Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет ___Автоматизації, комп'ютерно-інтегрованих технологій та систем___
(повна назва)

Кафедра __Комп’ютерно-інтегрованих технологій, автоматизації, робототехніки
_____та безпекової інженерії_____
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти _____перший (бакалаврський)_____

Розроблення автоматизованої системи моніторингу та диспетчеризації робочих
_____місць на базі комп’ютерно-інтегрованих технологій_____
(тема)

Виконав:
здобувач __4__ року навчання,
групи _____АКТАКІТ-22-1_____
_____Даніл ПОЛОВИНКІН_____
(власне ім’я, прізвище)
Спеціальність _151 «Автоматизація та_____
комп’ютерно-інтегровані технології»_____
(код і повна назва спеціальності)
Тип програми ____освітньо-професійна_____
Освітня програма Автоматизація та_____
комп’ютерно-інтегровані технології_____
(повна назва освітньої програми)

Керівник ____доц. Артем БРОННІКОВ_____
(посада, власне ім’я, прізвище)

Допускається до захисту

Зав. кафедри _____ Ігор НЕВЛЮДОВ
(підпис) (власне ім’я, прізвище)

2026 р.

Я, Половинкін Даніл Олександрович, як здобувач вищої освіти ХНУРЕ, розумію та підтримую політику закладу з академічної доброчесності. Я не надавав і не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Я не використовував штучний інтелект для підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.



"18" червень 2026 р. Даніл ПОЛОВИНКІН

Харківський національний університет радіоелектроніки

Факультет___Автоматизації, комп'ютерно-інтегрованих технологій та систем___

Кафедра __Комп’ютерно-інтегрованих технологій, автоматизації, робототехніки та безпекової інженерії_____

Рівень вищої освіти_____перший (бакалаврський)_____

Спеціальність___151 Автоматизація та комп`ютерно-інтегровані технології_____
(код і повна назва)

Тип програми _____освітньо-професійна_____

Освітня програма __Автоматизація та комп’ютерно-інтегровані технології____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)
«_____»_____ 2026_ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Половинкіну Данілу Олександровичу_____
(прізвище, ім’я, по батькові)

1. Тема роботи Розроблення автоматизованої системи моніторингу та_____
диспетчеризації робочих місць на базі комп’ютерно-інтегрованих технологій____

затверджена наказом університету від «04» 06 2026 р. № 472 ст
2. Термін подання здобувачем роботи до екзаменаційної комісії 17 07 2026 р.
3. Вихідні дані до роботи _____
3.1 Мови програмування та web-технології: JavaScript, HTML, CSS_____
3.2 Система керування базами даних MySQL_____
3.3 Технології взаємодії компонентів: REST API, JSON, Fetch API_____
3.4 Засоби моделювання та проєктування: UML, DFD, IDEF0, ER-модель_____
3.5 Методи аналізу ефективності співробітників із використанням ШІ_____
4. Перелік питань, що потрібно опрацювати в роботі _____
4.1 Аналіз предметної області та сучасних систем автоматизації виробництва____
4.2 Аналіз АСУП та АСУТП на сучасному підприємстві_____
4.3 Проєктування архітектури інформаційної системи та бази даних_____
4.4 Розробка алгоритмів управління замовленнями і виробничим графіком_____
4.5 Програмна реалізація веб-орієнтованої інформаційної системи_____

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій Графічний демонстраційний матеріал у форматі ppt, 12 арк. ф. А4

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз технічного стану виробництва зараз	30.04.2026	Виконано
2	Аналіз АСУТП та АСУП на сучасному виробництві	05.05.2026	Виконано
3	Сучасні підходи до керування підприємством	10.05.2026	Виконано
4	Управління замовленнями та графіком роботи підприємства	15.05.2026	Виконано
5	Теорія автоматичного управління	17.05.2026	Виконано
6	Вибір та обґрунтування мови програмування, бібліотек та технологій	18.05.2026	Виконано
7	Розробка структурної схема роботи додатку	26.05.2026	Виконано
8	Розробка блок-схеми роботи ПЗ	28.05.2026	Виконано
9	Створення програмного забезпечення	31.05.2026	Виконано
10	Створення документу «Охорона праці»	02.06.2026	Виконано
11	Оформлення пояснювальної записки	10.06.2026	Виконано
12	Захист кваліфікаційної роботи	17.07.2026	

Дата видачі завдання 04 06 2026 р.

Здобувач ____ _____ _____Даніл ПОЛОВИНКІН____
(підпис) (власне ім'я, прізвище)

Керівник роботи _____ __доц. Артем БРОННІКОВ_____
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 74 с., 14 табл., 24 рис., 1 дод., 47 джерел.

АНАЛІЗ, АСУП, АСУТП, ВЕБСИСТЕМА, ВИРОБНИЦТВО, ГРАФІК, ЗАМОВЛЕННЯ, ІНФОРМАЦІЙНА СИСТЕМА, ОЦІНКА ПЕРСОНАЛУ, ПІДПРИЄМСТВО, ПЛАНУВАННЯ, УПРАВЛІННЯ, ШТУЧНИЙ ІНТЕЛЕКТ, ШТУЧНИЙ ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ, ЕФЕКТИВНІСТЬ.

Об’єктом розробки є процеси моніторингу, контролю та диспетчеризації робочих місць підприємства в умовах використання інтегрованих технологій.

Предметом розробки є методи, моделі та програмно-технічні засоби автоматизації моніторингу стану робочих місць, збору й обробки даних про їх функціонування, а також диспетчеризації виробничих і організаційних процесів.

Метою роботи є підвищення ефективності функціонування автоматизованих систем управління підприємством шляхом розробки онлайн-системи обліку замовлень, планування виробничих завдань і контролю їх виконання.

Основна задача роботи полягає у створенні ІС для підвищення ефективності управління виробничими процесами за рахунок автоматизації збору, обробки та аналізу даних про замовлення і виробничу діяльність. Використання веб-технологій забезпечує централізований доступ до системи, а застосування ШІ дає змогу об’єктивно оцінювати ефективність роботи співробітників, виявляти проблемні ділянки виробництва та підвищувати якість управлінських рішень.

Отримані результати роботи відповідають цілі сталого розвитку 9 «Промисловість, інновації та інфраструктура», зокрема завданню 9.4 щодо розвитку високотехнологічних секторів промисловості та впровадження інформаційно-комунікаційних технологій у виробничу діяльність.

ABSTRACT

Explanatory note to the qualification work: 74 pages, 14 tables, 24 pictures, 1 appendix, 47 sources.

ANALYSIS, APCS, ARTIFICIAL INTELLIGENCE, ARTIFICIAL INTELLIGENCE ANALYSIS, AEMS, EFFICIENCY, ENTERPRISE, INFORMATION SYSTEM, MANAGEMENT, ORDER, PERSONNEL EVALUATION, PLANNING, PRODUCTION, SCHEDULE, WEB SYSTEM.

The object of development is the processes of monitoring, control and dispatching of workplaces of the enterprise in the conditions of using integrated technologies.

The subject of development is methods, models and software and hardware tools for automating monitoring of the state of workplaces, collecting and processing data on their functioning, as well as dispatching of production and organizational processes.

The purpose of the work is to increase the efficiency of the functioning of automated enterprise management systems by developing an online system for accounting for orders, planning production tasks and monitoring their implementation.

The main task of the work is to create IS to increase the efficiency of production process management by automating the collection, processing and analysis of data on orders and production activities. The use of web technologies provides centralized access to the system, and the use of AI makes it possible to objectively assess the efficiency of employee work and improve the quality of management decisions.

The results of the work correspond to the Sustainable Development Goal 9 "Industry, Innovation and Infrastructure", in particular task 9.4 on the development of high-tech sectors of industry and the introduction of information and communication technologies in production activities.

ЗМІСТ

Перелік скорочень ……. 8
Вступ …….. 9
1 Аналіз сучасних систем контролю на виробництві ……..... 11
1.1 Аналіз технічного стану виробництва зараз ……... 11
1.2 Аналіз АСУТП та АСУП на сучасному виробництві …… 17
1.3 Сучасні підходи до керування підприємством ……... 20
1.4 Управління замовленнями та графіком роботи підприємства ….. 23
2 Вибір та обґрунтуванням технічних засобів ……... 29
2.1 Теоретичні основи автоматичного управління процесами диспетчеризації виробничих завдань……... 29
2.2 Вибір та обґрунтування мови програмування, бібліотек та технологій …….. 32
2.3 Структурна схема роботи додатку …….. 37
2.4 Параметри оцінки ІС ……. 40
3 Розробка системи контролю процесу виконання робітником виробничих завдань …… 45
3.1 Блок-схема роботи ПЗ ……... 45
3.2 Створення програмного забезпечення …… 50
3.3 Робота додатку …….. 58
3.4 Охорона праці …… 67
Висновки …… 69
Перелік джерел посилання ……... 71
Додаток А Програмний код розробленого рішення …….. 75
Додаток Б Демонстраційний матеріал …… 85

ПЕРЕЛІК СКОРОЧЕНЬ

АСУП – автоматизована система управління підприємством

АСУТП – автоматизована система управління технологічними процесами

БД – база даних

ІС – інформаційна система

ПЗ – програмне забезпечення

ШІ – штучний інтелект

API – application programming interface

BPMN – business process model and notation

CRUD – create, read, update, delete (основні операції обробки даних)

DFD – data flow diagram

ERP – enterprise resource planning

IDEF0 – integration definition for function modeling

JSON – JavaScript object notation

SQL – structured query language

UML – unified modeling language

UX/UI – user experience / user interface

ВСТУП

Сучасний розвиток промислових підприємств супроводжується активним впровадженням ІС (інформаційних систем), АСУП (автоматизованих систем управління підприємством) та АСУТП (автоматизованих систем управління технологічними процесами), які забезпечують автоматизацію виробничих і управлінських процесів. В умовах зростання конкуренції та підвищення вимог до швидкості виконання замовлень підприємства потребують сучасних програмних рішень для централізованого управління замовленнями, планування виробничого графіка та контролю виконання робіт у режимі реального часу.

Традиційні методи управління, що базуються на паперовій документації або неінтегрованих програмних засобах, мають низку недоліків: значні витрати часу на обробку інформації, складність контролю виробничих процесів, високу ймовірність помилок та обмежені можливості аналітики. Відсутність єдиного інформаційного середовища ускладнює взаємодію між підрозділами та знижує ефективність управління підприємством.

Важливу роль у підвищенні ефективності виробництва відіграють web-орієнтовані ІС, які забезпечують централізоване зберігання даних, доступ до системи через браузер та інтеграцію функціональних модулів підприємства. Використання web-технологій дозволяє автоматизувати процеси створення та обробки замовлень, планування виробничих задач, розподілу робіт між співробітниками і контролю виконання виробничих операцій. Додатково інтеграція ШІ (штучного інтелекту) забезпечує аналіз ефективності роботи персоналу та формування аналітичних звітів.

Актуальність теми обумовлена необхідністю створення сучасної онлайн-ІС управління замовленнями та виробничим графіком підприємства, яка забезпечить автоматизацію бізнес-процесів, оперативний доступ до інформації, підвищення точності планування та ефективності контролю виконання робіт. Впровадження

такої системи дозволить скоротити час обробки замовлень, підвищити ефективність використання ресурсів підприємства та покращити управління виробничою діяльністю.

Отже, метою роботи є підвищення ефективності функціонування автоматизованих систем управління виробництвом шляхом розроблення комп'ютерно-інтегрованої системи моніторингу стану робочих місць та диспетчеризації завдань персоналу.

Об’єктом роботи є процес управління замовленнями та виробничими процесами підприємства.

Предметом розробки є програмне забезпечення онлайн-ІС управління замовленнями та виробничим графіком підприємства, реалізоване з використанням web-технологій, SQL-СУБД та засобів інтеграції ШІ.

Для досягнення поставленої мети необхідно вирішити такі задачі:

– провести аналіз сучасних ІС управління підприємством;

– виконати аналіз АСУП та АСУТП на сучасному виробництві;

– розробити структуру та архітектуру онлайн-ІС;

– створити ER-модель БД системи;

– розробити алгоритми та блок-схеми роботи ПЗ;

– обґрунтувати вибір технологій і засобів розробки;

– реалізувати web-інтерфейс системи та ШІ-модуль аналізу ефективності співробітників;

– провести тестування роботи системи.

Кваліфікаційна робота оформлена відповідно до вимог ДСТУ 3008:2015 [1] та методичних рекомендацій щодо виконання кваліфікаційних робіт [2].

1 АНАЛІЗ СУЧАСНИХ СИСТЕМ КОНТРОЛЮ НА ВИРОБНИЦТВІ

1.1 Аналіз технічного стану виробництва зараз

Сучасні виробничі підприємства функціонують в умовах постійного зростання обсягів замовлень, підвищення вимог до якості продукції та скорочення термінів її виготовлення [3]. У таких умовах ефективність діяльності підприємства значною мірою залежить від рівня автоматизації процесів управління замовленнями, планування виробництва та контролю виконання виробничих завдань. Проте на багатьох підприємствах значна частина цих процесів досі виконується вручну або з використанням окремих неінтегрованих програмних засобів без використання єдиної ІС та централізованої БД (бази даних).

Досліджуване підприємство є виробничою організацією, основною діяльністю якої є виконання індивідуальних і серійних замовлень клієнтів. Організаційна структура підприємства включає три основні категорії учасників виробничого процесу:

– клієнт формує замовлення на виготовлення продукції;

– менеджер здійснює обробку замовлення, планування виробництва та контроль виконання;

– співробітник виконує виробничі завдання відповідно до сформованого виробничого графіка.

Обмін інформацією між цими учасниками здійснюється переважно вручну або з використанням окремих програмних засобів, таких як електронні таблиці, документи або локальні програми, що не забезпечують централізованого управління даними.

Загальна структура процесу управління виробництвом представлена на рисунку 1.1 у вигляді контекстної IDEF0-діаграми.

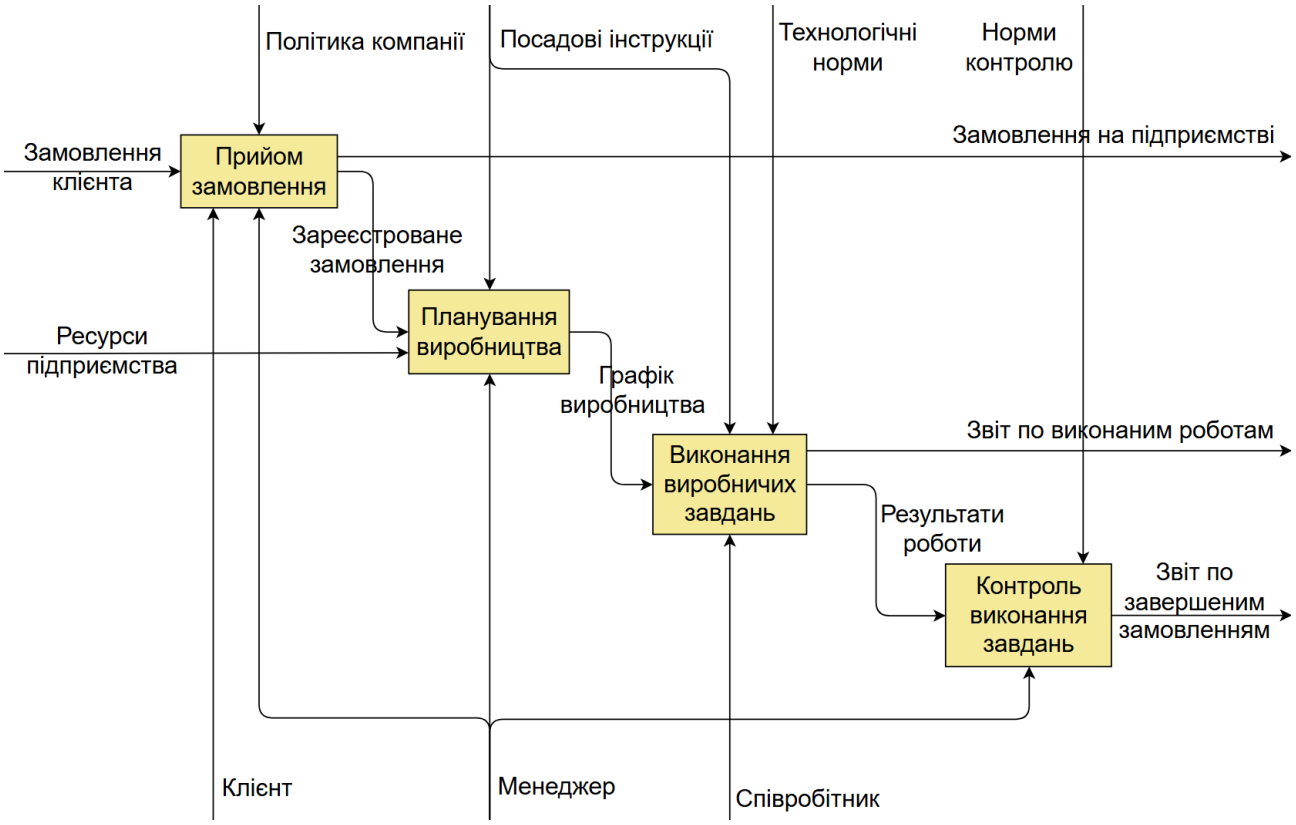


Рисунок 1.1 – Декомпозиція IDEF0-діаграми управління виробничим процесом

Основними функціональними блоками цього процесу є:

– прийом замовлення;

– планування виробництва;

– виконання виробничих завдань;

– контроль виконання завдань.

На першому етапі здійснюється прийом замовлення від клієнта та його реєстрація. Далі зареєстроване замовлення передається до підсистеми планування виробництва, де формується виробничий графік з урахуванням наявних ресурсів підприємства та технологічних норм. На основі сформованого графіка виробничі завдання передаються співробітникам для виконання. Після завершення виконання виробничих завдань результати передаються до підсистеми контролю, де здійснюється перевірка відповідності виконання встановленим нормам і формується звітність. Додатково на етапі контролю може виконуватись аналіз

причин відхилення від планових показників та оцінка ефективності використання ресурсів підприємства. Це дозволяє своєчасно виявляти проблемні ділянки виробничого процесу та підвищувати якість управлінських рішень.

Таким чином, IDEF0-діаграма дозволяє визначити функціональну структуру виробничого процесу і взаємозв'язки між його основними етапами [4].

Існуючі інформаційні потоки підприємства представлені на рисунку 1.2 у вигляді діаграми потоків даних [5] (DFD) AS-IS, яка відображає поточний стан інформаційної системи підприємства без використання інтегрованої ІС.

