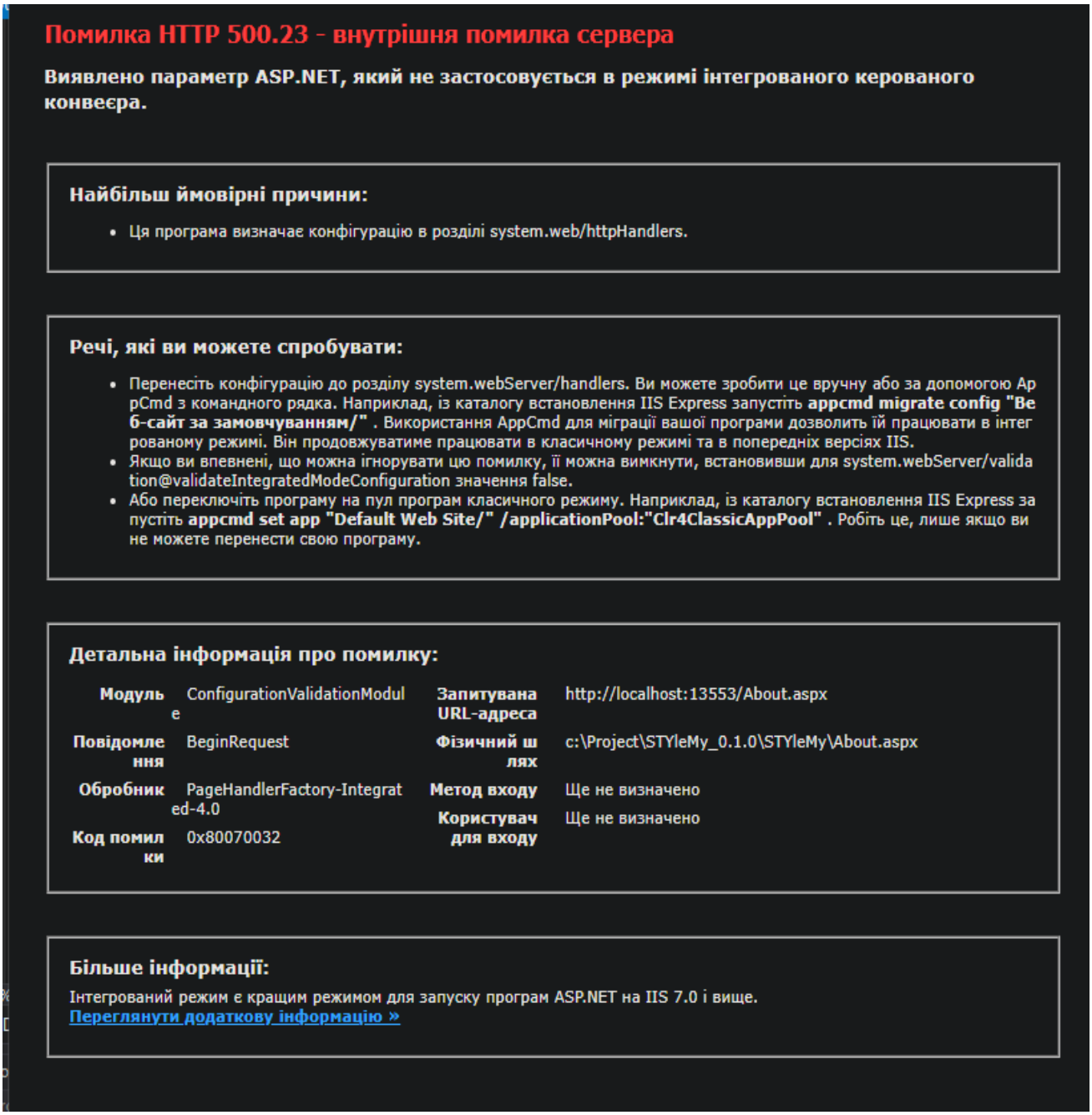


Рисунок 4.7 – Виявлення помилки при розгортанні сайту на сервері.

Загалом наявність добре розробленого плану випробувань, використання модульного тестування для внутрішнього тестування та використання інтеграційного тестування для перевірки викликів API так само важливо, як побудова архітектури під час розробки веб-API ASP.NET Core.

Під час проведення нами модульних та інтеграційних тестів було виявлено, що дане програмне забезпечення працює вірно та задовільняє усі вимоги та потреби до нашої інформаційної системи. Саме для цього нами було проведено

різні модульні тести для різних сторінок сайту та запитів, а також інтеграційні тести для визначення основних функцій програми.

У четвертому розділі здійснено вибір та обґрунтування методів тестування додатку. Розглянуто різні варіанти тестування та обґрунтовано при використанні ASP.NET Core 2.1 Web API використання інструменту xUnit. Проведено детальний процес тестування з допомогою інтегрованих засобів Microsoft Visual Studio за допомогою розроблених нами тестів модульного та ітераційного тестування програмного засобу.

ВИСНОВКИ

У ході дипломного проектування було розроблено інформаційну систему моніторингу діяльності салону з продажу мобільних телефонів, яка полегшує та спрощує роботу персоналу салону та надає більше інформації його клієнтам.

Зокрема, в першому розділі здійснений аналіз предметної області. В результаті аналізу визначено поняття самої системи, поняття моніторингу, його функцій та методологічних засад проведення. Здійснено аналіз останніх публікацій та наявного програмного забезпечення. Визначено задачі необхідні для реалізації проекту. Описано функціональне призначення та вимоги до розроблюваної інформаційної системи.

У другому розділі, в результаті проектування програмного забезпечення моніторингу діяльності салону з продажу мобільних телефонів, було розроблено архітектуру та функціональну структуру додатку. Побудовано дерево цілей системи. Розроблено структуру та функціональне призначення системи з визначенням основних модулів та їх функціональним призначенням. Спроектовано інфологічну модель, структур таблиць бази даних та побудовано саму базу даних. Далі здійснено вибір та обґрунтування мови та засобів розробки системи. Результатом проектування розділу було наведено структуру та функціональне призначення розроблювального додатку в середовищі Microsoft Visual Studio.

У третьому розділі наведено процес розробки програмних модулів інформаційної системи. Подано структуру основних запитів в системі. Деталізовано процес авторизації та розмежування прав доступу до даних. Наведено процес розробки інтерфейсу сайту інформаційної системи, з врахуванням вимог розробки інтерфейсу користувача, наведено технічні характеристики необхідні для серверної та клієнтської частини проектованої інформаційної системи.

У четвертому розділі здійснено вибір та обґрунтування методів тестування додатку. Розглянуто різні варіанти тестування та обґрунтовано при використанні ASP.NET Core 2.1 Web API використання інструменту xUnit. Проведено детальний процес тестування з допомогою інтегрованих засобів Microsoft Visual Studio за допомогою розроблених нами тестів модульного та ітераційного тестування програмного засобу.

У загальному система виконує основні задачі поставлені перед нею:

- зберігання і надання доступу до даних;
- ведення обліку товарів;
- формування бази співробітників підприємства;
- проводить аналіз роботи менеджерів;
- формує фінансовий та статистичний звіти про діяльність менеджера за будь-який період.

Отже, розроблена програма має велику кількість переваг у порівняні із аналогічними програмами, що робить її конкурентоспроможною.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Бедратюк Л. П. Дипломний проект : методичні вказівки щодо його виконання для студентів спеціальності 121 «Інженерія програмного забезпечення» / Л. П. Бедратюк, Г. І. Радельчук, Ю. В. Форкун, О. М. Яшина. – Хмельницький: ХНУ, 2020. – 77 с.
2. Форкун Ю. В. Курсове проектування. Методичні вказівки до виконання курсового проекту для студентів напряму підготовки “Програмна інженерія” з дисципліни «Архітектура програмного забезпечення систем» / Ю. В. Форкун – Хмельницький: ХНУ, 2016. –24 с.
3. Офіційна документація ASP.NET MVC 5 [Електронний ресурс] // Сайт Microsoft docs – Режим доступу: https://docs.microsoft.com/ru-ru/aspnet/mvc/overview/getting-started/introduction/getting-started (дата звернення 11.04.2022) – Назва з екрану.
4. Посібник по ASP.NET Core 5 [Електронний ресурс] // Сайт Metanit – Режим доступу: https://metanit.com/sharp/aspnet5/ (дата звернення 14.04.2022) – Назва з екрану.
5. Толстохатько В. А. Бази даних: проектування та використання для обліку нерухомогомайна : навч. посібник/ В. А. Толстохатько. О. Є. Поморцева,І. М. Патракеєв; Харк. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. –Х. : ХНУМГ, 2014. – 174 с.
6. Троелсен Ендрю. Мова програмування C# та платформа .Net і .Net Core, 8-е видання.: Пер. з англ / Ендрю Троелсен., 2018 – 1328с.
7. Client-Server Architecture for QA Engineers [Електронний ресурс] / Madnes – режим доступу до ресурсу: https://intellipaat.com/blog/what-is-client-serverarchitecture/ (дата звернення 23.03.2022)- Назва з екрана.
8. SQLite vs MySQL - Comparing 2 Popular Databases [Електронний ресурс] – Режим доступу до ресурсу: https://www.keycdn.com/support/sqlite-vs-mysql (дата звернення 29.04.2022) – Назва з екрану.

9. Zohreh Dehghani Champiri, Adeleh Asemi, and Salim Siti Salwah Binti. Meta-analysis of evaluation methods and metrics used in context-aware scholarly recommender systems. Knowledge and Information Systems, 2019. 61 (2):1147–1178p.

10. What is Visual Studio? [Електронний ресурс] – Режим доступу до ресурсу: www.incredibuild.com/integrations/visual-studio (дата звернення .05.2022) – Назва з екрану.

11. What is .NET? [Електронний ресурс] – Режим доступу до ресурсу: https://www.c-sharpcorner.com/article/what-is-net/ (дата звернення 18.05.2022) – Назва з екрану.

12. What Is Software Architecture & Software Security Design and How DoesIt Work? | Synopsys [Електронний ресурс] // Synopsys | EDA Tools, Semiconductor IP and Application Security Solutions. – Режим доступу:https://www.synopsys.com/glossary/what-is-software-architecture.html (дата звернення: 04.03.2022). – Назва з екрану.

ДОДАТОК А

(обов'язковий)

РОЗРОБКА ТЕХНІЧНОГО ЗАВДАННЯ

Введення

Робота виконується в рамках проекту розробки інформаційної системи моніторингу діяльності салону з продажу мобільних телефонів

Умовне позначення системи: Інформаційна система моніторингу діяльності салону з продажу мобільних телефонів.

2.4.1 Підстава для розробки

Підставою для розробки додатку є «Завдання на дипломне проектування», затверджене завідувачем ректором Хмельницького національного університету. Найменування розробки: Інформаційна система моніторингу діяльності салону з продажу мобільних телефонів.

2.4.2 Призначення розробки

Призначення роботи полягає у створенні простого і зрозумілого у користуванні програмного продукту для автоматизації роботи обліку та моніторингу салону з продажу мобільних телефонів, щоб підвищити рівень обслуговування, збільшити ефективність роботи та знизити витрати.

Користувачами програмного забезпечення є звичайні користувачі інтренет-магазинів та персонал салону мобільного зв’язку.

Експлуатаційне призначення, система може використовуватися на будь-якому браузері.

3 Вимоги до програми

3.1 Вимоги до функціональних характеристик

Перелік функціональних можливостей, які має надавати програма. Система повинна виконувати основні задачі поставлені перед нею:

- зберігання і надання доступу до даних;
- вести облік товарів;
- формувати базу співробітників підприємства;
- проводити аналіз роботи менеджерів
- формувати фінансовий та статистичний звіти про діяльність менеджера за будь-який період.

Створювана система повинна відповідати певним вимогам, основним положенням, принципам системного підходу, які є досить загальними твердженнями:

- принцип остаточної мети: створення системи моніторингу діяльності салону з продажу мобільних телефонів;
- принцип єдності: системи моніторингу діяльності салону з продажу мобільних телефонів повинна бути цілісною і функціонувати як єдина система, але одночасно повинна складатися з окремих компонентів, підсистем для: обробки і систематизації інформації, обслуговування клієнтів та ведення обліку продажу товарів, обліку товарів .
- принцип зв’язності: довільний елемент системи моніторингу діяльності салону з продажу мобільних телефонів повинен бути розглянутий сумісно з іншими елементами та зв’язками з ними.
- принцип модульності: в системі необхідно реалізувати декомпозицію на складові з різною функціональністю, кожна з яких має виконувати певні,

призначені їй дії: робота з базою даних, моніторинг роботи, створення звітності; необхідно розглянути всі модулі в сукупності та реалізувати зв'язки між ними.

- принцип ієрархії: система повинна мати ієрархічну будову, кожен функціональний елемент системи, в залежності від його важливості, повинен підпорядковуватися іншим елементам, які являються більш загальними і включають в себе необхідні функції.
- принцип функціональності: структура системи та її функції повинні розглядатися сумісно з пріоритетом функції над структурою; структура системи повинна бути побудована таким чином, щоб кожен елемент системи міг коректно працювати, не заважаючи роботі інших елементі системи.
- принцип розвитку: система повинна мати можливість для розвитку, враховуючи змінність системи, її розширення, заміну складових частин у майбутньому, при необхідності; систему необхідно реалізувати таким чином, щоб її розвиток не завдавав труднощів при її подальшій експлуатації, удосконаленні, заміні деяких її складових частин.
- принцип децентралізації: співвідношення між централізацією та децентралізацією, в управлінні системою, визначається призначенням та метою системи.
- принцип невизначеності: при побудові системи, необхідно взяти до уваги невизначеності та випадковості; потрібно врахувати можливі подальші шляхи розвитку діяльності салону з продажу мобільних телефонів у майбутньому, та врахувати їх у стратегію та тактику розвитку системи.

3.2 Вимоги до надійності

Інформаційна система моніторингу діяльності салону з продажу мобільних телефонів повиненна виконувати наступні вимоги до надійності:

- забезпечувати збереження інформації у випадку випадкового виключення інформаційної системи;
- передбачати блокування некоректних дій користувачів при роботі з інформаційною системою.

Інформаційна система зобов’язана пройти юніт-тести, системні тести та інтеграційні тести.

3.3 Умови експлуатації

Умови експлуатації інформаційної системи зобовязані відповідати санітарним і технічним нормам експлуатації персональних комп’ютерів, при температурі та відносній вологості навколишнього середовища, визначених для персональної обчислювальної техніки відповідно до державних стандартів.

Інформаційна система буде використовуватись на на персональних компютерах, смартфонах та планшетах.

3.4 Вимоги до складу та параметрів технічних засобів

Оскільки, розроблена система моніторингу діяльності салону з продажу мобільних телефонів працює на основі клієнт-серверної технології, окремо розглянемо вимоги до технічних засобів клієнта та сервера.

Програмне забезпечення сервера:

- операційна система Windows;
- повинен бути встановлений та налаштований сервер IIS;
- повинен бути встановлений Framework 2.0 або вище;

- MS SQL Server.

Апаратне забезпечення сервера:

- підключення до мережі Internet;
- обсяг оперативної пам'яті від 512 MB;
- процесор не нижче Intel Pentium;
- вільне місце на жорсткому диску для зберігання веб-додатку, що складає приблизно 20 MB;
- вільне місце на жорсткому диску для зберігання бази даних приблизно 20Мб;

Програмне забезпечення клієнта:

- операційна система Windows ,Vista,7;
- наявність веб-браузера, для повного функціонування має бути включена можливість виконання JavaScript скриптів;

Апаратне забезпечення клієнта:

- підключення до мережі Internet;
- обсяг оперативної пам'яті від 512 MB.

Дані вимоги є мінімальними вимогами, які забезпечують нормальне функціонування розробленої системи.

