

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) _____ автоматизації і комп'ютерних систем _____

Кафедра _____ інформаційних систем _____

Освітній ступінь _____ бакалавр _____

Спеціальність _____ 122 Комп'ютерні науки _____

(код і назва)

Освітньо-професійна програма _____ Комп'ютерні науки _____

(назва)

ЗАТВЕРДЖУЮ

Завідувач

кафедри _____

“ _____ ” _____ 20 _____ року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Рябого Віталія Олеговича

(прізвище, ім'я, по батькові)

1. Тема роботи _____ Розроблення Web-додатку для ТОВ "Цукорагропром" _____

керівник роботи _____ Харкянен Олена Валеріївна, доцент, к.т.н. _____

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від "29" квітня 2021 р. № 248-КС

2. Строк подання здобувачем роботи _____ 03.06.2021 р. _____

3. Вихідні дані до роботи опис діяльності ТОВ "Цукорагропром", результати дослідження діяльності ВП «Жданівський цукровий завод» в цілому і об'єкта автоматизації – відділу обробки замовлень, замовлення, договори

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) системний аналіз діяльності підприємства, функціональний аналіз діяльності відділу обробки замовлень, розрахунок економічного ефекту від впровадження системи, розробка бази даних в MS SQL Server, розробка графічного інтерфейсу в MS Visual Studio мовою C#, розробка інструкції користувача, опис заходів з охорони праці та техніки безпеки

5. Перелік графічного матеріалу

Організаційна структура ВП «Жданівський цукровий завод», функціональна модель процесу обробки замовлення підприємством, схеми бази даних, скріншоти інтерфейсу, програмний код

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Вступ	Харкянен О. В., доцент		
1	Харкянен О. В., доцент		
2	Харкянен О. В., доцент		
3	Харкянен О. В., доцент		

7. Дата видачі завдання 29.04.2021 р.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Вивчення діяльності підприємства ТОВ «ЦУКОРАГРОПРОМ»	29.04.2021	виконано
2	Проведення системного аналізу діяльності ТОВ «ЦУКОРАГРОПРОМ»	03.05.2021	виконано
3	Розроблення функціональної моделі діяльності ТОВ «ЦУКОРАГРОПРОМ»	05.05.2021	виконано
4	Дослідження та порівняльний аналіз систем аналогів розробки	08.05.2021	виконано
5	Обґрунтування вибору програмних засобів розробки	09.05.2021	виконано
7	Генерація фізичної моделі бази даних в MS SQL Server	13.05.2021	виконано
8	Розробка інтерфейсу користувача системи	16.05.2021	виконано
9	Написання інструкції користувача	24.05.2021	виконано
10	Розробка заходів з охорони праці та техніки безпеки	26.05.2021	виконано
11	Оформлення пояснювальної записки	27.05.2021	виконано
12	Оформлення презентації	28.05.2021	виконано

Здобувач _____ Рябий В.О.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Харкянен О.В.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Кваліфікаційна робота «Розроблення WEB-додатку для ТОВ «Цукорагропром»» розроблена Рябим В.О., складається з 93 сторінок, 14 таблиць, 9 рисунків, 5 додатків, 7 сторінок графічного матеріалу, 14 літературних джерел.

У кваліфікаційній роботі наведені результати дослідження діяльності ВП «Жданівський цукровий завод» ТОВ «Цукорагропром» вцілому і об'єкта автоматизації – відділу продажів підприємства. Для реалізацій функцій, описаних у функціональній моделі, розроблено веб-додаток, описано обрані технології для його розробки, розраховано економічний ефект від впровадження веб-додатку, розроблена інструкція користувача, описані заходи з охорони праці для робочого місця працівника ЕОМ.

Функціональна модель діяльності підприємства розроблена за допомогою CASE-засобу CA ERwin Process Modeler.

База даних розроблена в системі управління базами даних MS SQL Server, моделі бази створено за допомогою Microsoft SQL Server Management Studio 18.

КЛЮЧОВІ СЛОВА: ВЕБ-ДОДАТОК, ПРОДУКЦІЯ, ЦУКРОВИЙ ЗАВОД,
АВТОМАТИЗАЦІЯ

ANNOTATION

Qualification work "Development of a WEB-application for LLC" Tsukoragroprom "" was developed by Ryaby VO, consists of 93 pages, 14 tables, 9 figures, 5 appendices, 9 pages of graphic material, 14 literary sources.

The qualification work presents the results of a study of the activities of SE "Zhdanovsky Sugar Plant" LLC "Tsukoragroprom" as a whole and the object of automation - the order processing department of the enterprise. For the implementation of the functions described in the functional model, a web application is developed, the selected technologies for its development are described, the economic effect of the web application implementation is calculated, user instructions are developed, labor protection measures for the computer worker's workplace are described.

The functional model of the enterprise was developed using the CASE tool CA ERwin Process Modeler.

The database was developed in the MS SQL Server database management system, the database model was created using Microsoft SQL Server Management Studio 18.

KEY WORDS: WEB APPLICATION, PRODUCTS, SUGAR FACTORY, AUTOMATION

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ПРОЦЕСІВ ПІДПРИЄМСТВА ТОВ «ЦУКОРАГРОПРОМ» ТА ПОСТАНОВКА ЗАДАЧ НА ПРОЕКТУВАННЯ	10
1.1. Загальна характеристика ТОВ «Цукорагропром»	10
1.2. Організаційна структура ВП «Жданівський цукровий завод», роль і взаємодія підрозділів	12
1.3. Аналіз нинішнього стану автоматизації ВП «Жданівський цукровий завод»	14
1.4. Розроблення функціональної моделі та аналіз існуючих бізнес процесів під час реалізації продукції ВП «Жданівський цукровий завод».	14
1.4.1. Опис методології IDEF0	15
1.4.2. Опис функціональної моделі (AS-IS)	15
1.4.3. Виявлені недоліки	17
1.4.3. Задачі автоматизації	18
1.5. Огляд існуючих рішень для розв’язання виявлених проблем 18	
1.5.1. Програма «Універсальна Система Обліку»	18
1.5.2. Програма BAS ERP	19
1.5.3. Програма «1С:ПІДПРИЄМСТВО»	20
1.5.4. Порівняння систем-аналогів	21
1.6. Обґрунтування доцільності проектування веб-додатку для ВП «Жданівський цукровий завод»	22
1.5.1. 1.7. Опис функціональної моделі (TO-BE)	22

1.8. Розрахунок економічного ефекту від впровадження системи	24
РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА ПРОЕКТУВАННЯ.....	30
2.1. Загальні положення	30
2.2. Призначення і цілі створення системи	30
2.3. Характеристика об'єкта автоматизації	31
2.4. Вимоги до системи	31
2.5. Склад і зміст робіт по створенню системи	43
2.6. Порядок контролю і приймання системи	43
2.7. Вимоги до складу і змісту робіт із підготовки до введення системи в дію	44
2.8. Вимоги до документації	45
2.9. Джерела розробки	45
РОЗДІЛ 3. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ	47
3.1. Інформаційне забезпечення системи.....	47
3.2 Алгоритмізація та реалізація комплексу задач автоматизації....	49
3.3. Інструкція користувача	54
РОЗДІЛ 4. ОХОРОНА ПРАЦІ.....	59
4.1 Загальні питання охорони праці.....	59
4.2. Структура управління охороною праці на підприємстві	60
4.3 Загальна характеристика приміщення та робочого місця.....	60
4.4 Метеорологічні параметри робочої зони	62
4.5 Освітлення	63
4.6 Шум та вібрація у робочому приміщенні.....	64
4.7. Електробезпека.....	65
4.8. Ергономічні вимоги до робочого місця	67

ВИСНОВКИ	69
БІБЛІОГРАФІЧНИЙ СПИСОК	70
ДОДАТКИ.....	71
ДОДАТОК А. Організаційна структура ВП «Жданівський цукровий завод»	72
ДОДАТОК Б. Функціональна модель діяльності відділу продажів	73
ДОДАТОК Д. Схема бази даних в MS SQL Server	78
ДОДАТОК Ж. Скріншоти	79
ДОДАТОК К. Програмний код	82

ВСТУП

Інформатизація є важливим фактором зміни якості життя суспільства: адже в сучасних умовах рівень і спосіб життя починає все більшою мірою залежати від рівня та якості послуг, культури, освіти, організації роботи тощо. Інформатизація впливає на всі сфери людського життя: сприяє зростанню продуктивності праці, поліпшенню керованості економікою, розвитку наукоємних виробництв та високих технологій, збагаченню духовного життя та подальшій демократизації суспільства.

Управління будь-яким бізнесом сьогодні потребує активного використання інформаційних технологій. Світові тенденції використання сучасних інформаційно-комунікаційних технологій у функціонуванні підприємства характеризуються активним запровадженням найновітніших розробок не тільки у напрямку автоматизації управління та виробництва, а також для налагодження комунікацій з клієнтами, оптимізації маркетингу тощо.

Інформатизація процесів в ТОВ "Цукорагропром" є невід'ємною частиною глобального процесу активного формування та використання інформаційних ресурсів вказаного підприємства.

Створення та впровадження інформаційної системи вирішить питання доступу до публічної інформації про підприємство, а також дозволить автоматизувати процес створення та обробки замовлень. Така інновація спостить та пришвидшить процес створення та обробки замовлень на даному виробництві.

Тому розробка web-додатку для обробки замовлень продукції підприємства є актуальним інструментом для інформаційної підтримки діяльності як замовниками продукції підприємства так і менеджера.

РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ПРОЦЕСІВ ПІДПРИЄМСТВА ТОВ «ЦУКОРАГРОПРОМ» ТА ПОСТАНОВКА ЗАДАЧ НА ПРОЕКТУВАННЯ

1.1. Загальна характеристика ТОВ «Цукорагропром»

ТОВ «Цукорагропром» — один із провідних вертикально-інтегрованих агропромислових холдингів, основний бізнес якого пов'язаний із забезпеченням українських промислових споживачів та населення високоякісним цукром.

Основна діяльність ТОВ «Цукорагропром» полягає у вирощуванні цукрових буряків, виробництві високоякісного цукру та супутньої продукції (меляси та сухого гранульованого жому). Крім того, в Полтавській області ТОВ «Цукорагропром» має потужності із переробки сої (завод із переробки сої в м. Глобине) та виробництва біогазу (Біоенергетичний комплекс в м. Глобіно).

ТОВ «Цукорагропром», у виробництві цукру, в якості сирцю використовує власно-вирощений цукровий буряк або імпортує тростинний цукор-сирець. Вирощування цукрового буряку, для ТОВ «Цукорагропром», займається компанія-спеціаліст Агро-торгівельна фірма «Астарта-Київ».

Весь технологічний процес виробництва цукру-піску починається з вирощування цукрового буряка: підготовки насіння і ґрунту; своєчасного посіву, удобрення та поливу; збору і транспортування. ТОВ «Цукорагропром» тільки, контролює процес посіву, удобрення, поливу та зберігання.

Для збору урожаю, компанія використовує сучасну та автоматизовану техніку: System-Compactor 12/1200, біт-культиватор Tiger V8-3, біт-навантажувач Maus. Тростинний цукор-сирець імпортується компанією з Куби, Бразилії та Гватемали.

Жданівський цукровий завод — один із найстарших діючих цукрових заводів України. Історія заводу веде своє літочислення з грудня 1852 року. Саме тоді, в період бурхливого розвитку цукрової промисловості в Україні, поміщик Олександр Кушельов побудував цукровий завод у селі Війтівці (до недавнього

часу село Жданівка, Вінницької області, у 2014 році перейменоване в село Війтівці, Вінницької області).

У 2008 році на підприємстві замінено барабанну бурякомийку на коритоподібну кулачкову. А також було замінено конденсатні гідроколонки на збірники з повною автоматизацією.

У 2010 році було проведено більш масштабні зміни: замінено 16-ти рамні бурякорізки на 12-ти рамні, заміну дискових фільтрів густого сиропу на проточні ФСП-3-8, встановлено відстійники соку I та II сатурації, турбогенератор, конденсатор I продукту та модернізовано існуючі. Також було побудовано трансформаторну підстанцію та модернізовано котли I та II сатурації.

У 2011 році встановлено додатково два корпуси випарної станції типу А2-ВАГ-3000. Крім того, проводилась реконструкція у продуктовому та сокоочисному відділеннях.

16 березня 2011 року реконструкція очисних споруд вод 2 та 3 категорій.

У 2014 році проведено реконструкцію вакуум-апаратів I ступеню кристалізації. А також впроваджена схема стандарт-сиропу.

У 2015 році проведено реконструкцію вакуум-апаратів II-III ступенів кристалізації.

У 2016 році замінено існуючі центрифуги II та III продукту на центрифуги безперервної дії фірми ВМА К3300. а також розпочата реконструкція жомопресового відділення. Встановлено жомопрес глибокого віджиму «Mercier» ТМ-3000.

У 2017 році завершена реконструкція жомопресового відділення з встановленням другого преса глибокого віджиму «Mercier» ТМ-3000, проведена заміна двох існуючих центрифуг I продукту на центрифуги Е-1390 фірми ВМА.

У 2018 році встановлена ще одна центрифуга I продукту Е-1390 фірми ВМА, розпочато реконструкцію жомової ями в закритий склад зберігання жому.

У 2018 році добова потужність заводу склала 4,3 тис. т буряка.

Сьогодні це сучасне європейське підприємство, яке представляє нові технології та покращує виробництво. Потужність заводу збільшена до 550-600 т цукру на добу.

1.2. Організаційна структура ВП «Жданівський цукровий завод», роль і взаємодія підрозділів

Організаційна структура управління ВП «Жданівський цукровий завод» наведена у Додатку А.

Тип організаційної структури управління підприємством – лінійно-функціональний та повністю відповідає потребам галузі.

Майно підприємства складають виробничі і невиробничі фонди, а також інші цінності, вартість яких відображається в самостійному балансі підприємства.

Джерелами формування майна підприємства є:

- грошові та матеріальні інвестиції засновника;
- кошти які одержані від реалізації продукції, послуг;
- доходи від акцій підприємства;
- майно, придбане в інших суб'єктів господарювання, організації та громадян у встановленому законодавством порядку;
- інші легальні джерела .

Загальні відомості про діяльність підприємства наведено в табл. 1.1.

Таблиця 1.1 – Загальні відомості про діяльність підприємства

	Відомості
Повне найменування емітента	ВП «Жданівський цукровий завод»
Організаційно-правова форма	Дочірнє підприємство
Ідентифікаційний код за ЄДРПОУ	41169521

Місцезнаходження	Вінницька область, Хмільницький район, с. Війтівці, вул. Заводська 2.
Керівник	Зверков Євгеній Миколайович
Міжміський код, телефон, факс	04338 39518
Електронна поштова адреса	hmilnycke@astartakiev.kiev.ua

Розвиток підприємства здійснюється на основі зростання попиту споживачів на продукцію та послуги підприємства, що обумовлено такими факторами:

- зростанням доходів населення та відповідним підвищенням платоспроможного попиту;
- рекламою продукції та послуг підприємства на популярних джерелах;
- розширенням асортименту товарів і послуг національного виробництва та підвищенням їх якості;
- накопиченим досвідом роботи на ринку України;
- підвищення кваліфікації менеджерів і виробничого персоналу;
- упровадженням повного виробничого циклу обслуговування споживачів:

Виробнича структура стосується лише основного виробництва. У складі цехів відрізняють потокові лінії, які складаються з відділень, ті спеціалізуються на виконанні окремих стадій технологічного процесу. За кожним з них закріплена певна кількість робочих місць з визначеними операціями. До числа підрозділів інфраструктури підприємства належать ремонтно-механічна майстерня, компресорна, котельня, гараж та експедиція. Схема технологічного процесу включає у собі підготовку сировини до виробництва, приготування опари і тіста, випічку виробів, їх зберігання та реалізацію.

На підприємстві виробляється велика кількість різних видів цурку та похідних від нього продуктів. Сировину ВП «Жданівський цукровий завод»

отримує від різноманітних постачальників, але у підприємства є і постійні інвестори, які співпрацюють з підприємством протягом тривалого часу і є надійним джерелом коштів та ресурсів.

1.3. Аналіз нинішнього стану автоматизації ВП «Жданівський цукровий завод»

Під час аналізу роботи підприємства було досліджено рівень автоматизації та комп'ютеризації процесів виробництва. Так як вказаний підрозділ входить до складу великого агрохолдингу, об'єкт отримує належне фінансове та технологічне забезпечення. Це дозволяє постійно модернізувати процес виробництва, застосовуючи передові технології.

Проте, в процесі досліджень було виявлено проблему з реалізацією продукції для малого бізнесу та фізичних осіб, оскільки основне джерело реалізації продукту – компанії гіганти. Для потенційного клієнта ускладнюється процес придбання продукції, оскільки він не має достатньої інформації про сам продукт, а також про шляхи його поставок чи купівлі.

Тому стан автоматизації продажів продукції підприємства є незадовільним, оскільки клієнти є недостатньо проінформованими та не мають можливості придбати продукцію віддалено.

1.4. Розроблення функціональної моделі та аналіз існуючих бізнес процесів під час реалізації продукції ВП «Жданівський цукровий завод»

Для виявлення недоліків у процесах реалізації продукції ВП «Жданівський цукровий завод» потрібно провести аналіз процесу у такому вигляді яким він є на поточний момент. Для проведення такого аналізу розроблена функціональна модель, а саме модель AS-IS за допомогою методології IDEF0.

1.4.1. Опис методології IDEF0

IDEF0 - це методологія графічного опису систем і процесів діяльності організації як безлічі взаємозалежних функцій. Вона дозволяє досліджувати функції організації, не пов'язуючи їх з об'єктами, що забезпечують їх реалізацію.

Обрана модель володіє великою кількістю переваг, таких як:

Наочність. Зобразивши роботу компанії у вигляді схеми, стає зрозумілим, як працює компанія, де можуть виникнути проблеми і як попередити їх появу.

Взаєморозуміння, унеможливлення неправильного трактування схеми. Наочність і доступність функціональної моделі, що представляє роботу компанії у вигляді блоків і керуючих елементів, допоможуть вам під час обговорення з керівництвом функціонування їх компанії. До речі, в разі необхідності до функціональної моделі створюється глосарій, де зібрані всі терміни і умовні позначення. Таким чином, мінімізується можливість виникнення непорозумінь між вами і керівником, співробітниками компанії.

Простота і економія часу при створенні моделі. Звичайно, для того щоб добре володіти методикою функціонального моделювання, потрібно витратити багато часу. В першу чергу, потрібно навчитися представляти величезну кількість інформації у вигляді лаконічній схеми, тобто вміти фільтрувати і стискати вихідні дані. Але витрачені на навчання сили і час з лишком окупаються згодом. Адже на те, щоб створити функціональну модель і доступно представити її, вже не знадобиться багато часу.

Мінімальна вірогідність помилки. Робота за стандартом IDEF0 вимагає суворого дотримання його правил. Це дисциплінує виконавця і виключає можливість виникнення помилки. До того ж будь-яка невідповідність стандарту відразу ж стає помітним [2].

Саме завдяки вказаним перевагам було обрано IDEF0 в якості базового засобу аналізу та опису бізнес-процесів.

1.4.2. Опис функціональної моделі (AS-IS)

Розроблена функціональна модель описує процеси виробництва та реалізації продукції ВП «Жданівський цукровий завод» в тому вигляді, як вони відбуваються на підприємстві (AS-IS).

Контекстна діаграма наведена на Рис. 1 у додатку Б. Основними складовими діаграми є:

Управління моделі:

- Нормативно правові акти – це набір інструкцій та правил які описують процес та методи виробництва продукції.
- Прайс лист – перелік продукції та ціни на неї.

Входи:

- Сировина – матеріали з яких виготовляють продукцію.
- Замовлення в паперовому вигляді – заявка замовника на купівлю продукції підприємства.

Виходи:

- Виконане замовлення – документ з переліком замовленої продукції.
- Відвантажена продукція – продукція, яка відвантажена замовнику згідно замовлення.

Механізми:

- Замовник – об'єкт який здійснює замовлення.
- Менеджер – керує виконанням замовлень.
- Виробничий відділ – відділ підприємства, що відповідає за вироблення продукції
- ПК

Декомпована модель наведена на Рис. 2 у додатку Б. Дана модель являє собою чорирьохрівневу ієрархію упорядкованих і взаємозв'язаних діаграм, відображає існуючий порядок бізнес процесів та має статус «AS-IS».

На першому рівні декомпозиції діаграма складаються з чотирьох основних блоків:

- Виготовлення продукту;
- Передача замовлення на виробництво;

- Отримання продукції зі складу;
- Отримання замовником продукції.

Як можна побачити з описаної структури вся система є залежною від механізму обробки замовлення за певним принципом (управління), який надходить в паперовому форматі (вхід).

Вивчення та огляд процесів за допомогою моделі AS-IS здійснюється для того щоб визначити та показати існуючі недоліки у процесі обробки замовлення і чим саме цей процес це задовольняє нинішній рівень автоматизації.