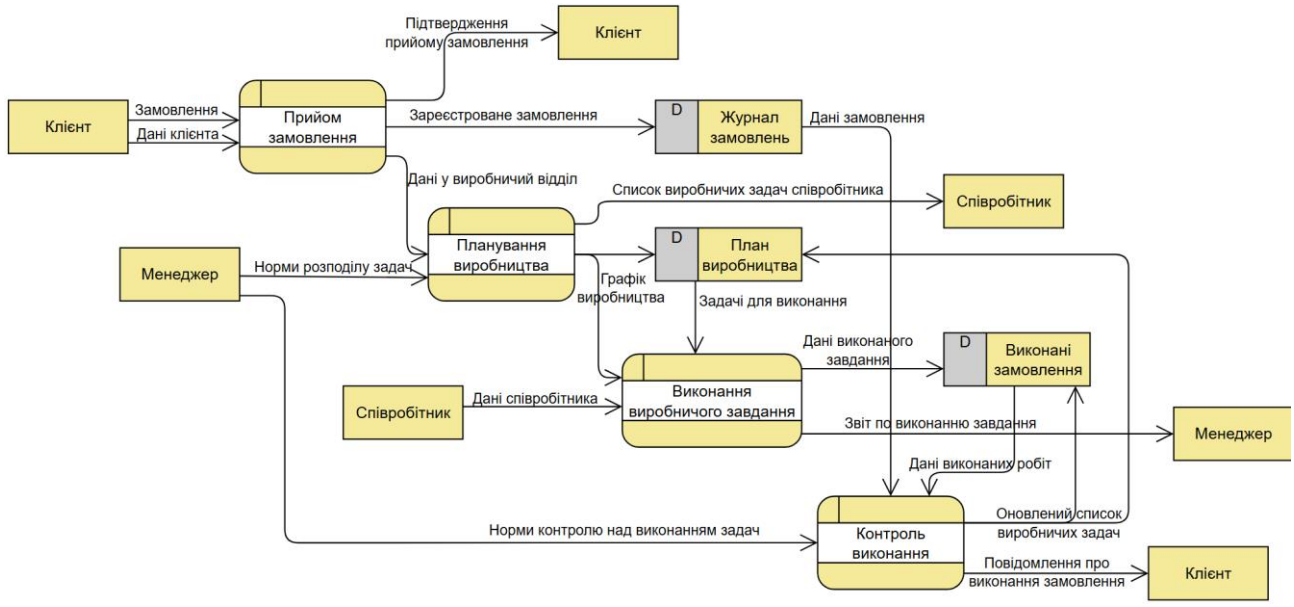


Рисунок 1.2 – Схема інформаційних потоків на підприємстві (DFD-модель AS-IS)

Основними процесами на діаграмі є:

– прийом замовлення;

– планування виробництва;

– виконання виробничого завдання;

– контроль виконання.

У процесі функціонування системи використовуються наступні сховища даних (таблиця 1.1).

Таблиця 1.1 – Сховища даних існуючої системи

Назва сховища	Призначення
Журнал замовлень	Зберігання інформації про зареєстровані замовлення
План виробництва	Зберігання виробничого графіка
Виконані замовлення	Зберігання інформації про завершені замовлення

Як показано на рисунку 1.2, після прийому замовлення інформація записується до сховища «Журнал замовлень». Далі ці дані використовуються для формування виробничого графіка, який зберігається у сховищі «План виробництва». Співробітники отримують виробничі завдання на основі цього графіка та виконують їх. Результати виконання записуються у сховище «Виконані замовлення» і використовуються для формування звітності.

Дана модель демонструє, що інформація передається між різними підрозділами підприємства через окремі сховища даних без використання єдиної інтегрованої ІС.

Аналіз існуючої системи показав наявність наступних недоліків [6]:

– відсутність єдиної ІС для управління виробництвом;

– відсутність централізованої БД;

– використання розрізнених програмних засобів;

– дублювання інформації;

– затримки при передачі інформації;

– складність контролю виконання виробничих завдань;

– відсутність автоматизованого аналізу ефективності роботи співробітників;

– ручне планування виробничого графіка.

Ці недоліки призводять до зниження ефективності управління підприємством, збільшення часу обробки замовлень і підвищення ймовірності

виникнення помилок. Існуюча система управління замовленнями і виробничим графіком характеризується низьким рівнем автоматизації. Основні процеси управління виконуються без використання інтегрованої ІС і централізованої БД, що ускладнює обробку інформації і контроль виробничих процесів. Це обґрунтовує необхідність розробки онлайн-ІС управління замовленнями та виробничим графіком, яка забезпечить автоматизацію процесів управління, централізоване зберігання даних і підвищення ефективності роботи підприємства.

Для аналізу технічного стану виробництва доцільно розглянути існуючі програмні рішення, що використовуються на підприємствах. В якості аналогів було обрано ERP-систему (на прикладі SAP ERP) та MES-система [7] (на прикладі SAP Digital Manufacturing). Ці системи є типовими представниками класів АСУП та АСУТП. Порівняльний аналіз між цими системами та розроблюваним модулем контролю наведено в таблиці 1.2.

Таблиця 1.2 – Порівняльний аналіз сучасних систем управління виробництвом

Критерій	ERP (SAP ERP)	MES (SAP Digital Manufacturing)	Розроблювана ІС
Основне призначення	Управління всіма бізнес-процесами підприємства	Оперативне управління виробництвом	Управління замовленнями + виробничим графіком
Рівень управління	Стратегічний і тактичний	Оперативний (виробництво)	Оперативний + частково тактичний
Інтеграція процесів	Висока, єдина БД	Часткова, між ERP та виробництвом	Повна інтеграція замовлення + виробництво

Продовження таблиці 1.2

Критерій	ERP (SAP ERP)	MES (SAP Digital Manufacturing)	Розроблювана ІС
Робота з виробничим графіком	Обмежена, через модулі	Присутня, але складна у налаштуванні	Гнучке управління графіком
Зручність UX/UI	Складний інтерфейс, потребує навчання	Орієнтований на технічний персонал	Орієнтований на користувача
Гнучкість налаштування	Висока, але складна	Складна та залежить від інтеграції	Висока, адаптована під задачу
Вартість впровадження	Дуже висока	Висока	Низька/середня
Швидкість впровадження	Довготривала	Довготривала	Швидка
Використання ШІ	Обмежене	Частково	Інтегрований ШІ-аналіз

ERP-системи є потужними інструментами, однак мають ряд суттєвих обмежень [8]:

– орієнтовані на загальне управління підприємством, а не на детальне управління виробничими процесами;

– складна структура та висока залежність між модулями;

– значні витрати на впровадження та підтримку;

– тривалий час адаптації персоналу.

MES-системи ефективні на рівні виробництва, але також мають обмеження [9]:

– складність інтеграції з іншими системами (ERP, CRM);

– вузька спеціалізація – відсутність повного управління підприємством;

– складність впровадження та налаштування;

– залежність від технічної інфраструктури.

Існуючі системи класів ERP та MES ефективно вирішують окремі задачі підприємства, однак не забезпечують комплексного, гнучкого та зручного управління замовленнями і виробничим графіком одночасно. ERP-системи є надто складними та дорогими, тоді як MES-системи мають обмежену функціональність і складну інтеграцію. Додатково використання окремих систем часто призводить до дублювання даних та ускладнення інформаційних потоків між підрозділами підприємства. Це негативно впливає на швидкість обробки замовлень і ефективність прийняття управлінських рішень.

Розроблена ІС усуває ці недоліки за рахунок наступних факторів:

– поєднання управління замовленнями та виробництвом в єдиній системі;

– спрощений інтуїтивно зрозумілий UX/UI;

– нативна інтеграція ШІ для аналізу ефективності.

1.2 Аналіз АСУТП та АСУП на сучасному виробництві

Сучасне виробниче підприємство функціонує як складна багаторівнева система управління, що включає управлінський і технологічний рівні. Автоматизація цих рівнів реалізується через АСУП та АСУТП, які виконують різні, але взаємопов’язані функції [10].

АСУП призначена для автоматизації управлінських процесів підприємства, тоді як АСУТП забезпечує автоматизацію технологічних процесів виробництва.

Основні характеристики цих систем наведено в таблиці 1.3.

Таблиця 1.3 – Порівняльна характеристика АСУП та АСУТП

Критерій	АСУП	АСУТП
Рівень управління	Управлінський	Технологічний
Основне призначення	Управління підприємством	Управління виробничими процесами
Об’єкт управління	Замовлення, персонал, ресурси	Обладнання, технологічні операції
Джерело даних	БД підприємства	Датчики, сенсори
Основний результат	Звіти, аналітика, графіки	Стан обладнання, виробничі параметри

Структура АСУП:

– підсистема отримання замовлень;

– підсистема управління замовленнями;

– підсистема управління персоналом;

– збереження технічних параметрів;

– БД підприємства.

АСУТП включає:

– модуль управління обладнанням;

– підсистему збору даних з датчиків;

– модуль контролю технологічних параметрів;

– модуль передачі даних до АСУП.

АСУП забезпечує централізоване зберігання інформації в БД та координацію всіх управлінських процесів. АСУТП працює в реальному часі та безпосередньо взаємодіє з виробничим обладнанням [11].

На рисунку 1.3 представлено структурну схему взаємодії підсистем управління підприємством.

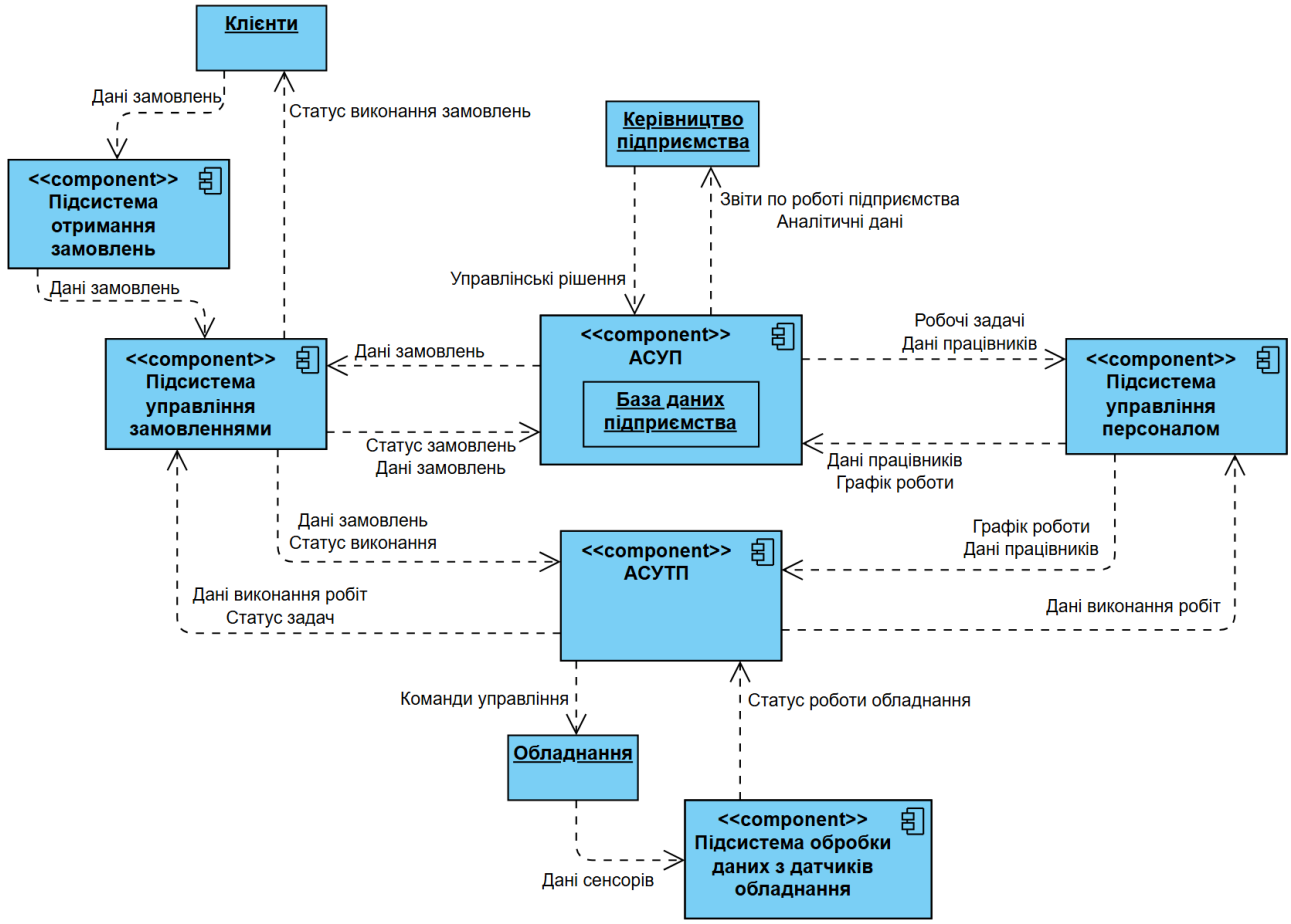


Рисунок 1.3 – Схема взаємодії підсистем управління підприємством

Центральним елементом є АСУП, яка інтегрує всі управлінські процеси через БД підприємства. Через неї проходять основні інформаційні потоки: дані замовлень, статус виконання, виробничі завдання, графік роботи, дані працівників, звітність та аналітичні дані.

АСУТП, у свою чергу, взаємодіє з обладнанням та підсистемою обробки даних з датчиків, забезпечуючи контроль технологічних параметрів і передачу статусу виробництва до АСУП.

Переваги впровадження АСУП та АСУТП:

– централізація управління;

– автоматизація обробки даних;

– зменшення людського фактору;

– підвищення оперативності управління;

– контроль виробничих процесів у реальному часі;

– можливість інтеграції ШІ для аналітики.

Недоліки існуючих реалізацій:

– складність інтеграції між підсистемами;

– висока вартість впровадження;

– низька гнучкість типових рішень;

– обмежена інтеграція з веб-ІС;

– відсутність розширеної аналітики на основі ШІ.

АСУП і АСУТП є фундаментальними компонентами сучасного виробництва, які забезпечують автоматизацію управлінських і технологічних процесів відповідно. Проте їх класичні реалізації часто не забезпечують повної інтеграції з сучасними веб-технологіями та не орієнтовані на ефективне управління замовленнями в режимі реального часу.

Онлайн-ІС управління забезпечить інтеграцію функцій АСУП і АСУТП, централізоване зберігання даних у БД, підтримку програмних інтерфейсів (API) та можливість використання ШІ для аналітики і підтримки прийняття рішень, а тому має цілком обґрунтовану потребу в створенні й інтеграції.

1.3 Сучасні підходи до керування підприємством

Сучасне виробниче підприємство є складною динамічною системою, ефективність функціонування якої визначається якістю управління інформаційними, виробничими та людськими ресурсами. В умовах зростання конкуренції, збільшення кількості замовлень і необхідності скорочення часу виробництва особливого значення набуває використання сучасних ІС управління. Основним завданням таких систем є забезпечення оперативного збору, обробки та аналізу інформації, необхідної для планування виробництва, контролю виконання замовлень і підтримки прийняття управлінських рішень.

Основою сучасного управління підприємством є централізація інформації та інтеграція всіх підсистем у межах єдиного інформаційного середовища [12]. Це дозволяє забезпечити узгодженість роботи різних підрозділів підприємства, підвищити точність планування та зменшити кількість помилки, пов’язаних із використанням неактуальних або неповних даних. У таких умовах ключову роль відіграє АСУП, яка забезпечує координацію роботи підприємства на основі даних, що зберігаються у централізованій БД.

Одним із найбільш поширених сучасних підходів до управління підприємством є використання ERP. ERP є комплексною ІС, яка забезпечує інтеграцію всіх основних процесів підприємства, включаючи управління замовленнями, виробництвом, персоналом і ресурсами [13].

Основною перевагою ERP є використання єдиної БД, що дозволяє уникнути дублювання інформації і забезпечує цілісність даних. Наприклад, після реєстрації замовлення інформація про нього стає доступною для підсистеми планування виробництва, підсистеми управління персоналом і системи контролю виконання без необхідності повторного введення даних. Це значно скорочує час обробки замовлення і підвищує ефективність управління підприємством. Крім того, ERP забезпечує формування аналітичних звітів, які дозволяють керівництву підприємства оцінювати ефективність роботи виробництва, аналізувати завантаженість співробітників і приймати обґрунтовані управлінські рішення.

Важливим сучасним підходом є процесний підхід, який передбачає розгляд діяльності підприємства як сукупності взаємопов’язаних бізнес-процесів. Основними такими процесами є прийом замовлення, планування виробництва, виконання виробничих завдань і контроль їх виконання.

Використання процесного підходу дозволяє чітко визначити послідовність виконання операцій, відповідальних осіб і інформаційні потоки між підсистемами. Для формалізації цих процесів використовуються методології IDEF0, DFD, UML і модель бізнес-процесів (BPMN), які дозволяють створити формалізовану модель функціонування підприємства і є основою для розробки ІС [14].

Одним із найбільш важливих сучасних напрямків є використання веб-ІС для управління підприємством. На відміну від традиційних систем, які встановлюються на окремі комп'ютери, веб-ІС функціонує на сервері і забезпечує доступ користувачів через веб-браузер.

Такий підхід має ряд важливих переваг. По-перше, забезпечується централізований доступ до даних, що дозволяє всім користувачам працювати з актуальною інформацією. По-друге, спрощується процес адміністрування системи, оскільки оновлення програмного забезпечення (ПЗ) виконується лише на сервері. По-третє, веб-ІС забезпечує можливість віддаленої роботи, що є особливо важливим для сучасних підприємств. Крім того, веб-ІС дозволяє реалізувати зручний UX/UI, який значно спрощує роботу користувачів із системою і зменшує ймовірність виникнення помилок.

Основою сучасних веб-ІС є клієнт-серверна архітектура, яка передбачає розподіл функцій між клієнтською і серверною частинами [15].

Серверна частина забезпечує:

– обробку даних;

– реалізацію бізнес-логіки;

– взаємодію з БД.

Клієнтська частина забезпечує:

– взаємодію користувача з системою;

– відображення інформації;

– передачу запитів до сервера через API.

Взаємодія між клієнтом і сервером здійснюється з використанням API, а обмін даними відбувається у форматі JSON [16]. Для роботи з БД використовується SQL.

Така архітектура забезпечує централізацію управління, високу продуктивність, масштабованість системи та підвищений рівень безпеки.

Одним із найбільш перспективних напрямків розвитку ІС є використання ШІ. На відміну від традиційних систем, які лише зберігають і відображають інформацію, ШІ дозволяє виконувати її аналіз і формувати рекомендації [17].

Використання ШІ дозволяє вирішувати наступні задачі:

– аналіз ефективності роботи співробітників;

– визначення перевантажених і недовантажених працівників;

– прогнозування термінів виконання замовлень;

– оптимізація виробничого графіка.

Наприклад, на основі даних про виконання попередніх замовлень ШІ може визначити середній час виконання задачі конкретним співробітником і використати цю інформацію для більш точного планування виробництва. Це дозволяє значно підвищити ефективність використання ресурсів підприємства і покращити якість управління.

1.4 Управління замовленнями та графіком роботи підприємства

Управління замовленнями та виробничим графіком є одним із ключових процесів функціонування виробничого підприємства. Ефективність виконання замовлень безпосередньо впливає на продуктивність підприємства, рівень використання ресурсів і якість обслуговування клієнтів. У сучасних умовах ці процеси реалізуються за допомогою ІС, яка забезпечує автоматизацію створення, планування, виконання і контролю замовлень [18].

Основною метою управління замовленнями є забезпечення своєчасного виконання виробничих завдань відповідно до встановлених термінів і доступних ресурсів. Для цього ІС повинна забезпечувати функції реєстрації замовлень, планування виробничого графіка, призначення завдань співробітникам, контролю виконання і формування звітності.