3.5 Вимоги до програмної документації

Попередній склад програмної документації встановлений відповідно до ДСТУ 3008–95 та Єдиної системи програмної документації. Наведемо перелік програмних документів та їх зміст:

- структурна схема інформаційної системи;
- програмний текст інформаційної системи – запис програмного коду зі всіма необхідними поясненнями та коментарями;

- опис програмного коду – відомості про логічну і фізичну модель системи, відомості про функціональні можливості інформаційної системи;
- програмний код та методика тестувань – вимоги, які підлягають перевірці при тестуванні інформаційної системи, а також порядок та методи контролю якості;
- технічне завдання – даний документ;
- записка пояснення – схема алгоритму;
- загальний опис алгоритму та функціональних властивостей інформаційної системи та обґрунтування ухвалених техніко-економічних рішень.

3.6 Стадії та етапи розробки

Стадії та етапи розробки проекту інформаційної системи моніторингу діяльності салону з продажу мобільних телефонів наведено в таблиці 2.

Таблиця 2 – Стадії та етапи розробки проекту

Стадії розробки	Етапи робіт	Зміст робіт
Технічне завдання 02.01.22 – 31.01.22	Обґрунтування необхідності розробки програми	Коротка характеристика програмного забезпечення по організації тестування; підстава і призначення розробки; вимоги до програмного комплексу і документація; стадії і етапи розробки програми; порядок контролю і приймання
Ескізний проект 01.02.22 – 14.02.22	Розробка ескізного проекту	Попередня розробка структури вхідних і вихідних даних; уточнення середовища програмування; розробка і опис загальної алгоритмічної структури системи, що буде розроблюватися
Технічний проект 15.02.22 – 28.02.22	Розробка технічного проекту	Уточнення структури вхідних і вихідних даних, визначення форми їх уявлення; розробка докладного алгоритму; розробка структури програми; остаточне визначення конфігурації

		технічних засобів; розробка заходів щодо упровадження програмного комплексу
Робочий проект 01.03.22– 10.05.22	Розробка програми	Реалізація програмного комплексу по тестуванню знань студентів; відладка; розробка методики випробувань; проведення попередніх випробувань (тестування); коректування програмного забезпечення; розробка документації
Розробка програмної документації 11.05.22 – 20.05.22	Розробка документації до програмного забезпечення	Розробка необхідної документації, передбаченої технічним завданням
Тестування системи 21.05.22 – 30.05.22	Проведення тестування програмного забезпечення	Розробка методики тестування; проведення основних тестів; коректування програмного забезпечення
Впровадження	Підготовка і передача програми.	Випуск додатку на платформі PlayMarket; внесення коректувань в програмне забезпечення і документацію

3.8 Порядок контролю та приймання

Контроль здійснюється кінцевими користувачами системи, підключеними на етапі тестування системи.

Після закінчення розробки системи повинні бути проведене тестування на захист від некоректного введення.

ДОДАТОК Б

(обов'язковий)

ФРАГМЕНТИ КОДУ ПРОГРАМНОЇ СИСТЕМИ

Код програми:

```
        Site.Master
<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site.Master.cs"
Inherits="STYleMy.SiteMaster" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <title></title>
    <link href="~/Styles/st.css" rel="stylesheet" type="text/css" />

</head>
<body>
    <form runat="server">
    </div>--%>
        <div id="main">
        <header>
      <div id="logo">
        <div id="logo_text">
          <!-- class="logo_colour", allows you to change the colour of the text -->
          <h1>ІС моніторингу діяльності<a href="Default.aspx"> <span class="logo_colour">салону
</span></a>з продажу мобільних телефонів</h1>
           <asp:contentplaceholder id="HeadContent" runat="server" />
        </div>
      </div>
      <nav>
            <div id="menu_container">
          <ul class="sf-menu" id="nav">
            <li><a href="/Default.aspx">Головна </a></li>

                <li><a href="#"> База даних </a>
              <ul>
                <li><a href="/MemberPages/Виробники.aspx">Виробники</a></li>
                <li><a href="/MemberPages/Товари.aspx">Товари</a></li>
                <li><a href="/MemberPages/Менеджери.aspx">Менеджери</a></li>
                <li><a href="/MemberPages/ОбсягПродажів.aspx">Обсяг продажів</a></li>
                <li><a href="/MemberPages/ПродажТоварів.aspx">Продаж товарів</a></li>
              </ul>

            </li>
            <li><a href="#"> Моніторинг роботи </a>
              <ul>
                <li><a href="/Середні показники по виробнику.aspx">Середні показники</a></li>
                <li><a href="/Сума продажу менеджерів.aspx"> Сума продажу</a></li>
                 <li><a href="/month.aspx">Місячний продаж</a></li>
              </ul>
            </li>

            <li><a href="#"> Звіти </a>
              <ul>
                <li><a href="/ВсіПродажі.aspx">Продажі</a></li>
                <li><a href="/Пошук по даті.aspx">Продажі за датою</a></li>
                <li><a href="/Пошук по прізвищу.aspx">Продажі менеджера</a></li>

              </ul>
            </li>

            <li><a href="/Help.aspx"> Довідка </a></li>
```

```
            <li><a href="/About.aspx"> Про програму <pre>              </pre></a></li>
           </ul>
         </div>


      </nav>
    </header>
                <asp:contentplaceholder id="MainContent" runat="server" />
         <footer>
      <p>КР &copy; ПІс-11-1 | <a href="http://www.css3templates.co.uk">Лавров В. В.</a></p>
    </footer>
    </div>
    <p>
         </p>
    <!-- javascript at the bottom for fast page loading -->
    <script type="text/javascript" src="/js/jquery.js"></script>
    <script type="text/javascript" src="/js/jquery.easing-sooper.js"></script>
    <script type="text/javascript" src="/js/jquery.sooperfish.js"></script>
    <script type="text/javascript">
        $(document).ready(function () {
            $('ul.sf-menu').sooperfish();
        });
    </script>
    </form>
</body>
</html>
```