1.4.3. Виявлені недоліки

Під час дослідження поточного стану процесу обробки замовлення виявлено наступні проблеми:

- Оскільки весь процес створення та обробки замовлення проходить в паперовому вигляді, сповільнюється процес взаємодії між відділами підприємства.
- Велика вірогідність похибок обчислень та помилок в оформленні замовлення через людський фактор.
- Замовник продукції не має можливості оперативно отримувати інформацію про стан виконання замовлення.

1.4.3. Задачі автоматизації

Для розв'язання виявлених проблем було вирішено автоматизувати процес створення та обробки замовлення надавши замовникам та відділам цукрового заводу необхідний їм доступ до замовлень. Передбачити можливість редагування та відстеження статусу замовлення згідно стадій його виконання.

Задачі для автоматизації:

- Створення загальнодоступного web-ресурсу з інформацією про продукцію ВП «Жданівський цукровий завод».
- Забезпечення можливості формування замовниками нових замовлень через web-додаток.
- Забезпечення можливості відстеження статусу виконання замовлення в режимі реального часу.
- Розробка адміністративної частини web-додатку для забезпечення обробки замовлень менеджерами ВП «Жданівський цукровий завод».

Розробка web-додатку для ВП «Жданівський цукровий завод» забезпечить збільшення потоку замовників за рахунок зручності формування та відстеження виконання замовлень на продукцію підприємства, зменшення помилок, які виникають при роботі персоналу підприємства, що безпосередньо вплине на підвищення якості обслуговування замовників.

1.5. Огляд існуючих рішень для розв'язання виявлених проблем

Для автоматизації процесу реалізації продукції підприємств існує велика кількість додатків та веб-ресурсів. Розглянуті рішення є web-додатками для автоматизації процесу виробництва, або ж площадками для продажу продукції.

1.5.1. Програма «Універсальна Система Обліку»

Програма поєднує в собі реалізацію і виробництво - дві основні складові єдиного процесу. Застосунок спрощує процеси виробництва в компанії, так як кожен із співробітників будуть працювати як один згуртований колектив, а

розмежування прав доступу допоможуть в рішеннях поставлених завдань, кожен з фахівців буде працювати з тією частиною програми, за що відповідає.

Основні можливості програми:

- Формується єдина база клієнтів з усіма необхідними контактними даними;
- Кожній продукції можна скласти калькуляцію, щоб програма автоматично розраховувала кількість необхідної сировини;
- Програма може працювати з будь-якою кількістю складів і підрозділів;
- Можна відслідковувати виконання робіт і контролювати ефективність;
- Легке внесення даних, безліч красивих шаблонів для оформлення вашого виду програми, легкість і простота в використанні.

Недоліки засобу:

- Висока вартість ліцензії для програмного забезпечення
- Немає прямого доступу клієнтів до списків продукції та її купівлі
- Програма може працювати тільки всередині мережі, обмежуючи коло охоплення робочих процесів

1.5.2. Програма BAS ERP

BAS ERP - це один з найпотужніших сучасних продуктів, метою використання якого є повна автоматизація багатьох робочих процесів у великих підприємствах з великою кількістю робочих місць, складним процесом виробництва. ERP система — це прикладне рішення, яке удосконалює, контролює і автоматизує найскладніші бізнес процеси великих компаній і виробництв.

Можливості та переваги ERP системи:

- Можливість використовувати кілька правил роботи з клієнтами – типових і індивідуальних
- Узгодження всіх замовлень, з статусів, контроль виконання
- Управління ефективністю процесів продажів та угод з клієнтом
- Самообслуговування клієнтів

- Моніторинг стану процесів продажів
- Оформлення замовлень постачальникам і контроль їх виконання.

Недоліки програмного пакету:

- Вартість програмного забезпечення
- Фокусується більше на роботі з клієнтами і не дає можливості відслідковувати внутрішні процеси виробництва.

1.5.3. Програма «1С:ПІДПРИЄМСТВО»

«1С:Підприємство» це програмний продукт, який призначений для автоматизації процесів обліку (бухгалтерський, кадровий, управлінський), а також управління виробничими процесами на підприємстві. За допомогою даної програми, завдяки її універсальності і багатофункціональності, можна автоматизувати бізнес-процеси використовуючи її типові і галузеві рішення.

Основні переваги:

- Можливість проведення автоматизації підприємств
- Зручність ведення зобов'язань по податках
- Можливість швидко отримати будь-яку інформацію
- Багатопрофільність, наявність індивідуальних рішень

Недоліки:

- Базові продукти не підходять для компаній роздрібною торгівлі
- Необхідність підтримки бухгалтерських рішень
- Низький рівень безпеки і захисту конфіденційної інформації.

1.5.4. Порівняння систем-аналогів

Порівнюємо системи-аналогами за наступними функціями: автоматизація процесу виробництва, незалежна система створення замовлень, можливість доступу за межами локальності системи, моніторинг стану виконання замовлення, доступ для клієнтів, створення інформаційних сторінок, вартість.

Таблиця 1.2 Порівняння систем-аналогів

Функції \ Системи	Універсальна Система Обліку	BAS ERP	1С:ПІДПРИЄМСТВО
Автоматизація процесу виробництва	+	+	-
Незалежна система створення замовлень	-	+	+
Можливість доступу за межами локальності системи	-	-	+
Моніторинг стану виконання замовлення	+	-	+
Доступ для клієнтів	+	+	-
Створення інформаційних сторінок	-	+	-
Вартість	1800 грн/міс.	2800 грн/міс.	Коливається залежно від обраних модулів

Висновок: після дослідження трьох варіантів програмного забезпечення: Універсальна Система Обліку, BAS ERP, 1С:ПІДПРИЄМСТВО, бачимо що всі програмні продукти мають широкий функціонал та можуть забезпечити автоматизацію процесів як на великих так і на малих підприємствах. Програми є досить вузькоспеціалізованими і не зможуть забезпечити комплексне вирішення поставлених завдань. Перевагою даних програмних пакетів є велика кількість функцій та модулів, а також зрозумілий та простий в використанні інтерфейс користувача.

1.6. Обґрунтування доцільності проектування веб-додатку для ВП

«Жданівський цукровий завод»

У наш час існує велика кількість засобів для автоматизації ведення і обліку замовлень. Є можливість користуватися загальними пакетами програмного забезпечення, або ж вибрати тільки модулі необхідні для вирішення актуальних проблем на підприємстві.

Наявні програмні пакети є дорогі, а дешевші або безкоштовні варіанти не містять достатньої кількості функціоналу. Також вище перелічені програмні засоби не зможуть повністю вирішити всі поставлені в розділі 1.4.3 задачі. Тому було вирішено розробити веб-додаток що буде забезпечувати підприємство ефективними засобами інформаційної підтримки при роботі з замовниками.

Створений засіб забезпечить інформування замовників про продукти які виробляє підприємство, а також дасть можливість придбати їх он-лайн.

1.5.1. 1.7. Опис функціональної моделі (ТО-ВЕ)

Метою створення веб-додатку є надання клієнту доступу до інформації про продукти які виробляє підприємство, а також можливість замовити цю продукцію он-лайн. Також спроститься процес обробки замовлення та процес відстеження термінів виготовлення продукції, оскільки всі підрозділи виробництва будуть володіти необхідною інформацією про об'єми і терміни виконання замовлень.

Знайдені недоліки в бізнес-процесах підприємства були виправлені та удосконалено існуючі процеси, що відображено у функціональній моделі статусу ТО-ВЕ, що описує процес обробки замовлення підприємством (див. Рис. Б.3 у додатку Б).

На даній діаграмі показано автоматизовані та нові процеси для обробки замовлень підприємством. При розробці даної функціональної моделі створені нові входи, виходи та спрощено механізми управління.

Входи:

- Акаунт клієнта – містить всю попередньо збережену інформацію про замовника.
- Інформація про замовлення – інформація про тип, обсяг і параметри продукції, яка необхідна замовнику.
- Опції доставки – містить інформацію про тип та терміни доставки готової продукції.

Виходи:

- Виконане замовлення – виконане та доставлене замовлення,

Побудована модель TO-BE процесу обробки замовлення компанією.

Модель складається з таких процесів:

- Обробка замовлення
- Передача замовлення на виробництво
- Отримання продукції зі складу
- Отримання замовником продукції

які показано на першому рівні декомпозиції (див. Рис. Б.4. у додатку Б).

Веб-додаток заповнюється даними про продукцію, яку виробляє компанія і має функції створення онлайн замовлення (вхід), а також його обробку та збереження результатів. Після створення та обробки замовлення менеджера користувач може відслідковувати статус свого замовлення та бачити історію замовлень, зроблених раніше. Основною перевагою створення моделі є те, що система більше не залежить від одного з процесів, а всі процеси отримують необхідні для неї дані незалежно, що дозволить оптимізувати та пришвидшити процес виробництва.

Також в майбутньому в межах розвитку додатку передбачена можливість генерування звітів по замовленнях клієнта, а також включити в додаток весь моніторинг всього життєвого циклу розробки продукту. Додаток є динамічним і може використовуватися не тільки для потреб заданого підприємства, а й бути задіяним в інших сферах виробництва, не потребуючи значних модифікацій.

1.8. Розрахунок економічного ефекту від впровадження системи

Економічний ефект від використання засобів автоматизації - це економія суспільної праці при виробництві будь-яких видів продукції. Економія праці або економія часу докорінно визначає спрямованість капіталовкладень.

Вихідні дані для розрахунку:

- Ступінь новизни розроблюваних задач – "В" – використання типових проектних рішень за умови їх змін.
- Група складності алгоритму – 1.
- Узагальнені дані вхідної та вихідної інформації веб-додатку для ВП «Жданівський цукровий завод» у табл.1.3.

Таблиця 1.3. Узагальнені дані

Вид інформації	Позначення	К-сть наборів даних
Змінна інформація	ЗІ	m = 9
Нормативно-довідкова інформація	НДІ	n = 5
База даних	БД	p = 1
Обробка в режимі реального часу	РЧ	так
Забезпечення телекомунікаційної обробки даних і управління віддаленими об'єктами	ТОУ	ні

Витрати часу на систему призначену для розробки ескізного проекту (передпроектного дослідження) T1 і технічного завдання T2, будуть наступні табл. 1.4.

Таблиця 1.4 Визначення витрат часу

Вид системи	Стадія розробки системи	
	Передпроектне дослідження	Технічне завдання
	В	В
Управління науково-технічною інформацією	T1 = 65	T2 = 28

Визначається базове значення витрат часу для стадій «Технічний проект», «Робочий проект» і «Впровадження». Вхідними даними для визначення є:

- кількість форм вхідної інформації $B_1 = 6$,
- кількість форм вихідної інформації $B_2 = 3$,
- базове значення витрат часу для стадій «Технічний проект»: $T_{B3} = 82$;
- базове значення витрат часу для стадій «Робочий проект»: $T_{B4} = 234$;
- базове значення витрат часу для стадій «Впровадження»: $T_{B5} = 68$.
- базове значення витрат часу T_B коригується за допомогою поправочних коефіцієнтів для всіх стадій розробки автоматизованої системи.

Розрахунок витрат часу для стадії «Технічний проект» (T_3).

Коефіцієнт трудомісткості робіт k_{Π} визначається за формулою:

$$K_{\Pi} = \frac{1.24*9+0.73*5+1.68*1}{9+5+1} = 1.097$$

Таблиця 1.5 Коефіцієнти k_1, k_2, k_3 для стадії «Технічний проект»

Вид використаної інформації	Ступінь новизни
	B
k_1 (ЗІ)	1.24
k_2 (НДІ)	0.73
k_3 (БД)	1.68

Таблиця 1.6. Коефіцієнт ступеню новизни проекту, k_0 для веб додатку ВП

«Жданівський цукровий завод»

Стадія розробки системи	Вид обробки	Ступінь новизни
		B
Технічний проект	РЧ	1.37
Робочий проект	РЧ	1.25
Впровадження	РЧ	1.18

Коефіцієнт ступеню новизни проекту, k_0 , що враховує вид обробки інформації для трьох стадій розробки системи визначається з урахуванням того, що обробка здійснюється в режимі реального часу.

Витрати часу для стадії «технічний проект» T_3 :

$$T_3 = T_{B3} * k_{\Pi} * k_0 = 65 * 1.097 * 1.37 = 97.687;$$

Розрахунок витрат часу для стадії «Робочий проект» (Т₄) для системи

Для того що б визначити витрат часу на стадії «Робочий проект» використовують формулу, де $k_{п}$ – коефіцієнт, що враховує вид використаної інформації і визначається за формулою:

$$K_{п} = \frac{1.1*9+0.65*5+0.54*1}{9+5+1} = 0,912$$

Таблиця 1.7. Коефіцієнти k_1, k_2, k_3 для стадії «Робочий проект».

Вид використаної інформації	Група складності алгоритму	Ступінь новизни
		В
k_1 (ЗІ)	1	1.05
k_2 (НДІ)	1	0.63
k_3 (БД)	1	0.44

Коефіцієнт, що враховує вид обробки інформації на стадії «Робочий проект» табл. 1.6. Коефіцієнт складності контролю вхідної та вихідної інформації і визначається на стадії «Робочий проект» і «Впровадження».

$$k_c = 1.00$$

Витрати часу T_4 вимірюються в людино-днях:

$$T_4 = T_{Б4} * k_{п} * k_o * k_c = 234 * 0.912 * 1.25 * 1.00 = 266.76;$$

Поправочні коефіцієнти мають такі ж значення, як і при обрахунку T_4 .

Для стадії визначення загальних витрат часу на «Впровадження» T_5 (люд-днів) використовують формулу:

$$T_5 = T_{Б5} * k_{п} * k_o * k_c = 68 * 0.912 * 1.18 * 1.00 = 73.178;$$

Таким чином, загальні витрати людської праці на проектування системи складають:

$$T_{\Sigma} = 67 + 24 + 97.687 + 266.76 + 73.178 = 528.625_{(люд-дн)}$$

Для кваліфікаційної роботи кількість робочих годин складає 540 із 8-годинним робочим днем, тому на розробку проекту виділено Φ , днів:

$$\Phi = \frac{540}{8} = 68$$

Для дипломного проекту $\Phi = 68$ днів. Тоді визначаємо кількість місяців із розрахунку 24 робочих днів. Кількість місяців на розробку, М:

$$M = \frac{\Phi}{24} = \frac{68}{24} = 2$$

Отже, для виконання такого проекту потрібно таку чисельність виконавців Ч, виконавців, обраховується за:

$$Ч = \frac{T\Sigma}{\Phi} = \frac{528.625}{68} \approx 8$$

Якщо прийняти, що оплата програміста здійснюється в розмірі 20 000 грн, то оплата праці всіх виконавців, яка підраховується за формулою складе:

$$V'_1 = Ч * M * 3П_{пр} = 8 * 2 * 2000 = 320000 \text{ грн.}$$

Витрати, пов'язані з розробкою програми на ПК

Розрахунок річного фонду часу роботи ПК.

Дійсний річний фонд часу ПК у годинах дорівнює числу робочих годин у році для оператора, за винятком часу на технічне обслуговування і ремонт ПК (в середньому 5год/міс + 6 роб.днів/рік).

$$T_{ПК} = 3723 - (6 * 8 + 5 * 12) = 3615 \text{ год.}$$

Так як під час виконання дипломного проекту (роботи) студент в середньому витрачає 450 год. машинного часу, то величина фонду часу ПК дорівнює

$$T_{ПК} = 3615 * \frac{450}{3723} = 436.9_{\text{год.}}$$

Поточні витрати на експлуатацію V1

ЦР – ринкова вартість ПК, орієнтовно складає 15000 грн., КУН – коефіцієнт, що враховує витрати на установку і налагодження ПК і дорівнює:

$$Ц_{ПК} = Ц_{р} * (1 + K_{ун}) = 15000 * (1 + 0.12) = 16800 \text{ грн.}$$

Амортизаційні відрахування використання ПК, ЗАМ, норма амортизаційних відрахувань, яка для ПК дорівнює НА = 5:

$$З_{АМ} = \frac{16800}{5} = 9360_{\text{грн.}}$$

Електровитрати визначаються :

$$З_{ЕЛ} = P_{ПК} * T'_{ПК} * Ц_{ЕЛ} * A = 0.032 * 436.9 * 2.1 * 0.9 = 26.42 \text{ грн}$$

ЗР – витрати на ремонт і ТО ПК визначаються як 6% від балансової вартості ПК, ЦПК.

$$ЗР = ЦПК * 0.06 = 16800 * 0.06 = 1008 \text{ грн.}$$

ЗМАТ – непрямі витрати, пов'язані з експлуатацією ПК, визначаються як 5% від балансової вартості ПК ЦПК.

$$З_{МАТ} = Ц_{ПК} * 0.05 = 16800 * 0.05 = 840 \text{ грн.}$$

Таким чином заробітна плата обслуговуючого персоналу:

$$З_{ОП} = 3360 \text{ грн, } З_{АМ} = 896 \text{ грн, } З_{ЕЛ} = 26.42 \text{ грн,}$$

Поточні експлуатаційні витрати V_1'' , грн, визначаються:

$$V_1'' = З_{ОП} + З_{АМ} + З_{ЕЛ} + З_{Р} + З_{МАТ} = 3360 + 3808 + 26.42 + 1008 + 840 = 9050 \text{ грн}$$

Отже, загальні витрати на розробку програмного забезпечення комп'ютерної системи розраховуються за формулою і складуть:

$$V = V_1' + V_1'' = 320000 + 9050 = 329050 \text{ грн}$$

Витрати на придбання і установку ПК V_2

Витрати на придбання і установку ПК (v_2):

$$V_2 = ЦПК = 18000 \text{ грн.}$$

Якщо немає потреби в купівлі ПК, то ці витрати дорівнюють "0".

Витрати на підготовку приміщення V_3 .

Це витрати які залежать від стану приміщення, де буде встановлюватися ПК. Так як пристосоване приміщення є, тому:

$$V_3 = 0 \text{ грн.}$$

Витрати на навчання персоналу V_4

В середньому навчання персоналу триватиме 3 дня, тому можна вважати, що:

$$V_4 = 2550 \text{ грн.}$$

Загальна вартість розробки і впровадження системи

Загальна вартість розробки і впровадження системи V_{Σ} , вираховується:

$$V_{\Sigma} = V_1 + V_2 + V_3 + V_4 = 320000 + 18000 + 0 + 2550 = 340550 \text{ грн.}$$

Так як норма амортизаційних втрат для комп'ютерних систем $N_A = 5$, то для обрахування річного економічного ефекту слід брати до розгляду величину:

$$V_p = \frac{340550}{5} = 68110 \text{ грн.}$$

Таблиця 1.7. Основні джерела прибутку від впровадження комп'ютерної системи і порядок його підрахунку.

№	Джерело прибутку	Сума
1	Скорочення витрат на друк матеріалів	12700 грн.
2	Оптимізація роботи менеджерів з продажів	12500 грн.
3	Збільшення кількості проданої продукції	31800 грн.
		66000 грн.

Коефіцієнт економічної ефективності розробки визначається за формулою:

$$K_{\text{ЕФ}} = \frac{Pr}{Vp} = \frac{66000}{68110} = 0.96$$

Термін окупності розробки дорівнює визначається за формулою:

$$T_{\text{ОК}} = \frac{1}{K_{\text{ЕФ}}} = \frac{1}{0.96} = 1,04$$

Таким чином, термін окупності інформаційної системи буде **1 рік і 1 місяць**.

РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА ПРОЕКТУВАННЯ

2.1. Загальні положення

2.1.1. Найменування системи: «Веб-додаток для підприємства ВП «Жданівський цукровий завод» ТОВ «Цукорагропром».

2.1.2. Результати робіт зі створення системи оформлюються згідно з вимогами ДСТУ на відповідні етапи розробки. Порядок оформлення і передачі результатів у даному випадку визначається змістом і календарним планом виконання розробки.

2.1.3. У випадку необхідності на наступних стадіях робіт по створенню системи окремі положення можуть уточнюватися і розвиватися.

2.2. Призначення і цілі створення системи

2.2.1 Призначення системи.

Система призначена для автоматизації роботи відділу реалізації продукції ВП «Жданівський цукровий завод». Система є веб-сторінкою підприємства, а також призначена для здійснення он-лайн замовлень продукції підприємства. Додаток містить інформацію про доступну для замовлення продукцію цукрозаводу, акаунти клієнтів та його замовлення і містить функціонал, який дозволяє створювати нові замовлення та обробляти існуючі.

2.2.2. Цілі створення системи.

Основною метою створення системи є забезпечення всього циклу обробки та виконання замовлення продукції підприємством. Це забезпечить створення умов для збільшення кількості проданої продукції та спрощення процесу опрацювання замовлення відділом продажів.

Завдання реалізації продукції ізольовано вважається неефективним та потребує процесів автоматизації. Тому, в умовах функціонування автоматизованої системи, у замовників з'явиться можливість отримувати всю інформацію про продукцію та здійснювати замовлення віддалено, а також відслідковувати весь процес обробки та виконання замовлення.