Функціональні можливості сучасної ІС управління замовленнями представлені на рисунку 1.4 у вигляді UML Use Case діаграми. Ця діаграма визначає основні сценарії взаємодії користувачів із системою [19].

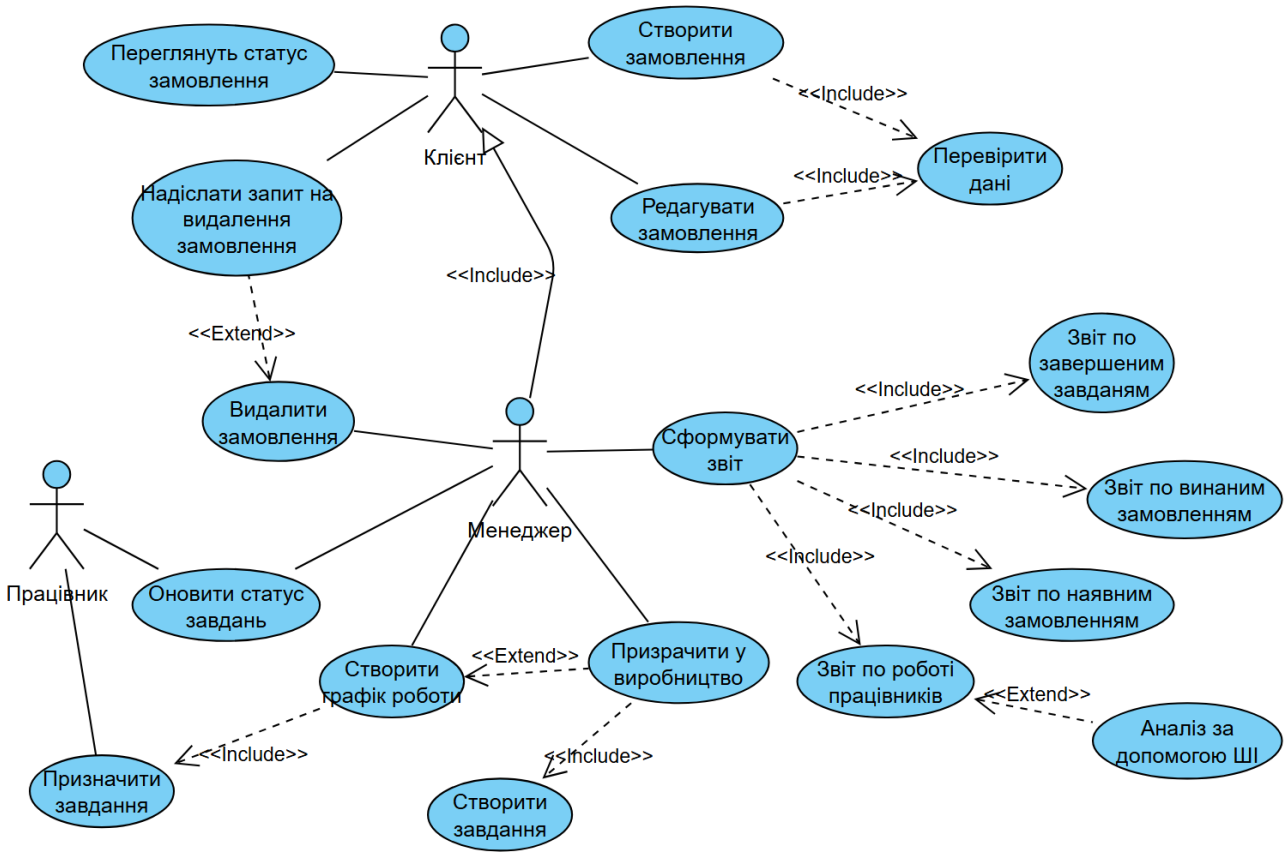


Рисунок 1.4 – UML Use Case діаграма системи управління замовленнями

На діаграмі визначено три основні ролі користувачів: клієнт, менеджер та працівник підприємства.

Клієнт взаємодіє із системою з метою створення і контролю замовлення. Він може створити замовлення, редагувати його, переглянути статус виконання і надіслати запит на видалення. Під час створення або редагування замовлення система виконує перевірку даних, що забезпечує коректність інформації.

Менеджер виконує основні функції управління виробництвом. Він може створювати виробничий графік, створювати виробничі завдання, призначати замовлення у виробництво і формувати звіти. Процес створення виробничого

графіка включає призначення завдань співробітникам і формування структури виробничих операцій.

Менеджер також має можливість формувати різні типи звітів:

– звіт по виконаним замовленням;

– звіт по завершеним завданням;

– звіт по наявним замовленням;

– звіт по роботі працівників.

Звіт по роботі працівників може бути додатково використаний для виконання аналізу за допомогою ШІ, що дозволяє оцінити ефективність роботи співробітників і оптимізувати виробничий процес.

Працівник взаємодіє із системою з метою виконання виробничих завдань. Він отримує призначені завдання і оновлює їх статус після виконання.

Варто, також, звернути увагу на процес управління виробничим графіком. Виробничий графік визначає послідовність виконання виробничих завдань, їх тривалість і відповідальних співробітників [20]. Використання ІС дозволяє автоматизувати цей процес і забезпечити ефективне використання ресурсів підприємства.

Процес формування виробничого графіка включає:

– аналіз наявних замовлень;

– визначення доступних ресурсів;

– створення виробничих завдань;

– призначення співробітників;

– контроль виконання.

Життєвий цикл замовлення представлений на рисунку 1.5 у вигляді UML діаграми станів [21].

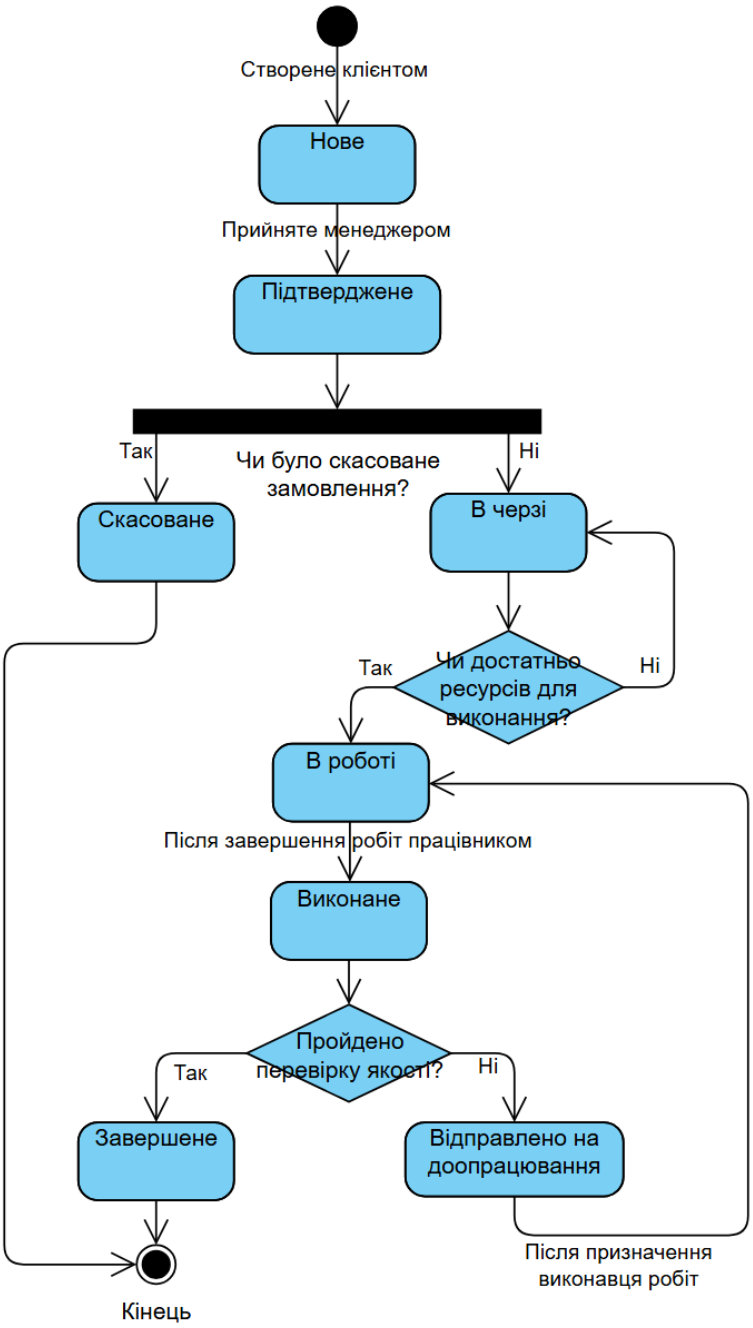


Рисунок 1.5 – Діаграма станів замовлення в системі

Процес починається зі стану «Нове», який означає, що замовлення створене клієнтом. Після перевірки менеджером замовлення переходить у стан «Підтверджене».

На наступному етапі можливі два варіанти розвитку подій: замовлення може бути скасоване і перейти у стан «Скасоване» або замовлення переходить у стан «В черзі», де очікує початку виконання.

Після перевірки наявності необхідних ресурсів замовлення переходить у стан «В роботі», де здійснюється його виконання співробітниками.

Після завершення виконання замовлення переходить у стан «Виконане», де здійснюється перевірка якості виконаних робіт.

У разі успішного проходження перевірки замовлення переходить у стан «Завершене», що означає повне завершення процесу виконання.

У разі виявлення помилок замовлення переходить у стан «Відправлено на доопрацювання», після чого повторно виконується співробітником і знову проходить перевірку.

Основні стани замовлення наведено в таблиці 1.4.

Таблиця 1.4 – Основні стани замовлення

Стан	Характеристика
Нове	Замовлення створене клієнтом
Підтверджене	Замовлення прийняте менеджером
В черзі	Замовлення очікує виконання
В роботі	Замовлення виконується
Виконане	Робота виконана
Завершене	Замовлення завершене
Скасоване	Замовлення скасоване
На доопрацюванні	Замовлення потребує виправлення

Аналіз UML Use Case діаграми і UML діаграми станів показав, що використання ІС дозволяє формалізувати процес управління замовленнями, визначити ролі користувачів і забезпечити контроль життєвого циклу замовлення.

Також ІС забезпечує інтеграцію всіх учасників виробничого процесу і дозволяє підвищити ефективність управління підприємством.

Узагальнюючи, використання ІС дозволяє забезпечити:

– автоматизацію управління замовленнями;

– централізоване зберігання даних у БД;

– автоматичне формування виробничого графіка;

– контроль виконання замовлень;

– аналіз ефективності роботи співробітників із використанням ШІ.

2 ВИБІР ТА ОБҐРУНТУВАННЯМ ТЕХНІЧНИХ ЗАСОБІВ

2.1 Теоретичні основи автоматичного управління процесами диспетчеризації виробничих завдань

Теорія автоматичного управління є науковою основою побудови сучасних ІС, зокрема АСУП та АСУТП, і визначає принципи створення систем, які забезпечують цілеспрямоване функціонування об’єкта управління без безпосереднього втручання людини або з його мінімальною участю. У контексті виробничого підприємства об’єктом управління є процес виконання виробничих завдань, а системою управління виступає ІС, яка забезпечує планування, розподіл і контроль виконання цих завдань.

Основною метою автоматичного управління є забезпечення досягнення заданого результату шляхом формування керуючих впливів на об’єкт управління на основі аналізу інформації про його поточний стан [22]. У виробничих системах таким результатом є своєчасне і якісне виконання замовлень, а керуючим впливом – виробничі завдання, сформовані АСУП.

Основні елементи системи автоматичного управління виробництвом наведено в таблиці 2.1.

Таблиця 2.1 – Основні елементи системи автоматичного управління виробництвом

Елемент	Опис
Об’єкт управління	Процес виконання виробничих завдань
Система управління	АСУП
Керуючий вплив	Виробничі завдання
Зворотний зв’язок	Дані про виконання завдань
Результат	Виконане замовлення

Структура системи автоматичного управління виробничими завданнями представлена на рисунку 2.1.

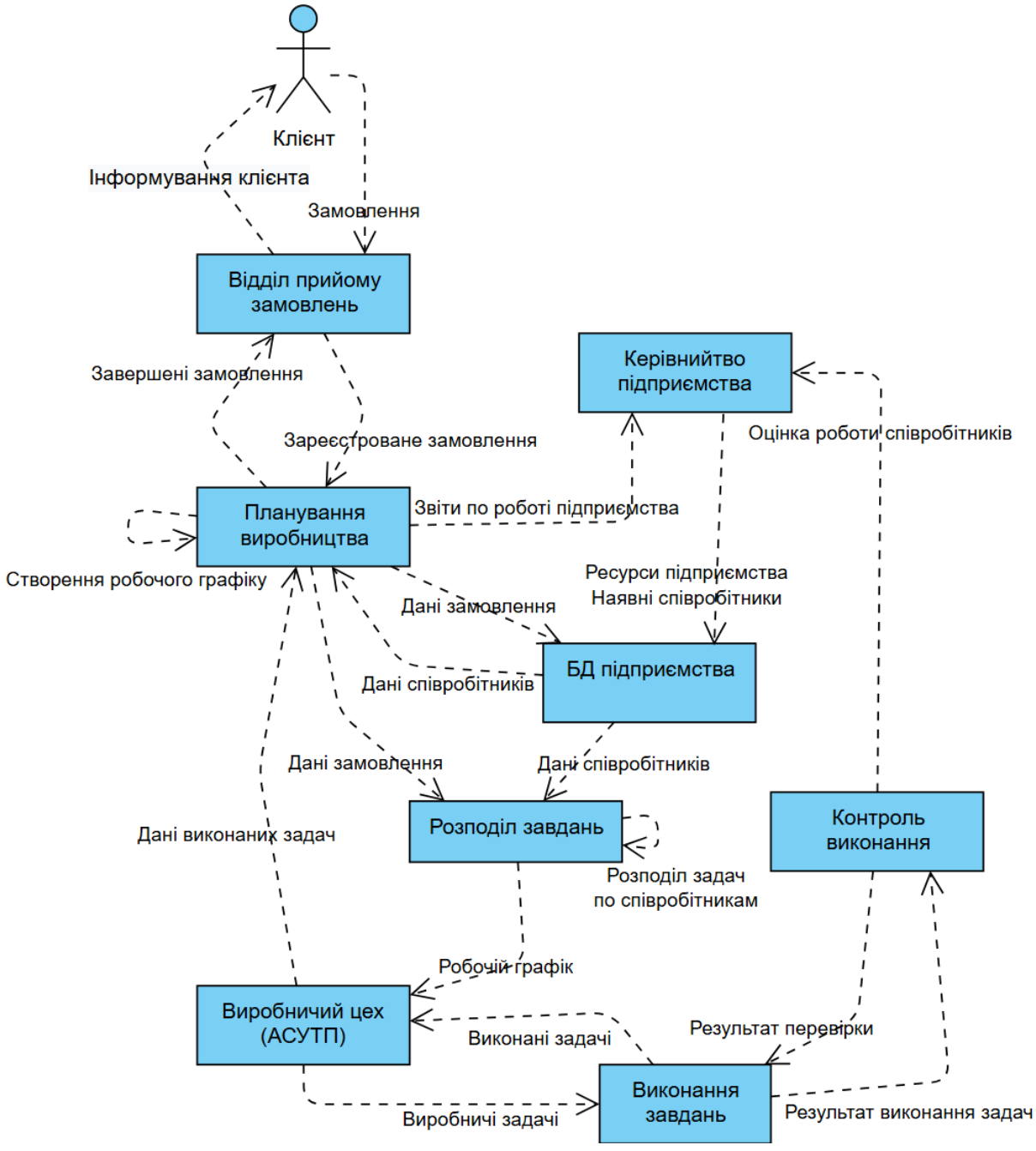


Рисунок 2.1 – Структурна схема процесу управління виробничими завданнями

На схемі показано повний цикл управління виробничим процесом, який включає формування замовлення, планування виробництва, розподіл завдань і контроль їх виконання.

Початковим елементом системи є клієнт, який формує замовлення. Замовлення надходить до відділу прийому замовлень, де виконується його реєстрація і передача до підсистеми планування виробництва. Після реєстрації замовлення дані передаються до БД підприємства, яка забезпечує централізоване зберігання інформації.

Підсистема планування виробництва використовує дані замовлення і інформацію про доступні ресурси підприємства, зокрема дані співробітників, які зберігаються у БД, для формування виробничого графіка. Сформований графік передається до підсистеми розподілу завдань, яка виконує призначення виробничих завдань конкретним співробітникам.

Далі виробничі завдання передаються до виробничого цеху, який представлений АСУТП. На цьому етапі здійснюється безпосереднє виконання виробничих операцій.

Після виконання завдань результати роботи передаються до підсистеми виконання завдань, де фіксується інформація про виконання. Ці дані передаються до підсистеми контролю виконання, яка здійснює перевірку результатів виконання завдань.

Результати контролю передаються до керівництва підприємства, яке використовує цю інформацію для оцінки ефективності роботи співробітників і прийняття управлінських рішень.

Що стосовно автоматичного управління, одним із ключових принципів є принцип зворотного зв’язку, який забезпечує контроль і коригування процесу управління. У виробничій ІС зворотний зв’язок реалізується через передачу наступних даних:

– статус виконання завдань;

– інформація про виконані роботи;

– інформація про продуктивність співробітників;

– результати контролю виконання.

Ця інформація передається до АСУП, де виконується аналіз ефективності виконання завдань і, при необхідності, виконується коригування виробничого графіка.

У сучасних ІС також застосовуються принципи автоматичного управління [23]. Вони реалізуються за допомогою програмних і технічних засобів, які забезпечують автоматичний контроль і управління виробничими процесами. АСУП виконує функції управління і прийняття рішень, використовуючи дані, що зберігаються у БД. АСУТП забезпечує виконання виробничих операцій і передачу інформації про їх виконання до АСУП. Взаємодія між АСУП і АСУТП здійснюється через API, що дозволяє забезпечити інтеграцію управлінських і технологічних процесів.

2.2 Вибір та обґрунтування мови програмування, бібліотек та технологій

Розробка онлайн-ІС управління замовленнями та виробничим графіком підприємства потребує використання сучасних технологій, які забезпечують високу швидкість роботи, надійність, масштабованість та можливість інтеграції з АСУП, АСУТП та ШІ-модулями. Основною вимогою є підтримка веб-архітектури, оскільки система реалізується на основі сайту та повинна забезпечувати віддалений доступ клієнтів, менеджерів і співробітників.

Як основну мову програмування обрано JavaScript, оскільки вона є стандартом для створення веб-додатків і забезпечує роботу як клієнтської, так і серверної частини системи [24]. Це дозволяє використовувати єдину технологію для всієї ІС, що значно спрощує розробку, супровід і масштабування.

JavaScript забезпечує:

– підтримку HTTP та API;

– ефективну роботу з JSON (а це саме той формат, з яким працює БД);

– швидку обробку запитів користувачів;

– просту інтеграцію з БД;

– можливість інтеграції ШІ-модулів.

Для зберігання інформації використовується реляційна БД з підтримкою SQL. Це обумовлено тим, що виробнича ІС працює зі структурованими даними, такими як замовлення, виробничі завдання, співробітники та виробничі графіки.

SQL забезпечує:

– надійне зберігання даних;

– швидкий доступ до інформації (особливо за великих об’ємів даних);

– підтримку CRUD (create, read, update, delete) операцій;

– цілісність даних.