ST.CSS

```
@font-face
{
    font-family: Jenna Sue;
    src: url('../fonts/JennaSue-webfont.eot');
    src: local("Jenna Sue"), url('../fonts/JennaSue-webfont.ttf');
    src: url('../fonts/JennaSue-webfont.eot');
}

@font-face
{

    font-family: StudioScriptCTT;
    src: url('../fonts/studio_script_ctt.ttf');
    src: local("StudioScriptCTT"), url('../fonts/StudioScriptCTT');
}




@font-face
{
    font-family: News Cycle;
    src: url('../fonts/NewsCycle-Regular.eot');
    src: local("News Cycle"), url('../fonts/NewsCycle-Regular.ttf');
}

html
{
    height: 100%;
}

*
{
    margin: 0;
    padding: 0;
}

/* tell the browser to render HTML 5 elements as block */
article, aside, figure, footer, header, hgroup, nav, section
```

```
{
    display: block;
}

body
{
    font: normal .80em arial, sans-serif;
    background: #000 url(../images/back.jpg) no-repeat center fixed;
    color: #FFF;
}

p
{
    padding: 0 0 20px 0;
    line-height: 1.7em;
}

img
{
    border-style: none;
    border-color: inherit;
    border-width: 0;
    margin-right: 0;
}

h1, h2, h3, h4, h5, h6
{
    color: #FFF;
    letter-spacing: 0em;
    padding: 0 0 5px 0;
}

h1, h2, h3
{
    font: normal 140% arial, sans-serif;
    margin: 0 0 15px 0;
    padding: 15px 0 5px 0;
    color: #FFF;
}

h2
{
    font-size: 160%;
    padding: 9px 0 5px 0;
    color: #D054FF;
}

h3
{
    font-size: 140%;
    padding: 5px 0 0 0;
}

h4, h6
{
    color: #D054FF;
    padding: 0 0 5px 0;
    font: normal 200% 'Jenna Sue' , 'StudioScriptCTT', sans-serif;
}

h5, h6
{
    font: italic 95% arial, sans-serif;
    letter-spacing: normal;
    padding: 0 0 15px 0;
}
```

```
h5
{
    color: #FFF;
}

a, a:hover
{
    outline: none;
    text-decoration: underline;
    color: #00E6FF;
}

a:hover
{
    text-decoration: none;
}

blockquote
{
    margin: 20px 0;
    padding: 10px 20px 0 20px;
    border: 1px solid #E5E5DB;
    background: #FFF;
}

ul
{
    margin: 2px 0 22px 17px;
}

ul li
{
    list-style-type: circle;
    margin: 0 0 6px 0;
    padding: 0 0 4px 5px;
    line-height: 1.5em;
}

ol
{
    margin: 8px 0 22px 20px;
}

ol li
{
    margin: 0 0 11px 0;
}

.left
{
    float: left;
    width: auto;
    margin-right: 10px;
}

.right
{
    float: right;
    width: auto;
    margin-left: 10px;
}

.center
{
    display: block;
```

```
    text-align: center;
    margin: 20px auto;
}

#main, nav, #container, #logo, #site_content, footer
{
    margin-left: auto;
    margin-right: auto;
}

#main
{
    margin: 50px auto;
    width: 1000px;
    border-radius: 13px 13px 13px 13px;
    -moz-border-radius: 13px 13px 13px 13px;
    -webkit-border: 13px 13px 13px 13px;
    background: transparent url(../images/main.png);
    padding-bottom: 30px;
}

header
{
    background: transparent;
    height: 179px;
}

#logo
{
    width: 930px;
    height: 179px;
    background: transparent;
    color: #888;
    padding: 0;
}

#logo h1, #logo h2
{
    font: normal 250% 'News Cycle' , Consolas, sans-serif;
    border-bottom: 0;
    text-transform: none;
    margin: 0;
}

#logo_text h1, #logo_text h1 a, #logo_text h1 a:hover
{
    padding: 20px 0 0 0;
    color: #FFF;
    text-decoration: none;
}

#logo_text h1 a .logo_colour
{
    color: #00E6FF;
}

#logo_text a:hover .logo_colour
{
    color: #FFF;
}

#logo_text h2
{
    font: 170% 'News Cycle' , Consolas, sans-serif;
    padding: 0 0 0 0;
}
```

```
nav
{
    height: 44px;
    width: 930px;
    margin: -34px auto 0 auto;
    -webkit-border-radius: 13px 13px 13px 13px;
    -moz-border-radius: 13px 13px 13px 13px;
    border-radius: 13px 13px 13px 13px;
    color: #FFF;
    border: solid 1px #003B56;
    background: #002A3E;
}

#menu_container
{
    width: 930px;
    margin: 0 auto 0 auto;
}

#site_content
{
    width: 930px;
    overflow: hidden;
    margin: 0px auto 0 auto;
    padding: 15px 0 15px 0;
}

#sidebar_container
{
    float: right;
    width: 224px;
    padding: 20px 0 0 0;
}

.sidebar
{
    float: right;
    width: 200px;
    margin: 0 0 17px 0;
    background: #22003C;
    border: 1px solid #4F0077;
    -webkit-border-radius: 13px 13px 13px 13px;
    -moz-border-radius: 13px 13px 13px 13px;
    border-radius: 13px 13px 13px 13px;
    padding: 0 15px 5px 13px;
    opacity: 0.8;
}

.sidebar h3, .content h1
{
    padding: 5px 15px 20px 0;
    font: 200% 'News Cycle' , Consolas, sans-serif;
    text-shadow: #000 1px 1px;
    margin: 0 1px;
    color: #FFF;
}

.sidebar h3
{
    padding: 9px 15px 14px 0;
    color: #FFF;
    font: 170% 'News Cycle' , Consolas, sans-serif;
}

.sidebar ul
```

```
{
    margin: 0 0 15px 0;
}

.sidebar li a, .sidebar li a:hover
{
    color: #DDD;
    text-decoration: none;
}

.sidebar li a:hover
{
    text-decoration: underline;
}

.sidebar_item, .content_item
{
    padding: 15px 0;
}

.paperclip
{
    float: left;
    position: relative;
    z-index: 0;
    vertical-align: middle;
    margin: -15px 0 -60px -40px;
}

.content
{
    text-align: left;
    width: 630px;
    margin: 0 0 15px 0;
    float: left;
    font-size: 120%;
    padding: 14px 0 0 0;
}

.content ul
{
    margin: 2px 0 22px 0px;
}

.content ul li, .sidebar ul li
{
    list-style-type: none;
    background: url(../images/bullet.png) no-repeat;
    margin: 0 0 0 0;
    padding: 0 0 4px 28px;
    line-height: 1.5em;
}

footer
{
    width: 930px;
    font: 190% 'Jenna Sue' , 'StudioScriptCTT', sans-serif;
    height: 30px;
    padding: 5px 0 20px 0;
    text-align: center;
    -webkit-border-radius: 13px 13px 13px 13px;
    -moz-border-radius: 13px 13px 13px 13px;
    border-radius: 13px 13px 13px 13px;
    color: #FFF;
    border: solid 1px #003B56;
    background: #002A3E;
```

```
}

footer p
{
    line-height: 1.7em;
    padding: 0 0 10px 0;
}

footer a
{
    color: #FFF;
    text-decoration: none;
}

footer a:hover
{
    color: #FFF;
    text-shadow: none;
    text-decoration: underline;
}

.form_settings
{
    margin: 15px 0 0 0;
}

.form_settings p
{
    padding: 0 0 4px 0;
}

.form_settings span
{
    float: left;
    width: 200px;
    text-align: left;
}

.form_settings input, .form_settings textarea
{
    padding: 5px;
    width: 299px;
    font: 100% arial;
    border: 1px solid #C6E7F0;
    background: #EFF8FB;
    color: #47433F;
    border-radius: 7px 7px 7px 7px;
    -moz-border-radius: 7px 7px 7px 7px;
    -webkit-border: 7px 7px 7px 7px;
}

.form_settings .submit
{
    font: 150% 'Jenna Sue' , 'StudioScriptCTT', sans-serif;
    border: 0;
    width: 99px;
    margin: 0 0 0 212px;
    height: 33px;
    padding: 0 0 3px 0;
    cursor: pointer;
    background: #1EC9F4;
    -webkit-border-radius: 9px 9px 9px 9px;
    -moz-border-radius: 9px 9px 9px 9px;
    border-radius: 9px 9px 9px 9px;
    -webkit-box-shadow: 0 1px 2px rgba(0,0,0,.2);
    -moz-box-shadow: 0 1px 2px rgba(0,0,0,.2);
```

```
    box-shadow: 0 1px 2px rgba(0,0,0,.2);
    color: #FFF;
    border: solid 1px #0D8AA9;
    background: #46C4DD;
    background: -webkit-gradient(linear, left top, left bottom, from(#63CFDF), to(#17B2D9));
    background: -moz-linear-gradient(top,  #63CFDF,  #17B2D9);
    text-shadow: 1px 1px #178497;
}

.form_settings textarea, .form_settings select
{
    font: 100% arial;
    width: 299px;
}

.form_settings select
{
    width: 310px;
}

.form_settings .checkbox
{
    margin: 4px 0;
    padding: 0;
    width: 14px;
    border: 0;
    background: none;
}

.separator
{
    width: 100%;
    height: 0;
    border-top: 1px solid #D9D5CF;
    border-bottom: 1px solid #FFF;
    margin:




table
{
    margin: 10px 0 30px 0;
}

 30px 0;
}

table tr th, table tr td
{
    background: #70D4E6;
    color: #FFF;
    padding: 7px 4px;
    text-align: left;
}

table tr td
{
    background: #D3F2F7;
    color: #47433F;
    border-top: 1px solid #FFF;
}


.style_TableSSS {
width: 400px;
```

```
font-size: 10px;
}



/**
Stylesheet for SooperFish by www.SooperThemes.com
Author: Jurriaan Roelofs
**/

/* Configuration of menu width */
html body ul.sf-menu ul, html body ul.sf-menu ul li
{
    width: 240px;
}

html body ul.sf-menu ul ul
{
    margin: 0 0 0 240px;
}

/* Framework for proper showing/hiding/positioning */
ul.sf-menu, ul.sf-menu *
{
    margin: 0;
    padding: 0;
}

ul.sf-menu
{
    display: block;
    position: relative;
}

ul.sf-menu li
{
    display: block;
    list-style: none;
    float: left;
    position: relative;
}

ul.sf-menu li:hover
{
    visibility: inherit; /* fixes IE7 'sticky bug' */
}

ul.sf-menu a
{
    display: block;
    position: relative;
}

ul.sf-menu ul
{
    position: absolute;
    left: 0;
    width: 150px;
    top: auto;
    left: -999999px;
}

ul.sf-menu ul a
{
    zoom: 1; /* IE6/7 fix */
```

```
}

ul.sf-menu ul li
{
    float: left; /* Must always be floated otherwise there will be a rogue 1px margin-bottom in
IE6/7 */
    width: 150px;
}

ul.sf-menu ul ul
{
    top: 0;
    margin: 0 0 0 150px;
}

ul.sf-menu li:hover ul, ul.sf-menu li:focus ul, ul.sf-menu li.sf-hover ul, ul.sf-menu ul li:hover
ul, ul.sf-menu ul li:focus ul, ul.sf-menu ul li.sf-hover ul, ul.sf-menu ul ul li:hover ul, ul.sf-
menu ul ul li:focus ul, ul.sf-menu ul ul li.sf-hover ul, ul.sf-menu ul ul ul li:hover ul, ul.sf-
menu ul ul ul li:focus ul, ul.sf-menu ul ul ul li.sf-hover ul
{
    left: auto;
}

ul.sf-menu li:hover ul ul, ul.sf-menu li:focus ul ul, ul.sf-menu li.sf-hover ul ul, ul.sf-menu ul
li:hover ul ul, ul.sf-menu ul li:focus ul ul, ul.sf-menu ul li.sf-hover ul ul, ul.sf-menu ul ul
li:hover ul ul, ul.sf-menu ul ul li:focus ul ul, ul.sf-menu ul ul li.sf-hover ul ul, ul.sf-menu ul
ul ul li:hover ul ul, ul.sf-menu ul ul ul li:focus ul ul, ul.sf-menu ul ul ul li.sf-hover ul ul
{
    left: -999999px;
}

/* autoArrows CSS */
span.sf-arrow
{
    width: 7px;
    height: 7px;
    position: absolute;
    top: 20px;
    right: 5px;
    display: block;
    background: url(../images/arrows-white.png) no-repeat 0 0;
    overflow: hidden; /* making sure IE6 doesn't overflow and expand the box */
    font-size: 1px;
}

ul ul span.sf-arrow
{
    right: 10;
    top: 24px;
    background-position: 0 100%;
}

/* Theming the menu */
ul#nav
{
    float: right;
}

ul#nav ul
{
    background: #0DBBD5;
    margin-top: -5px;
    padding-bottom: 15px;
}

ul#nav li a
```

```
{
    padding: 8px 20px 6px 20px;
    font: 190% 'Jenna Sue' , 'StudioScriptCTT', sans-serif;
    text-decoration: none;
    color: #FFF;
    margin-right: 2px;
}

ul#nav li a:hover, ul#nav li a:focus
{
    color: #D054FF;
    text-shadow: none;
}

ul#nav ul li a
{
    padding: 13px 20px 6px 20px;
}

ul#nav ul li a:hover, ul#nav ul li a:focus
{
    color: #000;
    text-shadow: none;
}

<%@ Page Title="Домашняя страница" Language="C#" MasterPageFile="~/Site.master"
AutoEventWireup="true"
    CodeBehind="Default.aspx.cs" Inherits="STYleMy._Default" %>

<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">
    <h2>  Головна сторінка  </h2>
   <h5 align="right">
        <asp:LoginView ID="LoginView1" runat="server">
           <AnonymousTemplate>
               <span id="result_box" class="" lang="uk"><span class="hps">Ви не</span>
               <span
    class="hps">ввійшли</span> <span class="hps">в</span> <span class="hps">систему</span><span
    class="">.</span> <span class="hps">Клацніть посилання</span> <span
class="hps">входу</span><span
    class="">,</span> <span class="hps">щоб</span> <span class="hps">увійти </span></span>
                <asp:HyperLink ID="HyperLink2" runat="server"
NavigateUrl="~/Register.aspx">Реєстрація</asp:HyperLink>
           </AnonymousTemplate>
           <LoggedInTemplate>
               <span ID="result_box0" class="short_text" lang="uk">><span class="hps">Ви</span>
               <span class="hps">увійшли</span> <span class="hps">в</span> <span class="hps">
               систему</span><span>.</span> <span class="hps">Ласкаво просимо</span><span
                   class="">, <asp:LoginName ID="LoginName1" runat="server" />
  </span>
               <asp:HyperLink ID="HyperLink3" runat="server"
                   NavigateUrl="~/MemberPages/ChangePassword.aspx">Зміна пароля</asp:HyperLink>
               </span>
                
           </LoggedInTemplate>
       </asp:LoginView>
       <asp:LoginStatus ID="LoginStatus1" runat="server" LoginText="Вхід"
           LogoutText="Вихід" />
       </h5>
</asp:Content>



<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">
    <div id="site_content">
        <div id="sidebar_container">
            <div class="sidebar">
```

```
                <h3>
                    Останні новини</h3>
                <h4>
                    Новий сайт запущений</h4>
                <h5>
                    11
                    жовтня 2012</h5>
                <p>
                    По пропозиціям що-до зміни сайту зверніться за посиланням<br />
                    <a href="#">Пропозиції</a></p>
            </div>
            <div class="sidebar">
                <h3>
                   Корисні посилання</h3>
                <ul>
                    <li><a href="https://www.google.com.ua/">
                        <asp:Image ID="Image1" runat="server"
                            ImageUrl="~/images/google.png" Height="25px" Width="25px" />
                        </a></li>
                    <li><a href="https://twitter.com/">
                        <asp:Image ID="Image2" runat="server" Height="25px"
                            ImageUrl="~/images/Twitter.png" Width="25px" /></a></li>
                    <li><a href="http://mail.ru/">
                        <asp:Image ID="Image3" runat="server" Height="25px"
                            ImageUrl="~/images/mailru.png" /></a></li>
                    <li><a href="http://uk.wikipedia.org">
                        <asp:Image ID="Image4" runat="server" Height="25px"
                            ImageUrl="~/images/wikipedia.png" Width="25px" /></a></li>
                </ul>
            </div>
        </div>
        <div class="content">
            <h1 align="center">
                    Ласкаво просимо </h1>
            <p>
                Сайт призначений для автоматизації роботи діяльності салону з продажу мобільних
                телефонів.</p>
            <p>
                За допомогою сайту можна здійснити моніторинг роботи, та пошук необхідної
                інформації </p>
            <p>
                 </p>
            <p>
                 </p>
        </div>
    </div>
</asp:Content>

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
    CodeBehind="Виробники.aspx.cs" Inherits="STYleMy.WebForm1" %>

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
    <h2>   Виробники</h2>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
    <asp:Label ID="Label1" runat="server" Text="Оберіть дію" Font-Bold="True" Font-Italic="True"
        Font-Names="Arimo" Font-Size="Large"></asp:Label>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$
ConnectionStrings:ConnectionString %>"
        SelectCommand="SELECT [КодВиробника], [НазваВтробника] FROM [Виробники]"
        DeleteCommand="DELETE FROM Виробники WHERE (КодВиробника = @Kvur3)"
        InsertCommand="INSERT INTO Виробники(НазваВтробника) VALUES (@nvur1)"
        UpdateCommand="UPDATE Виробники SET НазваВтробника = @nvur2 WHERE (КодВиробника =
@kvur2)">
        <DeleteParameters>
            <asp:ControlParameter ControlID="DropDownList3" Name="Kvur3"
```

```
                PropertyName="SelectedValue" />
        </DeleteParameters>
        <InsertParameters>
            <asp:ControlParameter ControlID="TextBox1" Name="nvur1" PropertyName="Text" />
        </InsertParameters>
        <UpdateParameters>
            <asp:ControlParameter ControlID="TextBox2" Name="nvur2" PropertyName="Text" />
            <asp:ControlParameter ControlID="DropDownList2" Name="kvur2"
                PropertyName="SelectedValue" />
        </UpdateParameters>
    </asp:SqlDataSource>
    <br />
    <asp:DropDownList ID="DropDownList1" runat="server"
OnSelectedIndexChanged="DropDownList1_SelectedIndexChanged"
        Height="25px" Width="103px" AutoPostBack="True">
        <asp:ListItem>Перегляд</asp:ListItem>
        <asp:ListItem Value="Додати">Додати</asp:ListItem>
        <asp:ListItem Value="Редагувати">Редагувати</asp:ListItem>
        <asp:ListItem Value="Видалити">Видалити</asp:ListItem>
    </asp:DropDownList>
      
    <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Виконати"
Visible="False" />
    <p>
        <asp:MultiView ID="MultiView1" runat="server" ViewStateMode="Enabled" ActiveViewIndex="0">
            <asp:View ID="View1" runat="server">
                <h2>
                    Виробники</h2>
                <br />
                <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
DataKeyNames="КодВиробника"
                    DataSourceID="SqlDataSource1" AllowSorting="True"
                    ForeColor="Black">
                    <Columns>
                        <asp:BoundField DataField="КодВиробника" HeaderText="КодВиробника"
InsertVisible="False"
                            ReadOnly="True" SortExpression="КодВиробника" />
                        <asp:BoundField DataField="НазваВтробника" HeaderText="НазваВтробника"
SortExpression="НазваВтробника" />
                    </Columns>
                </asp:GridView>
            </asp:View>
            <asp:View ID="View2" runat="server">
                <h2>
                    Додати нового виробника</h2>
                <asp:GridView ID="GridView2" runat="server" AutoGenerateColumns="False"
DataKeyNames="КодВиробника"
                    DataSourceID="SqlDataSource1" AllowSorting="True"
                    ForeColor="Black">
                    <Columns>
                        <asp:BoundField DataField="КодВиробника" HeaderText="КодВиробника"
InsertVisible="False"
                            ReadOnly="True" SortExpression="КодВиробника" />
                        <asp:BoundField DataField="НазваВтробника" HeaderText="НазваВтробника"
SortExpression="НазваВтробника" />
                    </Columns>
                </asp:GridView>
                <br />
                 
                <asp:Label ID="Label2" runat="server" Text="Назва  виробника"></asp:Label>
                <br />
                <asp:TextBox ID="TextBox1" runat="server" Width="122px"></asp:TextBox>
                  
                <asp:Button ID="Button2" runat="server" OnClick="Button2_Click" Text="Додати"
Width="59px" />
            </asp:View>
```

```
            <asp:View ID="View3" runat="server">
                <h2>
                    Редагувати назву виробника виробника</h2>
                <asp:GridView ID="GridView3" runat="server" AutoGenerateColumns="False"
DataKeyNames="КодВиробника"
                    DataSourceID="SqlDataSource1" AllowSorting="True"
                    ForeColor="Black">
                    <Columns>
                        <asp:BoundField DataField="КодВиробника" HeaderText="КодВиробника"
InsertVisible="False"
                            ReadOnly="True" SortExpression="КодВиробника" />
                        <asp:BoundField DataField="НазваВтробника" HeaderText="НазваВтробника"
SortExpression="НазваВтробника" />
                    </Columns>
                </asp:GridView>
                <br />
                <asp:Label ID="Label4" runat="server" Text="Видберіть виробника дня
редагування"></asp:Label>
                <br />
                <asp:DropDownList ID="DropDownList2" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource1"
                    DataTextField="НазваВтробника" DataValueField="КодВиробника"
OnSelectedIndexChanged="DropDownList2_SelectedIndexChanged">
                </asp:DropDownList>
                 
                <asp:Label ID="Label3" runat="server" Text="Label" Visible="False"></asp:Label>
                <br />
                <br />
                <asp:Label ID="Label5" runat="server" Text="Нове значення виробника"></asp:Label>
                <br />
                <asp:TextBox ID="TextBox2" runat="server" Width="135px"></asp:TextBox>
                  
                <asp:Button ID="Button3" runat="server" OnClick="Button3_Click" Text="Змінити" />
            </asp:View>
            <asp:View ID="View4" runat="server">
                <h2>
                    Видалити виробника</h2>
                <asp:GridView ID="GridView4" runat="server" AutoGenerateColumns="False"
DataKeyNames="КодВиробника"
                    DataSourceID="SqlDataSource1" AllowSorting="True"
                    ForeColor="Black">
                    <Columns>
                        <asp:BoundField DataField="КодВиробника" HeaderText="КодВиробника"
InsertVisible="False"
                            ReadOnly="True" SortExpression="КодВиробника" />
                        <asp:BoundField DataField="НазваВтробника" HeaderText="НазваВтробника"
SortExpression="НазваВтробника" />
                    </Columns>
                </asp:GridView>
                <br />
                <asp:Label ID="Label6" runat="server" Text="Вилучити виробника"></asp:Label>
                <br />
                <asp:DropDownList ID="DropDownList3" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource1"
                    DataTextField="НазваВтробника" DataValueField="КодВиробника">
                </asp:DropDownList>
                  
                <asp:Button ID="Button4" runat="server" OnClick="Button4_Click" Text="Вилучити" />
                <br />
                <br />
            </asp:View>
        </asp:MultiView>
    </p>
</asp:Content>

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
```

```