2.3. Характеристика об'єкта автоматизації

2.3.1. Короткі відомості про об'єкт автоматизації.

Об'єктом автоматизації є діяльність відділу з реалізації продукції ВП «Жданівський цукровий завод». Базовий об'єкт впровадження ТОВ «Цукорагропром».

2.4. Вимоги до системи

2.4.1. Вимоги до системи в цілому.

2.4.1.1. Вимоги до структури і функціонування системи.

2.4.1.1.1. Система повинна мати клієнт-серверну архітектуру, також як сховище даних повинна бути використана реляційна база даних.

Згідно з функціональною структурою веб-додаток повинен бути пов'язаний з учасниками створення та обробки замовлень, а саме:

- клієнтами;
- працівниками виробництва;
- працівниками відділу доставки;
- менеджерами.

2.4.1.1.2. Діагностування функціонування системи в мережі виробництва має передбачати виявлення відхилень від стандартного процес обробки замовлення і порушень в системі роботи сервера, а також помилок користувача веб-додатку, показуючи користувачам відповідні діагностичні повідомлення.

2.4.1.1.3. Розвиток та покращення системи повинні проводитися шляхом збільшення чи заміни функцій, які виконує система, модернізації технічних і програмних засобів по мірі впровадження нових технологій в проект.

Програмно-технічні засоби функціонування системи повинні мати програми з економіко-математичними та статистичними методами, методами моделювання, а також засоби табличного, текстового, графічного відображення даних. Програмна та інформаційна сумісність має забезпечуватися загальносистемним протоколом обміну, використанням проблемно-

орієнтованих пакетів прикладних програм міжмашинних зв'язків і єдиною системою класифікації і кодування.

2.4.1.1.4. Функціонування системи має забезпечувати діалогову мережу (розподілену) обробку даних.

2.4.1.2. Вимоги до чисельності і кваліфікації персоналу.

2.4.1.2.1. Персонал повинен дотримуватися таких вимог:

- пройти навчання і отримати базові навички роботи на ПК;
- дотримуватись технологічних інструкцій при роботі з системою;
- дотримуватись умов експлуатації ПК у відповідності з інструкціями по експлуатації;
- дотримуватись правил зберігання інформації і організації резервних копій БД;
- дотримуватись правил техніки безпеки при роботі з ПК.

2.4.1.2.2. Користувачами системи може виступати клієнт, працівник виробництва, менеджер, працівник відділу доставки. Вхід у систему повинен здійснюватися з застосуванням пароля, який відображає рівень користувача: із правом коригування БД і без права. Залежно від рівня користувач отримує повний чи обмежений функціонал.

2.4.1.3. Показники призначення.

2.4.1.3.1. Відповідно до п. 2.1, показники призначення повинні характеризувати як ступінь так і якість автоматизації планової, інформаційно-облікової і управлінської діяльності приймальної комісії для його оптимального функціонування. Перелік і допустимі значення показників, при яких зберігається цільове призначення системи, повинні бути визначені на стадії техноробочого проектування.

2.4.1.3.2. Система повинна мати можливість налаштування для параметрів об'єкта управління та периферійного обладнання при її модернізації та розвитку, а також зміні процесів та методів організаційного управління.

2.4.1.4. Вимоги до надійності.

2.4.1.4.1. Склад показників надійності системи в цілому

Рівень надійності повинен досягатися узгодженим застосуванням організаційних, організаційно-технічних заходів та програмно-апаратних засобів.

Система повинна відповідати наступним параметрам:

- середній час відновлення Системи після збоїв в роботі апаратного та програмного забезпечення встановлюється Замовником, але не повинен перевищувати 5 % від загального часу роботи системи за заданий календарний період.;

- коефіцієнт готовності повинен становити не менше 0,95 (визначається як результат відношення середнього напрацювання на відмову до суми середнього напрацювання на відмову та середнього часу відновлення);

-- час напрацювання на відмову визначається відповідним показником компонентів апаратного забезпечення Системи, відібраних Розробником та погоджених Замовником.

2.4.1.6. Вимоги з ергономіки та технічної естетики.

При розробці спеціалізованого програмного забезпечення інтерфейс програм повинен бути скомпонований з урахуванням зручності управління та найменшої стомлюваності оператора.

Підсистема формування та візуалізації звітності даних повинна забезпечувати зручний для кінцевого користувача інтерфейс, який відповідає наступним вимогам.

У частині зовнішнього оформлення:

- інтерфейси підсистем повинні бути типізовані;
- повинно бути забезпечено наявність локалізованого (україномовного) інтерфейсу користувача;
- повинен використовуватися шрифт згідно вимог Замовника;
- розмір шрифту повинен бути згідно вимог Замовника;
- колірна палітра повинна бути згідно вимог Замовника;
- в "шапці" звітів повинен використовуватися логотип Замовника.

У частині діалогу з користувачем:

- для найбільш частих операцій повинні бути передбачені «гарячі» клавіші;
- при виникненні помилок в роботі підсистеми на екран монітора повинне виводитися повідомлення українською мовою з назвою помилки та рекомендаціями щодо її усунення.

У частині процедур введення-виведення даних:

- повинна бути можливість багатовимірної аналізу даних в табличному і графічному видах.

До інших підсистем встановлюються наступні вимоги до ергономіки та технічної естетики:

У частині зовнішнього оформлення:

- інтерфейси підсистем повинні бути типізовані.

У частині діалогу з користувачем:

- для найбільш частих операцій повинні бути передбачені «гарячі» клавіші;
- при виникненні помилок в роботі підсистеми на екран монітора повинне виводитися повідомлення українською мовою з назвою помилки та рекомендаціями щодо її усунення.

У частині процедур введення-виведення даних:

повинна бути можливість отримання звітності з моніторингу роботи підсистем.

2.4.1.7. Вимоги по експлуатації, технічного обслуговування, ремонту і зберігання компонентів системи.

2.4.1.7.1. Види обслуговування системи визначаються у відповідності з ДСТУ EN 13306:2019. Загальні вимоги по експлуатації, технічному обслуговуванню і ремонту повинні відповідати ДСТУ 3576-97.

2.4.1.7.2. Для розміщення технічних засобів системи необхідні площі, визначені в ДБН В.2.2-9-2009. При цьому слід дотримуватися вимог, зазначених в експлуатаційній документації. Напруга живлення технічних засобів системи 220/380 В змінного струму, частотою (50±1) Гц. Допустиме відхилення напруги від +10 до -15%, тривалість перерв у живленні не повинна перевищувати 0,001 с.

2.4.1.7.3. Кількість, кваліфікація і режими роботи обслуговуючого персоналу повинні відповідати рекомендаціям, зазначеним в технічних умовах і інструкціях з експлуатації окремих ТЗ.

2.4.1.7.4. Склад, розміщення і умови зберігання компонентів технічних засобів системи визначається рекомендаціями які зазначені в експлуатаційній документації на ці елементи.

2.4.1.7.5. Регламент обслуговування повинен відповідати їх рівню і умовам роботи, щоб у випадку відмови системи забезпечити роботу в аварійному режимі.

2.4.1.8. Вимоги до захисту інформації від несанкціонованого доступу.

Система повинна забезпечувати захист комерційної інформації Замовника та персональних даних користувачів системи. Система інформаційної безпеки повинна задовольняти вимогам міжнародних стандартів серії ISO 27000.

Захист від несанкціонованого доступу може здійснюватися в різних складових інформаційної системи:

- прикладне й системне програмне забезпечення (ПЗ);
- апаратна частина серверів і робочих станцій;
- комунікаційне устаткування й канали зв'язку;
- периметр інформаційної системи.

Для захисту інформації на рівні прикладного й системного ПЗ мають використовуватися:

1. Системи розмежування доступу до інформації. У данному випадку будуть створені різні рівні ієрархії користувачів та адміністраторів.

2. Системи ідентифікації й аутентифікації. Вони використовуються для надання санкціонованого доступу учасникам інформаційної взаємодії до даних, що містяться в Системі. У системі має проводитись реєстрація нового користувача, якому буде наданий закритий ключ і сертифікат відкритого ключа за допомогою який його можна буде ідентифікувати і надати необхідний рівень доступу.

3. Системи аудиту й моніторингу. Мають використовуватись засоби

моніторингу для слідкування за активністю і подіями всередині системи, що дозволить аналізувати внутрішню роботу системи та користувачів, збирати статистику та виявляти підозрілу активність, що забезпечить можливість вияву небезпечних процесів.

4. Системи антивірусного захисту. Використання систем антивірусного захисту як на робочих станціях так і серверній частині дасть змогу захищатися від атак та несанкціонованого доступу на інформаційний периметр.

Для захисту інформації на рівні апаратного забезпечення використовуються:

- апаратні ключі;
- системи сигналізації;
- засоби блокування пристроїв і інтерфейсів вводу-виводу інформації.

Рекомендується створити синтез із систем сигналізації, відеоспостереження, датчиків, апаратних ключів, засобів блокування пристроїв та інтерфейсів вводу-виводу інформації для забезпечення комплексного захисту апаратної частини системи. Це стосується, насамперед, до серверів та мережевого обладнання, що мають мати максимальний ступінь захищеності. Проте, рекомендується провести аналогічні роботи й по забезпеченню захисту робочих станцій.

У комунікаційних системах використовуються наступні засоби мереженого захисту інформації:

- міжмережеві екрани (Firewall) — для блокування атак із зовнішнього середовища (Cisco PIX Firewall, Symantec Enterprise Firewall™, Contivity Secure Gateway і Alteon Switched Firewall від компанії Nortel Networks). Вони управляють проходженням мережевого трафіка відповідно до правил (policies) безпеки. Як правило, міжмережеві екрани встановлюються на вході мережі й розділяють внутрішні (частки) і зовнішні (загального доступу) мережі;

- системи виявлення вторгнень (IDS - Intrusion Detection System) — для виявлення спроб несанкціонованого доступу як ззовні, так і усередині мережі, захисту від атак типу "відмова в обслуговуванні" (Cisco Secure IDS,

Intruder Alert і NetProwler від компанії Symantec). Використовуючи спеціальні механізми, системи виявлення вторгнень здатні запобігати шкідливим діям, що дозволяє значно знизити час простою в результаті атаки й витрати на підтримку працездатності мережі;

- засоби створення віртуальних приватних мереж (VPN - Virtual Private Network) — для організації захищених каналів передачі даних через незахищене середовище (Symantec Enterprise VPN, Cisco IOS VPN, Cisco VPN concentrator). Віртуальні приватні мережі забезпечують прозоре для користувача з'єднання локальних мереж, зберігаючи при цьому конфіденційність і цілісність інформації шляхом її динамічного шифрування;

- засоби аналізу захищеності - для аналізу захищеності корпоративної мережі й виявлення можливих каналів реалізації погроз інформації (Symantec Enterprise Security Manager, Symantec NetRecon). Їхнє застосування дозволяє запобігти можливим атакам на корпоративну мережу, оптимізувати витрати на захист інформації й контролювати поточний стан захищеності мережі.

Захист інформації від її витоку технічними каналами зв'язку забезпечується наступними засобами й заходами:

- використанням екранованого кабелю й прокладкою проводів і кабелів в екранованих конструкціях;
- установкою на лініях зв'язку високочастотних фільтрів;
- побудовою екранованих приміщень ("капсул");
- використанням екранованого устаткування;
- установкою активних систем зашумлення.

З метою оцінки стану технічного захисту інформації, що обробляється або циркулює в автоматизованих системах, комп'ютерних мережах, системах зв'язку, і підготовки обґрунтованих висновків для прийняття відповідних рішень звичайно проводиться експертиза в сфері технічного захисту інформації.

2.4.1.9. Вимоги щодо збереження інформації при аваріях.

В Системі повинне бути забезпечене резервне копіювання та зберігання даних на зовнішніх носіях інформації (стріммери, диски тощо).

Резервні файлові сховища та сервера необхідно розташовувати у географічно віддалених приміщеннях задля уникнення фізичного ураження одразу усієї апаратної частини системи.

2.4.1.10. Вимоги по захисту від впливу зовнішніх діянь.

Система повинна мати можливість функціонування при коливаннях напруги електроживлення в межах від 155 до 265 В ($220 \pm 20\% - 30\%$);

Система повинна мати можливість функціонування в діапазоні допустимих температур навколишнього середовища, встановлених виробником апаратних засобів.

Система повинна мати можливість функціонування в діапазоні допустимих значень вібрацій, встановлених виробником апаратних засобів.

2.4.1.11. Вимоги до патентної чистоти.

При створенні даної системи патентні дослідження не проводяться.

2.4.1.12. Вимоги по стандартизації і уніфікації.

У системі кодування інформації необхідно проводити за світовим класифікатором і стандартом.

2.4.2. Вимоги до функцій.

2.4.2.1. Перелік функцій із зазначенням вхідної та вихідної інформації наведено в таблиці 2.1.

Функції мають забезпечити раціональну організацію роботи користувача на основі безперервної технології: заповнення БД, довідників, формування різнорідних звітів і виконання інших функцій, визначених чинним документом. При цьому пріоритетом є зручність введення та використання інформації користувачем за рахунок формування підказок і меню на екрані монітора.

Таблиця 2.1 – Перелік функцій, вхідної та вихідної інформації

№ п/п	Найменування функції	Вхідна інформація	Вихідна інформація
1	Формування та виведення списку категорій продукції	Таблиця БД "Category"	Форма зі списком категорій та типів продукції

№ п/п	Найменування функції	Вхідна інформація	Вихідна інформація
2	Формування та виведення асортименту продукції	Таблиця БД "Product"	Форма зі списком продукції підприємства
3	Формування та виведення списку працівників, клієнтів їх прав доступу та ролей в системі	Таблиці БД "Users" "UserRoles" "UserClaims"	Форми, списки та методи відображення користувачів та їх ролей в системі
4	Процес створення та обробки замовлення	Таблиці БД "OrderDetails" "OrderHeader" "ShoppingCart"	Форми та списки для створення та відображення нових та вже виконаних замовлень
5	Оплата замовлення	Сформоване замовлення Сервіс "Stripe"	Підтвердження про оплату
6	Зміна та відстеження статусу замовлення	Команда користувача Таблиці БД "OrderDetails" "OrderHeader"	Форми з відображенням поточного статусу виконання замовлення
7	Пошук необхідного замовлення за допомогою фільтрів	Запит користувача з комбінацією фільтрів	Форма з результатами пошуку

2.4.3. Вимоги до видів забезпечення.

2.4.3.1. Програмне забезпечення повинне бути представлено у виді комплексу програм, що забезпечують виконання задач.

2.4.3.2. Вимоги до інформаційного забезпечення (ІЗ).

2.4.3.2.1. Інформаційне забезпечення системи повинно містити дані, достатні для виконання всіх покладених на систему функцій. ІЗ повинно гарантувати раціональну організацію зберігання інформації та доступу до неї.

Заповнення БД інформацією покладається на замовника за методиками і формами, створеними розробниками системи.

Склад, структура і спосіб організації інформації представляються у логічній моделі БД і можуть уточнюватись на етапі технічного проектування.

2.4.3.2.2. Слід передбачити захист даних від руйнування при аваріях і порушеннях у енергоживленні системи — використання резервних копій БД.

2.4.3.3. Вимоги до лінгвістичного забезпечення (ЛЗ).

2.4.3.3.1. Для розробки програмних засобів, які реалізують виконання функцій і забезпечують сервіс користувачів повинні використовуватися мови високого рівня, які забезпечують створення структурних програм, а також мова обраної СУБД для здійснення доступу та маніпулювання даними.

2.4.3.3.2. Організація діалогу користувача до системи має будуватися на наборах меню і підказок, орієнтованих на виконання користувачем функцій. Запити користувача до системи повинні задаватись переважно природною мовою.

2.4.3.4. Вимоги до програмного забезпечення (ПЗ).

2.4.3.4.1. Загальносистемне ПЗ має забезпечувати надійне і якісне виконання функціональних завдань системи. До загальносистемного ПЗ належить:

- операційна система (далі ОС) — Windows 10;
- система управління БД (далі СУБД) — MS SQL Server .

2.4.3.4.2. Загальні вимоги до системного ПЗ:

- мінімальні вимоги до ресурсів технічних засобів (ТЗ);
- максимальна швидкодія;
- повне задоволення потреб функціональних завдань системи.

2.4.3.4.3. Вимоги до ОС:

- мінімальне використання ресурсів комп'ютера для власних потреб, передусім оперативної і дискової пам'яті;
- максимальна швидкодія при управлінні зовнішніми пристроями;
- ОС сервера — Windows Server, 10, ОС клієнта — Windows 7, 10.

2.4.3.4.4. Вимоги до СУБД:

- Мати можливість оновлення, поповнення та розширення БД.
- Забезпечити високу надійність зберігання інформації.
- Видавати повну та вірогідну інформацію на запити.
- Мати засоби, що забезпечують захист БД від несанкціонованого доступу.

2.4.3.4.5. Програмні засоби введення та виведення даних і ведення діалогу повинні забезпечувати:

- виведення необхідних даних на екран;
- Контроль введення даних і повідомлення користувачу про наявність помилок з можливістю їх виправлення під час введення даних;
- керований комп'ютером діалог при введенні даних;
- виведення даних у формі документа за запитом користувача.

2.4.3.4.6. При розробленні спеціального ПЗ слід виконати наступні вимоги:

- використані програми мають бути сумісні між собою та із загальносистемним ПЗ;
- ПЗ має розроблятися засобами об'єктно-орієнтованого програмування;
- забезпечити відповідність інтерфейсу користувача стандартам Windows;
- необхідна модульна структура програм;
- повинна бути передбачена можливість розширення функціоналу;
- ПЗ не повинно залежати від типу зовнішніх пристроїв (принтерів, дисків, сканерів тощо);

– діалог із користувачем повинен проводитись за допомогою клавіатури або миші з поясненням виконання дій і можливістю отримання підказки.

2.4.3.5. Вимоги до технічного забезпечення.

2.4.3.5.1. Технічні засоби системи (табл. 2.2) повинні забезпечувати виконання функцій, перерахованих в таблиці 2.1.

2.4.3.5.2. Засоби обчислювальної техніки повинні забезпечувати обмін інформації в об'ємах, приведених в п. 4.3.2.

Таблиця 2.2 – Вимоги до технічного забезпечення системи

Процесор	2 x Intel Xeon E5-2670V3 чи аналогічний
ОЗП	128ГБ DDR4
Дискові накопичувачі	2x480GB SSD + 4x8TB HDD
Апаратний RAID	Вбудований або Adaptec RAID 5085 чи аналогічний
Мережева карта	10 GbE + 10 GbE
Блок живлення	2x550 Вт
Джерело безперебійного живлення	APC Smart-UPS X 3000VA Rack/Tower LCD

2.4.3.6. Вимоги до метрологічного забезпечення.

Вимоги до метрологічного забезпечення встановлюють:

- попередній перелік вимірювальних каналів;
- вимоги до точності вимірювань параметрів та/або до метрологічних характеристик вимірювальних каналів;
- вимоги до метрологічної сумісності технічних засобів системи;
- перелік керівників та обчислювальних каналів системи, для яких необхідно оцінювати точності характеристики;
- вимоги до метрологічного забезпечення технічних і програмних засобів, що входять до складу вимірювальних каналів системи, засобів вбудованого контролю, метрологічної придатності вимірювальних каналів і засобів вимірювань, що використовуються при налагодженні та випробуваннях системи;
- вид метрологічної атестації (державна або відомча) із зазначенням

порядку її виконання і організацій, які проводять атестацію.

2.4.3.7. Вимоги до організаційного забезпечення.

2.4.3.7.1. Організаційне забезпечення системи розробляється в відповідності з вимогами державного стандарту по АСУП.

2.4.3.7.2. При впровадженні системи не передбачається збільшення штатної чисельності підприємства. Територіальне розміщення робочих місць, на яких буде встановлена система, визначається підприємством.

2.4.3.7.3. До функціонування системи висуваються наступні вимоги:

- визначається список співробітників, які мають доступ до системи (визначає директор);

- контроль і прийняття рішень при аварійних ситуаціях при експлуатації системи здійснює відповідальний за систему.

2.5. Склад і зміст робіт по створенню системи

2.5.1. Стадії створення системи і терміни виконання робіт наведені в таблиці 2.3.

Таблиця 2.3 – Найменування робіт при створенні системи

№ п/п	Найменування робіт	Строки виконання робіт
1	Передпроектне дослідження об'єкта автоматизації	30.04.2021
2	Технічне завдання	10.05.2021
3	Технічний проект	15.05.2021
4	Оформлення документації	25.05.2021

2.6. Порядок контролю і приймання системи

Контроль та приймання послуг здійснює приймальна комісія у складі представників Замовника та Виконавця.

Приймання результатів надання послуг здійснюється в процесі створення Системи згідно з календарним планом надання послуг, який є невід'ємною частиною договору, з підписанням актів здачі-приймання послуг.