На рисунку 2.2 представлена UML Component-діаграма, яка відображає основні програмні компоненти ІС та їх взаємодію.

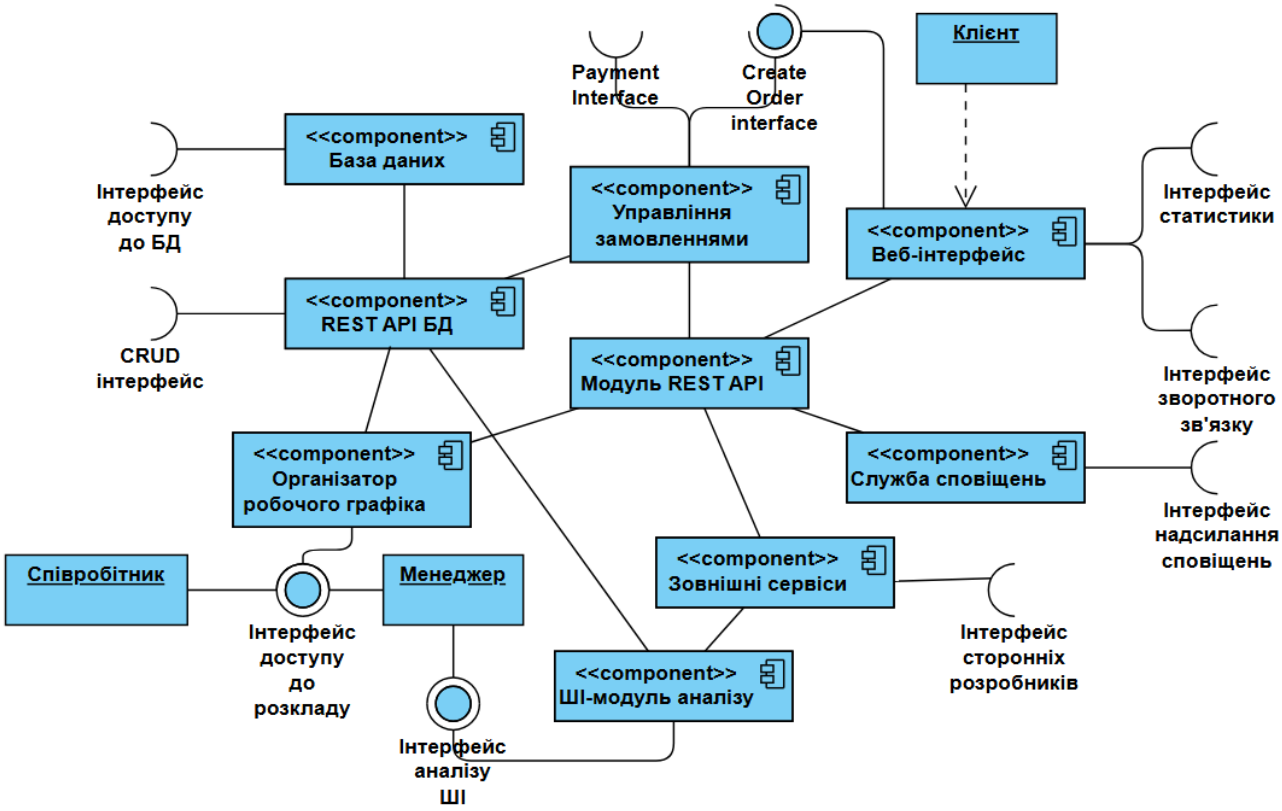


Рисунок 2.2 – Схема взаємодії компонентів (UML Component-діаграма)

Основним компонентом є веб інтерфейс, який забезпечує UX/UI та взаємодію користувача з системою. Через цей компонент клієнт створює замовлення, переглядає їх статус і отримує інформацію. Веб інтерфейс також забезпечує доступ до функцій управління виробничими процесами для менеджерів і співробітників підприємства. Додатково реалізовано адаптивний дизайн, що дозволяє працювати із системою на різних типах пристроїв без втрати функціональності.

Компонент «Управління замовленнями» виконує управління замовленнями, їх створення, зміну статусу та передачу в БД. Він взаємодіє з модулем «REST API», який забезпечує обмін даними між клієнтською частиною та серверною логікою.

Компонент «Організатор робочого графіку» відповідає за створення виробничого графіка та розподіл завдань між співробітниками. Це є ключовою функцією ІС управління виробництвом.

Компонент «REST API БД» забезпечує взаємодію з БД та виконання SQL запитів [25].

Особливе значення має ШІ-модуль аналізу, який виконує аналіз ефективності роботи співробітників і виробничих процесів. Це дозволяє підвищити ефективність управління підприємством. ШІ-модуль може використовувати накопичені історичні дані для прогнозування ефективності виконання майбутніх виробничих завдань та виявлення потенційних ризиків.

Компонент «Служба сповіщень» забезпечує інформування користувачів про стан замовлень.

На рисунку 2.3 представлена UML Deployment-діаграма, яка показує фізичне розміщення компонентів системи.

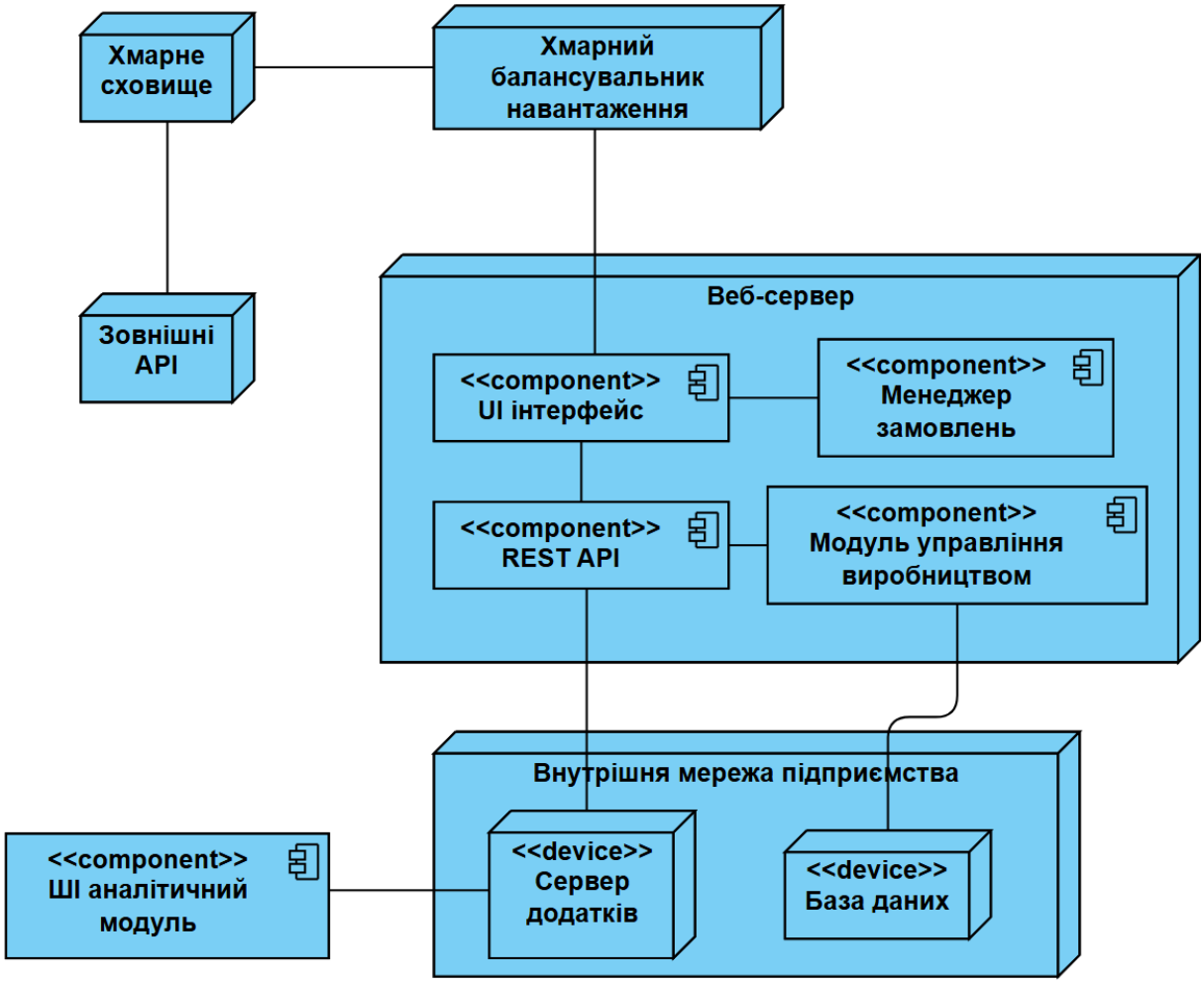


Рисунок 2.3 – UML Deployment діаграма автоматизованої системи

Основним вузлом є веб-сервер, на якому працює основна логіка ІС, REST API та UI інтерфейс.

Взаємодія з БД здійснюється через сервер додатків та сервер БД, що входять до внутрішньої мережі підприємства. Це забезпечує захист і централізоване зберігання даних.

Окремим вузлом є ШІ аналітичний модуль, який отримує дані з БД і виконує їх аналіз.

Для забезпечення стабільності роботи використовується хмарний балансувальник навантаження, який розподіляє запити між серверами.

На рисунку 2.4 представлена схема взаємодії клієнта і сервера.

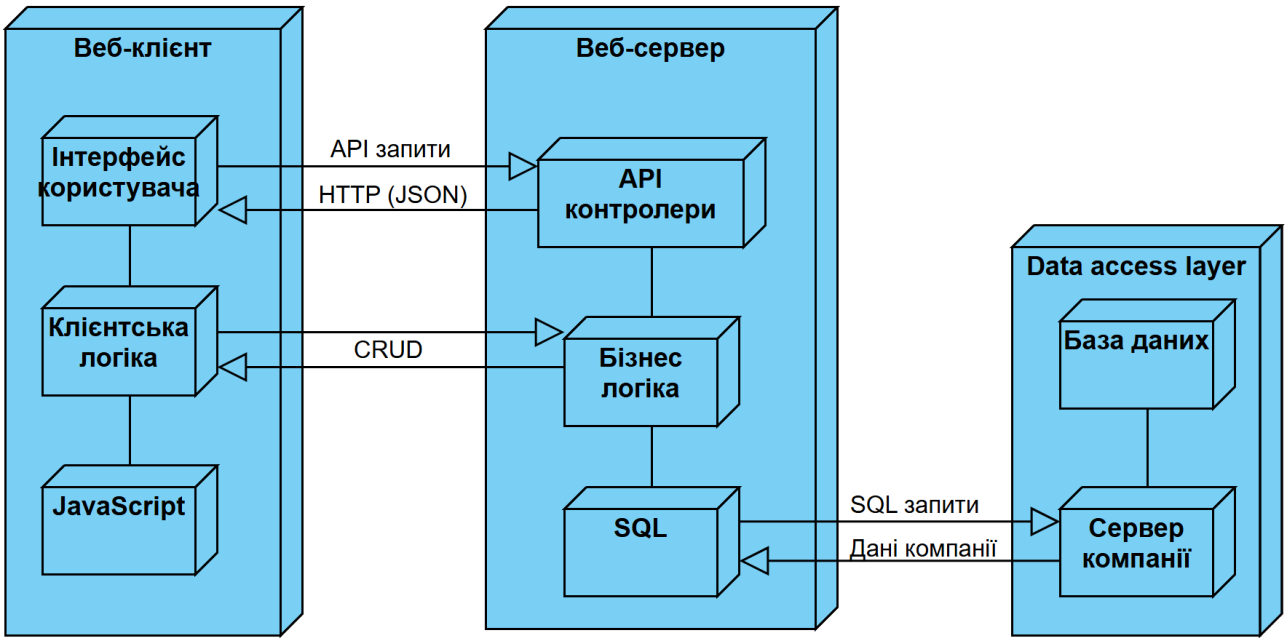


Рисунок 2.4 – Схема взаємодії клієнтської та серверної частини системи

Клієнтська частина реалізована на JavaScript і забезпечує UX/UI та взаємодію з користувачем [26]. Вона формує HTTP запити у форматі JSON і передає їх через API. Додатково клієнтська частина виконує первинну обробку та валідацію введених користувачем даних перед їх передачею на сервер.

Серверна частина містить API контролери і бізнес-логіку, які обробляють запити користувача і формують SQL запити до БД [27].

БД виконує SQL запити і повертає результати на сервер, який передає їх клієнту.

Використання REST API та JSON обумовлено тим, що ці технології є стандартом для веб-ІС і забезпечують швидку передачу даних, просту інтеграцію, сумісність з іншими ІС та підтримку розподіленої архітектури (що особливо важливо для інтеграції з АСУП та АСУТП). Такий підхід також спрощує масштабування системи та підключення додаткових модулів або зовнішніх сервісів у майбутньому.

2.3 Структурна схема роботи додатку

Структурна схема роботи додатку визначає логіку функціонування онлайн-ІС управління замовленнями та виробничим графіком підприємства, порядок взаємодії її компонентів, АСУП, БД та ШІ-модуля. Основною метою структурної організації ІС є забезпечення автоматизованого управління життєвим циклом замовлення, планування виробництва, контролю виконання завдань та аналізу ефективності роботи співробітників [28].

Структура роботи ІС базується на принципах централізованого управління даними, модульності, інтеграції з АСУП та використання ШІ для аналітичної підтримки прийняття управлінських рішень.

На рисунку 2.5 представлена декомпозиція DFD-діаграми системи після інтеграції з АСУП. Дана схема відображає повний цикл обробки замовлення.

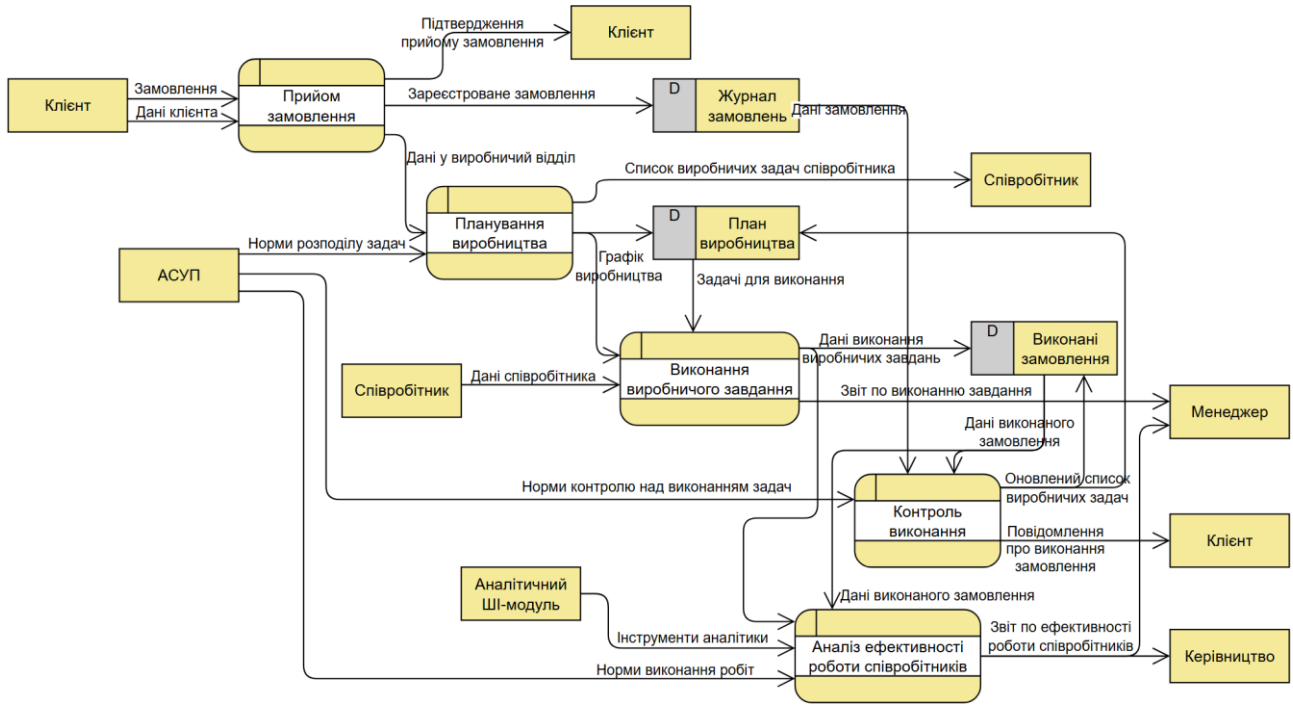


Рисунок 2.5 – Декомпозиція DFD діаграми системи після інтеграції АСУП

Основними процесами системи є:

– прийом замовлення, коли клієнт передає в систему замовлення та свої дані. Система виконує перевірку інформації та реєструє замовлення у сховищі даних «Журнал замовлень»;

– планування виробництва, коли зареєстроване замовлення передається у підсистему планування. Додатково використовуються дані з АСУП (норми розподілу задач, виробничі ресурси та норми виконання);

– виконання виробничого завдання, коли система передає співробітнику список виробничих завдань. Після виконання співробітник передає результат виконання в систему. Дані зберігаються у сховищі «Виконані замовлення»;

– контроль виконання. Система контролює виконання завдань, використовуючи: дані виконаних робіт та нормативи виконання з АСУП. Результатом є оновлення статусу замовлення та повідомлення клієнта;

– аналіз ефективності роботи співробітників в ході якого формується звіт ефективності роботи співробітників для керівництва.

На рисунку 2.6 представлено DFD-діаграму підсистеми аналізу ефективності роботи співробітників, яка призначена для автоматизованої оцінки продуктивності персоналу на основі фактичних даних виконання виробничих завдань та результатів аналітики ШІ.



Рисунок 2.6 – DFD діаграма підсистеми аналізу ефективності роботи співробітників

Вхідними даними для підсистеми є дані виконання виробничих завдань, що надходять із зовнішнього процесу виконання виробничого завдання, нормативи виконання робіт, які надає АСУП, а також інструменти аналізу, що надходять від аналітичного ШІ-модуля. На основі цих даних виконується процес збору інформації про виконання завдань, після чого система виконує розрахунок показників ефективності за встановленими формулами [29]. Додатково підсистема здійснює перевірку повноти та коректності отриманих даних перед початком аналітичної обробки. Це дозволяє зменшити ймовірність виникнення помилок під час розрахунку показників ефективності та формування звітності. Усі отримані дані зберігаються у БД для подальшого аналізу, формування статистики та побудови історичних звітів. Такий підхід забезпечує можливість відстеження динаміки продуктивності співробітників у різні часові проміжки.

Паралельно ШІ-модуль виконує інтелектуальний аналіз отриманих даних і формує аналітичні показники ефективності співробітника. Отримані розрахункові та аналітичні дані об’єднуються у процесі зіставлення аналітичних даних, що дозволяє визначити загальний рівень ефективності роботи співробітника. Для підвищення точності аналізу ШІ-модуль може враховувати історичні показники виконання завдань, складність робіт та середній час їх виконання. Це дозволяє формувати більш об’єктивну оцінку продуктивності персоналу та виявляти тенденції зміни ефективності роботи.