    CodeBehind="Товари.aspx.cs" Inherits="STYleMy.Товари" %>

<%@ Register Assembly="System.Web.DataVisualization, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"
    Namespace="System.Web.UI.DataVisualization.Charting" TagPrefix="asp" %>
<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
  <h2>  Товари</h2>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
    <asp:Label ID="Label1" runat="server" Font-Bold="True" Font-Italic="True" Font-Names="Arimo"
        Font-Size="Large" Text="Оберіть дію"></asp:Label>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$
ConnectionStrings:ConnectionString %>"
        SelectCommand="SELECT [КодТовару], [НазваТовару], [КодВиробника], [Тип], [Ціна],
[НаСкладі] FROM [Товари]"
        InsertCommand="INSERT INTO Товари(НазваТовару, КодВиробника, Тип, Ціна, НаСкладі) VALUES
(@nt, @vkd, @tp, @cn, @sk)"
        DeleteCommand="DELETE FROM Товари WHERE (КодТовару = @ktvr)" UpdateCommand="UPDATE Товари
SET НазваТовару = @nzt, КодВиробника = @rvur, Тип = @tup, Ціна = @cin, НаСкладі = @skl WHERE
(КодТовару = @ktov)">
        <DeleteParameters>
            <asp:ControlParameter ControlID="DropDownList7" Name="ktvr"
PropertyName="SelectedValue" />
        </DeleteParameters>
        <InsertParameters>
            <asp:ControlParameter ControlID="TextBox1" Name="nt" PropertyName="Text" />
            <asp:ControlParameter ControlID="DropDownList2" Name="vkd"
PropertyName="SelectedValue" />
            <asp:ControlParameter ControlID="DropDownList3" Name="tp" PropertyName="SelectedValue"
/>
            <asp:ControlParameter ControlID="TextBox2" Name="cn" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox3" Name="sk" PropertyName="Text" />
        </InsertParameters>
        <UpdateParameters>
            <asp:ControlParameter ControlID="TextBox4" Name="nzt" PropertyName="Text" />
            <asp:ControlParameter ControlID="DropDownList5" Name="rvur"
PropertyName="SelectedValue" />
            <asp:ControlParameter ControlID="DropDownList6" Name="tup"
PropertyName="SelectedValue" />
            <asp:ControlParameter ControlID="TextBox5" Name="cin" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox6" Name="skl" PropertyName="Text" />
            <asp:ControlParameter ControlID="DropDownList4" Name="ktov"
PropertyName="SelectedValue" />
        </UpdateParameters>
    </asp:SqlDataSource>
    <asp:SqlDataSource ID="SqlDataSource2" runat="server" ConnectionString="<%$
ConnectionStrings:ConnectionString %>"
        SelectCommand="SELECT [НазваВтробника], [КодВиробника] FROM
[Виробники]"></asp:SqlDataSource>
    <br />
    <asp:DropDownList ID="DropDownList1" runat="server" Height="25px" Width="103px"
AutoPostBack="True"
        OnSelectedIndexChanged="DropDownList1_SelectedIndexChanged">
        <asp:ListItem>Перегляд</asp:ListItem>
        <asp:ListItem Value="Додати">Додати</asp:ListItem>
        <asp:ListItem Value="Редагувати">Редагувати</asp:ListItem>
        <asp:ListItem Value="Видалити">Видалити</asp:ListItem>
    </asp:DropDownList>
    <br />
    <asp:MultiView ID="MultiView1" runat="server" ActiveViewIndex="0" ViewStateMode="Enabled">
        <asp:View ID="View1" runat="server">
            <h2 align="center">
                Товари</h2>
            <div align="center">
                <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
DataKeyNames="КодТовару"
```

```
                    DataSourceID="SqlDataSource1" AllowSorting="True" ForeColor="Black">
                    <Columns>
                        <asp:BoundField DataField="КодТовару" HeaderText="КодТовару"
InsertVisible="False"
                            ReadOnly="True" SortExpression="КодТовару" />
                        <asp:BoundField DataField="НазваТовару" HeaderText="НазваТовару"
SortExpression="НазваТовару" />
                        <asp:BoundField DataField="КодВиробника" HeaderText="КодВиробника"
SortExpression="КодВиробника" />
                        <asp:BoundField DataField="Тип" HeaderText="Тип" SortExpression="Тип" />
                        <asp:BoundField DataField="Ціна" HeaderText="Ціна" SortExpression="Ціна"
/>
                        <asp:BoundField DataField="НаСкладі" HeaderText="НаСкладі"
SortExpression="НаСкладі" />
                    </Columns>
                </asp:GridView>
            </div>
            <br />
        </asp:View>
        <asp:View ID="View2" runat="server">
            <h2 align="center">
                Додати товар</h2>
            <div align="center">
                <div align="left">

                &n
bsp;               &nbs
p;                
      
                    <asp:Label ID="Label2" runat="server" Text="Назва"></asp:Label>

                &n
bsp;        
                    <asp:Label ID="Label3" runat="server" Text="Виробник"></asp:Label>
                               
                    <asp:Label ID="Label4" runat="server" Text="Тип"></asp:Label>

                &n
bsp; 
                    <asp:Label ID="Label5" runat="server" Text="Ціна"></asp:Label>
                            
                    <asp:Label ID="Label6" runat="server" Text="На складі"></asp:Label></div>
                    <br />
                    <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
                    <asp:DropDownList ID="DropDownList2" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource2"
                        DataTextField="НазваВтробника" DataValueField="КодВиробника">
                    </asp:DropDownList>
                    <asp:DropDownList ID="DropDownList3" runat="server" AutoPostBack="True">
                        <asp:ListItem>Моноблок</asp:ListItem>
                        <asp:ListItem>QWERTY</asp:ListItem>
                        <asp:ListItem>Розкладний</asp:ListItem>
                        <asp:ListItem>Слайдер</asp:ListItem>
                    </asp:DropDownList>
                    <asp:TextBox ID="TextBox2" runat="server" Width="63px"></asp:TextBox>
                    <asp:TextBox ID="TextBox3" runat="server" Width="57px"></asp:TextBox>
                     
                    <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Додати"
/>

                    <br />

                <br />
                <asp:GridView ID="GridView2" runat="server" AllowSorting="True"
AutoGenerateColumns="False"
                    DataKeyNames="КодТовару" DataSourceID="SqlDataSource1" ForeColor="Black">
```

```
                    <Columns>
                        <asp:BoundField DataField="КодТовару" HeaderText="КодТовару"
InsertVisible="False"
                            ReadOnly="True" SortExpression="КодТовару" />
                        <asp:BoundField DataField="НазваТовару" HeaderText="НазваТовару"
SortExpression="НазваТовару" />
                        <asp:BoundField DataField="КодВиробника" HeaderText="КодВиробника"
SortExpression="КодВиробника" />
                        <asp:BoundField DataField="Тип" HeaderText="Тип" SortExpression="Тип" />
                        <asp:BoundField DataField="Ціна" HeaderText="Ціна" SortExpression="Ціна"
/>
                        <asp:BoundField DataField="НаСкладі" HeaderText="НаСкладі"
SortExpression="НаСкладі" />
                    </Columns>
                </asp:GridView>
            </div>
            <br />
            <br />
            <br />
        </asp:View>
        <asp:View ID="View3" runat="server">
            <h2 align="center">
                Редагувати дані товару</h2>
            <br />
   <div align="center">         <asp:Label ID="Label7" runat="server" Text="Виберіть
товар"></asp:Label>
            <br />
            <asp:DropDownList ID="DropDownList4" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource1"
                DataTextField="НазваТовару" DataValueField="КодТовару">
            </asp:DropDownList>
            <br />
            <br />
          <div align="left">
                &n
bsp;               &nbs
p;                
      
              <asp:Label ID="Label8" runat="server" Text="Назва"></asp:Label>

                &n
bsp;        
            <asp:Label ID="Label9" runat="server" Text="Виробник"></asp:Label>
                       
            <asp:Label ID="Label10" runat="server" Text="Тип"></asp:Label>

                &n
bsp; 
            <asp:Label ID="Label11" runat="server" Text="Ціна"></asp:Label>
                    
            <asp:Label ID="Label12" runat="server" Text="На складі"></asp:Label>   </div>
            <br />
            <asp:TextBox ID="TextBox4" runat="server"></asp:TextBox>
            <asp:DropDownList ID="DropDownList5" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource2"
                DataTextField="НазваВтробника" DataValueField="КодВиробника">
            </asp:DropDownList>
            <asp:DropDownList ID="DropDownList6" runat="server" AutoPostBack="True">
                <asp:ListItem>Моноблок</asp:ListItem>
                <asp:ListItem>QWERTY</asp:ListItem>
                <asp:ListItem>Розкладний</asp:ListItem>
                <asp:ListItem>Слайдер</asp:ListItem>
            </asp:DropDownList>
            <asp:TextBox ID="TextBox5" runat="server" Width="63px"></asp:TextBox>
            <asp:TextBox ID="TextBox6" runat="server" Width="57px"></asp:TextBox>
             
```

```
            <asp:Button ID="Button2" runat="server" Text="Змінити" OnClick="Button2_Click" />
            </div>
            <br />
          <div align="center">  <asp:GridView ID="GridView4" runat="server" AllowSorting="True"
AutoGenerateColumns="False"
                DataKeyNames="КодТовару" DataSourceID="SqlDataSource1" ForeColor="Black">
                <Columns>
                    <asp:BoundField DataField="КодТовару" HeaderText="КодТовару"
InsertVisible="False"
                        ReadOnly="True" SortExpression="КодТовару" />
                    <asp:BoundField DataField="НазваТовару" HeaderText="НазваТовару"
SortExpression="НазваТовару" />
                    <asp:BoundField DataField="КодВиробника" HeaderText="КодВиробника"
SortExpression="КодВиробника" />
                    <asp:BoundField DataField="Тип" HeaderText="Тип" SortExpression="Тип" />
                    <asp:BoundField DataField="Ціна" HeaderText="Ціна" SortExpression="Ціна" />
                    <asp:BoundField DataField="НаСкладі" HeaderText="НаСкладі"
SortExpression="НаСкладі" />
                </Columns>
            </asp:GridView></div>
            <br />

            <br />
            <br />
            <br />
        </asp:View>
        <asp:View ID="View4" runat="server">
            <h2 align="center">
                Видалити товар</h2>
                <div align="center">
                  <div align="left">

                &n
bsp;               &nbs
p;                
                &n
bsp;               &nbs
p;                
         <asp:Label ID="Label13" runat="server" Text="Виберіть
товар"></asp:Label>   </div>
                    <br />
                    <asp:DropDownList ID="DropDownList7" runat="server" AutoPostBack="True"
                        DataSourceID="SqlDataSource1" DataTextField="НазваТовару"
                        DataValueField="КодТовару">
                    </asp:DropDownList>
                     
                    <asp:Button ID="Button3" runat="server" OnClick="Button3_Click"
                        Text="Видалити" />
            </div>
                <br />
            <div align="center">   <asp:GridView ID="GridView5" runat="server" AllowSorting="True"
AutoGenerateColumns="False"
                DataKeyNames="КодТовару" DataSourceID="SqlDataSource1" ForeColor="Black">
                <Columns>
                    <asp:BoundField DataField="КодТовару" HeaderText="КодТовару"
InsertVisible="False"
                        ReadOnly="True" SortExpression="КодТовару" />
                    <asp:BoundField DataField="НазваТовару" HeaderText="НазваТовару"
SortExpression="НазваТовару" />
                    <asp:BoundField DataField="КодВиробника" HeaderText="КодВиробника"
SortExpression="КодВиробника" />
                    <asp:BoundField DataField="Тип" HeaderText="Тип" SortExpression="Тип" />
                    <asp:BoundField DataField="Ціна" HeaderText="Ціна" SortExpression="Ціна" />
                    <asp:BoundField DataField="НаСкладі" HeaderText="НаСкладі"
SortExpression="НаСкладі" />
                </Columns>
```

```
            </asp:GridView>
            <br />
            </div>
            <br />
            <br />
        </asp:View>
        <br />
    </asp:MultiView>
</asp:Content>

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
    CodeBehind="Менеджери.aspx.cs" Inherits="STYleMy.Менеджери" %>

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
 <h2>   Менеджери</h2>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
    <asp:Label ID="Label1" runat="server" Font-Bold="True" Font-Italic="True" Font-Names="Arimo"
        Font-Size="Large" Text="Оберіть дію"></asp:Label>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$
ConnectionStrings:ConnectionString %>"


        SelectCommand="SELECT [КодМенеджера], [Прізвище], [Імя], [ПоБатькові], [Адреса],
[РікПрийому], [Оклад] FROM [Менеджеди]"
        DeleteCommand="DELETE FROM Менеджеди WHERE (КодМенеджера = @km)"
        InsertCommand="INSERT INTO Менеджеди(Прізвище, Імя, ПоБатькові, Адреса, РікПрийому, Оклад)
VALUES (@pzv, @ima, @pbt, @adr, @rkp, @okl)"
        UpdateCommand="UPDATE Менеджеди SET Прізвище = @pr, Імя = @im, ПоБатькові = @pb, Адреса =
@ad, РікПрийому = @rp, Оклад = @ok WHERE (КодМенеджера = @km)">
        <DeleteParameters>
            <asp:ControlParameter ControlID="DropDownList3" Name="km"
                PropertyName="SelectedValue" />
        </DeleteParameters>
        <InsertParameters>
            <asp:ControlParameter ControlID="TextBox1" Name="pzv" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox2" Name="ima" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox3" Name="pbt" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox4" Name="adr" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox5" Name="rkp" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox6" Name="okl" PropertyName="Text" />
        </InsertParameters>
        <UpdateParameters>
            <asp:ControlParameter ControlID="TextBox8" Name="pr" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox9" Name="im" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox10" Name="pb" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox11" Name="ad" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox12" Name="rp" PropertyName="Text" />
            <asp:ControlParameter ControlID="TextBox13" Name="ok" PropertyName="Text" />
            <asp:ControlParameter ControlID="DropDownList2" Name="km"
                PropertyName="SelectedValue" />
        </UpdateParameters>
    </asp:SqlDataSource>
    <br />
    <asp:DropDownList ID="DropDownList1" runat="server" Height="25px" Width="103px"
AutoPostBack="True"
        OnSelectedIndexChanged="DropDownList1_SelectedIndexChanged">
        <asp:ListItem>Перегляд</asp:ListItem>
        <asp:ListItem Value="Додати">Додати</asp:ListItem>
        <asp:ListItem Value="Редагувати">Редагувати</asp:ListItem>
        <asp:ListItem Value="Видалити">Видалити</asp:ListItem>
    </asp:DropDownList>
    <br />
    <p>
        <asp:MultiView ID="MultiView1" runat="server" ActiveViewIndex="0" ViewStateMode="Enabled">
            <asp:View ID="View1" runat="server">
```

```
                <h2>
                    Менеджери</h2>
                <br />
                <asp:GridView ID="GridView1" runat="server" AllowSorting="True"
                    AutoGenerateColumns="False" DataKeyNames="КодМенеджера"
                    DataSourceID="SqlDataSource1" ForeColor="Black">
                    <Columns>
                        <asp:BoundField DataField="КодМенеджера" HeaderText="КодМенеджера"
InsertVisible="False"
                            ReadOnly="True" SortExpression="КодМенеджера" />
                        <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
SortExpression="Прізвище" />
                        <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя" />
                        <asp:BoundField DataField="ПоБатькові" HeaderText="ПоБатькові"
SortExpression="ПоБатькові" />
                        <asp:BoundField DataField="Адреса" HeaderText="Адреса"
SortExpression="Адреса" />
                        <asp:BoundField DataField="РікПрийому" HeaderText="РікПрийому"
SortExpression="РікПрийому" />
                        <asp:BoundField DataField="Оклад" HeaderText="Оклад"
SortExpression="Оклад" />
                    </Columns>
                </asp:GridView>
            </asp:View>
            <asp:View ID="View2" runat="server">
                <h2>
                    Додати нового менеджера</h2>
                <asp:GridView ID="GridView2" runat="server" AllowSorting="True"
DataSourceID="SqlDataSource1"
                    ForeColor="Black" AutoGenerateColumns="False"
                    DataKeyNames="КодМенеджера">
                    <Columns>
                        <asp:BoundField DataField="КодМенеджера" HeaderText="КодМенеджера"
                            InsertVisible="False" ReadOnly="True" SortExpression="КодМенеджера" />
                        <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
                            SortExpression="Прізвище" />
                        <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя" />
                        <asp:BoundField DataField="ПоБатькові" HeaderText="ПоБатькові"
                            SortExpression="ПоБатькові" />
                        <asp:BoundField DataField="Адреса" HeaderText="Адреса"
                            SortExpression="Адреса" />
                        <asp:BoundField DataField="РікПрийому" HeaderText="РікПрийому"
                            SortExpression="РікПрийому" />
                        <asp:BoundField DataField="Оклад" HeaderText="Оклад"
SortExpression="Оклад" />
                    </Columns>
                </asp:GridView>
                <br />
                 <asp:Label ID="Label2" runat="server" Text="Прізвище"></asp:Label>
                            
                <asp:Label ID="Label3" runat="server" Text="Ім'я"></asp:Label>

               &n
bsp;           
                <asp:Label ID="Label4" runat="server" Text="По-батькові"></asp:Label>

               &n
bsp;
                <asp:Label ID="Label5" runat="server" Text="Адреса"></asp:Label>

               &n
bsp;              &nbs
p;           <asp:Label
                    ID="Label6" runat="server" Text="Рік прийому"></asp:Label>
                    <asp:Label ID="Label7" runat="server"
Text="Оклад"></asp:Label>
```

```
                <br />
                <asp:TextBox ID="TextBox1" runat="server" Width="111px"></asp:TextBox>
                <asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>
                <asp:TextBox ID="TextBox3" runat="server"></asp:TextBox>
                <asp:TextBox ID="TextBox4" runat="server" Width="220px"></asp:TextBox>
                <asp:TextBox ID="TextBox5" runat="server" Width="87px"></asp:TextBox>
                <asp:TextBox ID="TextBox6" runat="server" Width="81px"></asp:TextBox>
                 <asp:Button ID="Button1" runat="server" OnClick="Button1_Click"
Text="Додати" />
                <br />
                <br />
            </asp:View>
            <asp:View ID="View3" runat="server">
                <h2>
                    Редагувати дані менеджера</h2>
                <asp:GridView ID="GridView3" runat="server" AllowSorting="True"
DataSourceID="SqlDataSource1"
                    ForeColor="Black" AutoGenerateColumns="False"
                    DataKeyNames="КодМенеджера">
                    <Columns>
                        <asp:BoundField DataField="КодМенеджера" HeaderText="КодМенеджера"
                            InsertVisible="False" ReadOnly="True" SortExpression="КодМенеджера" />
                        <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
                            SortExpression="Прізвище" />
                        <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя" />
                        <asp:BoundField DataField="ПоБатькові" HeaderText="ПоБатькові"
                            SortExpression="ПоБатькові" />
                        <asp:BoundField DataField="Адреса" HeaderText="Адреса"
                            SortExpression="Адреса" />
                        <asp:BoundField DataField="РікПрийому" HeaderText="РікПрийому"
                            SortExpression="РікПрийому" />
                        <asp:BoundField DataField="Оклад" HeaderText="Оклад"
SortExpression="Оклад" />
                    </Columns>
                </asp:GridView>
                <br />
                <asp:Label ID="Label8" runat="server" Text="Оберіть менеджера"></asp:Label>
                <br />
                <asp:DropDownList ID="DropDownList2" runat="server" AutoPostBack="True"
                    DataSourceID="SqlDataSource1" DataTextField="Прізвище"
                    DataValueField="КодМенеджера" Height="16px"
                    onselectedindexchanged="DropDownList2_SelectedIndexChanged" Width="123px">
                </asp:DropDownList>
                 
                <asp:Label ID="Label9" runat="server" Text="КодМенеджера: "></asp:Label>
                <asp:TextBox ID="TextBox7" runat="server" Enabled="False"
Width="24px">3</asp:TextBox>
                <br />