Виконавець повинен надати Замовнику всі необхідні документи за договором, а також необхідне програмне забезпечення.

Замовник здійснює приймання послуг згідно умов договору.

Передача програмного забезпечення, документації, інших результатів надання послуг, а також товарів по окремим етапам робіт здійснюється супроводжувальним документами Виконавця та затверджується Замовником.

Здача-приймання наданих послуг здійснюється відповідно до "Програми та методики випробувань", яка розробляється Виконавцем.

По закінченні чергового етапу створення Системи Виконавець надає Замовнику Акт здачі-приймання наданих послуг з доданням до нього комплекту прикладного програмного забезпечення, техноробочої та/або іншої документації, передбаченої умовами договору та технічного завдання на етап, який завершено. Приймання послуг здійснюється поетапно, із підписанням актів здачі-приймання послуг по кожному етапу.

Контроль та приймання робіт проводяться шляхом випробувань, які полягають в перевірці працездатності програмного забезпечення в цілому або його складових частин з використанням проектної та експлуатаційної документації з метою встановлення їх відповідності технічному завданню та проектній документації.

Випробування здійснюються у відповідності з програмою випробувань. Результати оформляються протоколами випробувань.

В разі необхідності проводиться доопрацювання програмного забезпечення та документації.

Випробування проводяться на виробничих площах Замовника.

2.7. Вимоги до складу і змісту робіт із підготовки до введення системи в дію

Для введення в дію замовник виконує ряд робіт із підготовки об'єкта:

- проводить укомплектування технічних засобів;
- організовує навчання користувачів системи роботі на ПК і вивчення інструкції з її експлуатації;
- проводить дослідну експлуатацію і вводить систему в дію.

2.8. Вимоги до документації

Склад і зміст документів, що розроблюються, визначаються згідно з ГОСТ 34.201-89, ГОСТ 34.602-89 та РД 50-34.698-90.

До складу документів обов'язково входять:

- Технічне завдання;
- Керівництво адміністратора;
- Керівництво користувача;
- Програма та методика випробувань.

Конкретний склад та зміст документації можуть бути розширені Виконавцем за згодою Замовника. Кожен примірник виконується державною мовою. Науково-технічна документація надається на паперових та електронних носіях.

Склад технічної документації повинен забезпечувати можливість експлуатації Системи персоналом замовника.

Конкретний склад і зміст документації може бути розширений Виконавцем за погодженням Замовника.

Технічна документація надається на паперових і електронних носіях.

Розробник надає Замовнику вихідні тексти прикладного програмного забезпечення Системи з їх описом.

2.9. Джерела розробки

2.9.1. При розробленні технічного завдання на систему використано наступні документи:

- ДСТУ 3008-2015. Документація. Звіти у сфері науки і техніки. Структура та правила оформлювання;
- ДСТУ 3973–2000 Система розроблення та поставлення продукції на виробництво;

– ДСТУ Б В.2.5–82:2016 Електробезпека в будівлях і спорудах. Вимоги до захисних заходів від ураження електричним струмом.

РОЗДІЛ 3. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ

3.1. Інформаційне забезпечення системи

На основі функціональної моделі діяльності відділу обробки замовлень ВП «Жданівський цукровий завод» (наведеної у Додатку А), розроблювались моделі бази даних. Модель і БД були написані за допомогою Code First. Підхід, званий Code First (спочатку код) передбачає мінімальну участь в проектуванні сутностей бази даних програмістом. Програміст пише код, інше робить Entity і Visual Studio.

Він підходить у випадках якщо головне в проекті - бізнес логіка, а база даних - це спосіб зберігання даних. Або у випадках якщо проект вже написаний, але в якості джерел даних використані списки, масиви, колекції. Code First дозволяє з мінімальними зусиллями змінити проект з використанням баз даних в якості джерел даних замість стандартних колекцій .NET. Розроблена фізична модель складається з наступних таблиць, які є відображенням системного каталогу бази даних в СУБД MS SQL Server 2008 (таблиця 3.1).

Таблиця 3.1 – Структура таблиць бази даних системи

Назва таблиці	Поле	Тип даних
«Category»	Id	int (Ключове поле)
	Name	nvarchar(Max)
«Product»	Id	int(Ключове поле)
	Name	nvarchar(Max)
	Description	nvarchar(Max)
	Image	nvarchar(Max)
	CategoryId	int
	Price	float
«ShoppingCart»	Id	int(Ключове поле)
	UserId	nvarchar(Max)
	ProductId	int

Назва таблиці	Поле	Тип даних
	Count	int
«OrderDetails»	Id OrderId ProductId Count Name Description Price	int(Ключове поле) int int int nvarchar(Max) nvarchar(Max) float
«OrderHeader»	Id UserId Date TotalPrice Status PaymentStatus Comments Name PhoneNumber	int(Ключове поле) nvarchar(Max) datetime2(7) float datetime2(7) nvarchar(Max) nvarchar(Max) nvarchar(Max) nvarchar(Max)
«Roles»	Id Name NormalizedName	nvarchar(450) nvarchar(256) nvarchar(256)
«Users»	Id Name Email PasswordHash PhoneNumber StreetAddress	nvarchar(450) nvarchar(MAX) nvarchar(256) nvarchar(MAX) nvarchar(MAX) nvarchar(MAX)

Назва таблиці	Поле	Тип даних
	City	nvarchar(MAX)
	State	nvarchar(MAX)
	PostalCode	nvarchar(MAX)

Генерація бази даних системи виконується за допомогою підходу CodeFist з структури попередньо створеної в середовищі ERwin. Схема бази даних згенерована в СУБД MS SQL Server наведена у Додатку В.

3.2 Алгоритмізація та реалізація комплексу задач автоматизації

Для реалізації поставлених задач автоматизації було використано одні з найновіших засобів для моделювання та реалізації веб-додатків. Вся розробка базується на стеку технологій .Net.

Для створення веб-додатку було використано технологію Asp.net core. Дана технологія є продовженням розвитку Active Server Page (ASP). Вона представляє універсальну платформу для розробки веб-додатків корпоративного рівня. Asp.net Core пропонує новий підхід до програмування та структури додатку, які дозволяють створювати захищені і легко розширювані веб-додатки[5].

Додаток розроблено за допомогою шаблону проектування MVC 5. Шаблон передбачає поділ додатку на три компоненти:

Контролер (controller) – клас, який забезпечує зв'язок між користувачем і системою, представленням і зберіганням даних. Він отримує введені дані від користувача та опрацьовує їх. В залежності від результатів обробки контролер відправляє користувачу дані, наприклад у вигляді представлення.

Представлення (view) – візуальна частина, або ж користувацький інтерфейс додатку. Зазвичай це html сторінку, яку бачить користувач коли користується сайтом.

Модель (model) – це клас, який описує логіку даних, які використовуються.

Завдяки такому підходу реалізується концепція розділення відповідальності, як результат спрощується процес роботи над окремими

компонентами. Також, при використанні такого підходу, додаток легко тестувати, оскільки ми можемо тестувати клієнтську частину незалежно від логіки. Asp.net містить багато елементів управління та підходів до розробки, які були використані під час виконання даної роботи.

Для зберігання даних було обрано СУБД MS SQL Server.

Microsoft SQL Server 2008 – одна з найкращих пропозицій в області баз даних і аналізу даних для швидкого створення масштабних вирішень електронної комерції, бізнесу і сховищ даних. Вона дозволяє значно скоротити час виходу готових рішень на ринок, одночасно забезпечуючи масштабітність, що відповідає найвищим вимогам. SQL Server включає підтримку мови XML і протоколу HTTP, засобів підвищення швидкодії і доступності, що дозволяють розподілити навантаження і забезпечити безперебійну роботу. Крім того, SQL Server повністю використовує всі можливості операційної системи Windows, включаючи підтримку різних процесорів та різних версій операційних систем[8].

Вся розробка системи проводилась в середовищі Microsoft Visual Studio 2019. Microsoft Visual Studio - лінійка продуктів компанії Microsoft, що включають інтегроване середовище розробки програмного забезпечення та ряд інших інструментальних засобів. IDE MS Visual Studio .NET – це набір інструментів і засобів розробки різного роду застосувань (консольних, Windows, мобільних, Web-застосувань) та сервісів. MS Visual Studio є мультипрограмним середовищем, що підтримує декілька мов програмування, зокрема, C++, C#. Visual Studio надає шаблони для проектів найбільш поширених типів. Використання проектів і їх шаблонів дозволяє користувачеві зосередитися на реалізації окремої функції, в той час як проект буде виконувати загальне управління та завдання побудови.

В процесі створення веб-додатку було розроблено та реалізовано алгоритми для представлення, створення, редагування та видалення даних, а також алгоритми для оплати замовлення за допомогою онлайн сервісу:

Для роботи з базою даних використовувалась бібліотека Entity Framework Core. Рядок під'єднання до бази даних знаходиться в файлі конфігурацій appsettings.json і має вигляд:

```
"DefaultConnection": "Server= .; Database = Sugar Factory; Trusted Connection = True;MultipleActiveResultSets=true".
```

Приклад обробки запиту додавання нового продукту:

```
[HttpPost, ActionName("Create")]
[ValidateAntiForgeryToken]
public async Task<IActionResult> CreatePOST()
{
    if(!ModelState.IsValid)
    {
        return View(ProductVM);
    }

    _db.Product.Add(ProductVM.Product);
    await _db.SaveChangesAsync();

    string webRootPath = _hostingEnvironment.WebRootPath;
    var files = HttpContext.Request.Form.Files;

    var productFromDb = await _db.Product.FindAsync(ProductVM.Product.Id);

    if(files.Count>0)
    {
        var uploads = Path.Combine(webRootPath, "images");
        var extension = Path.GetExtension(files[0].FileName);

        using (var filesStream = new
FileStream(Path.Combine(uploads, ProductVM.Product.Id + extension), FileMode.Create))
        {
            files[0].CopyTo(filesStream);
        }
        productFromDb.Image = @"\images\" + ProductVM.Product.Id + extension;
    }
    else
    {
        var uploads = Path.Combine(webRootPath, @"\images\" + SD.DefaultFoodImage);
        if (!System.IO.File.Exists(webRootPath + @"\images\" + ProductVM.Product.Id
+ ".png"))
        {
            System.IO.File.Copy(uploads, webRootPath + @"\images\" +
ProductVM.Product.Id + ".png");
        }
        productFromDb.Image = @"\images\" + ProductVM.Product.Id + ".png";
    }

    await _db.SaveChangesAsync();

    return RedirectToAction(nameof(Index));
}
```

Приклад обробки запиту виведення списку історії замовлень:

```
[Authorize]
public async Task<IActionResult> OrderHistory(int productPage=1)
{
    var claimsIdentity = (ClaimsIdentity)User.Identity;
```

```

var claim = claimsIdentity.FindFirst(ClaimTypes.NameIdentifier);

OrderListViewModel orderListVM = new OrderListViewModel()
{
    Orders = new List<OrderDetailsViewModel>()
};

List<OrderHeader> OrderHeaderList = await _db.OrderHeader.Include(o =>
o.ApplicationUser).Where(u => u.UserId == claim.Value).ToListAsync();

foreach (OrderHeader item in OrderHeaderList)
{
    OrderDetailsViewModel individual = new OrderDetailsViewModel
    {
        OrderHeader = item,
        OrderDetails = await _db.OrderDetails.Where(o => o.OrderId ==
item.Id).ToListAsync()
    };
    orderListVM.Orders.Add(individual);
}

var count = orderListVM.Orders.Count;
orderListVM.Orders = orderListVM.Orders.OrderByDescending(p => p.OrderHeader.Id)
    .Skip((productPage - 1) * PageSize)
    .Take(PageSize).ToList();

orderListVM.PagingInfo = new PagingInfo
{
    CurrentPage = productPage,
    ItemsPerPage = PageSize,
    TotalItem = count,
    urlParam = "/Customer/Order/OrderHistory?productPage="
};

return View(orderListVM);
}

```

Приклад обробки запиту видалення існуючої категорії продукту:

```

public async Task<IActionResult> Delete(int? id)
{
    if (id == null)
    {
        return NotFound();
    }
    var category = await _db.Category.FindAsync(id);
    if (category == null)
    {
        return NotFound();
    }
    return View(category);
}

[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public async Task<IActionResult> DeleteConfirmed(int? id)
{
    var category = await _db.Category.FindAsync(id);

    if(category ==null)
    {
        return View();
    }
    _db.Category.Remove(category);
    await _db.SaveChangesAsync();
    return RedirectToAction(nameof(Index));
}

```

```
}
```

Приклад обробки запиту підтвердження та оплати нового замовлення:

```
[HttpPost]
[ValidateAntiForgeryToken]
[ActionName("Summary")]
public async Task<IActionResult> SummaryPost(string stripeToken)
{
    var claimsIdentity = (ClaimsIdentity)User.Identity;
    var claim = claimsIdentity.FindFirst(ClaimTypes.NameIdentifier);

    detailCart.listCart = await _db.ShoppingCart.Where(c => c.ApplicationUserId ==
claim.Value).ToListAsync();

    detailCart.OrderHeader.PaymentStatus = SD.PaymentStatusPending;
    detailCart.OrderHeader.OrderDate = DateTime.Now;
    detailCart.OrderHeader.UserId = claim.Value;
    detailCart.OrderHeader.Status = SD.PaymentStatusPending;
    detailCart.OrderHeader.PickUpTime =
Convert.ToDateTime(detailCart.OrderHeader.PickUpDate.ToShortDateString() + " " +
detailCart.OrderHeader.PickUpTime.ToShortTimeString());

    List<OrderDetails> orderDetailsList = new List<OrderDetails>();
    _db.OrderHeader.Add(detailCart.OrderHeader);
    await _db.SaveChangesAsync();

    detailCart.OrderHeader.OrderTotalOriginal = 0;

    foreach (var item in detailCart.listCart)
    {
        item.Product = await _db.Product.FirstOrDefaultAsync(m => m.Id ==
item.ProductId);
        OrderDetails orderDetails = new OrderDetails
        {
            ProductId = item.ProductId,
            OrderId = detailCart.OrderHeader.Id,
            Description = item.Product.Description,
            Name = item.Product.Name,
            Price = item.Product.Price,
            Count = item.Count
        };
        detailCart.OrderHeader.OrderTotalOriginal += orderDetails.Count *
orderDetails.Price;
        _db.OrderDetails.Add(orderDetails);
    }

    _db.ShoppingCart.RemoveRange(detailCart.listCart);
    await _db.SaveChangesAsync();

    var options = new ChargeCreateOptions
    {
        Amount = Convert.ToInt32(detailCart.OrderHeader.OrderTotalOriginal * 100),
        Currency = "uah",
        Description = "Order ID : " + detailCart.OrderHeader.Id,
        Source = stripeToken
    };

    var service = new ChargeService();
    Charge charge = service.Create(options);

    if (charge.BalanceTransactionId == null)
```

```

    {
        detailCart.OrderHeader.PaymentStatus = SD.PaymentStatusRejected;
    }
    else
    {
        detailCart.OrderHeader.TransactionId = charge.BalanceTransactionId;
    }

    if (charge.Status.ToLower() == "succeeded")
    {
        detailCart.OrderHeader.PaymentStatus = SD.PaymentStatusApproved;
        detailCart.OrderHeader.Status = SD.StatusSubmitted;
    }
    else
    {
        detailCart.OrderHeader.PaymentStatus = SD.PaymentStatusRejected;
    }

    await _db.SaveChangesAsync();
    return RedirectToAction("Index", "Home");
}
}

```

3.3. Інструкція користувача

Так як сам додаток ще не підключений до головного сайту холдингу, <https://astartaholding.com>, тому для початку роботи потрібно запустити проект.

Для початку роботи з системою користувачу необхідно пройти авторизацію (пункт меню «Вхід в систему»).

Після входу у систему на екран виводиться сторінка за замовчуванням, яка представляє собою список продуктів, які виготовляє ВП «Жданівський цукровий завод» ТОВ «Цукорагропром». Також користувачу доступна сторінка «Про нас» де наведена детальна інформація про підприємство та адреси його знаходження.

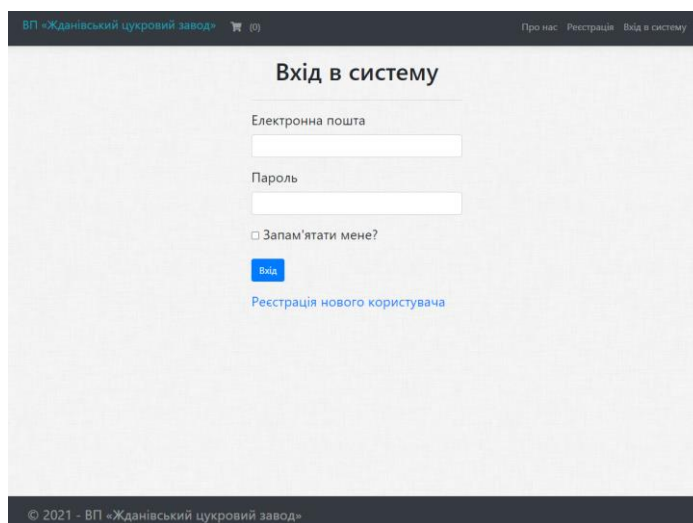


Рис. 3.1. Сторінка для входу в систему

У випадку, якщо користувач ще немає доступу до системи, у нього є можливість створити новий акаунт обравши пункт меню «Реєстрація».

Користувачі системи мають акаунти різних типів, в залежності від виконуваних ними функцій, це:

- клієнти;
- працівники виробництва;
- працівники відділу доставки продукції;
- менеджери.

Кожен з акаунтів містить всі функції для клієнта, тобто створювати, редагувати та відслідковувати замовлення.

Для створення нового замовлення клієнт повинен перейти на головну сторінку, де розташований список доступної для замовлення продукції.

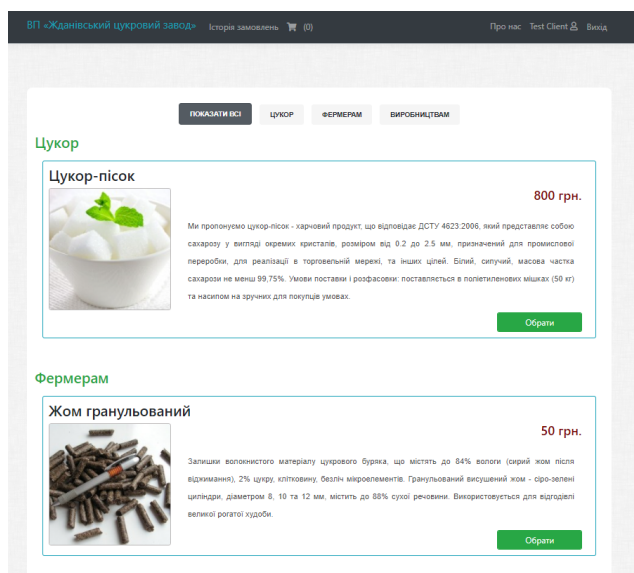


Рис. 3.2. Список доступної продукції

Для оформлення замовлення потрібно обрати необхідну продукцію та перейти за посиланням на кнопці «Обрати». У наступному вікні, вказати необхідну кількість продукції та обрати опцію «Додати до замовлення».

Вся додана користувачем продукція буде додана до кошика і сформує список замовлення.

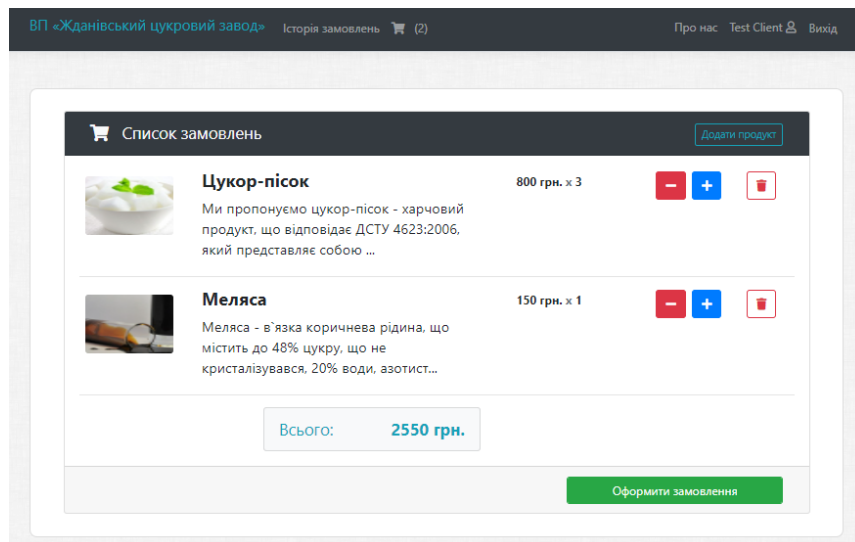


Рис. 3.3. Список замовлення

Процес оформлення замовлення включає в себе надання додаткових даних про клієнта: П.І.Б. того хто буде приймати замовлення, телефон клієнта (заповнюються автоматично з даних акаунту, при необхідності можна змінити) та додаткових параметрів замовлення, таких як, бажана дата та час доставки та примітки стосовно замовлення.