Результатом роботи підсистеми є формування звіту ефективності роботи співробітників, який передається менеджеру та використовується керівництвом підприємства для прийняття управлінських рішень. Сформовані звіти можуть використовуватись для оцінки продуктивності персоналу, планування навантаження та виявлення проблемних етапів виробничого процесу.

На рисунку 2.7 представлена UML Sequence-діаграма, яка відображає послідовність взаємодії клієнта, веб-додатку, системи та БД під час створення та виконання замовлення.

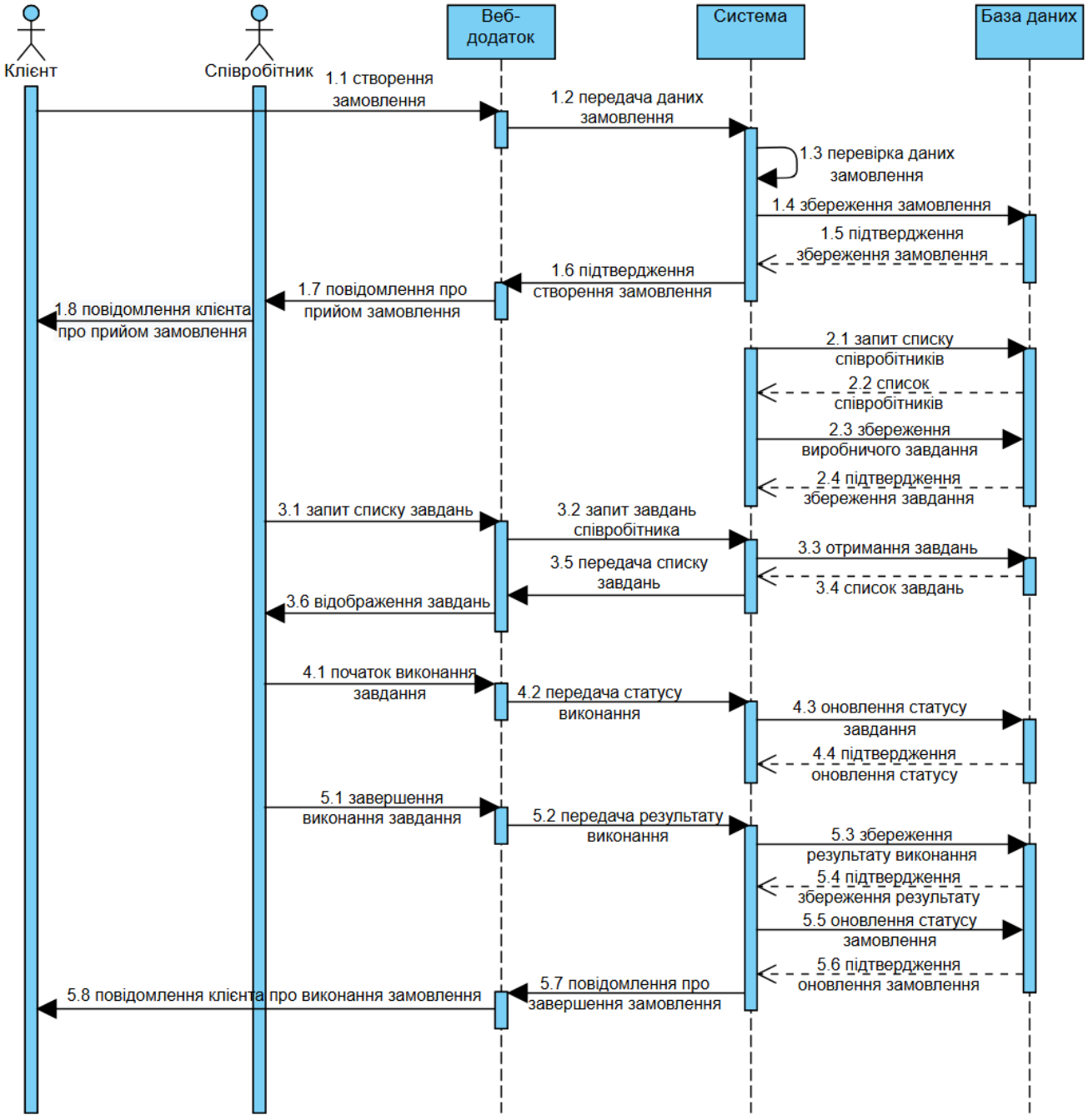


Рисунок 2.7 – UML Sequence-діаграма процесу створення та виконання замовлення

Процес починається з того, що клієнт через веб-додаток створює замовлення, після чого веб-додаток передає відповідні дані в систему. Система виконує перевірку отриманої інформації, формує запис замовлення та зберігає його в БД.

Після підтвердження збереження система передає відповідь у веб-додаток, який повідомляє клієнта про успішне прийняття замовлення.

Далі система отримує з БД інформацію про співробітників, формує виробниче завдання і зберігає його у БД. Співробітник через веб-додаток отримує список призначених завдань, після чого виконує виробниче завдання і передає результат виконання у систему.

На рисунку 2.8 представлена UML Sequence-діаграма роботи ШІ-модуля.

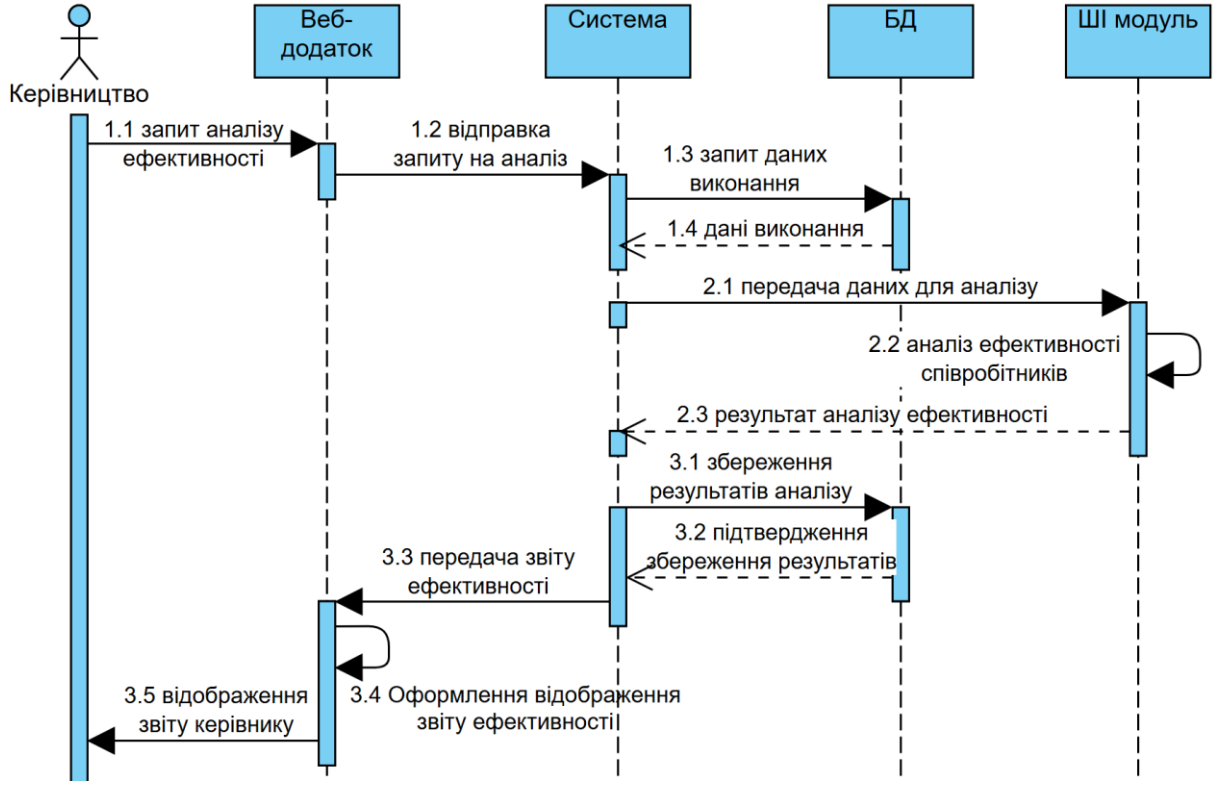


Рисунок 2.8 – UML Sequence-діаграма роботи модулю аналізу з використанням ШІ

Процес починається з того, що керівництво підприємства через веб-додаток формує запит аналізу ефективності роботи співробітників. Веб-додаток, в цей час, звертається до БД для отримання даних про виконання виробничих завдань.

Отримані дані система передає у ШІ-модуль, який виконує аналіз ефективності роботи співробітників та формує відповідні результати. Після цього

система зберігає результати аналізу у БД та передає сформований звіт у веб-додаток.

На завершальному етапі веб-додаток відображає звіт керівництву підприємства, що дозволяє оцінити ефективність роботи персоналу та прийняти відповідні управлінські рішення.

2.4 Параметри оцінки ІС

Для оцінки загального часу проходження інформації між модулями онлайн-ІС використовується формула сумарного часу обробки даних. Вона дозволяє визначити швидкість взаємодії між клієнтською частиною, сервером та БД [30].

Час проходження інформації між модулями системи визначається за формулою (2.1):

$$T_{sys} = T_{cli} - T_{api} - T_{db}, \quad (2.1)$$

де T_{cli} – час обробки на клієнтській частині;

T_{api} – час обробки REST API;

T_{db} – час виконання SQL-запиту в БД.

Для розробленої ІС були отримані наступні значення:

– $T_{cli} = 0{,}18$ с;

– $T_{api} = 0{,}24$ с;

– $T_{db} = 0{,}11$ с.

Тоді:

$$T_{sys} = 0{,}18 + 0{,}24 + 0{,}11 = 0{,}53 \text{ с.}$$

Отриманий результат свідчить про високу швидкодію онлайн-ІС та відповідність сучасним вимогам до web-додатків.

Для оцінки стабільності роботи онлайн-системи використовується коефіцієнт доступності системи. Показник демонструє частку часу, протягом якого система була доступною для користувачів.

Коефіцієнт доступності системи обчислюється за формулою (2.2):

$$A = \frac{T_{\text{р}}}{T_{\text{р}} - T_{\text{в}}}, \tag{2.2}$$

де $T_{\text{р}}$ – час стабільної роботи системи;

$T_{\text{в}}$ – час відмов або простоїв.

Для аналізу результативності роботи підприємства застосовується коефіцієнт успішного виконання замовлень. Формула дозволяє визначити відсоток замовлень, які були успішно завершені без порушення виробничого процесу. Формула розрахунку цього коефіцієнту (2.3):

$$K_{\text{усп}} = \frac{N_{\text{у}}}{N_{\text{з}}} \cdot 100, \tag{2.3}$$

де $N_{\text{у}}$ – кількість успішно завершених замовлень;

$N_{\text{з}}$ – загальна кількість замовлень.

Для оцінки швидкості роботи онлайн-системи управління замовленнями використовується середній час обробки замовлення. Формула дозволяє визначити середню тривалість виконання замовлення від моменту створення до завершення.

Середній час обробки замовлення обчислюється за формулою (2.4):

$$T_{ord} = \frac{\sum_{n}^{i=1}(T_{\text{з}} - T_{\text{с}})}{n}, \tag{2.4}$$

де $T_{\text{с}}$ – дата та час створення замовлення;

$T_{\text{з}}$ – дата та час завершення замовлення;

n – кількість завершених замовлень.

Для тестового періоду у роботи ІС сумарний час виконання 150 замовлень становив 794 дні (19052,8 годин роботи), а кількість завершених замовлень (n = 104).

Тоді:

$$T_{ord} = \frac{794}{104} = 7{,}63.$$

Отриманий результат відповідає нормальним показникам роботи виробничої ІС – виконання замовлення до 24-х робочих днів.

Для оцінки ефективності використання ресурсів підприємства застосовується коефіцієнт використання ресурсів (2.5). Показник дозволяє визначити ступінь залучення доступних виробничих ресурсів у процес виконання замовлень.

$$K_{\text{р}} = \frac{R_{\text{в}}}{R_{\text{д}}}, \tag{2.5}$$

де $R_{\text{в}}$ – кількість задіяних ресурсів;

$R_{\text{д}}$ – загальна кількість доступних ресурсів.

3 РОЗРОБКА СИСТЕМИ КОНТРОЛЮ ПРОЦЕСУ ВИКОНАННЯ РОБІТНИКОМ ВИРОБНИЧИХ ЗАВДАНЬ

3.1 Блок-схема роботи ПЗ

Блок-схеми часто застосовуються при проєктуванні ПЗ. Вони виступають зручним інструментом інженерного підходу до розробки ІС, оскільки вони забезпечують формалізований опис алгоритмів відповідно до таких стандартів як ISO та UML Activity/Flow. Блок-схеми дозволяють відобразити не лише послідовність виконання процесів, але й логіку прийняття рішень, обробку помилок, альтернативні сценарії та точки взаємодії з БД, API і зовнішніми підсистемами [31]. Важливою перевагою є можливість перевірки коректності бізнес-логіки до етапу реалізації, що знижує ризики помилок у ПЗ. Додатково блок-схеми спрощують документування програмної системи та подальший супровід програмного забезпечення після впровадження. Їх використання дозволяє швидше адаптувати нових розробників до структури проєкту та логіки функціонування окремих модулів системи.

Крім того, блок-схеми виступають універсальним засобом комунікації між аналітиками, розробниками та замовниками, що особливо актуально для складних ІС, таких як АСУП з інтеграцією ШІ. У межах даного проєкту блок-схеми використовуються для опису ключових процесів: авторизації, управління замовленнями, планування виробництва та аналітики. Додатково вони дозволяють візуалізувати взаємозв’язки між окремими компонентами ІС та потоками обробки даних у системі.

Нижче на рисунку 3.1 представлено блок-схему алгоритму авторизації користувача.

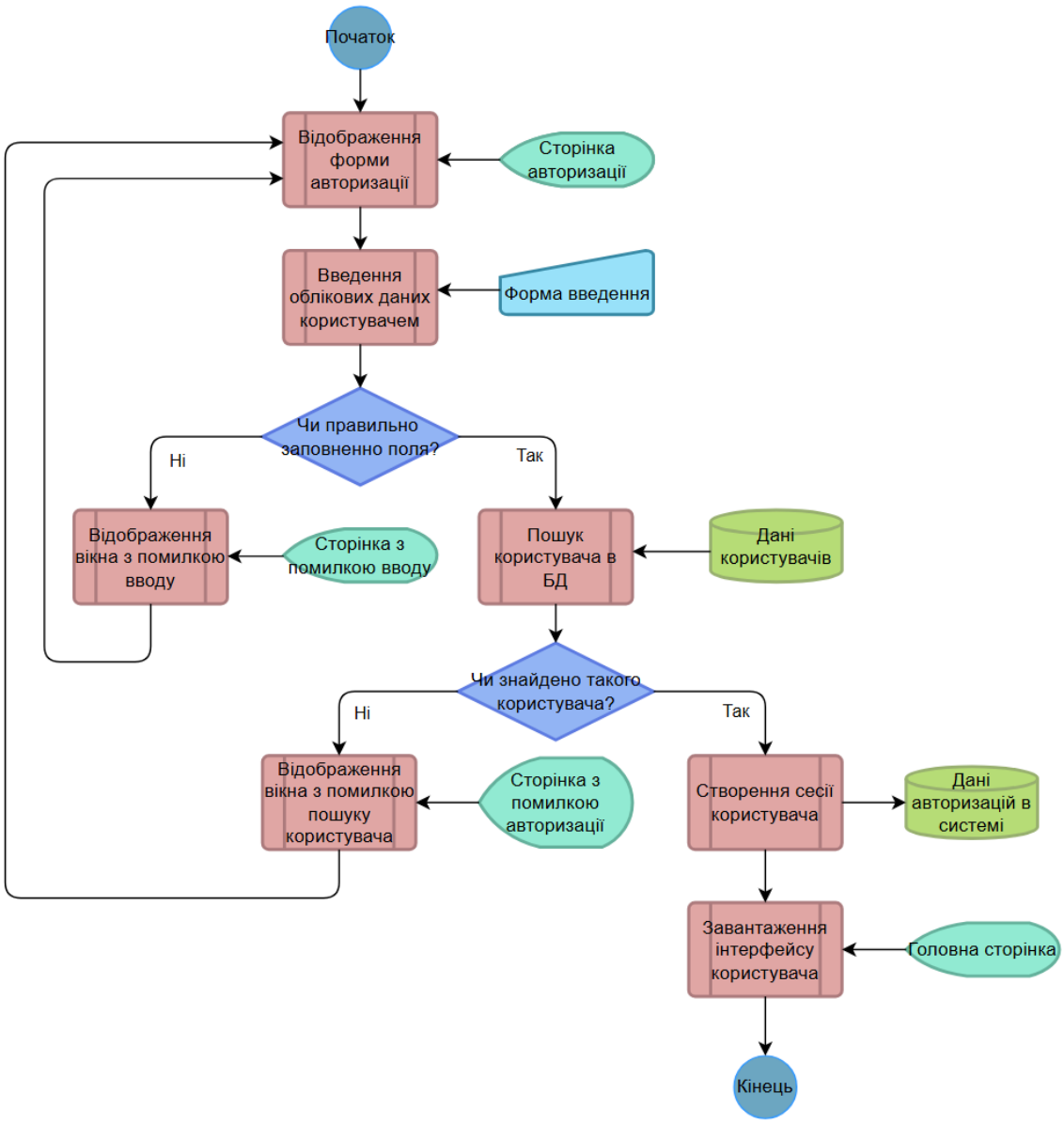


Рисунок 3.1 – Блок-схема алгоритму авторизації користувача

Ця схема демонструє типовий процес аутентифікації з урахуванням перевірки вхідних даних, взаємодії з БД та обробки помилок. Особливістю є наявність двох рівнів перевірки: синтаксичної (коректність заповнення полів) та семантичної (перевірка існування користувача в БД). У схемі чітко виділено точки прийняття рішень (ромби), що відповідають умовам валідації, а також передбачено повернення до попередніх етапів у разі помилки, що відповідає принципам UI/UX.

Завершальним етапом є створення сесії користувача та ініціалізація його робочого середовища, що відображає інтеграцію з механізмами безпеки ІС.

На рисунку 3.2 представлено блок-схему процесу створення замовлення, яка відображає взаємодію користувача з ІС через web-UI та подальшу обробку даних на серверному рівні.