                <br />
                 <asp:Label ID="Label10" runat="server" Text="Прізвище"></asp:Label>
                            
                <asp:Label ID="Label11" runat="server" Text="Імя"></asp:Label>

               &n
bsp;           
                <asp:Label ID="Label12" runat="server" Text="По-батькові"></asp:Label>

               &n
bsp;
                <asp:Label ID="Label13" runat="server" Text="Адреса"></asp:Label>

               &n
bsp;              &n
bsp;          
                <asp:Label ID="Label14" runat="server" Text="Рік прийому"></asp:Label>
                    
```

```
                <asp:Label ID="Label15" runat="server" Text="Оклад"></asp:Label>
                <br />
                <asp:TextBox ID="TextBox8" runat="server" Width="111px"></asp:TextBox>
                <asp:TextBox ID="TextBox9" runat="server"></asp:TextBox>
                <asp:TextBox ID="TextBox10" runat="server"></asp:TextBox>
                <asp:TextBox ID="TextBox11" runat="server" Width="220px"></asp:TextBox>
                <asp:TextBox ID="TextBox12" runat="server" Width="87px"></asp:TextBox>
                <asp:TextBox ID="TextBox13" runat="server" Width="81px"></asp:TextBox>
                 <asp:Button ID="Button2" runat="server" Text="Змінити"
                    onclick="Button2_Click2" />
                <br />
            </asp:View>
            <asp:View ID="View4" runat="server">
                <h2>
                    Видалити менеджера</h2>
                <asp:GridView ID="GridView4" runat="server" AllowSorting="True"
                    AutoGenerateColumns="False" DataKeyNames="КодМенеджера"
                    DataSourceID="SqlDataSource1" ForeColor="Black">
                    <Columns>
                        <asp:BoundField DataField="КодМенеджера" HeaderText="КодМенеджера"
                            InsertVisible="False" ReadOnly="True" SortExpression="КодМенеджера" />
                        <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
                            SortExpression="Прізвище" />
                        <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя" />
                        <asp:BoundField DataField="ПоБатькові" HeaderText="ПоБатькові"
                            SortExpression="ПоБатькові" />
                        <asp:BoundField DataField="Адреса" HeaderText="Адреса"
                            SortExpression="Адреса" />
                        <asp:BoundField DataField="РікПрийому" HeaderText="РікПрийому"
                            SortExpression="РікПрийому" />
                        <asp:BoundField DataField="Оклад" HeaderText="Оклад"
SortExpression="Оклад" />
                    </Columns>
                </asp:GridView>
                <br />
                <asp:Label ID="Label16" runat="server" Text="Оберіть менеджера"></asp:Label>
                <br />
                <asp:DropDownList ID="DropDownList3" runat="server" AutoPostBack="True"
                    DataSourceID="SqlDataSource1" DataTextField="Прізвище"
                    DataValueField="КодМенеджера"
                    onselectedindexchanged="DropDownList3_SelectedIndexChanged">
                </asp:DropDownList>
 
                <asp:Label ID="Label17" runat="server" Text="КодМенеджера: "></asp:Label>
                <asp:TextBox ID="TextBox14" runat="server" Enabled="False"
Width="24px">3</asp:TextBox>
                <br />
                <br />
                <asp:Button ID="Button3" runat="server" onclick="Button3_Click"
                    Text="Видалити" />
                <br />
            </asp:View>
            <br />
        </asp:MultiView>
    </p>
</asp:Content>

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
    CodeBehind="ОбсягПродажів.aspx.cs" Inherits="STYleMy.ОбсягПродажів" %>

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
  <h2>  Обсяг продажів</h2>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
    <p>
```

```
        <asp:Label ID="Label1" runat="server" Font-Bold="True" Font-Italic="True" Font-
Names="Arimo"
            Font-Size="Large" Text="Оберіть дію"></asp:Label>
        <br />
        <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$
ConnectionStrings:ConnectionString %>"
            SelectCommand="SELECT [№ продажу] AS [№ продажу], [КодПродажу], [КодТовару],
[Кількість] FROM [ОбсягПродажів]"
            DeleteCommand="DELETE FROM ОбсягПродажів WHERE ([№ продажу] = @нп1)"
InsertCommand="INSERT INTO ОбсягПродажів(КодПродажу, КодТовару, Кількість) VALUES (@КП1, @КТ1,
@КСТ1)">
            <DeleteParameters>
                <asp:ControlParameter ControlID="DropDownList4" Name="нп1"
PropertyName="SelectedValue" />
            </DeleteParameters>
            <InsertParameters>
                <asp:ControlParameter ControlID="DropDownList2" Name="КП1"
PropertyName="SelectedValue" />
                <asp:ControlParameter ControlID="DropDownList3" Name="КТ1"
PropertyName="SelectedValue" />
                <asp:ControlParameter ControlID="TextBox1" Name="КСТ1" PropertyName="Text" />
            </InsertParameters>
        </asp:SqlDataSource>
        <asp:SqlDataSource ID="SqlDataSource2" runat="server" ConnectionString="<%$
ConnectionStrings:ConnectionString %>"
            SelectCommand="SELECT [КодТовару], [НазваТовару] FROM [Товари]"></asp:SqlDataSource>
        <asp:SqlDataSource ID="SqlDataSource3" runat="server" ConnectionString="<%$
ConnectionStrings:ConnectionString %>"
            SelectCommand="SELECT [КодПродажу], [Дата] FROM [ПродажТоварів]"></asp:SqlDataSource>
        <asp:SqlDataSource ID="SqlDataSource4" runat="server" ConnectionString="<%$
ConnectionStrings:ConnectionString %>"
            SelectCommand="SELECT ПродажТоварів.КодПродажу, Менеджеди.Прізвище, Менеджеди.Імя,
ПродажТоварів.Дата FROM Менеджеди INNER JOIN ПродажТоварів ON Менеджеди.КодМенеджера =
ПродажТоварів.КодМенеджера">
        </asp:SqlDataSource>
        <asp:DropDownList ID="DropDownList1" runat="server" Height="25px" Width="103px"
AutoPostBack="True"
            OnSelectedIndexChanged="DropDownList1_SelectedIndexChanged">
            <asp:ListItem>Перегляд</asp:ListItem>
            <asp:ListItem Value="Додати">Додати</asp:ListItem>
            <asp:ListItem Value="Видалити">Видалити</asp:ListItem>
        </asp:DropDownList>
        <asp:MultiView ID="MultiView1" runat="server" ActiveViewIndex="0" ViewStateMode="Enabled">
            <asp:View ID="View1" runat="server">
                <h2 align="center">
                    Обсяг продажів</h2>
                <div align="center">
                    <asp:GridView ID="GridView1" runat="server" AllowSorting="True"
AutoGenerateColumns="False"
                        DataSourceID="SqlDataSource1" ForeColor="Black" DataKeyNames="№ продажу">
                        <Columns>
                            <asp:BoundField DataField="№ продажу" HeaderText="№ продажу"
InsertVisible="False"
                                ReadOnly="True" SortExpression="№ продажу" />
                            <asp:BoundField DataField="КодПродажу" HeaderText="КодПродажу"
SortExpression="КодПродажу" />
                            <asp:BoundField DataField="КодТовару" HeaderText="КодТовару"
SortExpression="КодТовару" />
                            <asp:BoundField DataField="Кількість" HeaderText="Кількість"
SortExpression="Кількість" />
                        </Columns>
                    </asp:GridView>
                </div>
                <br />
            </asp:View>
            <asp:View ID="View2" runat="server">
```

```
                <h2 align="center">
                    Додати продаж</h2>
                <h2 align="center">
                    Продажі менеджерів</h2>
                <div align="center">
                    <asp:GridView ID="GridView4" runat="server" AllowSorting="True"
AutoGenerateColumns="False"
                        DataKeyNames="КодПродажу" DataSourceID="SqlDataSource4" ForeColor="Black">
                        <Columns>
                            <asp:BoundField DataField="КодПродажу" HeaderText="КодПродажу"
InsertVisible="False"
                                ReadOnly="True" SortExpression="КодПродажу" />
                            <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
SortExpression="Прізвище" />
                            <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя"
/>
                            <asp:BoundField DataField="Дата" DataFormatString="{0:dd.MM.yyyy}"
HeaderText="Дата"
                                SortExpression="Дата" />
                        </Columns>
                    </asp:GridView>
                </div>
                <br />
                <br />
                <div align="left">

                &n
bsp;               &nbs
p;                
                &n
bsp;               
                    <asp:Label ID="Label4" runat="server" Text="Код продажу"></asp:Label>
                     <asp:Label ID="Label2" runat="server" Text="Товар"></asp:Label>

                &n
bsp;    
                    <asp:Label ID="Label3" runat="server" Text="Кількість"></asp:Label>
                    <br />
                </div>
                <div align="center">
                    <asp:DropDownList ID="DropDownList2" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource3"
                        DataTextField="КодПродажу" DataValueField="КодПродажу" Height="16px"
Width="80px">
                    </asp:DropDownList>
                     <asp:DropDownList ID="DropDownList3" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource2"
                        DataTextField="НазваТовару" DataValueField="КодТовару" Height="16px"
Width="125px">
                    </asp:DropDownList>
                    <asp:TextBox ID="TextBox1" runat="server" Width="54px"></asp:TextBox>
                     
                    <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Додати"
/></div>
                <br />
                <br />
                <div align="center">
                    <asp:GridView ID="GridView2" runat="server" AllowSorting="True"
AutoGenerateColumns="False"
                        DataKeyNames="№ продажу" DataSourceID="SqlDataSource1" ForeColor="Black">
                        <Columns>
                            <asp:BoundField DataField="КодПродажу" HeaderText="КодПродажу"
SortExpression="КодПродажу" />
                            <asp:BoundField DataField="КодТовару" HeaderText="КодТовару"
SortExpression="КодТовару" />
```

```
                            <asp:BoundField DataField="Кількість" HeaderText="Кількість"
SortExpression="Кількість" />
                        </Columns>
                    </asp:GridView>
                </div>
                <br />
            </asp:View>
            <asp:View ID="View3" runat="server">
                <h2 align="center">
                    Видалити продаж</h2>
                <div align="center">
                    <asp:Label ID="Label5" runat="server" Text="№ продажу"></asp:Label>
                      
                    <asp:DropDownList ID="DropDownList4" runat="server" AutoPostBack="True"
                        DataSourceID="SqlDataSource1" DataTextField="№ продажу"
                        DataValueField="№ продажу" Height="16px" Width="79px">
                    </asp:DropDownList>
                       
                    <asp:Button ID="Button2" runat="server" OnClick="Button2_Click"
                        Text="Видалити" />
                </div>
                <br />
                <div align="center">
                    <asp:GridView ID="GridView3" runat="server" AllowSorting="True"
AutoGenerateColumns="False"
                        DataSourceID="SqlDataSource1" ForeColor="Black" DataKeyNames="№ продажу">
                        <Columns>
                            <asp:BoundField DataField="№ продажу" HeaderText="№ продажу"
InsertVisible="False"
                                ReadOnly="True" SortExpression="№ продажу" />
                            <asp:BoundField DataField="КодПродажу" HeaderText="КодПродажу"
SortExpression="КодПродажу" />
                            <asp:BoundField DataField="КодТовару" HeaderText="КодТовару"
SortExpression="КодТовару" />
                            <asp:BoundField DataField="Кількість" HeaderText="Кількість"
SortExpression="Кількість" />
                        </Columns>
                    </asp:GridView>
                </div>
                <br />
                <br />
            </asp:View>
        </asp:MultiView>
    </p>
</asp:Content>

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
    CodeBehind="ПродажТоварів.aspx.cs" Inherits="STYleMy.ПродажТоварів" %>

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
 <h2>   Продаж товарів</h2>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
    <p>
        <asp:Label ID="Label1" runat="server" Font-Bold="True" Font-Italic="True" Font-
Names="Arimo"
            Font-Size="Large" Text="Оберіть дію"></asp:Label>
        <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$
ConnectionStrings:ConnectionString %>"
            SelectCommand="SELECT [КодПродажу], [КодМенеджера], [Дата] FROM [ПродажТоварів]"
            DeleteCommand="DELETE FROM ПродажТоварів WHERE (КодПродажу = @nkpr)"
InsertCommand="INSERT INTO ПродажТоварів(КодМенеджера, Дата) VALUES (@kmnd, @Dta)"
            UpdateCommand="UPDATE ПродажТоварів SET КодМенеджера = @mnk, Дата = @tada WHERE
(КодПродажу = @pdn)">
            <DeleteParameters>
```

```
                <asp:ControlParameter ControlID="DropDownList5" Name="nkpr"
PropertyName="SelectedValue" />
            </DeleteParameters>
            <InsertParameters>
                <asp:ControlParameter ControlID="DropDownList2" Name="kmnd"
PropertyName="SelectedValue" />
                <asp:ControlParameter ControlID="Calendar1" Name="Dta" PropertyName="SelectedDate"
/>
            </InsertParameters>
            <UpdateParameters>
                <asp:ControlParameter ControlID="DropDownList4" Name="mnk"
PropertyName="SelectedValue" />
                <asp:ControlParameter ControlID="Calendar2" Name="tada"
PropertyName="SelectedDate" />
                <asp:ControlParameter ControlID="DropDownList3" Name="pdn"
PropertyName="SelectedValue" />
            </UpdateParameters>
        </asp:SqlDataSource>
        <asp:SqlDataSource ID="SqlDataSource2" runat="server" ConnectionString="<%$
ConnectionStrings:ConnectionString %>"
            SelectCommand="SELECT [КодМенеджера], [Прізвище], [Імя], [ПоБатькові] FROM
[Менеджеди]">
        </asp:SqlDataSource>
        <br />
        <asp:DropDownList ID="DropDownList1" runat="server" Height="25px" Width="103px"
AutoPostBack="True"
            OnSelectedIndexChanged="DropDownList1_SelectedIndexChanged">
            <asp:ListItem>Перегляд</asp:ListItem>
            <asp:ListItem Value="Додати">Додати</asp:ListItem>
            <asp:ListItem Value="Редагувати">Редагувати</asp:ListItem>
            <asp:ListItem Value="Видалити">Видалити</asp:ListItem>
        </asp:DropDownList>
        <asp:MultiView ID="MultiView1" runat="server" ActiveViewIndex="0" ViewStateMode="Enabled">
            <asp:View ID="View1" runat="server">
                <h2 align="center">
                    Продаж товарів</h2>
                <div align="center">
                    <asp:GridView ID="GridView1" runat="server" AllowSorting="True"
AutoGenerateColumns="False"
                        DataKeyNames="КодПродажу" DataSourceID="SqlDataSource1" ForeColor="Black">
                        <Columns>
                            <asp:BoundField DataField="КодПродажу" HeaderText="КодПродажу"
InsertVisible="False"
                                ReadOnly="True" SortExpression="КодПродажу" />
                            <asp:BoundField DataField="КодМенеджера" HeaderText="КодМенеджера"
SortExpression="КодМенеджера" />
                            <asp:BoundField DataField="Дата" HeaderText="Дата"
DataFormatString="{0:dd.MM.yyyy}"
                                SortExpression="Дата" />
                        </Columns>
                    </asp:GridView>
                </div>
                <br />
            </asp:View>
            <asp:View ID="View2" runat="server">
                <h2 align="center">
                    Додати продаж</h2>
                <div align="center">
                    <div align="left">

                &n
bsp;               &nbs
p;                
                &n
bsp;               &nbs
```

```
p;                
 
                        Виберіть менеджера</div>
                    <br />
                    <asp:DropDownList ID="DropDownList2" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource2"
                        DataTextField="Прізвище" DataValueField="КодМенеджера">
                    </asp:DropDownList>
                        
                    <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Додати"
/>
                    <br />
                    <br />
                    Виберіть дату<br />
                    <asp:Calendar ID="Calendar1" runat="server" BackColor="White"
BorderColor="White"
                        BorderWidth="1px" Font-Names="Vijaya" Font-Size="12pt" ForeColor="Black"
Height="10px"
                        Width="180px">
                        <DayHeaderStyle Font-Bold="True" Font-Size="8pt" />
                        <NextPrevStyle Font-Bold="True" Font-Size="8pt" ForeColor="#333333"
VerticalAlign="Bottom" />
                        <OtherMonthDayStyle ForeColor="#999999" />
                        <SelectedDayStyle BackColor="#333399" ForeColor="White" />
                        <TitleStyle BackColor="White" BorderColor="Black" BorderWidth="4px" Font-
Bold="True"
                            Font-Size="12pt" ForeColor="#333399" />
                        <TodayDayStyle BackColor="#CCCCCC" />
                        <WeekendDayStyle Wrap="True" />
                    </asp:Calendar>
                </div>
                <br />
                <div align="center">
                    <asp:GridView ID="GridView2" runat="server" AllowSorting="True"
AutoGenerateColumns="False"
                        DataKeyNames="КодПродажу" DataSourceID="SqlDataSource1" ForeColor="Black">
                        <Columns>
                            <asp:BoundField DataField="КодПродажу" HeaderText="КодПродажу"
InsertVisible="False"
                                ReadOnly="True" SortExpression="КодПродажу" />
                            <asp:BoundField DataField="КодМенеджера" HeaderText="КодМенеджера"
SortExpression="КодМенеджера" />
                            <asp:BoundField DataField="Дата" DataFormatString="{0:dd.MM.yyyy}"
HeaderText="Дата"
                                SortExpression="Дата" />
                        </Columns>
                    </asp:GridView>
                </div>
                <br />
                <br />
            </asp:View>
            <asp:View ID="View3" runat="server">
                <h2 align="center">
                    Редагувати продаж</h2>
                <br />
                <div align="center">
                    <asp:GridView ID="GridView3" runat="server" AllowSorting="True"
AutoGenerateColumns="False"
                        DataKeyNames="КодПродажу" DataSourceID="SqlDataSource1" ForeColor="Black">
                        <Columns>
                            <asp:BoundField DataField="КодПродажу" HeaderText="КодПродажу"
InsertVisible="False"
                                ReadOnly="True" SortExpression="КодПродажу" />
                            <asp:BoundField DataField="КодМенеджера" HeaderText="КодМенеджера"
SortExpression="КодМенеджера" />
```

```
                            <asp:BoundField DataField="Дата" HeaderText="Дата"
DataFormatString="{0:dd.MM.yyyy}"
                                SortExpression="Дата" />
                        </Columns>
                    </asp:GridView>
                </div>
                <br />
                <div align="left">
                    Виберіть № продажу<br />
                    <asp:DropDownList ID="DropDownList3" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource1"
                        DataTextField="КодПродажу" DataValueField="КодПродажу">
                    </asp:DropDownList>
                    <br />
                    <br />
                    <asp:Label ID="Label2" runat="server" Text="Виберіть менеджера"></asp:Label>
                    <br />
                    <asp:DropDownList ID="DropDownList4" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource2"
                        DataTextField="Прізвище" DataValueField="КодМенеджера">
                    </asp:DropDownList>
                     
                    <asp:Button ID="Button2" runat="server" OnClick="Button2_Click" Text="Змінити"
/>
                    <br />
                    <br />
                    <asp:Label ID="Label3" runat="server" Text="Виберіть дату"></asp:Label>
                    <br />
                    <asp:Calendar ID="Calendar2" runat="server" BackColor="White"
BorderColor="White"
                        BorderWidth="1px" Font-Names="Vijaya" Font-Size="12pt" ForeColor="Black"
Height="10px"
                        Width="180px">
                        <DayHeaderStyle Font-Bold="True" Font-Size="8pt" />
                        <NextPrevStyle Font-Bold="True" Font-Size="8pt" ForeColor="#333333"
VerticalAlign="Bottom" />
                        <OtherMonthDayStyle ForeColor="#999999" />