Наступним кроком є оплата оформленого замовлення, для цього вказуємо електронну пошту на яку отримаємо квитанцію про оплату та дані платіжної картки.

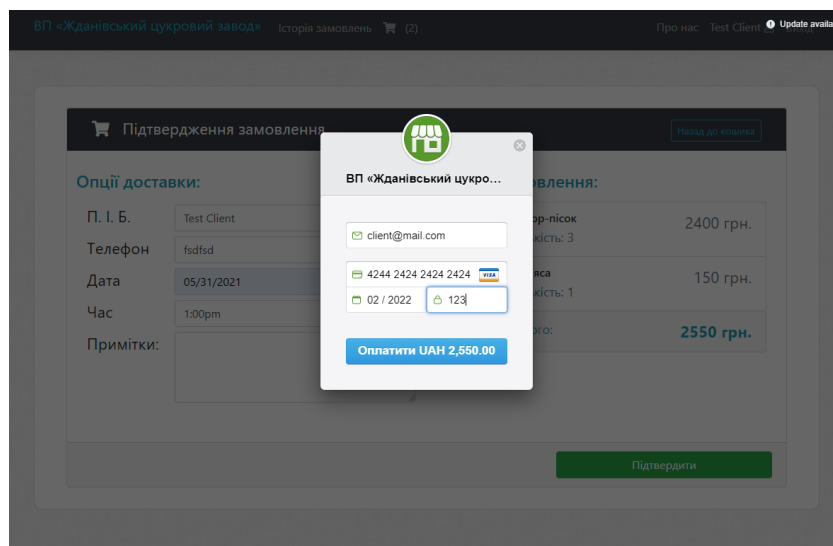


Рис. 3.4. Процес оплати замовлення

Після проведення оплати, здійснені замовлення можна побачити перейшовши за пунктом меню історія замовлень. На створеному замовленні є дві кнопки: «Деталі» - детальний опис здійсненого замовлення, а також інформація

про оплату, «Статус» - інформація про поточний статус виконання замовлення (змінюється в режимі реального часу).

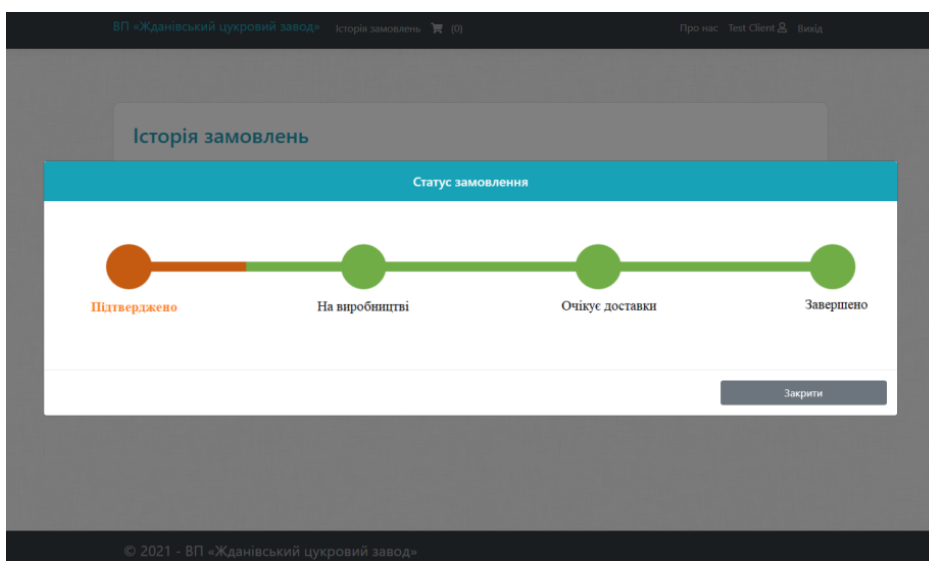


Рис. 3.5. Статус виконання замовлення

Акаунт працівника виробництва відрізняється наявністю нового пункту меню «Замовлення => Виробництво». На вказаній сторінці працівник вказує статус виконання замовлення. Якщо виробництво замовлення розпочато працівник обирає опцію «Почати виробництво». Після завершення всіх виробничих процесів працівник обирає опцію «Замовлення виконано».

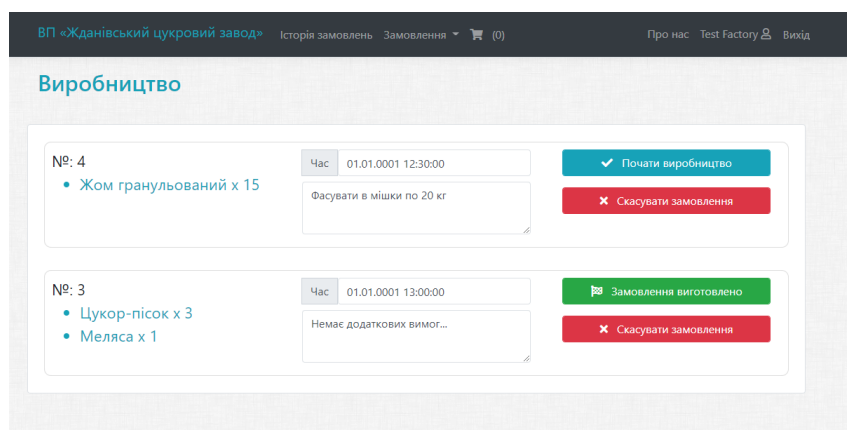


Рис. 3.6. Сторінка «Виробництво»

Акаунт працівника доставки включає в себе опцію меню «Замовлення => Доставка». На цій сторінці працівнику будуть показані виготовлені замовлення, які необхідно доставити, на сторінці доступні фільтри по П.І.Б., телефону та адресі електронної пошти клієнта. Кур'єр має можливість переглянути деталі замовлення, а також вказати що замовлення було доставлено.

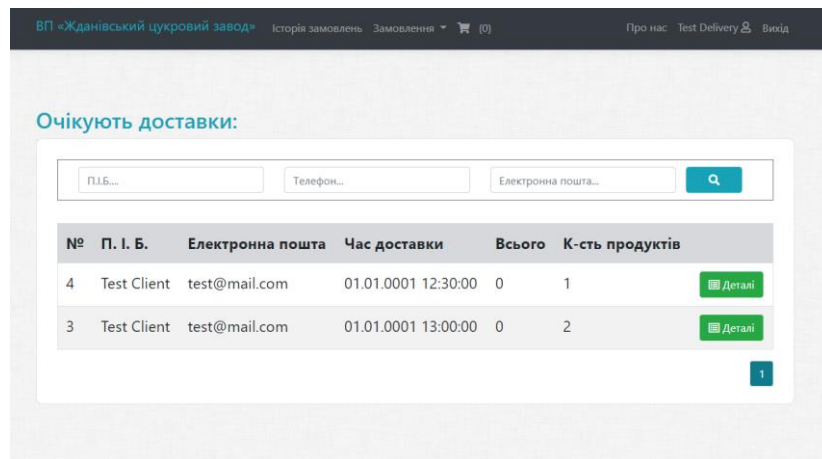


Рис. 3.7. Сторінка «Доставка»

Акаунт менеджера включає в себе всі вище перелічені можливості, а також надає можливість налаштувати роботу всієї системи. Для цього в нього доступний пункт меню «Налаштування» та підпункти «Категорії», «Продукти», «Користувачі».

Підпункт «Категорії» відповідає за створення, редагування, видалення категорій товарів. Підпункт «Продукти» дозволяє додавати нову продукцію до системи, а також редагувати чи видаляти її. Підпункт користувачі дозволяє створювати нових працівників (акаунти працівника виробництва, відділу доставки, або ж менеджера), а також блокувати доступ до системи вибраним акаунтам.

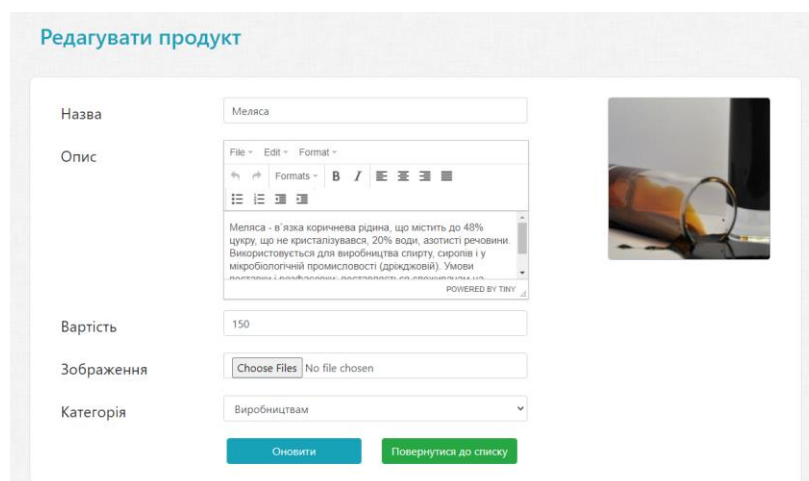


Рис. 3.8. Сторінка редагування продукції

Вихід з програми виконується за допомогою опції меню «Вихід».

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

4.1 Загальні питання охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівниками, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» визначається, що охорона праці - система збереження життя, здоров'я і працездатності працівників у процесі трудової діяльності, що включає в себе правові, соціально-економічні, організаційні, технічні, санітарно-гігієнічні, лікувально-профілактичні, реабілітаційні та інші заходи. При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Повітря забруднюється шкідливими хімічними речовинами антропогенного походження за рахунок деструкції полімерних матеріалів, які використовуються для обробки приміщень та обладнання. Неправильна організація робочого місця сприяє загальному і локальній напрузі м'язів шиї, тулуба, верхніх кінцівок, викривлення хребта і розвитку остеохондрозу.

На всіх підприємствах, в установах, організаціях повинні створюватися служби охорони праці. Згідно з Типовим положенням на службу охорони праці підприємства покладаються такі завдання:

- відпрацювання ефективної системи управління охорони праці на підприємстві та сприяння вдосконаленню діяльності в цьому напрямку кожного структурного підрозділу і кожного працівника;
- забезпечення професійної підтримки рішень роботодавця щодо цих питань;
- організація проведення профілактичних заходів, спрямованих на усунення шкідливих і небезпечних виробничих факторів, запобігання нещасним

випадкам на виробництві, професійним захворюванням та іншим випадкам загрози життю або здоров'ю працівників;

- вивчення та сприяння впровадженню у виробництво досягнень науки і техніки, прогресивних і безпечних технологій, сучасних засобів колективного та індивідуального захисту працівників;
- контроль за додержанням працівниками вимог законів та інших нормативно-правових актів з охорони праці, положень (за наявності) галузевої угоди, розділу «Охорона праці» колективного договору та актів з охорони праці, що діють у межах підприємства.
- інформування та надання роз'яснень працівникам підприємства з питань охорони праці.

У даній дипломній роботі питання охорони праці розглядаються стосовно підприємства, де виконується безпосередньо робота за напрямом диплому та за умовами праці які визначені завданням.

4.2. Структура управління охороною праці на підприємстві

Система управління охороною праці (СУОП) є комплексом дій з підготовки, прийняття та реалізації рішень з метою виконання організаційних, технічних, санітарно-гігієнічних і лікувально-профілактичних заходів.

Створення СУОП – це, в першу чергу, визначення керівництвом підприємства політики в галузі охорони праці, а саме працезахоронної політики стосовно зобов'язань, намірів та заходів в сфері охорони праці, визначення мети роботи СУОП, об'єкта та органів управління, завдань і заходів щодо охорони праці, функцій і методів управління, побудови організаційної структури управління, створення ефективно діючих систем мотивації, контролю та обліку, аналізу, аудиту і моніторингу СУОП, оцінка ризиків, розробка та впровадження комп'ютеризованих технологій управління, забезпечення відповідного контролю за ефективністю роботи СУОП, створення умов для її надійного функціонування і подальшого вдосконалення.

4.3 Загальна характеристика приміщення та робочого місця

Приміщення, в якому проводяться розробка та тестування розробленої системи наведені у таблиці 4.1.

Таблиця 4.1 – Загальна характеристика умов праці

Номер за журналом групи	Шкідливі та небезпечні фактори на робочому місці	Джерела утворення небезпек	Примітка (данні наведені для технічного відділу)
1	Електрична напруга вище 127 В; Шум; Випромінювання – електромагнітні, радіаційні, теплові; Статична електрика; Іонізація повітря; Пожежна безпека у приміщенні; Не якісне освітлення.	Кондиціонер, 6-ЕОМ, Сервери, Світильники (лампи)	Розміри приміщення (м) : Довжина – 10; Ширина – 5; Висота – 3. Кількість працюючих – 6.

Згідно з НПАОП 0.00-1.28-2010 в лабораторії може перебувати 6 працівників. Мінімальна припустима площа приміщення на 1 людину повинна складати від 6,0 м². Висновок - за умовами завдання це виконується повністю. В приміщенні відсутні умови, які можуть створювати підвищену або особливо підвищену небезпеку, так як воно відноситься до класу звичайних приміщень (згідно ПУЕ[4]). Джерелом живлення є трифазна мережа напруги 380/220 В з глухо заземленою нейтралю, з частотою 50 Гц (згідно НПАОП 0.00-1.28-2010). За пожежною вибухонебезпекою приміщення лабораторії відноситься до класу В. У таблиці 4.2 наведена загальна характеристика приміщення щодо вибухопожежної небезпеки та за важкістю робіт.

Таблиця 4.2 - Загальна характеристика приміщення щодо вибухопожежної небезпеки та за важкістю робіт

Характеристика приміщень за вибухопожежною категорією та класом зони	Загальна характеристика приміщення	Категорія за важкістю робіт згідно ГН 3.3.5-8.6.6.1-2002
--	------------------------------------	--

<p>В – пожеже небезпечна, Клас П-ІІ</p>	<p>Звичайне, без ознак хімічного забруднення та нормальної вологості за санітарними вимогами</p>	<p>1адо 139 Вт/м² 1б 140-174 Вт/м² Клас умов праці - <u>Оптимальний</u> <u>Окремі показники напруженості</u> <u>трудового процесу</u> – ступінь ризику для власного життя виключено; ступінь відповідальності за безпеку інших осіб – виключено. Ступінь відповідальності за результат своєї діяльності. Значущість помилки - допустимий: (напруженість праці середнього ступеня), а саме – несе відповідальність за функціональну якість допоміжних завдань. Вимагає додаткових зусиль з боку керівництва, спостереження за екраном відео терміналу (годин на зміну) 2-3.</p>
---	--	--

4.4 Метеорологічні параметри робочої зони

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючої на організм людини з'єднанням температури, вологості, швидкості переміщення повітря.

В даному приміщенні проводяться роботи, що виконуються сидячи і не потребують фізичного напруження, то для нього відповідає категорія робіт Іа. Отже оптимальні значення для температури, відносної вологості й рухливості повітря відповідають і зазначені в таблиці 4.3:

Таблиця 4.3 - Оптимальні параметри мікроклімату

Категорія робіт	Період року	Температура, °С	Відносна вологість, %	Швидкість руху повітря, м/с
Легка (Іб)	холодний	21-23	40-60	не більше 0,1
Легка (Іб)	теплий	22-24	40-60	не більше 0,2

Дане приміщення має бути обладнане системами опалення,

кондиціонування повітря, або припливно-витяжною вентиляцією. У приміщеннях на робочих місцях мають забезпечуватись оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря у відповідності до. Рівні позитивних і негативних іонів у повітрі мають відповідати.

Для забезпечення оптимальних параметрів мікроклімату в приміщенні проводяться перерви в роботі співробітників, з метою його провітрювання. Існують спеціальні системи кондиціонування, які забезпечують підтримання в приміщенні балансу оптимальних параметрів мікроклімату.

Контроль параметрів мікроклімату в холодний і теплий період року здійснюється не менше 3-х разів на зміну (на початку, середині, в кінці).

4.5 Освітлення

Основним завданням виробничого освітлення є підтримання на робочому місці освітленості, що відповідає характеру зорової роботи. Збільшення освітленості робочої поверхні покращує видимість об'єктів за рахунок підвищення їх яскравості, збільшує швидкість розрізнення деталей, що позначається на зростанні продуктивності праці.

Виробниче освітлення повинно забезпечувати відсутність в поле зору працюючого різких тіней. Наявність різких тіней спотворює розміри і форми об'єктів відмінності і тим самим підвищує стомлюваність, знижує продуктивність праці. Особливо шкідливі рухомі тіні, які можуть призвести до травм.

Для поліпшення видимості об'єктів у полі зору працюючого має бути відсутня пряма і відбита блесність. Блесність обмежують зменшенням яскравості джерел світла, правильним вибором захисного кута світильника, збільшенням висоти підвісу світильників, правильним напрямком світлового потоку на робочу поверхню, а також зміною кута нахилу робочої поверхні.

При організації виробничого освітлення слід вибрати необхідний спектральний склад світлового потоку. Ця вимога особливо важливо для забезпечення правильної передачі кольору. Оптимальний спектральний склад забезпечує природне освітлення.

Освітлювальні установки повинні бути зручні і прості в експлуатації, довговічні, відповідати вимогам естетики, електробезпеки, а також не повинні бути причиною виникнення вибуху або пожежі.

4.6 Шум та вібрація у робочому приміщенні

Допустимий рівень вібрацій на робочому місці:

- Для 1 ступеня шкідливості до 3 дБ;
- Для 2-3 - 1-6 дБ;
- Для 3 - більше 6 дБ.

Допустимі значення октавних рівнів звукового тиску, рівнів звуку на робочих місцях у приміщеннях з комп'ютерною технікою слід приймати згідно з таблицею 4.4.

Таблиця 4.4 – Норми параметрів значення октавних рівнів

Призначення приміщення й умови	Рівні звукового тиску, дБ, в октавних дугах частот зсередньгеометричними частотами, Гц										Рівні звуку, дБА
	16	31,5	63	125	250	500	1000	2000	4000	8000	
Приміщення без робот иЕОМ	-	-	63	52	45	39	35	32	30	28	40
Приміщення при роботі ЕОМ	85	75	67	57	49	44	40	37	35	33	45

У приміщеннях з ЕОМ коректований рівень звукової потужності не перевищує 45 дБА.

Для зниження шуму на шляху його поширення передбачається розміщення в приміщенні штучних поглиначів. Віброізоляція здійснюється за допомогою спеціальної прокладки під системний блок, який послаблює передачу вібрацій робочого столу.

Таблиця 4.5 - Гранично допустимі рівні вібрацій на робочому місці, дБ

Нормований параметр	Середньгеометричні частоти октанових дуг, Гц						Кориговані і еквівалентні рівні, дБ
	2	4	8	16	31,5	63	
Віброшвидкість	79	73	67	67	67	67	72
Віброприскорення	25	25	25	31	37	43	30

Основними заходами боротьби з шумом згідно з ГОСТ 12.1.029-80 ССБТ є ліквідація або ослаблення джерела шуму шляхом застосування звукопоглинаючих матеріалів у конструкціях механізмів, використання коштів звукопоглинання і раціональна планування виробничого приміщення.

Вібрація на робочих місцях, яка створюється ЕОМ, не вище значень, які представлені в таблиці 4.5., приміщення що розглядається відповідає нормам.

4.7. Електробезпека

Електробезпека - це комплекс заходів і дій, орієнтованих на мінімізацію ризику заподіяння шкоди електрострумом, а також магнітні й електричні поля або статичною електрикою. Якщо заподіяння такої шкоди з об'єктивних причин запобігти не вдалося, система заходів з електробезпеки зобов'язана давати максимальне скорочення його наслідків. Склад цього комплексу залежить від особливостей технологічного циклу конкретного підприємства і встановленого обладнання.

Ключові правила електробезпеки при експлуатації електроустановок вимагають, щоб застосування такої апаратури в ході технологічного процесу проводилося відповідно до рекомендацій і вказівок виробника. Це означає, що співробітники, які виробляють його експлуатацію, зобов'язані бути ознайомлені з інструкцією або керівництвом, яке описує правила роботи з цим типом техніки.

Крім цього, для безпечної роботи апаратури важливо стежити за виконанням наступних умов:

своєчасне виконання профілактичних випробувань і планового технічного обслуговування, що забезпечує безперербійне функціонування обладнання протягом усього терміну його служби;

невідкладна зупинка і виконання поточного та капітального ремонту при виході з ладу вузлів, компонентів або деталей конструкції техніки, оскільки продовження експлуатації з такою проблемою може стати причиною більш істотною поломки або аварії із заподіянням значної шкоди матеріальним активам, а також здоров'ю та життю працівників;

допуск до роботи на обладнанні тільки тих співробітників, які пройшли профільну підготовку і отримали навички експлуатації такої техніки.