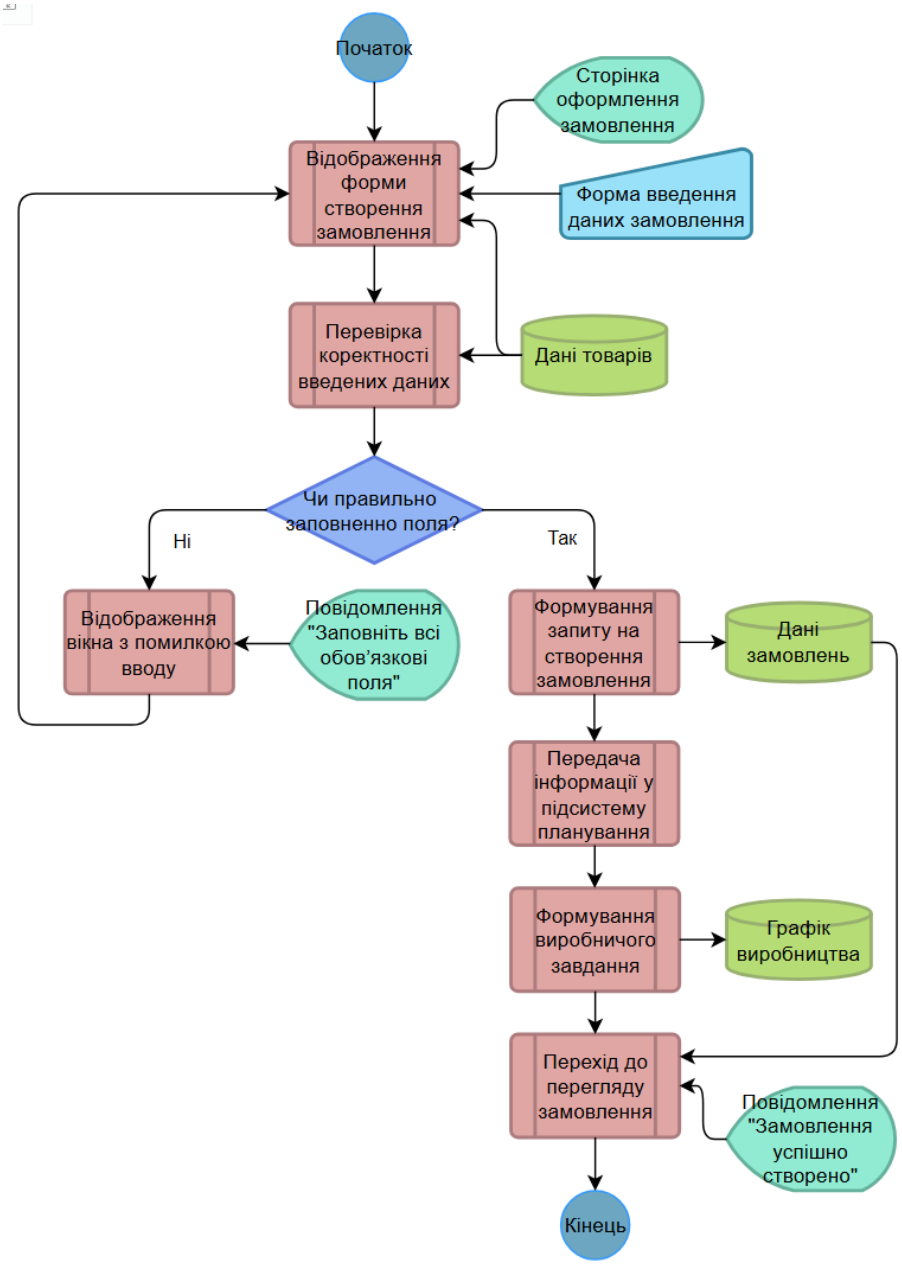


Рисунок 3.2 – Блок-схема процесу створення замовлення

Схема демонструє повний цикл обробки введених даних: від їх перевірки до збереження в БД та ініціації пов’язаних процесів. Важливим аспектом є те, що після створення замовлення відразу запускається механізм інтеграції з підсистемою планування, що забезпечує автоматизацію переходу від бізнес-процесу до виробничого. Таким чином, блок-схема відображає не лише CRUD-операцію створення запису, але й бізнес-логіку взаємодії між модулями ІС.

На рисунку 3.3 зображено блок-схему процесу управління виробничим графіком, яка є центральною з точки зору функціонування АСУП.

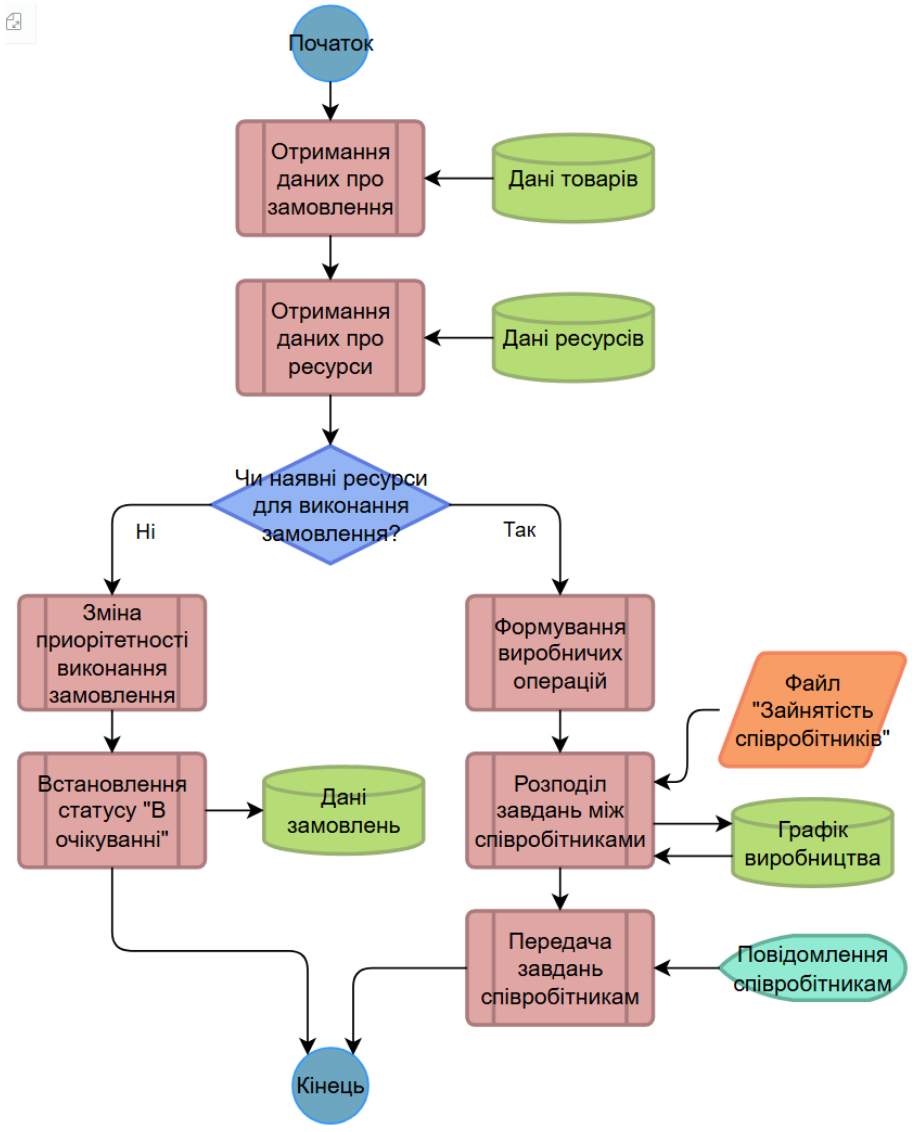


Рисунок 3.3 – Блок-схема процесу управління виробничим графіком

У схемі відображено ключові етапи планування: аналіз вхідних даних, перевірка ресурсів, формування виробничих операцій та розподіл задач. Додатково блок-схема дозволяє простежити логіку прийняття рішень у випадку нестачі ресурсів або зміни пріоритетів виробничих завдань. Це забезпечує більш наочне представлення процесу управління виробничим графіком та взаємодії між окремими модулями системи.

На рисунку 3.4 представлено блок-схему процесу виконання замовлень, яка відображає динамічну поведінку системи під час реалізації виробничих задач.

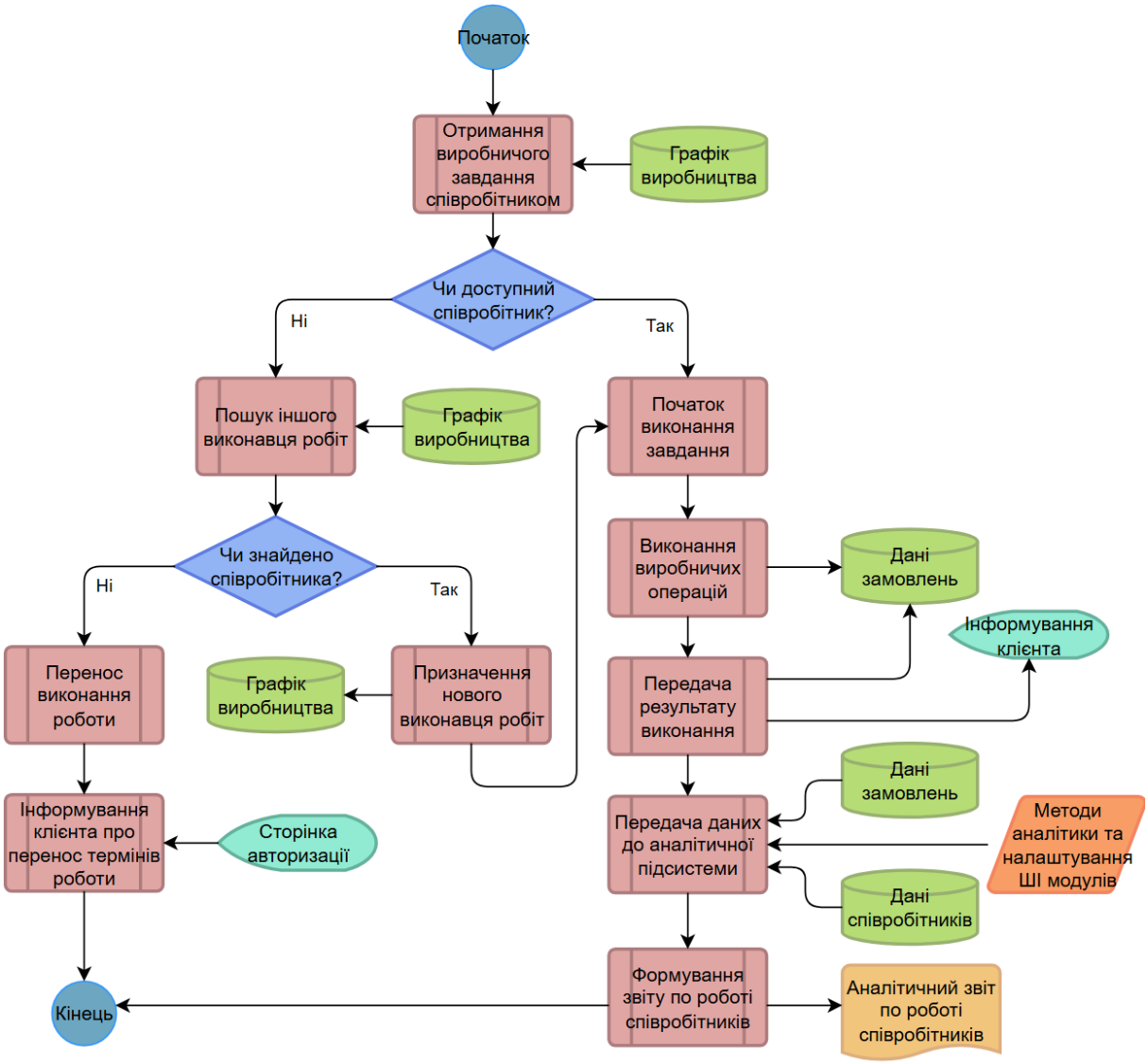


Рисунок 3.4 – Блок-схема процесу виконання замовлень в системі

Схема з рисунку 3.4 включає перевірку доступності співробітника, механізм перепризначення задач та обробку виключних ситуацій, таких як відсутність виконавця. Значною перевагою є інтеграція з аналітичною підсистемою, де результати виконання передаються до ШІ-модуля для подальшого аналізу. Це дозволяє не лише фіксувати факт виконання задач, але й оцінювати ефективність роботи співробітників, формувати рекомендації та підтримувати прийняття управлінських рішень. Завершення процесу супроводжується оновленням стану замовлення та інформуванням клієнта, що забезпечує прозорість роботи ІС.

У цілому, саме ці 4 блок-схеми формують цілісну модель функціонування ПЗ, яка охоплює всі рівні ІС: від користувацького інтерфейсу до рівня БД і аналітики. Вони демонструють узгодженість між бізнес-логікою, даними та процесами управління, які були описані в попередніх розділах.

3.2 Створення програмного забезпечення

Створення ПЗ для онлайн-ІС управління замовленнями та виробничим графіком підприємства виконано з урахуванням принципів модульності, масштабованості та інтеграції з АСУП, БД, API та ШІ. Архітектура системи побудована за клієнт-серверною моделлю, де веб-інтерфейс (UX/UI) взаємодіє з серверною частиною через REST API, а обробка даних виконується з використанням SQL-запитів до БД [32]. Особливу увагу приділено узгодженості структури даних із бізнес-логікою, що забезпечує ефективне управління замовленнями та виробничими процесами.

Після завершення етапів аналізу та проєктування було розпочато безпосередню реалізацію онлайн-ІС управління замовленнями та виробничим графіком підприємства на основі сайту. Першим етапом створення системи стало формування структури frontend-проєкту та налаштування файлової архітектури веб-додатку. Для цього було створено окремі HTML-сторінки для кожного

функціонального модуля системи, а також сформовано структуру папок для стилів, зображень, іконок, шрифтів та тестової БД.

На рисунку 3.5, який наведено нижче, представлено ER-модель БД системи, яка відображає структуру даних та зв’язки між основними сутностями.

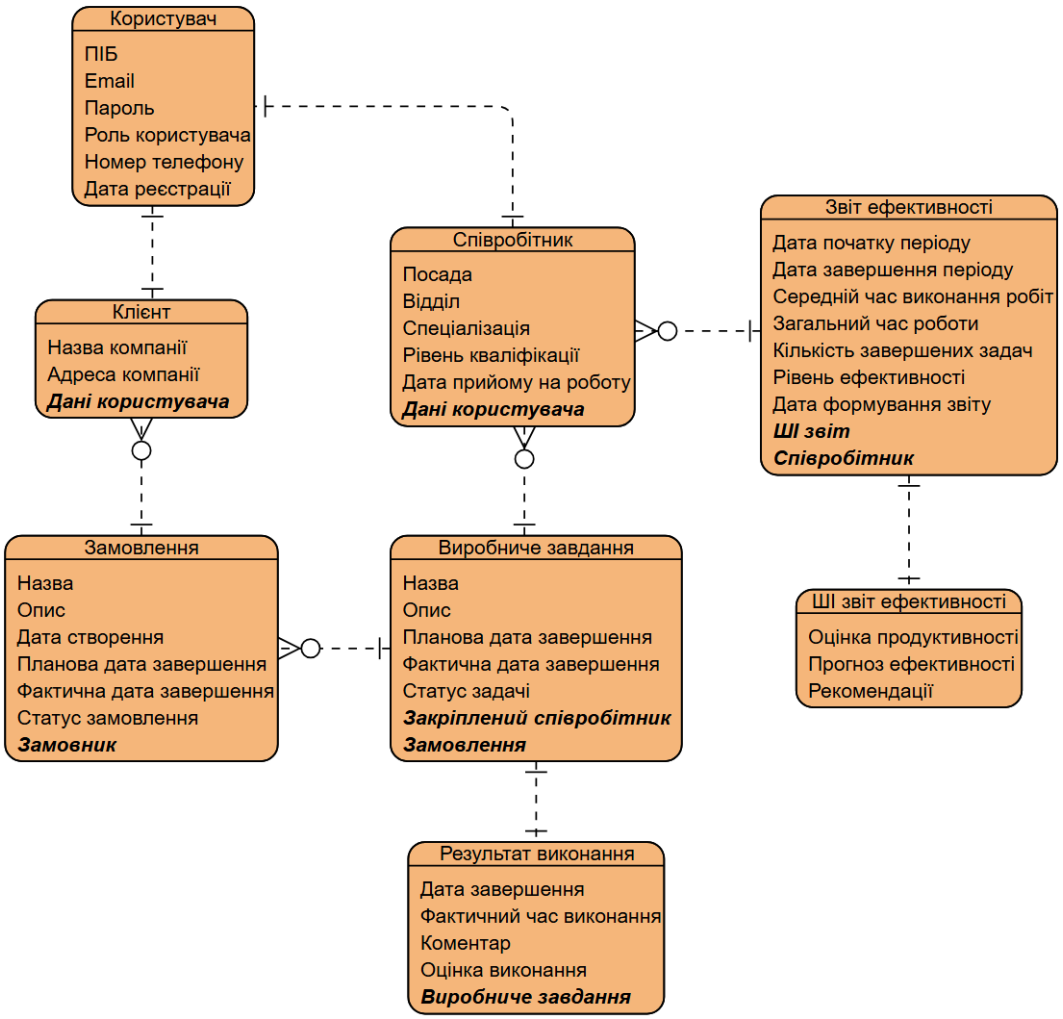


Рисунок 3.5 – ER-модель бази даних системи

Центральною сутністю для роботи із системою є таблиця Users, яка пов’язана з Clients та Employees зв’язками типу 1:1. Сутність Orders пов’язана з клієнтами (1:N) і є джерелом формування Production_tasks, які, у свою чергу, пов’язані зі співробітниками. Результати виконання задач зберігаються у Task_results, що забезпечує фіксацію фактичних даних виконання. Для аналітики використовується

сутність Performance_reports, яка агрегує показники ефективності, а також пов’язана з AI_performance_reports, що містить результати ШІ-аналізу.

На рисунку 3.6 представлено діаграму класів програмної системи, яка демонструє відповідність між логічною моделлю БД та програмною реалізацією. Кожна сутність БД відображається у вигляді окремого класу, а зв’язки між ними реалізуються через асоціації та залежності. Класи взаємодіють через API-контролери, що реалізують CRUD-операції, а бізнес-логіка винесена в окремий рівень, що відповідає принципам багатошарової архітектури.