                        <SelectedDayStyle BackColor="#333399" ForeColor="White" />
                        <TitleStyle BackColor="White" BorderColor="Black" BorderWidth="4px" Font-
Bold="True"
                            Font-Size="12pt" ForeColor="#333399" />
                        <TodayDayStyle BackColor="#CCCCCC" />
                        <WeekendDayStyle Wrap="True" />
                    </asp:Calendar>
                </div>
                <br />
            </asp:View>
            <asp:View ID="View4" runat="server">
                <h2 align="center">
                    Видалити продаж</h2>
                <div align="center">

                        <asp:Label ID="Label4" runat="server" Text="Виберіть №
продажу"></asp:Label>
                         
                        <asp:DropDownList ID="DropDownList5" runat="server" AutoPostBack="True"
DataSourceID="SqlDataSource1"
                            DataTextField="КодПродажу" DataValueField="КодПродажу">
                        </asp:DropDownList>
                         
                        <asp:Button ID="Button3" runat="server" OnClick="Button3_Click"
Text="Видалити" />
                        <br />
                    </div>
                    <br />
```

```
                <div align="center">    <asp:GridView ID="GridView4" runat="server"
AllowSorting="True" AutoGenerateColumns="False"
                        DataKeyNames="КодПродажу" DataSourceID="SqlDataSource1" ForeColor="Black">
                        <Columns>
                            <asp:BoundField DataField="КодПродажу" HeaderText="КодПродажу"
InsertVisible="False"
                                ReadOnly="True" SortExpression="КодПродажу" />
                            <asp:BoundField DataField="КодМенеджера" HeaderText="КодМенеджера"
SortExpression="КодМенеджера" />
                            <asp:BoundField DataField="Дата" HeaderText="Дата"
DataFormatString="{0:dd.MM.yyyy}"
                                SortExpression="Дата" />
                        </Columns>
                    </asp:GridView>
                </div>
                <br />
                <br />
            </asp:View>
            <br />
        </asp:MultiView>
        <br />
    </p>
</asp:Content>

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
CodeBehind="month.aspx.cs" Inherits="STYleMy.month" %>
<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
        ConnectionString="<%$ ConnectionStrings:ConnectionString %>"
        SelectCommand="SELECT Менеджеди.Прізвище, Менеджеди.Імя, Менеджеди.ПоБатькові,
Виробники.НазваВтробника, Товари.НазваТовару, ОбсягПродажів.Кількість, Товари.Ціна,
ПродажТоварів.Дата FROM ПродажТоварів INNER JOIN Менеджеди ON ПродажТоварів.КодМенеджера =
Менеджеди.КодМенеджера INNER JOIN ОбсягПродажів ON ПродажТоварів.КодПродажу =
ОбсягПродажів.КодПродажу INNER JOIN Товари ON ОбсягПродажів.КодТовару = Товари.КодТовару INNER
JOIN Виробники ON Товари.КодВиробника = Виробники.КодВиробника ">
    </asp:SqlDataSource>
    <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
        DataSourceID="SqlDataSource1" Visible="False">
        <Columns>
            <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
                SortExpression="Прізвище" />
            <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя" />
            <asp:BoundField DataField="ПоБатькові" HeaderText="ПоБатькові"
                SortExpression="ПоБатькові" />
            <asp:BoundField DataField="НазваВтробника" HeaderText="НазваВтробника"
                SortExpression="НазваВтробника" />
            <asp:BoundField DataField="НазваТовару" HeaderText="НазваТовару"
                SortExpression="НазваТовару" />
            <asp:BoundField DataField="Кількість" HeaderText="Кількість"
                SortExpression="Кількість" />
            <asp:BoundField DataField="Ціна" HeaderText="Ціна" SortExpression="Ціна" />
            <asp:BoundField DataField="Дата" HeaderText="Дата" SortExpression="Дата" />
        </Columns>
    </asp:GridView>
    <asp:SqlDataSource ID="SqlDataSource2" runat="server"
        ConnectionString="<%$ ConnectionStrings:ConnectionString %>"
        SelectCommand="SELECT Менеджеди.Прізвище, Менеджеди.Імя, Менеджеди.ПоБатькові,
Виробники.НазваВтробника, Товари.НазваТовару, Товари.Ціна * ОбсягПродажів.Кількість AS Сума,
ПродажТоварів.Дата FROM ПродажТоварів INNER JOIN Менеджеди ON ПродажТоварів.КодМенеджера =
Менеджеди.КодМенеджера INNER JOIN ОбсягПродажів ON ПродажТоварів.КодПродажу =
ОбсягПродажів.КодПродажу INNER JOIN Товари ON ОбсягПродажів.КодТовару = Товари.КодТовару INNER
JOIN Виробники ON Товари.КодВиробника = Виробники.КодВиробника WHERE (MONTH(ПродажТоварів.Дата) =
@321)">
        <SelectParameters>
```

```
            <asp:ControlParameter ControlID="DropDownList1" DefaultValue="1" Name="321"
                PropertyName="SelectedValue" />
        </SelectParameters>
    </asp:SqlDataSource>
   <h2 align="center"> Оберіть місяць</h2>
  <div align="center">  <asp:DropDownList ID="DropDownList1" runat="server" AutoPostBack="True"
        Height="16px" Width="94px">
        <asp:ListItem Value="1">Січень</asp:ListItem>
        <asp:ListItem Value="2">Лютий</asp:ListItem>
        <asp:ListItem Value="3">Березень</asp:ListItem>
        <asp:ListItem Value="4">Квітень</asp:ListItem>
        <asp:ListItem Value="5">Травень</asp:ListItem>
        <asp:ListItem Value="6">Червень</asp:ListItem>
        <asp:ListItem Value="7">Липень</asp:ListItem>
        <asp:ListItem Value="8">Серпень</asp:ListItem>
        <asp:ListItem Value="9">Вересень</asp:ListItem>
        <asp:ListItem Value="10">Жовтень</asp:ListItem>
        <asp:ListItem Value="11">Листопад</asp:ListItem>
        <asp:ListItem Value="12">Грудень</asp:ListItem>
    </asp:DropDownList>
    <br />
    <asp:Button ID="Button1" runat="server" onclick="Button1_Click1" Text="Button"
        Visible="False" />
    <asp:DropDownList ID="DropDownList2" runat="server" AutoPostBack="True"
        Height="16px" Visible="False" Width="149px">
        <asp:ListItem Value="1">1</asp:ListItem>
        <asp:ListItem Value="2">2</asp:ListItem>
        <asp:ListItem>3</asp:ListItem>
        <asp:ListItem>4</asp:ListItem>
        <asp:ListItem>5</asp:ListItem>
        <asp:ListItem>6</asp:ListItem>
        <asp:ListItem>7</asp:ListItem>
        <asp:ListItem>8</asp:ListItem>
        <asp:ListItem>9</asp:ListItem>
        <asp:ListItem>10</asp:ListItem>
        <asp:ListItem>11</asp:ListItem>
        <asp:ListItem>12</asp:ListItem>
    </asp:DropDownList>
    <br />
    <asp:GridView ID="GridView2" runat="server" AllowSorting="True"
        AutoGenerateColumns="False" DataSourceID="SqlDataSource2" ForeColor="Black">
        <Columns>
            <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
                SortExpression="Прізвище" />
            <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя" />
            <asp:BoundField DataField="ПоБатькові" HeaderText="ПоБатькові"
                SortExpression="ПоБатькові" />
            <asp:BoundField DataField="НазваВтробника" HeaderText="НазваВтробника"
                SortExpression="НазваВтробника" />
            <asp:BoundField DataField="НазваТовару" HeaderText="НазваТовару"
                SortExpression="НазваТовару" />
            <asp:BoundField DataField="Сума" HeaderText="Сума" ReadOnly="True"
                SortExpression="Сума" />
            <asp:BoundField DataField="Дата" DataFormatString="{0:dd.MM.yyyy}"
                HeaderText="Дата" SortExpression="Дата" />
        </Columns>
    </asp:GridView></div>
    <br />
   <h2 align="center"> Діаграма продажів менеджерів за місяць</h2><br />
<br />
   <div align="center"> <asp:Chart ID="Chart1" runat="server" DataSourceID="SqlDataSource2">
        <Series>
            <asp:Series Name="Series1" XValueMember="Прізвище" YValueMembers="Сума">
            </asp:Series>
        </Series>
        <ChartAreas>
```

```
            <asp:ChartArea Name="ChartArea1">
            </asp:ChartArea>
        </ChartAreas>
    </asp:Chart>
    </div>
<br />
<br />
<br />
</asp:Content>
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
CodeBehind="Сума продажу менеджерів.aspx.cs" Inherits="STYleMy.Сума_продажу_менеджерів" %>
<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
    <h2>   Сума продажу менеджерів</h2>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">

        <asp:SqlDataSource ID="SqlDataSource1" runat="server"
            ConnectionString="<%$ ConnectionStrings:ConnectionString %>"
            SelectCommand="SELECT Менеджеди.Прізвище, Менеджеди.Імя, Менеджеди.ПоБатькові,
SUM(ОбсягПродажів.Кількість * Товари.Ціна) AS Сума FROM ПродажТоварів INNER JOIN Менеджеди ON
ПродажТоварів.КодМенеджера = Менеджеди.КодМенеджера INNER JOIN ОбсягПродажів ON
ПродажТоварів.КодПродажу = ОбсягПродажів.КодПродажу INNER JOIN Виробники INNER JOIN Товари ON
Виробники.КодВиробника = Товари.КодВиробника ON ОбсягПродажів.КодТовару = Товари.КодТовару GROUP
BY Менеджеди.Прізвище, Менеджеди.Імя, Менеджеди.ПоБатькові">
        </asp:SqlDataSource>
      <div align="center"> <h2> Ефективність менеджерів </h2>
        <br />
        <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
            DataSourceID="SqlDataSource1" AllowSorting="True" ForeColor="Black"
            Width="627px">
            <Columns>
                <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
                    SortExpression="Прізвище" />
                <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя" />
                <asp:BoundField DataField="ПоБатькові" HeaderText="ПоБатькові"
                    SortExpression="ПоБатькові" />
                <asp:BoundField DataField="Сума" HeaderText="Сума" ReadOnly="True"
                    SortExpression="Сума" />
            </Columns>
        </asp:GridView>
        <br />
        <br />
        <asp:Chart ID="Chart1" runat="server" DataSourceID="SqlDataSource1"
            Width="595px">
            <Series>
                <asp:Series ChartType="Bar" Name="Series1" XValueMember="Прізвище"
                    YValueMembers="Сума">
                </asp:Series>
            </Series>
            <ChartAreas>
                <asp:ChartArea Name="ChartArea1">
                </asp:ChartArea>
            </ChartAreas>
        </asp:Chart>

        <br />
        <br />

        <br /></div>

</asp:Content>
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
CodeBehind="Пошук по даті.aspx.cs" Inherits="STYleMy.Пошук_по_даті" %>
<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
    <p>
       <h2> Перегляд продажів за датою</h2></p>
```

```
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
    <div align="center"> <br />
  <h4>  <asp:Label ID="Label1" runat="server" Text="Оберіть дату"></asp:Label></h4>
    <asp:Calendar ID="Calendar1" runat="server"></asp:Calendar>
    <br />
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
           ConnectionString="<%$ ConnectionStrings:ConnectionString %>"
           SelectCommand="SELECT Менеджеди.Прізвище, Менеджеди.Імя, Менеджеди.ПоБатькові,
Виробники.НазваВтробника, Товари.НазваТовару, Товари.Ціна * ОбсягПродажів.Кількість AS [Сума
продажу], ПродажТоварів.Дата, ОбсягПродажів.Кількість FROM ПродажТоварів INNER JOIN Менеджеди ON
ПродажТоварів.КодМенеджера = Менеджеди.КодМенеджера INNER JOIN ОбсягПродажів ON
ПродажТоварів.КодПродажу = ОбсягПродажів.КодПродажу INNER JOIN Товари ON ОбсягПродажів.КодТовару =
Товари.КодТовару INNER JOIN Виробники ON Товари.КодВиробника = Виробники.КодВиробника WHERE
(ПродажТоварів.Дата = @dtsort)">
        <SelectParameters>
            <asp:ControlParameter ControlID="Calendar1" Name="dtsort"
                PropertyName="SelectedDate" />
        </SelectParameters>
       </asp:SqlDataSource>
    <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
           DataSourceID="SqlDataSource1" AllowSorting="True" ForeColor="Black">
        <Columns>
            <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
                SortExpression="Прізвище" />
            <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя" />
            <asp:BoundField DataField="ПоБатькові" HeaderText="ПоБатькові"
                SortExpression="ПоБатькові" />
            <asp:BoundField DataField="НазваВтробника" HeaderText="НазваВтробника"
                SortExpression="НазваВтробника" />
            <asp:BoundField DataField="НазваТовару" HeaderText="НазваТовару"
                SortExpression="НазваТовару" />
            <asp:BoundField DataField="Кількість" HeaderText="Кількість"
                SortExpression="Кількість" />
            <asp:BoundField DataField="Сума продажу" HeaderText="Сума продажу"
                ReadOnly="True" SortExpression="Сума продажу" />
            <asp:BoundField DataField="Дата" DataFormatString="{0:dd.MM.yyyy}"
                HeaderText="Дата" SortExpression="Дата" />
        </Columns>
    </asp:GridView>
    <br />
    </div>
</asp:Content>
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
CodeBehind="ВсіПродажі.aspx.cs" Inherits="STYleMy.ВсіПродажі" %>
<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
 <h2>   Загальна таблиця продажів</h2>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
    <p>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
        ConnectionString="<%$ ConnectionStrings:ConnectionString %>"
        SelectCommand="SELECT Менеджеди.Прізвище, Менеджеди.Імя, Менеджеди.ПоБатькові,
Виробники.НазваВтробника, Товари.НазваТовару, ОбсягПродажів.Кількість, ПродажТоварів.Дата FROM
ПродажТоварів INNER JOIN Менеджеди ON ПродажТоварів.КодМенеджера = Менеджеди.КодМенеджера INNER
JOIN ОбсягПродажів ON ПродажТоварів.КодПродажу = ОбсягПродажів.КодПродажу INNER JOIN Товари ON
ОбсягПродажів.КодТовару = Товари.КодТовару INNER JOIN Виробники ON Товари.КодВиробника =
Виробники.КодВиробника">
    </asp:SqlDataSource>
    <br />
      <div align="center"> <asp:GridView ID="GridView1" runat="server" AllowSorting="True"
        AutoGenerateColumns="False" DataSourceID="SqlDataSource1" ForeColor="Black">
        <Columns>
            <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
                SortExpression="Прізвище" />
            <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя" />
```

```
            <asp:BoundField DataField="ПоБатькові" HeaderText="ПоБатькові"
                SortExpression="ПоБатькові" />
            <asp:BoundField DataField="НазваВтробника" HeaderText="Втробник"
                SortExpression="НазваВтробника" />
            <asp:BoundField DataField="НазваТовару" HeaderText="Модель"
                SortExpression="НазваТовару" />
            <asp:BoundField DataField="Кількість" HeaderText="Кількість"
                SortExpression="Кількість" />
            <asp:BoundField DataField="Дата" HeaderText="Дата продажу"
                SortExpression="Дата" DataFormatString="{0:dd.MM.yyyy}" />
        </Columns>
    </asp:GridView>   </div>
</p>
</asp:Content>

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
    CodeBehind="Пошук по прізвищу.aspx.cs" Inherits="STYleMy.Пошук_по_прізвищу" %>

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
    <h2>
        Пошук по прізвищу</h2>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
    <div align="center">
        <br />
        <h4>
            <asp:Label ID="Label1" runat="server" Text="Введіть прізвище"></asp:Label></h4>
    </div>
    <div align="center">
        <h4>
            <asp:TextBox ID="TextBox1" runat="server" Width="164px"
                ondatabinding="TextBox1_DataBinding"
ontextchanged="TextBox1_TextChanged"></asp:TextBox>
         <asp:Button ID="Button1" runat="server" onclick="Button1_Click1"
                Text="Знайти" />
        </h4>
    </div>
    <div align="center">

            <br />
            <asp:SqlDataSource ID="SqlDataSource2" runat="server"
                ConnectionString="<%$ ConnectionStrings:ConnectionString %>"
                SelectCommand="SELECT Менеджеди.Прізвище, Менеджеди.Імя, Менеджеди.ПоБатькові,
Виробники.НазваВтробника, Товари.НазваТовару, ОбсягПродажів.Кількість, Товари.Ціна,
ПродажТоварів.Дата FROM Менеджеди INNER JOIN ПродажТоварів ON Менеджеди.КодМенеджера =
ПродажТоварів.КодМенеджера INNER JOIN ОбсягПродажів ON ПродажТоварів.КодПродажу =
ОбсягПродажів.КодПродажу INNER JOIN Товари ON ОбсягПродажів.КодТовару = Товари.КодТовару INNER
JOIN Виробники ON Товари.КодВиробника = Виробники.КодВиробника WHERE (Менеджеди.Прізвище LIKE '%'
+ @Прізвище + '%')">
                <SelectParameters>
                    <asp:ControlParameter ControlID="TextBox1" DefaultValue="" Name="Прізвище"
                        PropertyName="Text" Type="String" />
                </SelectParameters>
            </asp:SqlDataSource>
            <asp:GridView ID="GridView2" runat="server" AutoGenerateColumns="False"
                DataSourceID="SqlDataSource2" AllowSorting="True" ForeColor="Black">
                <Columns>
                    <asp:BoundField DataField="Прізвище" HeaderText="Прізвище"
                        SortExpression="Прізвище" />
                    <asp:BoundField DataField="Імя" HeaderText="Імя" SortExpression="Імя" />
                    <asp:BoundField DataField="ПоБатькові" HeaderText="ПоБатькові"
                        SortExpression="ПоБатькові" />
                    <asp:BoundField DataField="НазваВтробника" HeaderText="НазваВтробника"
                        SortExpression="НазваВтробника" />
                    <asp:BoundField DataField="НазваТовару" HeaderText="НазваТовару"
                        SortExpression="НазваТовару" />
```

```
                    <asp:BoundField DataField="Кількість" HeaderText="Кількість"
                        SortExpression="Кількість" />
                    <asp:BoundField DataField="Ціна" HeaderText="Ціна" SortExpression="Ціна" />
                    <asp:BoundField DataField="Дата" DataFormatString="{0:dd.MM.yyyy}"
                        HeaderText="Дата" SortExpression="Дата" />
                </Columns>
            </asp:GridView>
        <br />
    </div>
</asp:Content>
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
CodeBehind="Середні показники по виробнику.aspx.cs" Inherits="STYleMy.WebForm2" %>
<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
  <h2>  Середні показники виробника</h2>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">




        <asp:SqlDataSource ID="SqlDataSource1" runat="server"
            ConnectionString="<%$ ConnectionStrings:ConnectionString %>"
            SelectCommand="SELECT Виробники.НазваВтробника, AVG(Товари.Ціна) AS [Середня ціна],
AVG(Товари.НаСкладі) AS [Сер на складі], AVG(ОбсягПродажів.Кількість) AS [Сер продано] FROM
Виробники INNER JOIN Товари ON Виробники.КодВиробника = Товари.КодВиробника INNER JOIN
ОбсягПродажів ON Товари.КодТовару = ОбсягПродажів.КодТовару GROUP BY Виробники.НазваВтробника">
        </asp:SqlDataSource>




        <br />
    <br />
    <div align="center">
        <div align="center"> 
            <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
            DataSourceID="SqlDataSource1" AllowSorting="True"
    ForeColor="Black" Width="808px">
            <Columns>
                <asp:BoundField DataField="НазваВтробника" HeaderText="НазваВтробника"
                    SortExpression="НазваВтробника" />
                <asp:BoundField DataField="Середня ціна" HeaderText="Середня ціна"
                    ReadOnly="True" SortExpression="Середня ціна" DataFormatString="{0:F2}" />
                <asp:BoundField DataField="Сер на складі" HeaderText="Сер на складі"
                    ReadOnly="True" SortExpression="Сер на складі" />
                <asp:BoundField DataField="Сер продано" HeaderText="Сер продано"
                    ReadOnly="True" SortExpression="Сер продано" />
            </Columns>
        </asp:GridView></div>
        <br />

        <br />

       <h2> Середня ціна на товар виробника</h2>
        <asp:Chart ID="Chart1" runat="server" DataSourceID="SqlDataSource1"
    TextAntiAliasingQuality="Normal">
            <Series>
                <asp:Series Name="Series1" XValueMember="НазваВтробника"
                    YValueMembers="Середня ціна">
                </asp:Series>
            </Series>
            <ChartAreas>
```

```
                <asp:ChartArea Name="ChartArea1">
                </asp:ChartArea>
            </ChartAreas>
        </asp:Chart>
        <br />
        <br />
      <h2> Продано у середньому виробником</h2>

        <asp:Chart ID="Chart2" runat="server" DataSourceID="SqlDataSource1">
            <Series>
                <asp:Series Name="Series1" XValueMember="НазваВтробника"
                    YValueMembers="Сер продано">
                </asp:Series>
            </Series>
            <ChartAreas>
                <asp:ChartArea Name="ChartArea1">
                </asp:ChartArea>
            </ChartAreas>
        </asp:Chart>
        <br />
      <h2> Товарів виробника на складі</h2>
        <asp:Chart ID="Chart3" runat="server" DataSourceID="SqlDataSource1"
            Palette="Pastel">
            <Series>
                <asp:Series ChartType="Pie" Name="Series1" XValueMember="НазваВтробника"
                    YValueMembers="Сер на складі">
                </asp:Series>
            </Series>
            <ChartAreas>
                <asp:ChartArea Name="ChartArea1">
                </asp:ChartArea>
            </ChartAreas>
        </asp:Chart>
        </div>
    <br />
    <br />


</asp:Content>
```