Виконання перерахованих вимог контролюється уповноваженими органами. Державний нагляд за дотриманням вимог електробезпеки проводиться інспекторами в порядку, встановленому законодавством.

Ще одна вимога, яке обов'язково в рамках системи заходів щодо забезпечення електробезпеки, - це надання працівникам захисних засобів, які скорочують ризик ураження електричним струмом. Порядок їх надання, номенклатура конкретних ЗІЗ, норми їх видачі прописані в окремих нормативних документах - це міжгалузеві правила з електробезпеки, а також Наказ Мінпраці від 09.12.2014 N 997н. Роботодавець має право збільшувати норми або частоту видачі захисних засобів, проте погіршувати становище працівників у цьому відношенні в порівнянні з положеннями чинного законодавства він не має права.

Зараз основні правила електробезпеки в частині організації виконання робіт включають такі вимоги:

оформлення необхідної документації, що регламентує порядок виконання операцій, включаючи особливо складні дії, що вимагають оформлення додаткових документів, наприклад, нарядів-допусків;

надання працівникам регламентованих перерв, дотримання режимів праці та відпочинку;

здійснення нагляду за виконанням роботи співробітниками старшим майстром або іншим безпосереднім керівником;

підбір електрообладнання, що відповідає поточним технологічним завданням підприємства;

розробка посадових інструкцій для працівників, зайнятих в експлуатації електроустановок, в яких ретельно прописані їхні права, обов'язки і порядок виконання робіт;

виконання вказівок інспекторів державного нагляду.

Забезпечення електробезпеки на підприємстві - це зона спільної відповідальності роботодавця і працівника, тому останній зобов'язаний так само ретельно ставитися до виконання своїх завдань в цій області, щоб забезпечити результат такої роботи.

4.8. Ергономічні вимоги до робочого місця

Для забезпечення нормальних умов праці, норма виробничого приміщення, для одного працівника має складати не менше 15 м³, площа приміщення вигороджених стінами або глухими перегородками не менше 4,5 м² при висоті стелі не менше 3,3 м. У відповідності з будівельними нормами і правилами, підлоги в приміщенні повинні бути виконані з негорючих матеріалів і задовольняти гігієнічним, технологічним і експлуатаційним вимогам.

Важливу роль відіграє планування робочого місця, яка повинна задовольняти вимогам зручності та економії енергії і часу, раціонального використання зайнятого простору і зручності обслуговування пристроїв ЕОМ, дотримання правил охорони праці. При організації робочого місця взяті до уваги дані антропометрії. Робочий простір організовано на основі даних про охоплення рук людини - 40 см. Близькій зоні відповідає область, що охоплюється рукою з притиснутим до тулуба ліктем, далекій зоні - область витягнутої руки. При цьому руху працівника виходять такими, що групи його м'язів навантажені рівномірно, а зайві непродуктивні руху усунені. Сидіти від дисплея потрібно не ближче 60-70 см (приблизно на відстані витягнутої руки). Не має бути яскравих джерел

світла за екраном монітора (вікон, джерел штучного освітлення і т.п.). Крім того, яскраві джерела світла не повинні створювати відблисків на екрані монітора.

ВИСНОВКИ

В кваліфікаційній роботі досліджено роботу ВП «Жданівський цукровий завод» ТОВ «Цукорагропром» і розроблено веб-додаток для покращення ефективності його діяльності.

Основною метою розроблення веб-додатку є автоматизація процесу реалізації продукції підприємством, відстеження етапів виконання замовлення, забезпечення швидкого доступу до історії замовлення та статусу його виконання, а також надання замовнику можливості віддалено в режимі реального часу отримувати інформацію про продукцію, яку виробляє підприємство, створювати нові замовлення та здійснювати їх оплату.

В процесі розробки здійснено системний аналіз діяльності відділу продажів підприємства та розроблена функціональна модель.

Функціональний аналіз підприємства дозволив виявити основні проблеми та задачі автоматизації, на основі яких було розроблене технічне завдання на проектування веб-додатку. Технічне завдання містить основні вимоги до структури, функцій та принципів роботи веб-додатку.

Приведено розрахунок економічного ефекту від впровадження розробки на підприємстві та визначено термін її окупності, який складає 1 рік і 1 місяць.

Розробка веб-додатку реалізована з використанням сучасних підходів до проектування веб-систем, а саме ASP.Net Core та шаблону проектування MVC. Системою управління базою даних є MS SQL Server 2008, який керується за допомогою бібліотеки EntityFramework Core. Інтерфейс реалізовано мовою програмування C#.

Впровадження веб-додатку надасть можливість замовникам здійснювати замовлення продукції цукрового заводу віддалено та відслідковувати весь процес його виконання в режимі реального часу, працівники виробництва отримають можливість бачити нові замовлення та змінювати статус їх виконання незалежно, в режимі реального часу.

БІБЛОГРАФІЧНИЙ СПИСОК

1. Марка Д. Методология структурного анализа и проектирования / Д. Марка, К. Макгоуэн. – М. : МетаТехнология, 1993.
2. Маклаков С. В. Создание информационных систем с ALLFusion Modeling Suite / С. В. Маклаков. – М. : Диалог – МИФИ, 2005. – 432 с.
3. Кватрани Т. Rational Rose 2000 и UML. Визуальное моделирование / Т. Кватрани. – М. : ДМК Пресс, 2001. – 176 с.
4. Гайна Г. А. Основы проектування баз даних : навч. посіб. – К. : КНУБА, 2005. – 204 с.
5. Microsoft Developer Network [Електронний ресурс]. Режим доступу: <https://msdn.microsoft.com/ru-ru>.
6. Троелсен Е.В. Язык программирования С# 5.0 и платформа .NET 4.5. Навчальний посібник. / Троелсен Е.В; пер. с англ. Ю.Р. Артеменко – Москва: Диалектика, 2013. - 1306 с.
7. Metsker S. T. Design Patterns in C#; Moscow, 2011. - 480 с.
8. С. Байдачный, Д. Маленко, Ю. Лозинский. SQL Server 2005: новые возможности для разработчиков – М.: СОЛОН – Пресс, 2006 – 208с.
9. ДСТУ 3008:2015. Документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. – К.: ДП «УкрНДНЦ», 2016. – 31 с.
10. ДСТУ ISO/IEC 12207:2016 (ISO/IEC 12207:2008, IDT). Інженерія систем і програмного забезпечення. Процеси життєвого циклу програмного забезпечення.
11. ДСТУ 1.0:2003. СТУ 1.0:2003. Національна стандартизація. Основні положення.
12. ДСТУ ISO 6309:2007. Пожежна безпека. Загальні вимоги.
13. НПАОП 0.00-1.28-10 Правила охорони праці під час експлуатації електронно-обчислювальних машин/ Зареєстровано в Міністерстві юстиції України 19 квітня 2010 р. за N 293/17588.
14. ДСТУ ISO/IEC 12207:2014. Інженерія систем і програмного забезпечення. Процеси життєвого циклу програмного забезпечення.

ДОДАТКИ

ДОДАТОК А. Організаційна структура ВП «Жданівський цукровий завод»



Рис. А.1. Організаційна структура ВП «Жданівський цукровий завод»

ДОДАТОК Б. Функціональна модель діяльності відділу продажів

Перший рівень декомпозиції

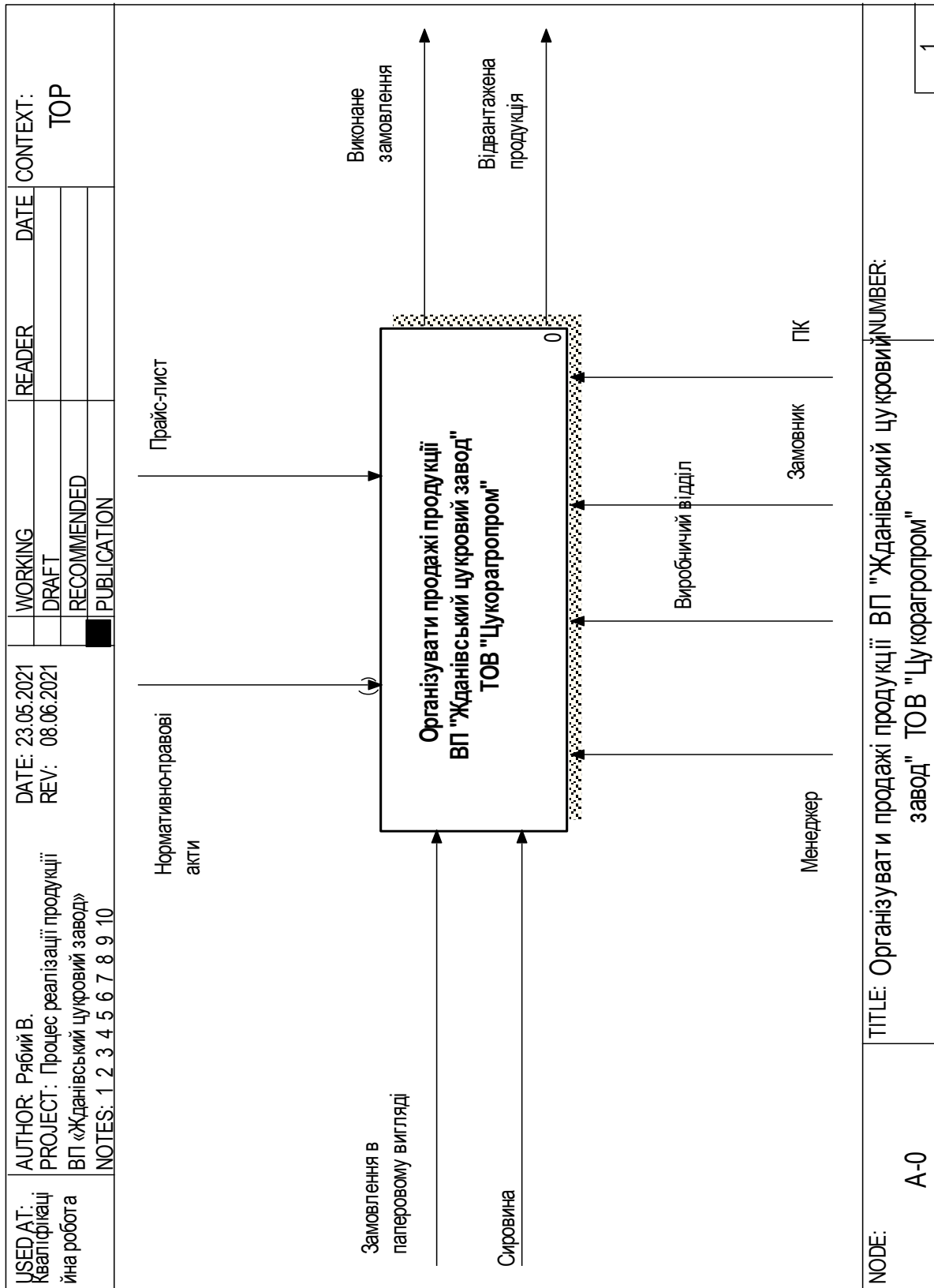


Рис. Б.1. Функціональна модель діяльності відділу продажів

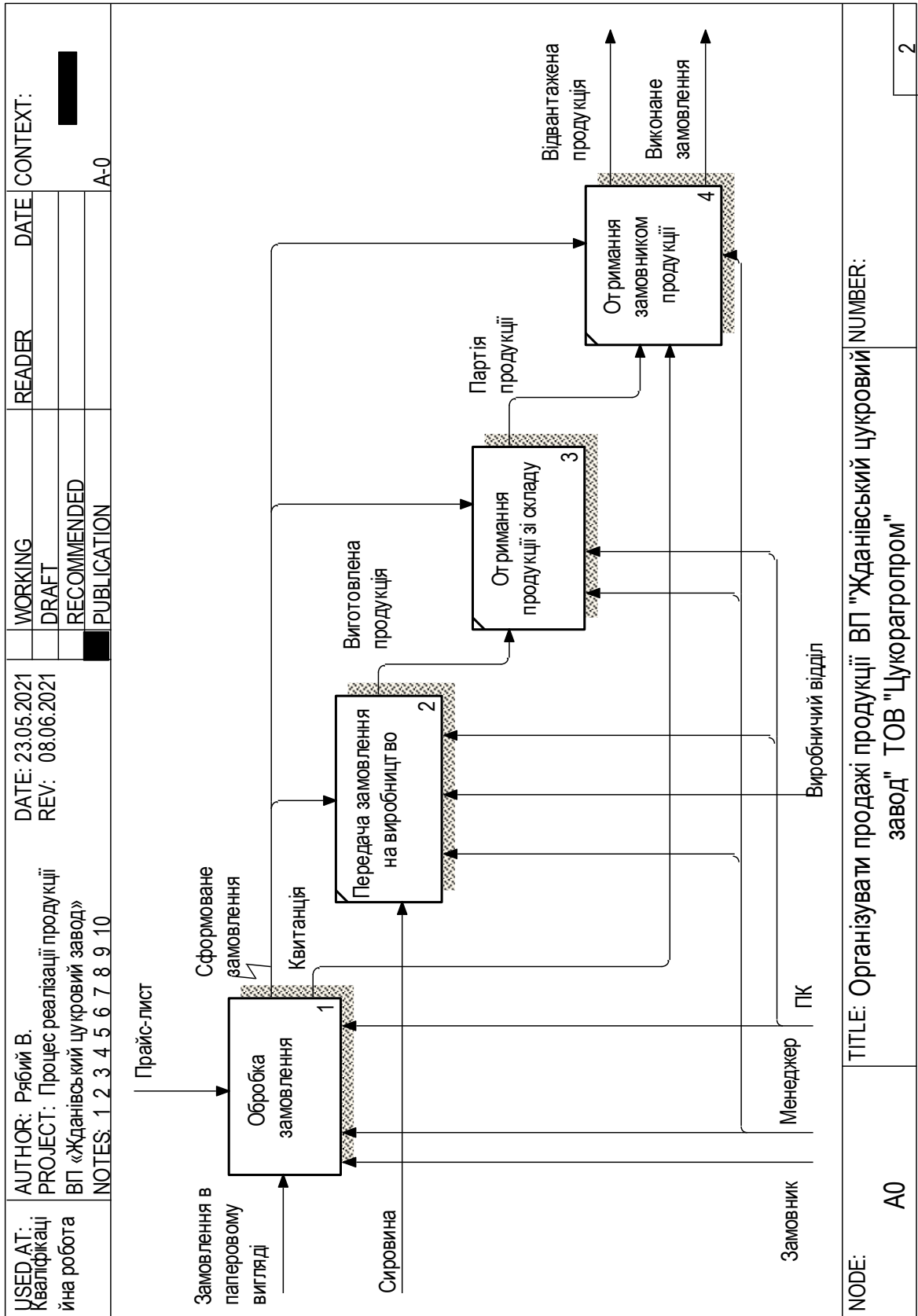
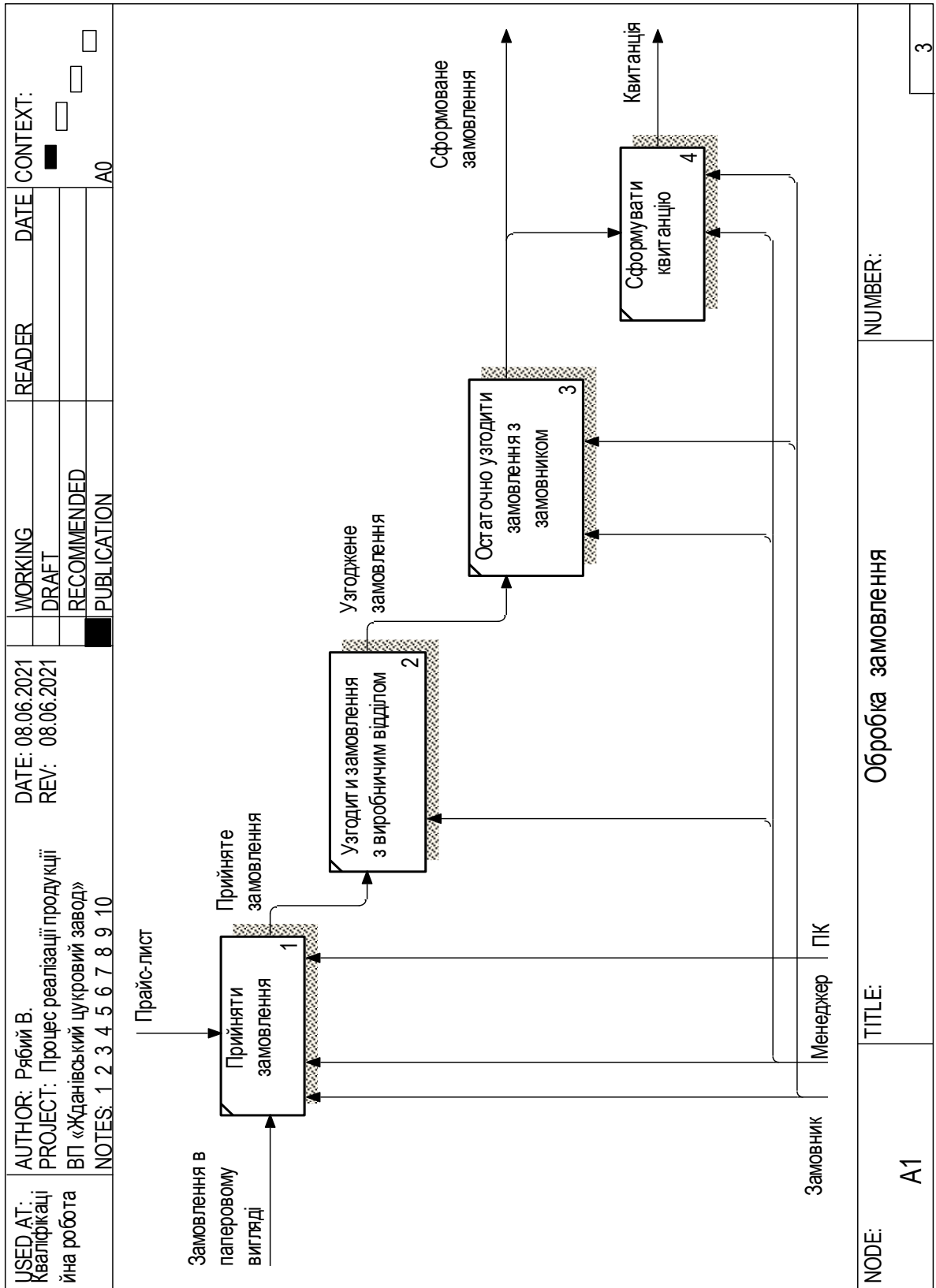


Рис. Б.2. Перший рівень декомпозиції



NUMBER:

Обробка замовлення

TITLE:

A1

3

Рис.Б.3 Другий рівень декомпозиції

Функціональна модель ТО-ВЕ

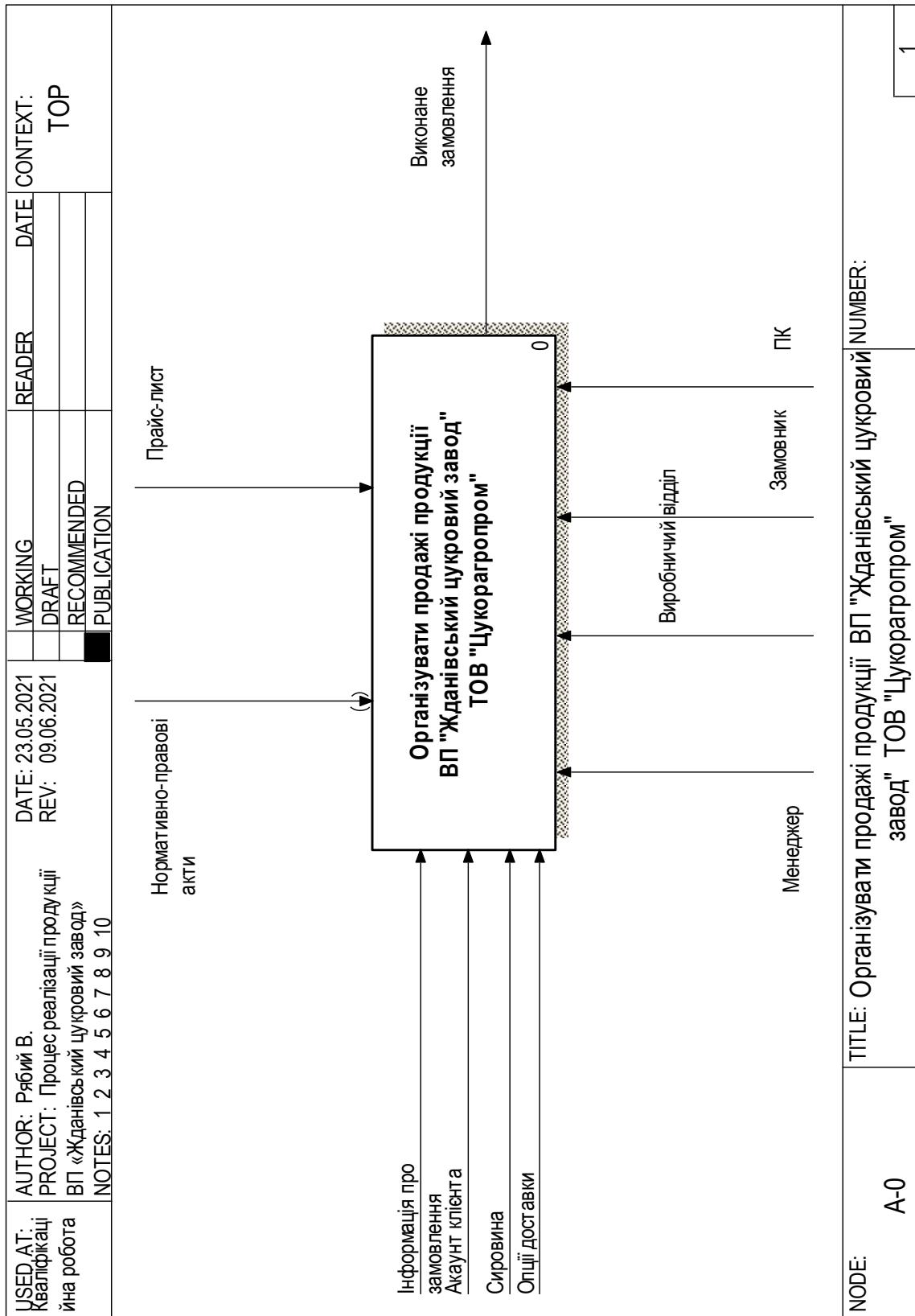


Рис. Б.3. Функціональна модель ТО-ВЕ

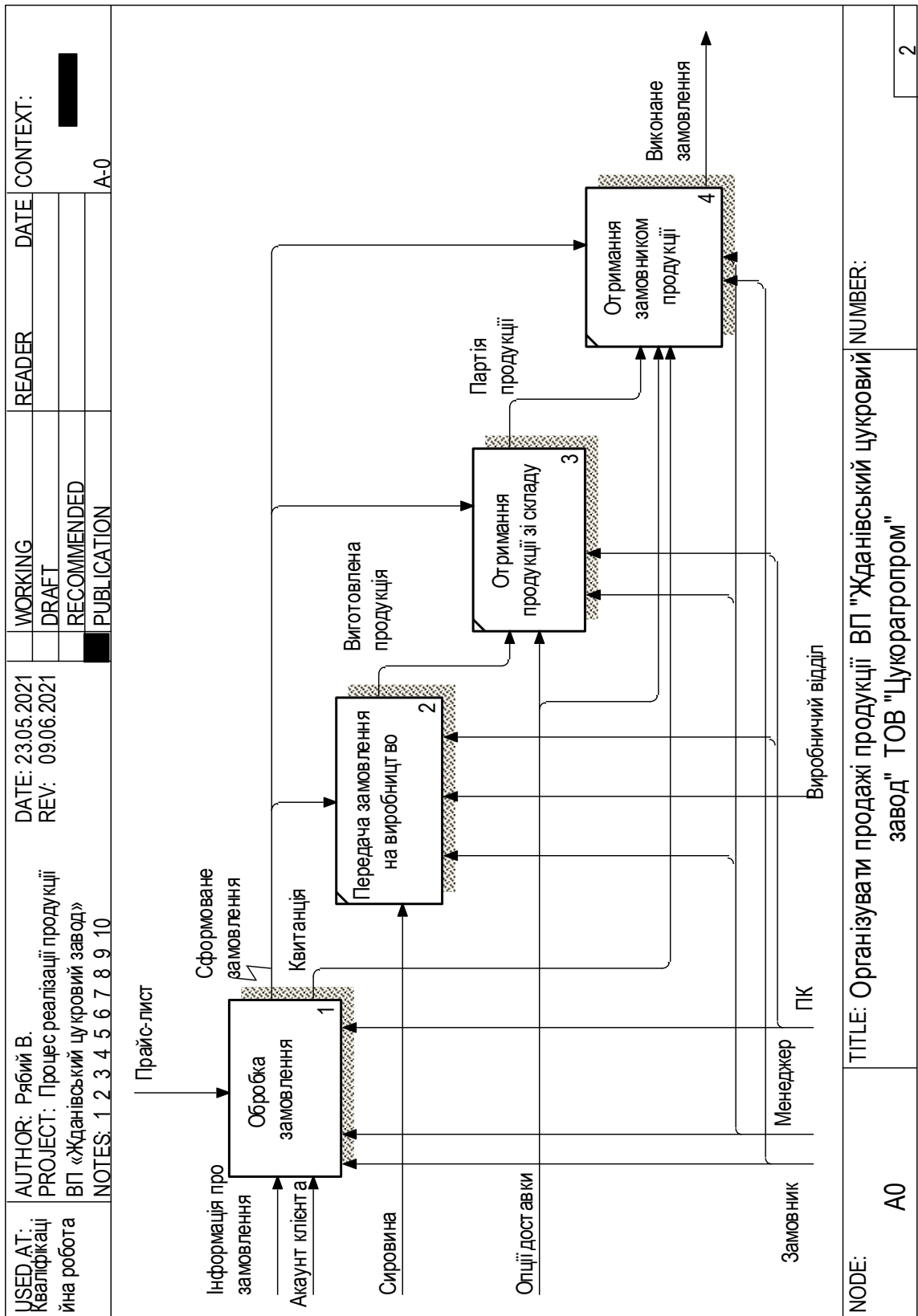


Рис. Б.4. Перший рівень декомпозиції

ДОДАТОК Д. Схема бази даних в MS SQL Server

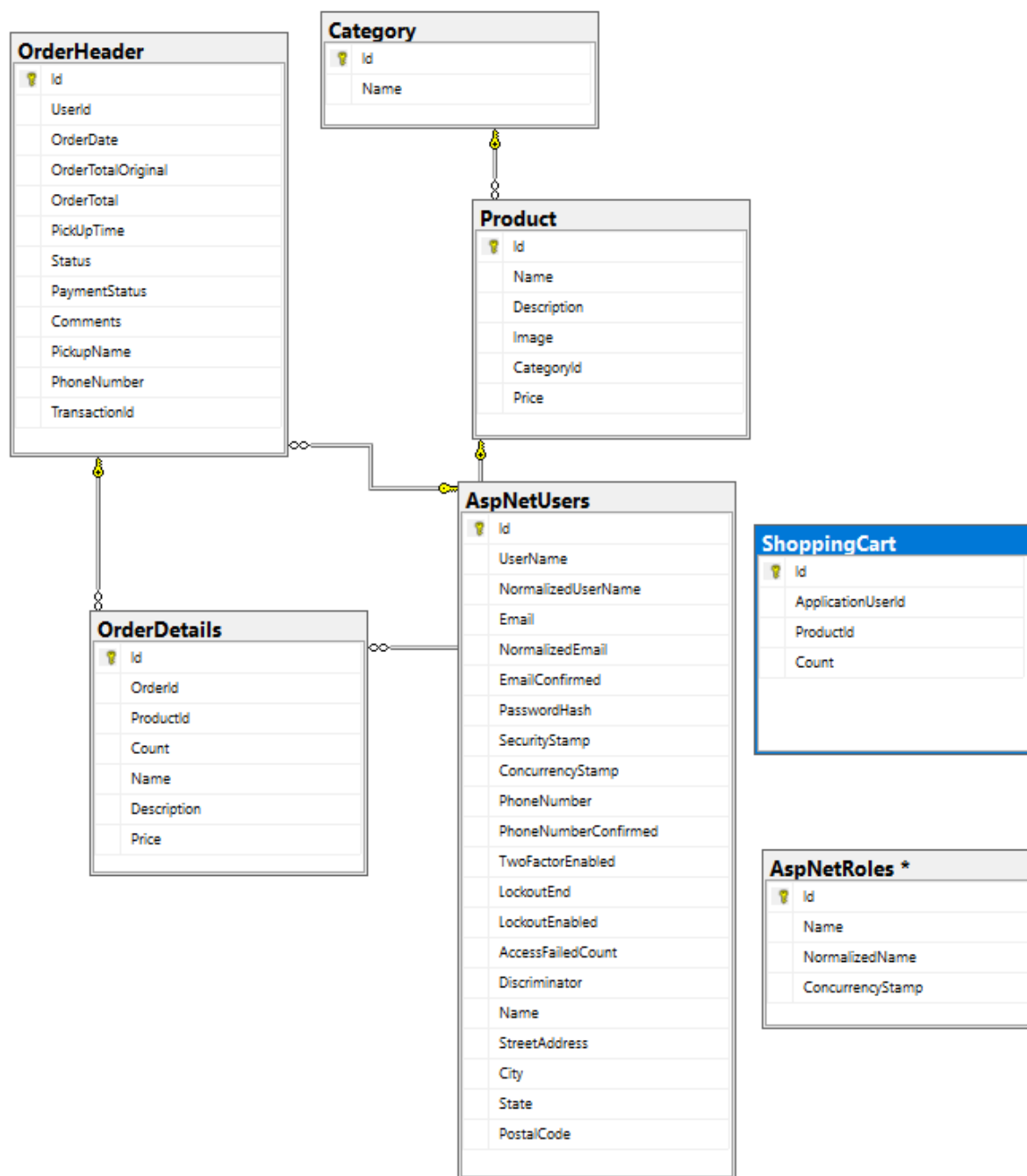


Рис. Б.1. Схема бази даних

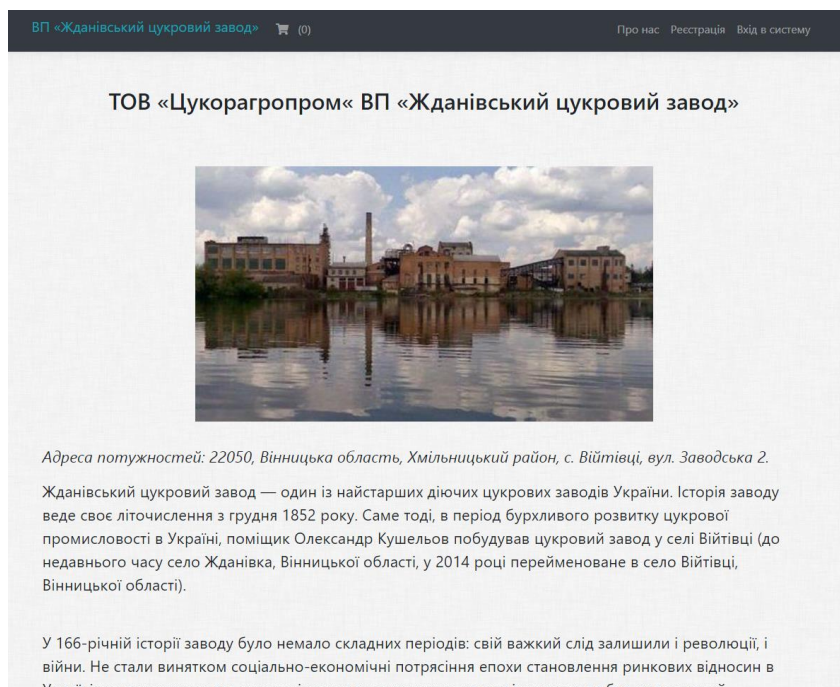


Рис. Г.1. Сторінка «Про нас»

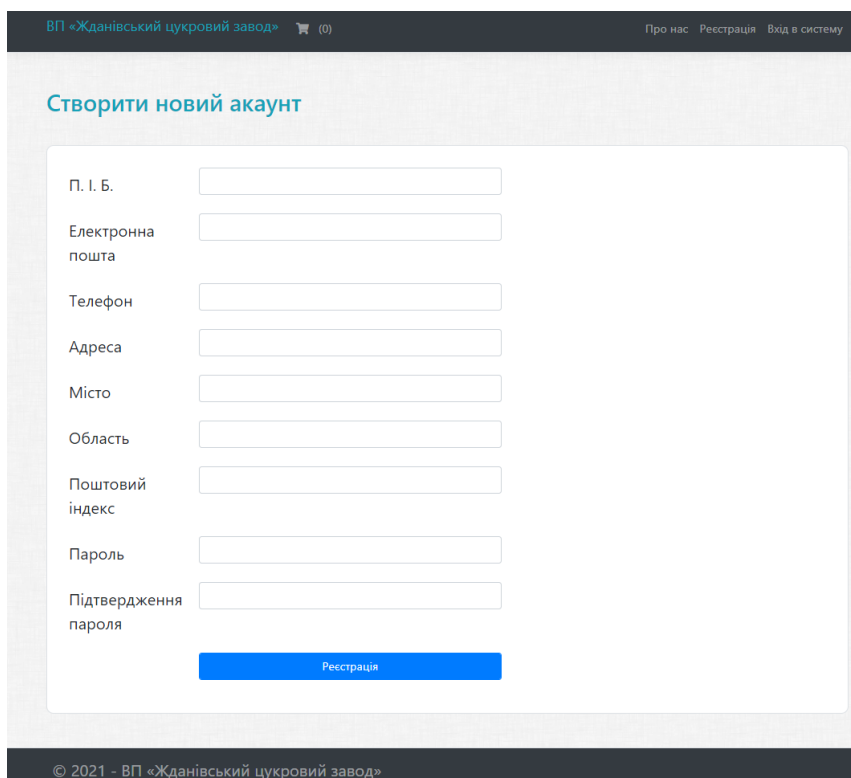


Рис. Г.2. Сторінка реєстрації нового користувача

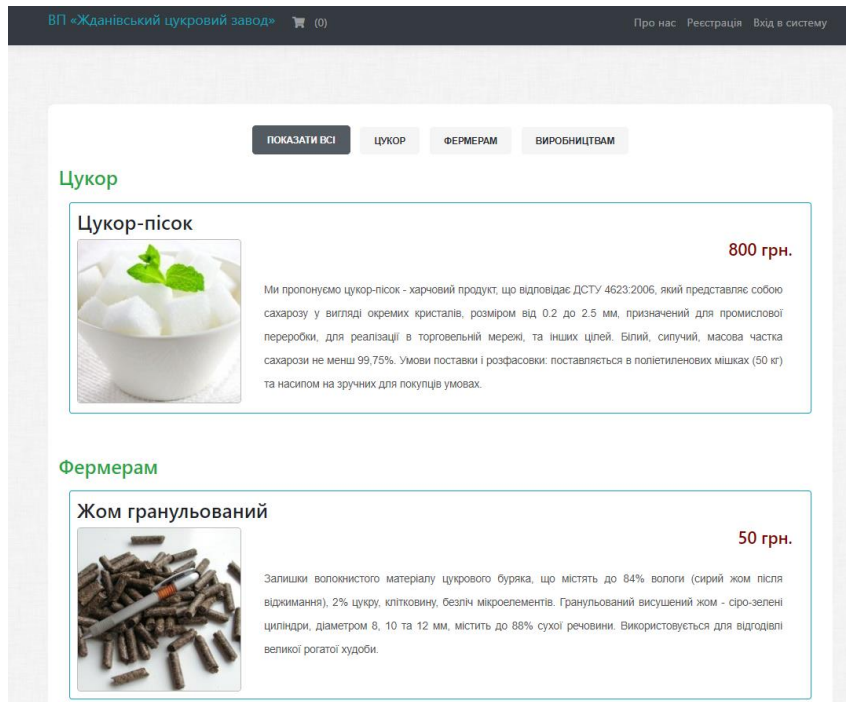


Рис. Г.3. Сторінка «Продукція виробництва»

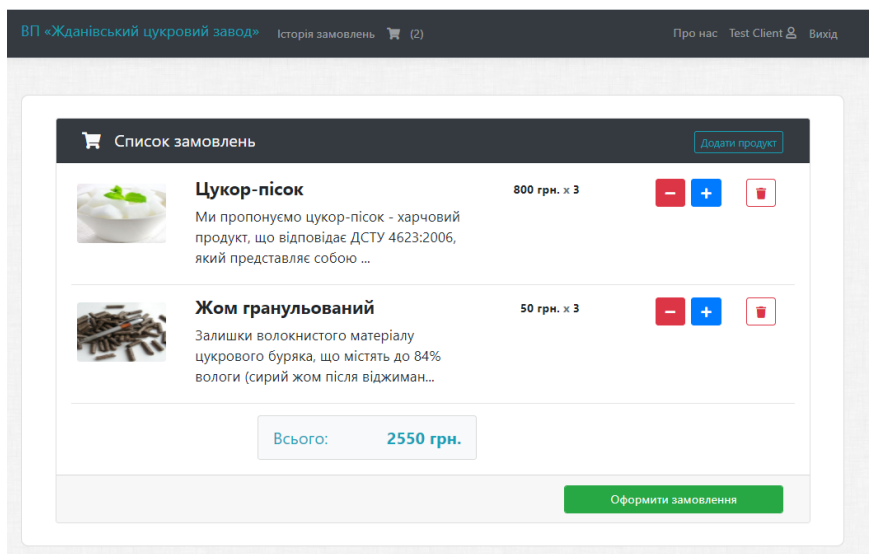


Рис. Г.4. Поточне замовлення

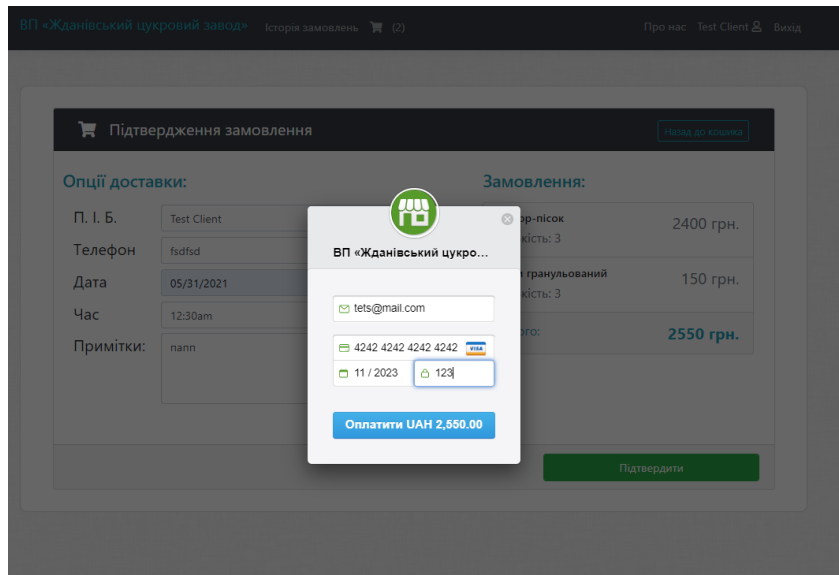


Рис. Г.5. Оплата замовлення

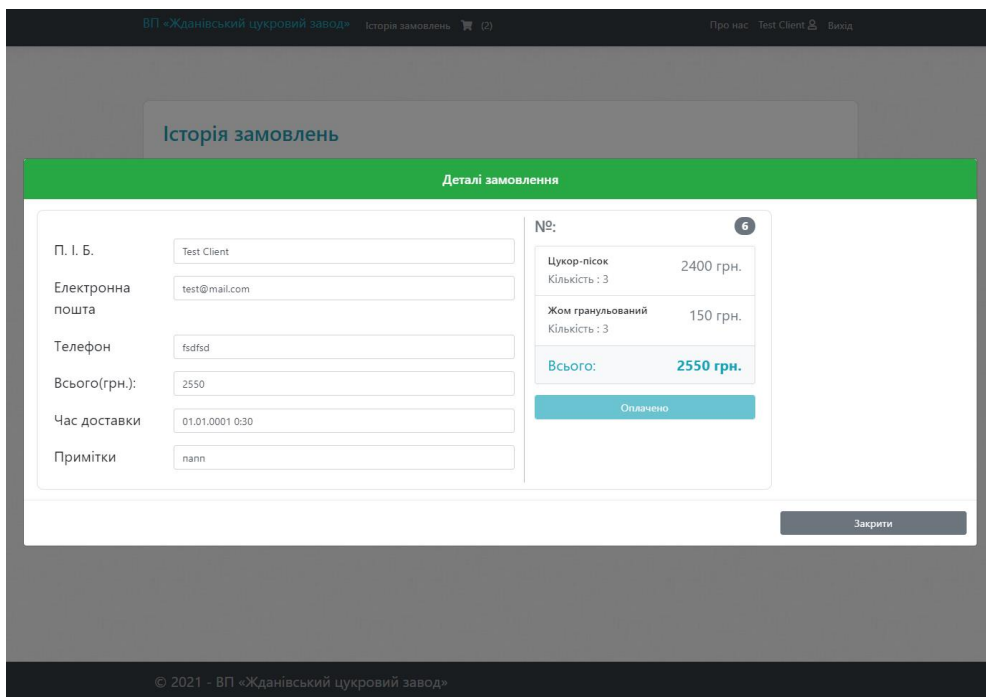


Рис. Г.6. Деталі замовлення

Модуль «Продукти»:

```

public class ProductController : Controller
{
    private readonly ApplicationDbContext _db;
    private readonly IWebHostEnvironment _hostingEnvironment;

    [BindProperty]
    public ProductViewModel ProductVM { get; set; }

    public ProductController(ApplicationDbContext db, IWebHostEnvironment
hostingEnvironment)
    {
        _db = db;
        _hostingEnvironment = hostingEnvironment;
        ProductVM = new ProductViewModel()
        {
            Category = _db.Category,
            Product = new Models.Product()
        };
    }
    public async Task<IActionResult> Index()
    {
        var products = await _db.Product.Include(m=>m.Category).ToListAsync();
        return View(products);
    }
    public IActionResult Create()
    {
        return View(ProductVM);
    }
    [HttpPost, ActionName("Create")]
    [ValidateAntiForgeryToken]
    public async Task<IActionResult> CreatePOST()
    {
        if(!ModelState.IsValid)
        {
            return View(ProductVM);
        }

        _db.Product.Add(ProductVM.Product);
        await _db.SaveChangesAsync();

        string webRootPath = _hostingEnvironment.WebRootPath;
        var files = HttpContext.Request.Form.Files;

        var productFromDb = await _db.Product.FindAsync(ProductVM.Product.Id);

        if(files.Count>0)
        {
            var uploads = Path.Combine(webRootPath, "images");
            var extension = Path.GetExtension(files[0].FileName);

            using (var filesStream = new
FileStream(Path.Combine(uploads,ProductVM.Product.Id + extension), FileMode.Create))
            {
                files[0].CopyTo(filesStream);
            }
            productFromDb.Image = @"\images\" + ProductVM.Product.Id + extension;
        }
        else
        {
            var uploads = Path.Combine(webRootPath, @"\images\" + SD.DefaultFoodImage);