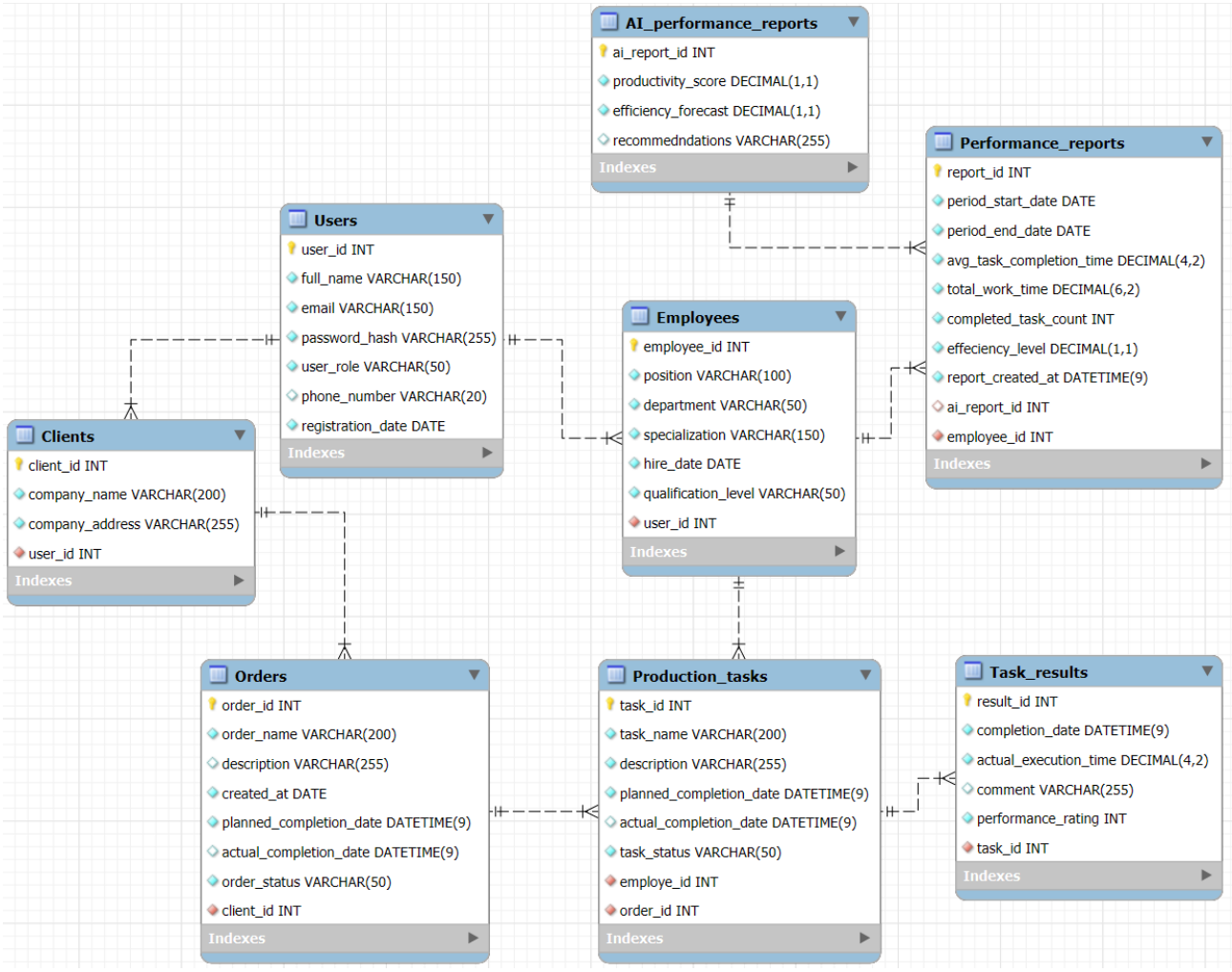


Рисунок 3.6 – Діаграма класів програмної системи

Опис полів сутностей БД наведено у таблицях 3.1 – 3.8.

Таблиця 3.1 – Сутність Users

Назва	Опис	Тип даних	NULL
user_id	унікальний ідентифікатор користувача	INT PK	Ні
full_name	повне ім’я користувача	VARCHAR(150)	Ні
email	унікальна електронна адреса	VARCHAR(150)	Ні
password_hash	хешований пароль користувача	VARCHAR(255)	Ні
user_role	роль користувача в системі (клієнт, менеджер, співробітник)	VARCHAR(50)	Ні
phone_number	контактний номер телефону	VARCHAR(20)	Так
registration_date	дата та час реєстрації	DATE	Ні

Таблиця 3.2 – Сутність Clients

Назва	Опис	Тип даних	NULL
client_id	унікальний ідентифікатор клієнта	INT PK	Ні
company_name	офіційна назва компанії клієнта	VARCHAR(200)	Ні
company_address	юридична або фактична адреса компанії	VARCHAR(255)	Ні
user_id	посилання на обліковий запис користувача	INT FK	Ні

Таблиця 3.3 – Сутність Orders

Назва	Опис	Тип даних	NULL
order_id	унікальний ідентифікатор замовлення	INT PK	Ні
order_name	назва замовлення	VARCHAR(200)	Ні
description	детальний опис замовлення	VARCHAR(255)	Так
created_at	дата створення замовлення	DATE	Ні
planned_completion_date	запланована дата завершення	DATETIME	Ні
actual_completion_date	фактична дата завершення	DATETIME	Так
order_status	поточний статус замовлення	VARCHAR(50)	Ні
client_id	ідентифікатор клієнта-замовника	INT FK	Ні

Таблиця 3.4 – Сутність AI_performance_reports

Назва	Опис	Тип даних	NULL
ai_report_id	унікальний ідентифікатор ШІ-звіту	INT PK	Ні
productivity_score	оцінка продуктивності	DECIMAL(1,1)	Ні
efficiency_forecast	прогнозована ефективність	DECIMAL(1,1)	Ні
recommendations	текстові рекомендації	VARCHAR(255)	Так

Таблиця 3.5 – Сутність Employees

Назва	Опис	Тип даних	NULL
employee_id	унікальний ідентифікатор співробітника	INT PK	Ні
position	посада співробітника в компанії	VARCHAR(100)	Ні
department	відділ співробітника	VARCHAR(50)	Ні
specialization	професійна спеціалізація	VARCHAR(150)	Ні
hire_date	дата прийому на роботу	DATE	Ні
qualification_level	рівень кваліфікації або досвіду	VARCHAR(50)	Ні
user_id	обліковий запис користувача	INT FK	Ні

Таблиця 3.6 – Сутність Production_tasks

Назва	Опис	Тип даних	NULL
task_id	унікальний ідентифікатор виробничого завдання	INT PK	Ні
task_name	назва виробничого завдання	VARCHAR(200)	Ні
description	детальний опис задачі	VARCHAR(255)	Так
planned_completion_date	запланований строк виконання	DATETIME	Ні
actual_completion_date	фактичний строк виконання	DATETIME	Так
task_status	поточний статус виконання задачі	VARCHAR(50)	Ні
employee_id	відповідальний співробітник	INT FK	Ні
order_id	ідентифікатор замовлення	INT FK	Ні

Таблиця 3.7 – Сутність Performance_reports

Назва	Опис	Тип даних	NULL
report_id	унікальний ідентифікатор звіту	INT PK	Ні
period_start_date	дата початку звітного періоду	DATE	Ні
period_end_date	дата завершення звітного періоду	DATE	Ні
avg_task_completion_time	середній час виконання задач	DECIMAL(4,2)	Ні
total_work_time	сумарний відпрацьований час	DECIMAL(6,2)	Ні
completed_tasks_count	кількість завершених задач	INT	Ні
efficiency_level	інтегральний показник ефективності	DECIMAL(1,1)	Ні
report_created_at	дата формування звіту	DATETIME	Ні
ai_report_id	посилання на ШІ-аналітичний звіт	INT FK	Так
employee_id	ідентифікатор співробітника	INT FK	Ні

Таблиця 3.8 – Сутність Task_results

Назва	Опис	Тип даних	NULL
result_id	унікальний ідентифікатор результату	INT PK	Ні
completion_date	дата завершення виконання задачі	DATETIME	Ні
actual_execution_time	фактичний час виконання задачі (години)	DECIMAL(4,2)	Ні
comment	текстовий коментар щодо виконання	VARCHAR(255)	Так
performance_rating	оцінка якості виконання задачі	INT	Ні
task_id	посилання на виробниче завдання	INT FK	Ні

Після визначення всіх сутностей та залежностей було створено тестову БД у файлі database.json. У БД були реалізовані сутності users, clients, orders, employees, production_tasks, task_results, performance_reports та ai_performance_reports. Для кожної сутності були сформовані відповідні атрибути, зв’язки та тестові записи для подальшої інтеграції frontend-частини із серверною логікою.

Розроблене ПЗ базується на чітко структурованій БД, узгодженій із логікою роботи ІС та UML-моделями. Використання такої архітектури дозволяє ефективно реалізувати CRUD-операції, забезпечити масштабованість системи та інтеграцію з аналітичними модулями ШІ.

Реалізацію ключових компонентів ПЗ у вигляді програмного коду представлено в додатку А. В цьому додатку містить JS-код роботи з БД та відображення даних на сторінці з реалізацією сортування та пагінації.

3.3 Робота додатку

Наступним етапом стала реалізація базової структури веб-інтерфейсу. Було створено бічне меню навігації, верхню панель керування та основний контейнер для відображення інформації. Навігація системи була розподілена на окремі категорії: керування, ресурси, аналітика та системні модулі. Для забезпечення адаптивності інтерфейсу використовувались Flexbox та CSS Grid-макети.

Робота онлайн-ІС управління замовленнями та виробничим графіком підприємства реалізована у вигляді веб-додатку, що забезпечує взаємодію користувачів із системою через UI/UX та API [33]. Основні функціональні можливості системи відповідають раніше визначеним бізнес-процесам: авторизація, управління замовленнями, планування, виконання задач та аналітика з ШІ.

На рисунку 3.7 представлено інтерфейс управління співробітниками. На цій сторінці наявні всі співробітники та задачі, які вони зараз виконують. На сторінці реалізовано функції сортування та пошуку конкретних даних.

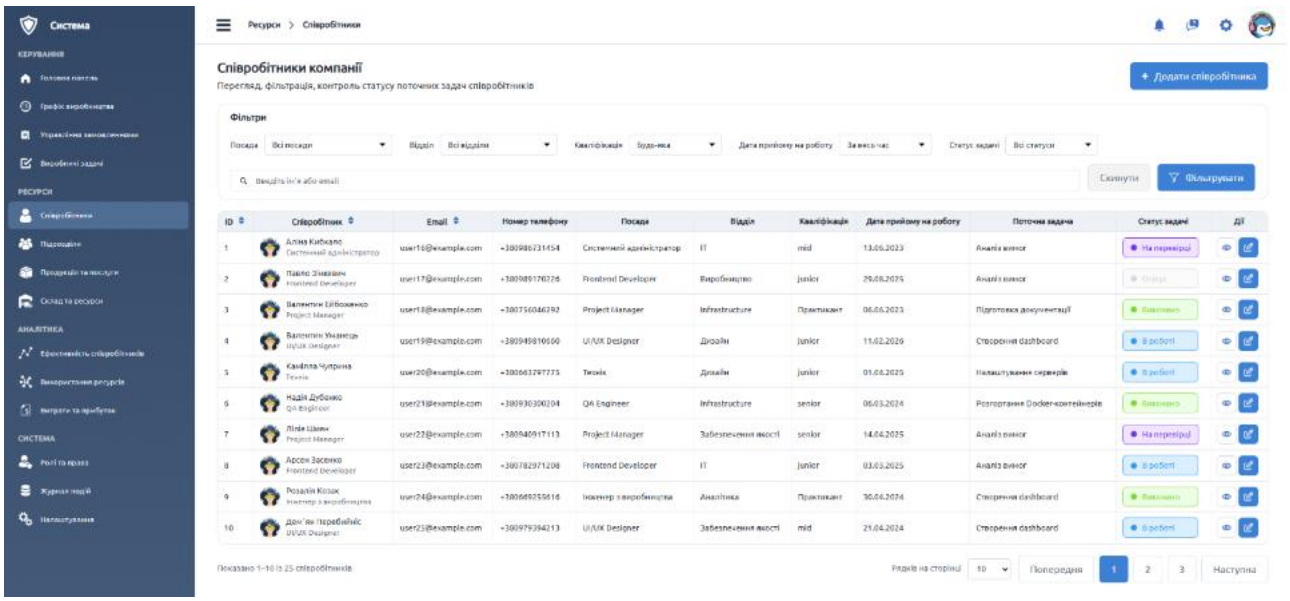


Рисунок 3.7 – Інтерфейс управління співробітниками

Наступним етапом стала розробка сторінки управління замовленнями. Було реалізовано таблицю замовлень із можливістю фільтрації, пошуку, перегляду статусів та редагування інформації. На сторінці відображаються дані про назву замовлення, клієнта, дату створення, планову дату завершення та поточний статус виконання. Для роботи з даними використовувались JSON-структури, що імітують взаємодію з БД [34].

На рисунку 3.8 представлено інтерфейс управління замовленнями. Користувач має можливість створювати, переглядати та редагувати замовлення, а також контролювати їх статус. Інтерфейс взаємодіє з БД через CRUD-операції та відображає актуальні дані з таблиці orders.

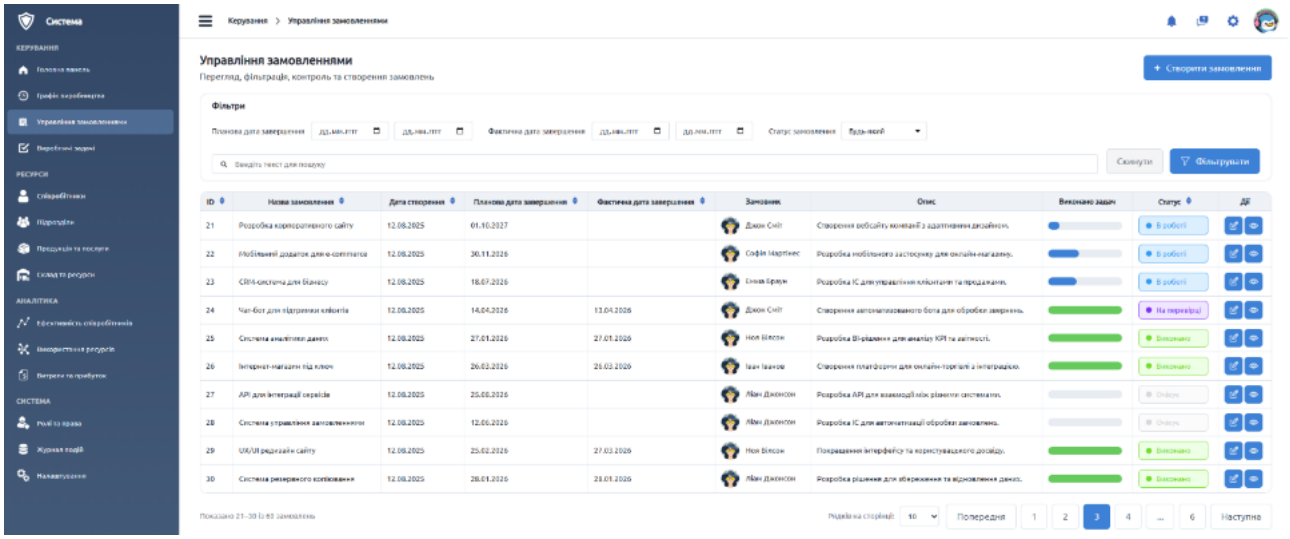


Рисунок 3.8 – Інтерфейс управління замовленнями

На рисунку 3.9 представлено інтерфейс управління виробничим графіком. У цьому модулі реалізовано функціонал планування та розподілу задач між співробітниками з урахуванням ресурсів підприємства. Інтерфейс відображає графік виробництва та забезпечує взаємодію з підсистемою АСУП.

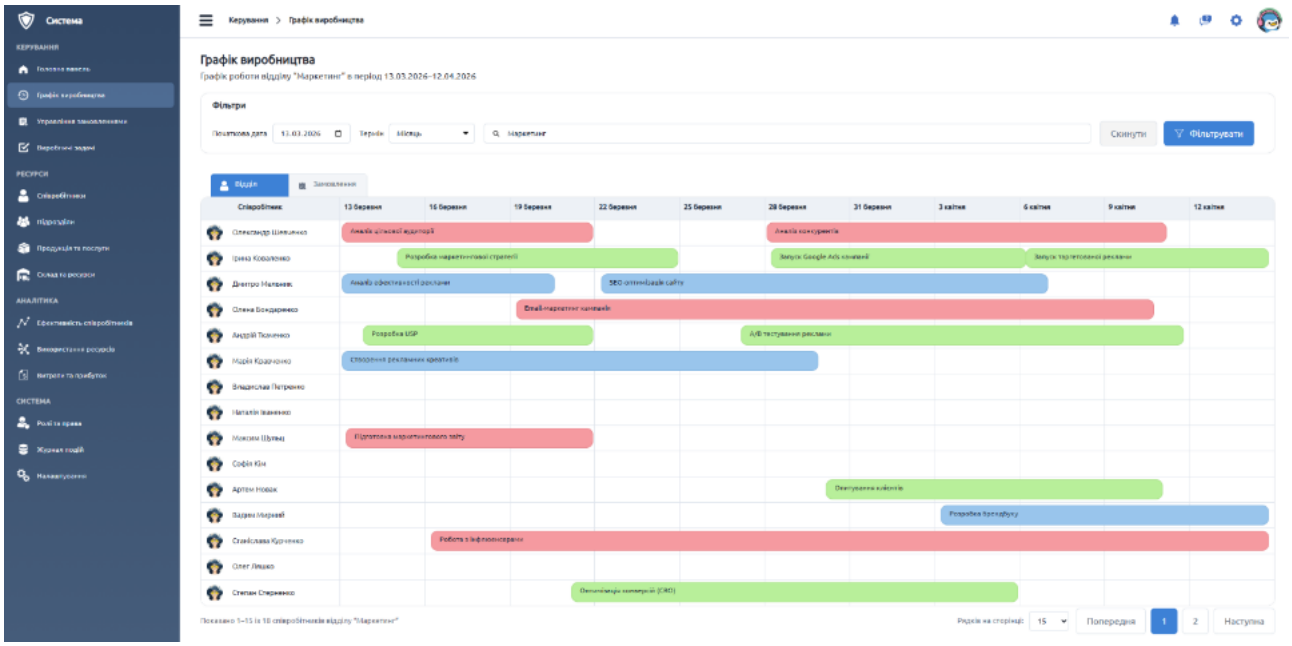


Рисунок 3.9 – Інтерфейс управління виробничим графіком

На рисунку 3.10 представлено інтерфейс перегляду виробничих завдань. Співробітники отримують доступ до призначених задач, можуть змінювати їх статус та передавати результати виконання до БД. Це забезпечує актуалізацію даних у таблицях production_tasks та task_results.

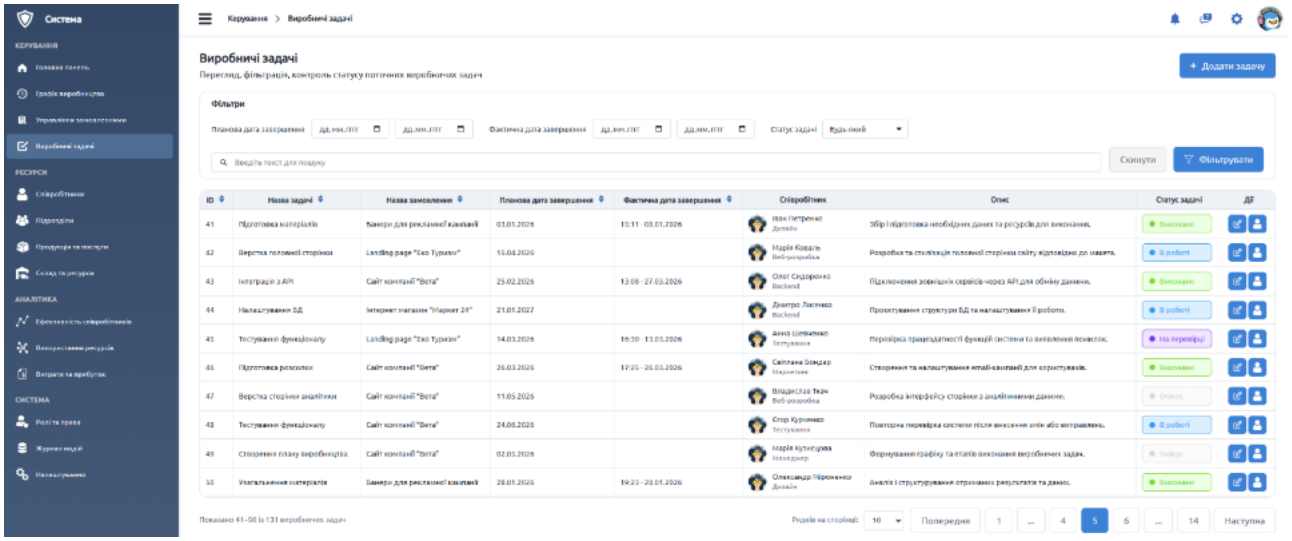


Рисунок 3.10 – Інтерфейс перегляду виробничих завдань

Окремим етапом стала реалізація аналітичних модулів системи. Було створено сторінки аналізу ефективності співробітників, використання ресурсів та фінансової аналітики. Модуль employee-performance.html забезпечує формування статистики продуктивності працівників, побудову графіків виконання задач та відображення AI-рекомендацій щодо оптимізації виробничих процесів.