ДОДАТОК В

(обов'язковий)

ПРЕЗЕНТАЦІЙНІ СЛАЙДИ

ДИПЛОМНИЙ ПРОЕКТ

«ІНФОРМАЦІЙНА СИСТЕМА МОНІТОРИНГУ ДІЯЛЬНОСТІ САЛОНУ З ПРОДАЖУ МОБІЛЬНИХ ТЕЛЕФОНІВ»

Виконав: студент IV курсу, групи ІПЗ-18-1 Король В.В.

Керівник: канд. тех. наук, доцент Форкун Ю.В.

2

Вступ

В наш час поняття смартфон і мобільний телефон так швидко увійшли до навколишнього світу, що уявити процес надання послуг без застосування смартфонів не можливо. Тим паче, в такій висококонкурентній галузі як салону продажу. Також на звичний порядок надання послуг вплинула пандемія короновірусу, і це ще одна причина з якої власники салону з продажу мобільних телефонів мають вдосконалювати процес надання послуг в торговельній галузі, якщо бажають зберегти свій бізнес. В цьому випадку на допомогу приходять інформаційні технології за допомогою яких здійснюється залучення та втримання існуючих клієнтів.

Торговельна галузь - це галузь, яка найбільше постраждала під час пандемії короновірусу. Тому актуальним є питання вдосконалення та розробки програмного забезпечення, яке б відповідало викликам часу та надавало конкурентну перевагу за рахунок зручності використання та потрібної функціональності.

3

Мета, об’єкт та предмет дослідження

Головною метою проекту є є розробка системи моніторингу діяльності салону з продажу мобільних телефонів..

Об’єктом дослідження роботи виступають веб-сервіси, що спеціалізуються на моніторингу продажу та обслуговування мобільних телефонів менеджерами магазинів.

Предметом дослідження виступають алгоритми та методи моніторингу діяльності, які використовуються для прогнозування діяльності роботи менеджерів та роботи магазину загалом.

4

Аналіз предметної області, її структурних та функціональних особливостей

Моніторинг комплекс наукових, технічних, технологічних, організаційних та інших засобів, які забезпечують систематичний контроль (стеження) за станом та тенденціями розвитку досліджуваних процесів. [1]

Методологічно моніторинг - це проведення ряду однотипних замірів при цьому головна інформація полягає навіть не в самих значеннях результатів, а в їх зміні, динаміці від одного заміру до іншого.

Основним завданням є збір та аналіз інформації по діяльності певного суб’єкта, для прийняття подальших рішень що - до підвищення конкретних показників, зі зростанням яких зросте загальна ефективність діяльності підприємства в цілому.

Мета системи та дерево цілей

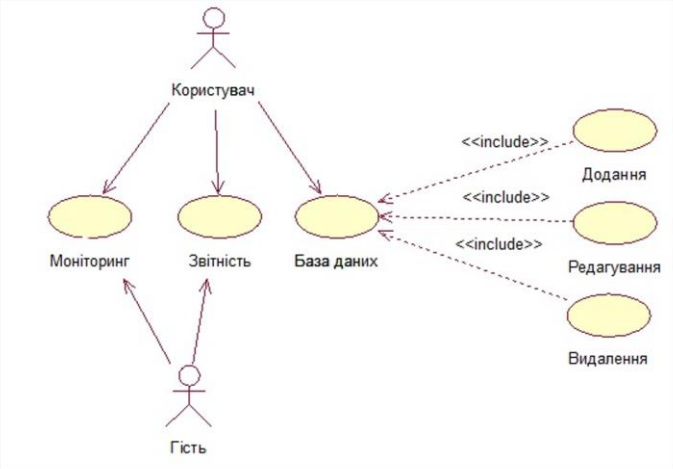
8

Мета системи конкретизується за допомогою цілей. Одним зі способів розкриття внутрішньої структури мети є побудова дерева цілей.

Дерево цілей системи складається з трьох рівнів.

Діаграма варіантів використання

9



12

Вибір технологій і методів реалізації системи

Для реалізації нашого проекту з множини засобів чітко можна виділити два основних напрямки:

- Linux - Apache - MySQL;
- MS Windows - IIS - ASP -MSSQL/MS Access

Для розробки системи було обрано другий напрям, цей вибір ґрунтується на ряді переваг asp.Net.

14

Оскільки наш проект має клієнт-серверну архітектуру, то оперувати базою даних буде сервер в залежності від запитів клієнта. В результаті потрібно два застосунки, зокрема сам сервер та клієнтський застосунок, який буде з ним взаємодіяти.

Для розробки даного проекту я обрав мову програмування C#. C# має багато можливостей для зручної роботи з різними СКБД.

C# використовується, як для розробки програм для настільних ПК, так і для розробки веб-сервісів, сайтів та інших застосунків. Дана мова є універсальною. На ній можна розробляти, як і прості проекти, так і досить потужні додатки.

Серверну частину пропонується розробляти на основі технології ASP.NET Core на основі шаблону WEB API.

Клієнтську частину пропонується розробляти для платформи .NET Framework, як веб-додаток.