```

```

        if (!System.IO.File.Exists(webRootPath + @"\images\" + ProductVM.Product.Id
+ ".png"))
        {
            System.IO.File.Copy(uploads, webRootPath + @"\images\" +
ProductVM.Product.Id + ".png");
        }
        productFromDb.Image = @"\images\" + ProductVM.Product.Id + ".png";
    }

    await _db.SaveChangesAsync();

    return RedirectToAction(nameof(Index));
}
public async Task<IActionResult> Edit(int? id)
{
    if(id==null)
    {
        return NotFound();
    }

    ProductVM.Product = await _db.Product.Include(m =>
m.Category).SingleOrDefaultAsync(m => m.Id == id);

    if(ProductVM.Product ==null)
    {
        return NotFound();
    }
    return View(ProductVM);
}

[HttpPost, ActionName("Edit")]
[ValidateAntiForgeryToken]
public async Task<IActionResult> EditPOST(int? id)
{
    if(id==null)
    {
        return NotFound();
    }

    if (!ModelState.IsValid)
    {
        return View(ProductVM);
    }
    string webRootPath = _hostingEnvironment.WebRootPath;
    var files = HttpContext.Request.Form.Files;

    var productFromDb = await _db.Product.FindAsync(ProductVM.Product.Id);

    if (files.Count > 0)
    {
        var uploads = Path.Combine(webRootPath, "images");
        var extension_new = Path.GetExtension(files[0].FileName);
        var imagePath = Path.Combine(webRootPath,
productFromDb.Image.TrimStart('\'));

        if(System.IO.File.Exists(imagePath))
        {
            System.IO.File.Delete(imagePath);
        }
        using (var filesStream = new FileStream(Path.Combine(uploads,
ProductVM.Product.Id + extension_new), FileMode.Create))
        {
            files[0].CopyTo(filesStream);
        }
        productFromDb.Image = @"\images\" + ProductVM.Product.Id + extension_new;
    }
}

```

```

    }

    productFromDb.Name = ProductVM.Product.Name;
    productFromDb.Description = ProductVM.Product.Description;
    productFromDb.Price = ProductVM.Product.Price;
    productFromDb.CategoryId = ProductVM.Product.CategoryId;

    await _db.SaveChangesAsync();

    return RedirectToAction(nameof(Index));
}
public async Task<IActionResult> Details(int? id)
{
    if (id == null)
    {
        return NotFound();
    }

    ProductVM.Product = await _db.Product.Include(m =>
m.Category).SingleOrDefaultAsync(m => m.Id == id);

    if (ProductVM.Product == null)
    {
        return NotFound();
    }

    return View(ProductVM);
}
public async Task<IActionResult> Delete(int? id)
{
    if (id == null)
    {
        return NotFound();
    }
    ProductVM.Product = await _db.Product.Include(m =>
m.Category).SingleOrDefaultAsync(m => m.Id == id);

    if (ProductVM.Product == null)
    {
        return NotFound();
    }
    return View(ProductVM);
}
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public async Task<IActionResult> DeleteConfirmed(int id)
{
    string webRootPath = _hostingEnvironment.WebRootPath;
    Product product = await _db.Product.FindAsync(id);

    if (product != null)
    {
        var imagePath = Path.Combine(webRootPath, product.Image.TrimStart('\'));

        if (System.IO.File.Exists(imagePath))
        {
            System.IO.File.Delete(imagePath);
        }
        _db.Product.Remove(product);
        await _db.SaveChangesAsync();
    }
    return RedirectToAction(nameof(Index));
}
}

```

Модуль «Замовлення»:

```
public class OrderController : Controller
{
    private readonly ApplicationDbContext _db;
    private int PageSize = 5;
    public OrderController(ApplicationDbContext db)
    {
        _db = db;
    }

    [Authorize]
    public async Task<IActionResult> Confirm(int id)
    {
        var claimsIdentity = (ClaimsIdentity)User.Identity;
        var claim = claimsIdentity.FindFirst(ClaimTypes.NameIdentifier);

        OrderDetailsViewModel orderDetailsViewModel = new OrderDetailsViewModel()
        {
            OrderHeader = await _db.OrderHeader.Include(o =>
o.ApplicationUser).FirstOrDefaultAsync(o => o.Id == id && o.UserId == claim.Value),
            OrderDetails = await _db.OrderDetails.Where(o => o.OrderId ==
id).ToListAsync()
        };

        return View(orderDetailsViewModel);
    }

    public IActionResult Index()
    {
        return View();
    }

    public IActionResult GetOrderStatus(int Id)
    {
        return PartialView("_OrderStatus", _db.OrderHeader.Where(m => m.Id ==
Id).FirstOrDefault().Status);
    }

    [Authorize]
    public async Task<IActionResult> OrderHistory(int productPage=1)
    {
        var claimsIdentity = (ClaimsIdentity)User.Identity;
        var claim = claimsIdentity.FindFirst(ClaimTypes.NameIdentifier);

        OrderListViewModel orderListVM = new OrderListViewModel()
        {
            Orders = new List<OrderDetailsViewModel>()
        };

        List<OrderHeader> OrderHeaderList = await _db.OrderHeader.Include(o =>
o.ApplicationUser).Where(u => u.UserId == claim.Value).ToListAsync();

        foreach (OrderHeader item in OrderHeaderList)
        {
            OrderDetailsViewModel individual = new OrderDetailsViewModel
            {
                OrderHeader = item,
                OrderDetails = await _db.OrderDetails.Where(o => o.OrderId ==
item.Id).ToListAsync()
            };
            orderListVM.Orders.Add(individual);
        }
    }
}
```

```

var count = orderListVM.Orders.Count;
orderListVM.Orders = orderListVM.Orders.OrderByDescending(p => p.OrderHeader.Id)
    .Skip((productPage - 1) * PageSize)
    .Take(PageSize).ToList();

orderListVM.PagingInfo = new PagingInfo
{
    CurrentPage = productPage,
    ItemsPerPage = PageSize,
    TotalItem = count,
    urlParam = "/Customer/Order/OrderHistory?productPage="
};

return View(orderListVM);
}

[Authorize(Roles = SD.FactoryUser + "," + SD.ManagerUser)]
public async Task<IActionResult> ManageOrder(int productPage = 1)
{
    List<OrderDetailsViewModel> orderDetailsVM = new List<OrderDetailsViewModel>();

    List<OrderHeader> OrderHeaderList = await
_db.OrderHeader.Where(o=>o.Status==SD.StatusSubmitted ||
o.Status==SD.StatusInProgress).OrderByDescending(u=>u.PickUpTime).ToListAsync();

    foreach (OrderHeader item in OrderHeaderList)
    {
        OrderDetailsViewModel individual = new OrderDetailsViewModel
        {
            OrderHeader = item,
            OrderDetails = await _db.OrderDetails.Where(o => o.OrderId ==
item.Id).ToListAsync()
        };
        orderDetailsVM.Add(individual);
    }

    return View(orderDetailsVM.OrderBy(o=>o.OrderHeader.PickUpTime).ToList());
}

public async Task<IActionResult> GetOrderDetails(int Id)
{
    OrderDetailsViewModel orderDetailsViewModel = new OrderDetailsViewModel()
    {
        OrderHeader = await _db.OrderHeader.Include(e1 =>
e1.ApplicationUser).FirstOrDefaultAsync(m => m.Id == Id),
        OrderDetails = await _db.OrderDetails.Where(m => m.OrderId ==
Id).ToListAsync()
    };
    return PartialView("_IndividualOrderDetails", orderDetailsViewModel);
}

[Authorize(Roles =SD.FactoryUser + ","+ SD.ManagerUser)]
public async Task<IActionResult> OrderPrepare(int OrderId)
{
    OrderHeader orderHeader = await _db.OrderHeader.FindAsync(OrderId);
    orderHeader.Status = SD.StatusInProgress;
    await _db.SaveChangesAsync();
    return RedirectToAction("ManageOrder", "Order");
}

[Authorize(Roles = SD.FactoryUser + "," + SD.ManagerUser)]
public async Task<IActionResult> OrderReady(int OrderId)

```

```

    {
        OrderHeader orderHeader = await _db.OrderHeader.FindAsync(OrderId);
        orderHeader.Status = SD.StatusReady;
        await _db.SaveChangesAsync();

        return RedirectToAction("ManageOrder", "Order");
    }
    [Authorize(Roles = SD.FactoryUser + "," + SD.ManagerUser)]
    public async Task<IActionResult> OrderCancel(int OrderId)
    {
        OrderHeader orderHeader = await _db.OrderHeader.FindAsync(OrderId);
        orderHeader.Status = SD.StatusCancelled;
        await _db.SaveChangesAsync();

        return RedirectToAction("ManageOrder", "Order");
    }

    [Authorize]
    public async Task<IActionResult> OrderPickup(int productPage = 1, string
    searchEmail=null, string searchPhone = null, string searchName = null)
    {
        OrderListViewModel orderListVM = new OrderListViewModel()
        {
            Orders = new List<OrderDetailsViewModel>()
        };

        StringBuilder param = new StringBuilder();
        param.Append("/Customer/Order/OrderPickup?productPage=");
        param.Append("&searchName=");
        if(searchName!=null)
        {
            param.Append(searchName);
        }
        param.Append("&searchEmail=");
        if (searchEmail != null)
        {
            param.Append(searchEmail);
        }
        param.Append("&searchPhone=");
        if (searchPhone != null)
        {
            param.Append(searchPhone);
        }
        List<OrderHeader> OrderHeaderList = new List<OrderHeader>();
        if (searchName != null || searchEmail != null || searchPhone != null)
        {
            var user = new ApplicationUser();

            if(searchName!=null)
            {
                OrderHeaderList = await _db.OrderHeader.Include(o => o.ApplicationUser)
                    .Where(u =>
u.PickupName.ToLower().Contains(searchName.ToLower()))
                    .OrderByDescending(o =>
o.OrderDate).ToListAsync();
            }
            else
            {
                if (searchEmail != null)
                {
                    user = await _db.ApplicationUser.Where(u =>
u.Email.ToLower().Contains(searchEmail.ToLower())).FirstOrDefaultAsync();

```

```

        OrderHeaderList = await _db.OrderHeader.Include(o =>
o.ApplicationUser)
                                                    .Where(o=>o.UserId==user.Id)
                                                    .OrderByDescending(o =>
o.OrderDate).ToListAsync();
    }
    else
    {
        if (searchPhone != null)
        {
            OrderHeaderList = await _db.OrderHeader.Include(o =>
o.ApplicationUser)
                                                    .Where(u =>
u.PhoneNumber.Contains(searchPhone))
                                                    .OrderByDescending(o =>
o.OrderDate).ToListAsync();
        }
    }
}
else
{
    OrderHeaderList = await _db.OrderHeader.Include(o =>
o.ApplicationUser).Where(u => u.Status == SD.StatusReady).ToListAsync();
}

foreach (OrderHeader item in OrderHeaderList)
{
    OrderDetailsViewModel individual = new OrderDetailsViewModel
    {
        OrderHeader = item,
        OrderDetails = await _db.OrderDetails.Where(o => o.OrderId ==
item.Id).ToListAsync();
    };
    orderListVM.Orders.Add(individual);
}

var count = orderListVM.Orders.Count;
orderListVM.Orders = orderListVM.Orders.OrderByDescending(p => p.OrderHeader.Id)
    .Skip((productPage - 1) * PageSize)
    .Take(PageSize).ToList();

orderListVM.PagingInfo = new PagingInfo
{
    CurrentPage = productPage,
    ItemsPerPage = PageSize,
    TotalItem = count,
    urlParam = param.ToString()
};

return View(orderListVM);
}
[Authorize(Roles =SD.FrontDeskUser + ","+ SD.ManagerUser)]
[HttpPost]
[ActionName("OrderPickup")]
public async Task<IActionResult> OrderPickupPost(int orderId)
{
    OrderHeader orderHeader = await _db.OrderHeader.FindAsync(orderId);
    orderHeader.Status = SD.StatusCompleted;
    await _db.SaveChangesAsync();

    return RedirectToAction("OrderPickup", "Order");
}
}
}

```

Модуль «Створення замовлення»:

```
public class CartController : Controller
{
    private readonly ApplicationDbContext _db;

    [BindProperty]
    public OrderDetailsCart detailCart { get; set; }

    public CartController(ApplicationDbContext db)
    {
        _db = db;
    }

    public async Task<IActionResult> Index()
    {
        detailCart = new OrderDetailsCart()
        {
            OrderHeader = new Models.OrderHeader()
        };

        detailCart.OrderHeader.OrderTotal = 0;

        var claimsIdentity = (ClaimsIdentity)User.Identity;
        var claim = claimsIdentity.FindFirst(ClaimTypes.NameIdentifier);

        var cart = _db.ShoppingCart.Where(c => c.ApplicationUserId == claim.Value);
        if (cart != null)
        {
            detailCart.listCart = cart.ToList();
        }

        foreach (var list in detailCart.listCart)
        {
            list.Product = await _db.Product.FirstOrDefaultAsync(m => m.Id ==
list.ProductId);
            detailCart.OrderHeader.OrderTotal = detailCart.OrderHeader.OrderTotal +
(list.Product.Price * list.Count);
            list.Product.Description = SD.ConvertToRawHtml(list.Product.Description);
            if (list.Product.Description.Length > 100)
            {
                list.Product.Description = list.Product.Description.Substring(0, 99) +
"...";
            }
        }
        detailCart.OrderHeader.OrderTotalOriginal = detailCart.OrderHeader.OrderTotal;

        return View(detailCart);
    }

    public async Task<IActionResult> Summary()
    {
        detailCart = new OrderDetailsCart()
        {
            OrderHeader = new Models.OrderHeader()
        };

        detailCart.OrderHeader.OrderTotal = 0;

        var claimsIdentity = (ClaimsIdentity)User.Identity;
        var claim = claimsIdentity.FindFirst(ClaimTypes.NameIdentifier);
```

```

        ApplicationUser applicationUser = await _db.ApplicationUser.Where(c => c.Id ==
claim.Value).FirstOrDefaultAsync();
        var cart = _db.ShoppingCart.Where(c => c.ApplicationUserId == claim.Value);
        if (cart != null)
        {
            detailCart.listCart = cart.ToList();
        }

        foreach (var list in detailCart.listCart)
        {
            list.Product = await _db.Product.FirstOrDefaultAsync(m => m.Id ==
list.ProductId);
            detailCart.OrderHeader.OrderTotal = detailCart.OrderHeader.OrderTotal +
(list.Product.Price * list.Count);
        }
        detailCart.OrderHeader.OrderTotalOriginal = detailCart.OrderHeader.OrderTotal;
        detailCart.OrderHeader.PickupName = applicationUser.Name;
        detailCart.OrderHeader.PhoneNumber = applicationUser.PhoneNumber;
        detailCart.OrderHeader.PickUpTime = DateTime.Now;

        return View(detailCart);
    }

    [HttpPost]
    [ValidateAntiForgeryToken]
    [ActionName("Summary")]
    public async Task<IActionResult> SummaryPost(string stripeToken)
    {
        var claimsIdentity = (ClaimsIdentity)User.Identity;
        var claim = claimsIdentity.FindFirst(ClaimTypes.NameIdentifier);

        detailCart.listCart = await _db.ShoppingCart.Where(c => c.ApplicationUserId ==
claim.Value).ToListAsync();

        detailCart.OrderHeader.PaymentStatus = SD.PaymentStatusPending;
        detailCart.OrderHeader.OrderDate = DateTime.Now;
        detailCart.OrderHeader.UserId = claim.Value;
        detailCart.OrderHeader.Status = SD.PaymentStatusPending;
        detailCart.OrderHeader.PickUpTime =
Convert.ToDateTime(detailCart.OrderHeader.PickUpDate.ToShortDateString() + " " +
detailCart.OrderHeader.PickUpTime.ToShortTimeString());

        List<OrderDetails> orderDetailsList = new List<OrderDetails>();
        _db.OrderHeader.Add(detailCart.OrderHeader);
        await _db.SaveChangesAsync();

        detailCart.OrderHeader.OrderTotalOriginal = 0;

        foreach (var item in detailCart.listCart)
        {
            item.Product = await _db.Product.FirstOrDefaultAsync(m => m.Id ==
item.ProductId);
            OrderDetails orderDetails = new OrderDetails
            {
                ProductId = item.ProductId,
                OrderId = detailCart.OrderHeader.Id,
                Description = item.Product.Description,
                Name = item.Product.Name,
                Price = item.Product.Price,
            }
        }
    }

```

```

        Count = item.Count
    };
    detailCart.OrderHeader.OrderTotalOriginal += orderDetails.Count *
orderDetails.Price;
    _db.OrderDetails.Add(orderDetails);
}

_db.ShoppingCart.RemoveRange(detailCart.listCart);
await _db.SaveChangesAsync();

var options = new ChargeCreateOptions
{
    Amount = Convert.ToInt32(detailCart.OrderHeader.OrderTotalOriginal * 100),
    Currency = "uah",
    Description = "Order ID : " + detailCart.OrderHeader.Id,
    Source = stripeToken
};
var service = new ChargeService();
Charge charge = service.Create(options);

if (charge.BalanceTransactionId == null)
{
    detailCart.OrderHeader.PaymentStatus = SD.PaymentStatusRejected;
}
else
{
    detailCart.OrderHeader.TransactionId = charge.BalanceTransactionId;
}

if (charge.Status.ToLower() == "succeeded")
{
    detailCart.OrderHeader.PaymentStatus = SD.PaymentStatusApproved;
    detailCart.OrderHeader.Status = SD.StatusSubmitted;
}
else
{
    detailCart.OrderHeader.PaymentStatus = SD.PaymentStatusRejected;
}

await _db.SaveChangesAsync();
return RedirectToAction("Index", "Home");
}

public async Task<IActionResult> Plus(int cartId)
{
    var cart = await _db.ShoppingCart.FirstOrDefaultAsync(c => c.Id == cartId);
    cart.Count += 1;
    await _db.SaveChangesAsync();
    return RedirectToAction(nameof(Index));
}

public async Task<IActionResult> Minus(int cartId)
{
    var cart = await _db.ShoppingCart.FirstOrDefaultAsync(c => c.Id == cartId);
    if (cart.Count == 1)
    {
        _db.ShoppingCart.Remove(cart);
        await _db.SaveChangesAsync();

        var cnt = _db.ShoppingCart.Where(u => u.ApplicationUserId ==
cart.ApplicationUserId).ToList().Count;

```

```

        HttpContext.Session.SetInt32(SD.ssShoppingCartCount, cnt);
    }
    else
    {
        cart.Count -= 1;
        await _db.SaveChangesAsync();
    }

    return RedirectToAction(nameof(Index));
}

public async Task<IActionResult> Remove(int cartId)
{
    var cart = await _db.ShoppingCart.FirstOrDefaultAsync(c => c.Id == cartId);

    _db.ShoppingCart.Remove(cart);
    await _db.SaveChangesAsync();

    var cnt = _db.ShoppingCart.Where(u => u.ApplicationUserId ==
cart.ApplicationUserId).ToList().Count;
    HttpContext.Session.SetInt32(SD.ssShoppingCartCount, cnt);

    return RedirectToAction(nameof(Index));
}

```