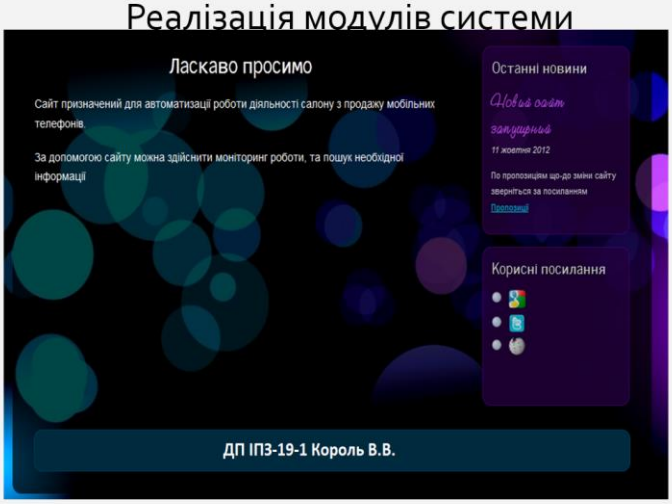
16

Для реалізації нашої системи ми будемо створювати веб-додаток використовуючи сучасні технології проектування, зокрема спроектуємо наш додаток використовуючи концепцію MVC (модель – представлення - контролер).

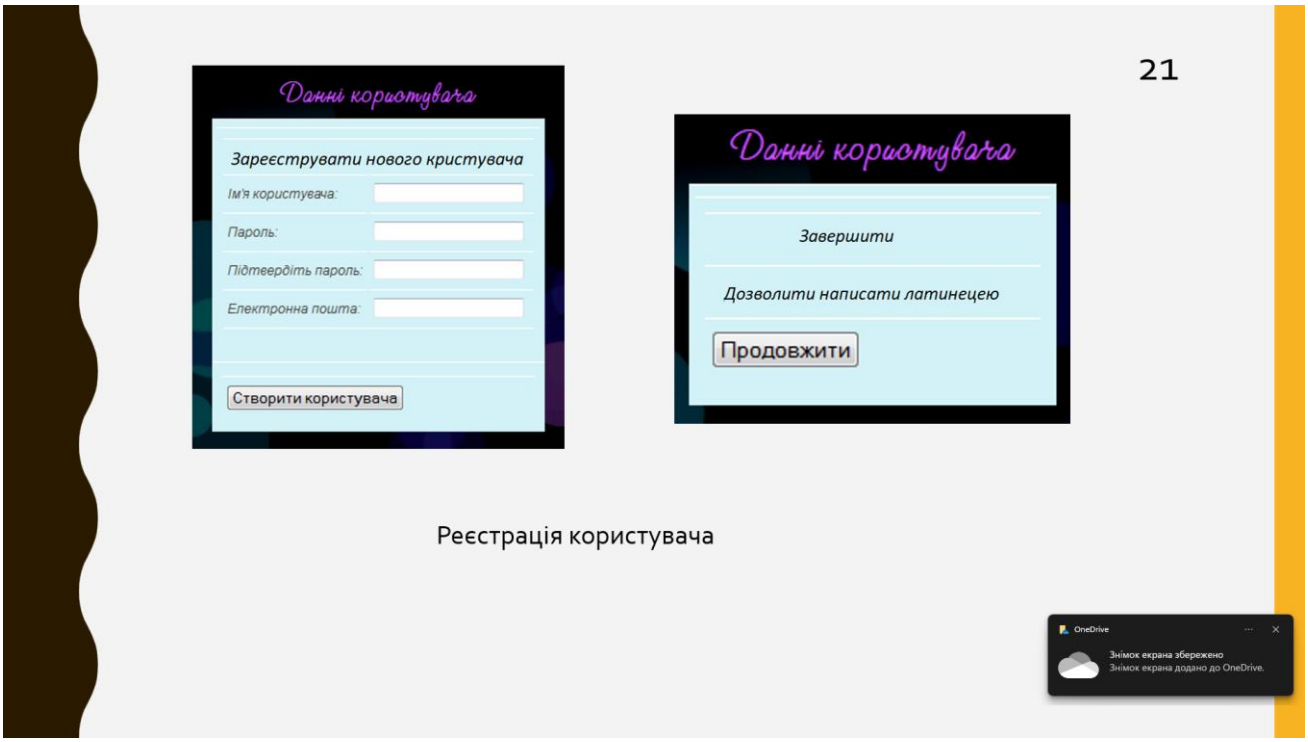
Цей шаблон передбачає поділ системи на три взаємопов'язані частини: модель даних, вигляд (інтерфейс користувача) та модуль керування. Застосовується для відокремлення даних (моделі) від інтерфейсу користувача (вигляду) так, щоб зміни інтерфейсу користувача мінімально впливали на роботу з даними, а зміни в моделі даних могли здійснюватися без змін інтерфейсу користувача.

18

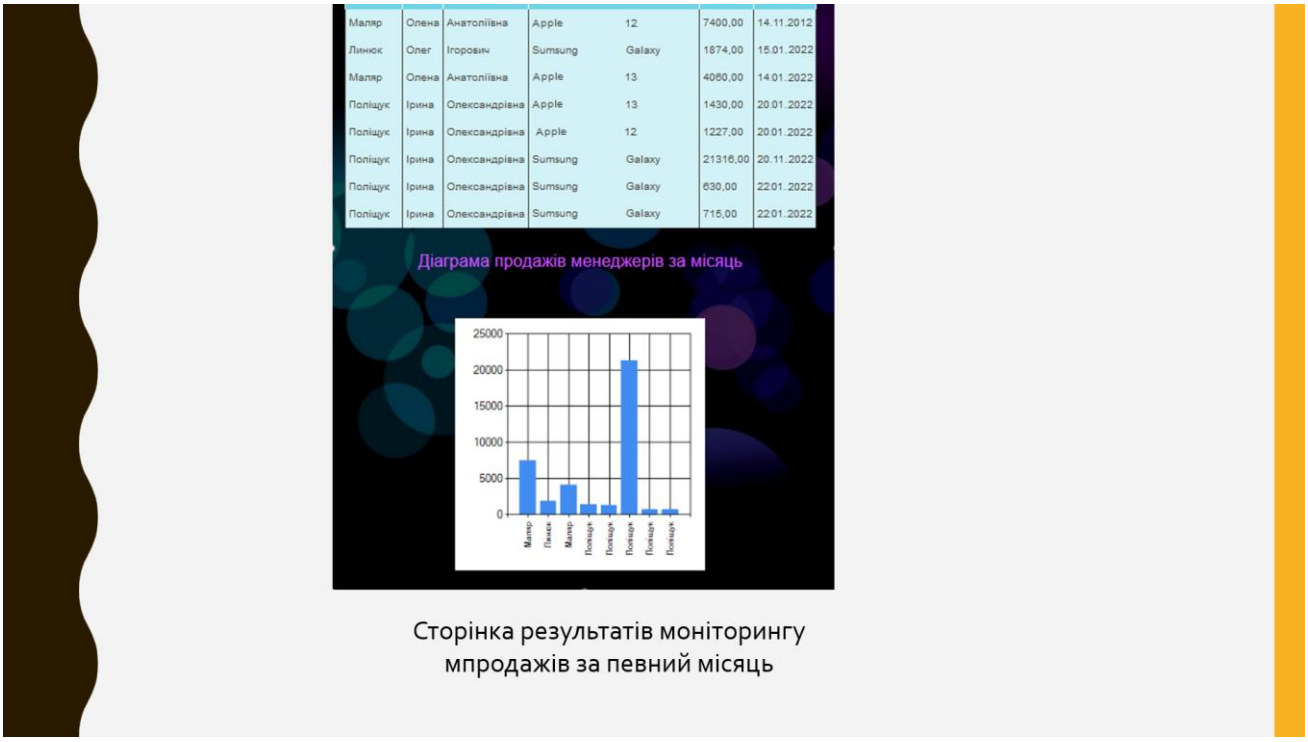
Реалізація модулів системи



Головна сторінка системи моніторингу діяльності салону з продажу мобільних телефонів

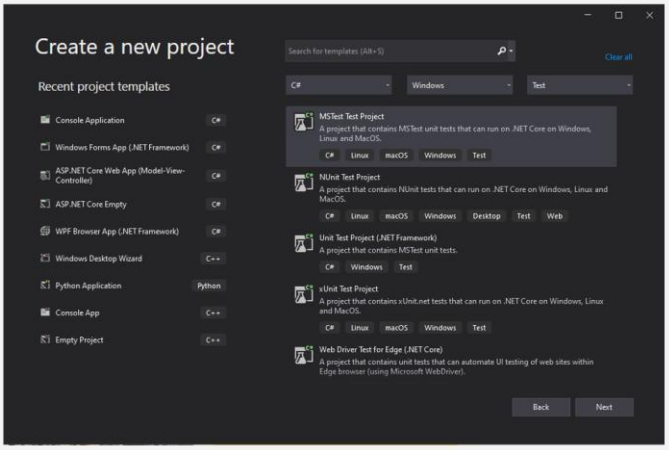


Реєстрація користувача



Сторінка результатів моніторингу
мпродажів за певний місяць

Сторінка результатів звітності за певну дату



Тестування xUnit в MS Visual Studio

23

Висновки

У ході дипломного проектування було розроблено інформаційну систему моніторингу діяльності салону з продажу мобільних телефонів, яка полегшує та спрощує роботу персоналу салону та надає більше інформації його клієнтам.

Система повинна виконувати основні задачі поставлені перед нею:

- зберігання і надання доступу до даних;
- ведення обліку товарів;
- формування бази співробітників підприємства;
- проводить аналіз роботи менеджерів;
- формує фінансовий та статистичний звіти про діяльність менеджера за будь-який період.

Отже, розроблена програма має велику кількість переваг у порівняні із аналогічними програмами, що робить її конкурентоспроможною.

24

Дякую за увагу!

Завідувачу кафедри інженерії програмного забезпечення проф. Бедратюку Л. П.

здобувача вищої освіти
Короля В.В.
Прізвище, ініціали

факультет ІТ, 4 курс, група ІПЗ-18-1

ЗАЯВА

З правилами чинного Положення «Про дотримання академічної доброчесності в Хмельницькому національному університеті» від 26.09.2020 (зі змінами від 26.11.2020), згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений. Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

06.06.22
дата

підпис

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 25.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилок в документах: 9%**

ID: 104689 Назва: Інформаційна система моніторингу діяльності салону з продажу мобільних телефонів Додано в БД: 2022-06-07 Автора: В. В. Король Керівники: Ю.В. Форкун Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	58691	508	21953 (37%)	193 (38%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми
103276	Назва: Звіт з переддипломної практики Додано в БД: 2022-05-04 Автора: В. В. Король Керівники: Форкун Ю. В. Консультанти: Опоненти:	14873 (25.0%)	113 (22.0%)

UNICHECK

Ім'я користувача:
Кафедра ІПЗ

ID перевірки:
1011493158

Дата перевірки:
07.06.2022 17:10:27 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
07.06.2022 17:11:10 EEST

ID користувача:
100005589

Назва документа: DiplomKorol_VD

Кількість сторінок: 68 Кількість слів: 9188 Кількість символів: 74996 Розмір файлу: 8.10 MB ID файлу: 1011369370

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

19.9% Схожість

Найбільша схожість: 5.61% з джерелом з Бібліотеки (ID файлу: 1011341903)

10.8% Джерела з Інтернету | 248 ..... Сторінка 70

10.8% Джерела з Бібліотеки | 143 ..... Сторінка 72

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи | 7

Підозріле форматування | 21 сторінка

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА ДИПЛОМНИЙ ПРОЕКТ
освітнього ступеня «Бакалавр»

Дипломник Король Володимир Володимирович

Тема Інформаційна система моніторингу діяльності салону з продажу мобільних телефонів

Спеціальність 121 – Інженерія програмного забезпечення

Обсяг дипломного проекту:

Кількість листів креслень _____; кількість сторінок записки _____

1.Короткий зміст пояснювальної записки та прийнятих рішень У дипломному проекті проведено аналіз інформаційного забезпечення предметної області та інформаційних потоків, розроблена структура застосування. Для розробки програмного продукту використано мову C# та середовище розробки Microsoft Visual Studio 2019. За допомогою цих засобів розроблено програмне забезпечення моніторингу діяльності салону з продажу мобільних телефонів. Було проведено тестування програми, за результатами якого доведено, що розроблений сайт працює коректно та готове до експуатації.

2. Висновок про відповідність проекту поставленому завданню Дипломний проект виконаний відповідно до поставленого завдання та з дотриманням всіх вимог.

3. Характеристика виконання кожного розділу проекту, ступінь використання останніх досягнень науки і техніки та передових методів роботи У вступі доведено актуальність теми, визначено мету та завдання дипломноого проектування. У першому розділі здійснено змістовний аналіз предметної області. Визначено поняття системи, моніторингу, його функції та методологічних засад його проведення. Здійснено аналіз останніх публікацій та наявного програмного забезпечення, з визначенням задач необхідних для реалізації проекту. Також визначено функціональне призначення та вимоги до розроблюваного програмного забезпечення. У другому розділі проведено розроблеено архітектуру інформаціної системи та спроектовано базу даних. У третьому розділі та виконано практичну розробку програмних модулів з описом їх особливостей, в результаті чого створено інформаційну ситему. В четвертому розділі було виконано xUnit тестування системи та проведено його у відповідності до функціональних вимог, в результаті було підтверджено коректну роботу програми.

4. Позитивні сторони проекту Тематика дипломного проекту є актуальною, оскільки на моніторингу діяльності об'єктів торгівельної діяльності. Дозволить не тільки підвищити імідж компанії, але бути конкуретноздатним на ринку

5. Негативні сторони проекту У проекті пошук публікацій був реалізований лише за категоріями – було б доцільно додати розширений пошук..

6. Оцінка графічного оформлення та пояснювальної записки проекту Графічне оформлення виконано відповідно до теми дипломного проекту та подано у вигляді діаграм і рисунків. Пояснювальна записка оформлена згідно вимог чинних стандартів.

7. Відгук про дипломний проект в цілому Дипломний проект заслуговує позитивної оцінки. Матеріал пояснювальної записки структурований, послідовний, чіткий та простий, що дозволяє чітко зрозуміти викладений матеріал у рамках тематики дипломного проекту. Графічний матеріал дає можливість наочно побачити деталі проектування системи.

8. Інші зауваження

9. Оцінка дипломного проекту Дипломний проект виконаний у повному обсязі, відповідає поставленій задачі та заслуговує на оцінку «добре».

РЕЦЕНЗЕНТ Чешуя Віктор Миколайович
КТН, доцент кафедри кібербезпеки

"8" 06 2022 р. (підпис)

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ
КАФЕДРИ ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: «Інформаційна система моніторингу діяльності салону з продажу мобільних телефонів»

Автор: Король Володимир Володимирович
Спеціальність: 121 – Інженерія програмного забезпечення
Освітня програма: Освітньо-професійна програма «Інженерія програмного забезпечення»
Науковий керівник: Форкун Юрій Вікторович, кандидат технічних наук, доцент
Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	**відповідає**
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
5	Інше:	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

1) у тексті дипломної роботи системами перевірки на плагіат виявлено схожість з деякими документами в частині загальновживаних обов'язкових словосполучень у стандартних бланках (титулка, бланк завдання), у структурі змісту, назвах розділів/підрозділів тощо та в назвах публікацій у переліку джерел посилання;

2) в якості запозичень системою було зафіксовано деякі послідовності вихідного коду і посиланя на бібліотеки, які є стандартними мовними конструкціями програмування та не можуть розглядатися як об'єкт авторських прав і, відповідно, їх порушення;

3) усі запозичення є фрагментарними або мають належним чином оформленні посилання;

4) виявлені модифікації тексту не впливають на відсоток схожості.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів ідентичності/ схожості, складає 25% і адресується до 1 джерела, а саме звіту з передипломниої практики самого студента, що, з урахуванням характеру звіту з переддипломної практики, який включає в себе частину дипломного проекту відповідає характеру теми і свідчить на користь дипломної роботи.

Керівник _____ Ю.В. Форкун

Гарант ОП _____ Л. П. Бедратюк

Завідувач кафедри _____ Л. П. Бедратюк