На рисунку 3.11 представлено інтерфейс отримання аналітичної інформації та звіту від ШІ щодо ефективності співробітників. У цьому модулі відображаються аналітичні показники, сформовані на основі даних БД та оброблені ШІ. Користувач отримує оцінку продуктивності та рекомендації.

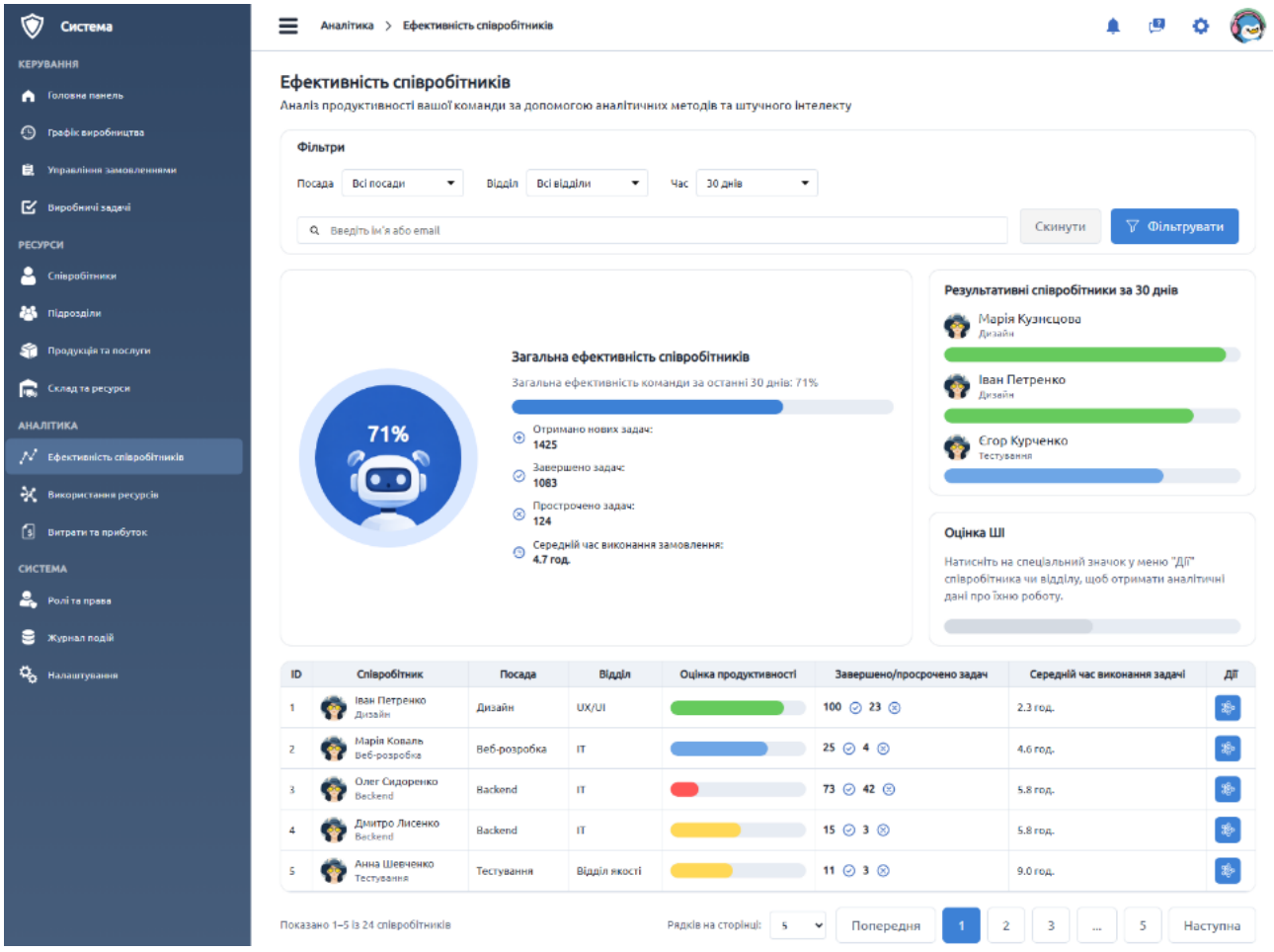


Рисунок 3.11 – Інтерфейс отримання інформації стосовно ефективності співробітників

Наступним етапом стало створення JavaScript-модулів для отримання даних із database.json та відображення їх у таблицях і компонентах веб-додатку. Було реалізовано механізми fetch-запитів [35], динамічного формування таблиць, оновлення статусів та роботи фільтрів [36].

У файлі layout-sidebar-toggle.js реалізовано функцію toggleSidebar(), яка відповідає за згортання та розгортання бічної панелі навігації сайту. Основний принцип роботи функції полягає у додаванні або видаленні CSS-класу app-shell--sidebar-collapsed для головного контейнера системи. Після виклику функції змінюється ширина sidebar-панелі та перебудовується структура інтерфейсу. Також у скрипті реалізовано автоматичне створення кнопки керування меню у випадку її відсутності в DOM-структурі сторінки.

У файлі order-management-data.js ключовою функцією є normalizeRelationalOrders(database). Дана функція виконує формування структури замовлень на основі взаємопов’язаних таблиць БД. Загальний алгоритм функції працює наступним чином:

– зчитування таблиць orders, clients, users, employees та production_tasks;

– формування Map-структур для швидкого доступу до записів [37];

– пошук клієнта для кожного замовлення;

– пошук відповідального менеджера;

– обчислення кількості виконаних задач;

– розрахунок відсотка завершення замовлення;

– автоматичне визначення пріоритету замовлення;

– автоматичний розрахунок бюджету.

Функція getOrderPriority(order) виконує автоматичний аналіз пріоритету замовлення залежно від дедлайну. Якщо дата завершення наближається або прострочена – замовлення отримує високий пріоритет. Якщо запас часу великий – пріоритет знижується. Таким чином система автоматично визначає критичність виконання замовлення.

Функція createProgressCell(order) створює progress-bar виконання замовлення. Значення прогресу визначається на основі кількості завершених виробничих задач. Колір progress-bar автоматично змінюється залежно від статусу замовлення.

Функція renderOrdersTable() виконує побудову HTML-таблиці замовлень. На початку функція визначає режим роботи таблиці – детальний або прогрес-режим. Після цього виконується:

– отримання активної сторінки;

– формування масиву видимих замовлень;

– генерація HTML-рядків таблиці;

– оновлення пагінації;

– оновлення лічильника записів.

У файлі production-tasks-data.js основною функцією є normalizeRelationalTasks(database). Дана функція виконує об’єднання даних із production_tasks, orders, employees та users в єдину структуру виробничих задач [38]. Для кожної задачі система автоматично визначає:

– назву замовлення;

– виконавця;

– роль співробітника;

– планову дату;

– фактичну дату завершення;

– поточний статус задачі.

Функція createEmployeeCell(task) створює інформаційний блок співробітника [39] у таблиці виробничих задач. У функції виконується генерація аватара на основі ініціалів працівника, а також відображення його ПІБ та посади.

Функція applyFilters() виконує фільтрацію виробничих задач за датами, статусами та пошуковим рядком. На початку функція зчитує параметри фільтрів із форми інтерфейсу, після чого перевіряє кожну задачу на відповідність умовам.

Після завершення фільтрації автоматично виконується сортування та оновлення таблиці задач.

У файлі employees-data.js ключовою функцією є normalizeRelationalEmployees(database). Дана функція виконує формування структури співробітників із використанням таблиць users, employees та production_tasks. Для кожного працівника система автоматично визначає:

– ПІБ;

– email;

– номер телефону;

– посаду;

– відділ;

– кваліфікацію;

– поточну задачу;

– статус виконання задачі.

Функція getCurrentTaskForEmployee(employeeId, tasks) виконує визначення активної задачі співробітника. Алгоритм функції працює наступним чином:

– пошук усіх задач конкретного співробітника [40];

– сортування задач за пріоритетом статусу;

– визначення найважливішої задачі;

– повернення назви задачі та статусу виконання.

Для сортування використовується TASK_STATUS_PRIORITY, де найвищий пріоритет мають задачі зі статусом «В роботі».

Функція applyFilters() у employees-data.js забезпечує комплексну фільтрацію співробітників. У процесі роботи функція перевіряє:

– відповідність посади;

– відповідність відділу;

– рівень кваліфікації;

– статус задачі;

– дату найму;

– пошуковий запит користувача.

Після фільтрації виконується повторне сортування та оновлення таблиці співробітників.

У файлі employee-performance-filters.js центральною функцією є calculateEmployeeMetrics(employee, startDateValue, periodValue). Дана функція виконує аналітичний розрахунок продуктивності співробітника. Загальний алгоритм роботи функції включає:

– вибір задач за вказаний часовий період;

– визначення завершених задач;

– визначення прострочених задач;

– розрахунок середнього часу виконання;

– розрахунок показника продуктивності;

– формування інтегрального рейтингу ефективності працівника.

Функція renderAiAdvice(employeeId) відповідає за генерацію AI-рекомендацій для співробітника. На початку функція знаходить останній аналітичний звіт працівника, після чого:

– визначає показник продуктивності;

– формує текст рекомендацій;

– оновлює progress-bar ефективності;

– відображає прогноз продуктивності співробітника.

Код звернення до ШІ модулю представлено нижче [41]:

```
app.post("/api/ai-assistant", async (employeeId, res) => {
  const employee = employeePerformanceState.allEmployees.find((item) => Number(item.id) === Number(employeeId));
  const prompt = `
  Проаналізуй ефективність роботи співробітника.
  Ім'я: ${employee.name}
  Посада: ${employee.position}
  Кількість виконаних задач: ${employee.completed_tasks}
```

```
        Протермінованих задач: ${employee.late_tasks}
        эффективність: ${employee.efficiency_level}%
        Середня эффективність роботи: ${employee.length ? Math.round(totalScore / employee.length) : 0}%
        Попередні звіти ефективності: ${GetLastEmployeesReports(employee)}
        Дай власну оцінку ефективності та рекомендації для покращення роботи.
        `;
        const response = await client.chat.completions.create({
          model: "gpt-4.1-mini",
          messages: [
            {
              role: "user",
              content: prompt
            }
          ]
        });
        res.json({
          answer: response.choices[0].message.content
        });
    });
```

У випадку відсутності достатньої кількості даних система автоматично виводить повідомлення про неможливість проведення аналізу.

Функція updateSummaryCard() виконує формування загальної статистики роботи підприємства. У процесі роботи функція:

– обчислює середню ефективність команди;

– визначає кількість нових задач;

– визначає кількість завершених задач;

– визначає кількість прострочених задач;

– розраховує середній час виконання робіт;

– оновлює інформаційні блоки dashboard-панелі.

На завершальному етапі проводилось тестування всіх сторінок системи, перевірка коректності відображення даних, адаптивності інтерфейсу та взаємодії між компонентами. Також виконувалась оптимізація структури CSS, перевірка навігації між сторінками та підготовка системи до подальшої backend-інтеграції [42].

3.4 Охорона праці

У процесі розробки та експлуатації онлайн-ІС управління замовленнями та виробничим графіком підприємства необхідно забезпечити належний рівень охорони праці користувачів, які працюють із ПЗ. Основна увага приділяється організації безпечних умов праці при роботі з комп'ютерною технікою, а також дотриманню санітарно-гігієнічних та ергономічних вимог.

Робота користувачів ІС здійснюється за персональними комп'ютерами, що відноситься до категорії робіт із підвищеним навантаженням на зорову систему, опорно-руховий апарат та нервову систему [43]. Основними небезпечними та шкідливими факторами є електромагнітне випромінювання, тривала статична поза, зорове напруження, а також психоемоційне навантаження при роботі з інформаційними потоками.

З метою мінімізації впливу зазначених факторів необхідно дотримуватись наступних вимог:

– організація робочого місця відповідно до ергономічних стандартів [44];

– забезпечення оптимального освітлення як природного, так і штучного;

– регламентування часу роботи за ПК;

– використання сучасного обладнання з низьким рівнем випромінювання.

Основні вимоги до організації робочого місця наведено в таблиці 3.9.

Таблиця 3.9 – Вимоги до організації робочого місця користувача

Параметр	Вимога
Відстань до монітора	50 – 70 см
Висота розташування монітора	на рівні очей
Освітлення	300 – 500 лк
Кут огляду	15 – 20 градусів нижче горизонталі
Тривалість безперервної роботи	не більше 2 годин

Важливим аспектом є забезпечення режиму праці та відпочинку. Для користувачів ІС рекомендовано встановлювати регламентовані перерви, що дозволяють зменшити втому та підвищити продуктивність роботи. При роботі більше 4 годин на день необхідно передбачити перерви тривалістю 5 – 10 хвилин після кожної години роботи.

З точки зору електробезпеки, обладнання повинно відповідати вимогам стандартів безпеки та мати належне заземлення [45]. Забороняється експлуатація пошкодженого обладнання або використання несправних кабелів живлення. Усі роботи з обслуговування техніки повинні виконуватись кваліфікованим персоналом.

Окрему увагу слід приділити пожежній безпеці. Робочі приміщення повинні бути обладнані засобами пожежогасіння [46], а також мати вільний доступ до евакуаційних виходів. Забороняється перевантаження електромережі та використання несертифікованих пристроїв.

З урахуванням специфіки ІС важливим є також аспект інформаційної безпеки, який опосередковано впливає на умови праці. Використання механізмів авторизації, захисту даних та контроль доступу знижує ризики несанкціонованого втручання та забезпечує стабільну роботу ПЗ.

ВИСНОВКИ

У межах виконаної роботи розроблено онлайн-ІС управління замовленнями та виробничим графіком підприємства, орієнтовану на автоматизацію ключових процесів планування, виконання та контролю виробничих завдань.

На початковому етапі було проведено аналіз предметної області та існуючих підходів до автоматизації, що дозволило визначити недоліки традиційних рішень і сформувати обґрунтовані вимоги до ІС. Особливу увагу приділено інтеграції управлінських (АСУП) і технологічних (АСУТП) процесів, що є критично важливим для сучасних підприємств.

Далі було виконано проєктування системи, у межах якого сформовано архітектуру ПЗ, обрано технологічний стек та розроблено моделі взаємодії компонентів. Використання UML, DFD та IDEF0 дозволило формалізувати логіку роботи ІС та забезпечити узгодженість між бізнес-процесами і програмною реалізацією.

Наступним етапом стала розробка ПЗ, яка включала створення алгоритмів функціонування системи, проєктування БД на основі ER-моделі та реалізацію основних модулів ІС. Система забезпечує повний цикл роботи із замовленнями: від їх створення до виконання та подальшого аналізу результатів. Окремим важливим компонентом є ШІ-модуль, що дозволяє оцінювати ефективність роботи співробітників і формувати рекомендації для підвищення продуктивності.

Результатом роботи є функціональна ІС, яка поєднує управління замовленнями, планування виробництва та аналітичну підтримку прийняття рішень [47]. Запропоноване рішення характеризується структурованістю, можливістю масштабування та адаптації до умов реального підприємства.

Також, отримані результати роботи можно віднести до Цілі сталого розвитку 9 “Промисловість, інновації та інфраструктура”, а саме п.9.4 “Сприяти прискореному розвитку високо- та середньовисокотехнологічних секторів

переробної промисловості, які формуються на основі використання ланцюгів «освіта – наука – виробництво» та кластерного підходу за напрямами: розвиток інноваційної екосистеми; розвиток інформаційнотелекомунікаційних технологій (ІКТ); застосування ІКТ в АПК, енергетиці, транспорті та промисловості; високотехнологічне машинобудування; створення нових матеріалів; розвиток фармацевтичної та біоінженерної галузей”.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. – Київ : ДП «УкрНДНЦ», 2016. – 31 с.

2. Невлюдов І. Ш., Филипенко О. І., Токарєва О. В., Новосьолов С. П., Сичова О. В. Навчальний посібник з підготовки кваліфікаційної роботи першого (бакалаврського) рівня вищої освіти за спеціальністю 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «АКІТ». – Харків : Видавництво Іванченка І. С., 2023. – 151 с.

3. Козак В. І. Автоматизовані системи управління виробництвом. – Київ : КНЕУ, 2019. – 312 с.

4. Литвин В. В. Інтелектуальні інформаційні системи. – Львів : Видавництво Львівської політехніки, 2020. – 406 с.

5. Олійник А. О. Інформаційні системи і технології на підприємстві. – Харків : ХНЕУ, 2018. – 292 с.

6. Бойко О. В. Системи автоматизації виробничих процесів. – Київ : КПІ ім. Ігоря Сікорського, 2021. – 350 с.

7. Pressman R. S. Software Engineering: A Practitioner's Approach. – 8th ed. – New York : McGraw-Hill Education, 2015. – 970 p.

8. Kendall K. E., Kendall J. E. Systems Analysis and Design. – 10th ed. – Boston : Pearson, 2019. – 768 p.

9. Fowler M. UML Distilled : A Brief Guide to the Standard Object Modeling Language. – 3rd ed. – Boston : Addison-Wesley, 2004. – 208 p.

10. Eriksson H., Penker M. Business Modeling with UML. – New York : Wiley, 2000. – 459 p.

11. Date C. J. An Introduction to Database Systems. – 8th ed. – Boston : Addison-Wesley, 2003. – 1024 p.

12. Elmasri R., Navathe S. B. Fundamentals of Database Systems. – 7th ed. – Boston : Pearson, 2016. – 1272 p.

13. Oracle Corporation. MySQL 8.0 Reference Manual [Електронний ресурс]. – Режим доступу: https://dev.mysql.com/doc/refman/8.0/en/ (дата звернення: 21.04.2026).

14. Welling L., Thomson L. PHP and MySQL Web Development. – 5th ed. – Boston : Addison-Wesley, 2017. – 768 p.

15. Mozilla Foundation. JavaScript Guide [Електронний ресурс]. – Режим доступу: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide (дата звернення: 21.04.2026).

16. Duckett J. HTML and CSS: Design and Build Websites. – Indianapolis : Wiley, 2011. – 490 p.

17. Flanagan D. JavaScript: The Definitive Guide. – 7th ed. – Sebastopol : O'Reilly Media, 2020. – 704 p.

18. Fielding R. T. Architectural Styles and the Design of Network-based Software Architectures. – Irvine : University of California, 2000. – 162 p.

19. Red Hat Inc. REST API Design Guidelines [Електронний ресурс].

20. Martin R. C. Clean Architecture: A Craftsman's Guide to Software Structure and Design. – Boston : Prentice Hall, 2017. – 432 p.

21. Gamma E., Helm R., Johnson R., Vlissides J. Design Patterns: Elements of Reusable Object-Oriented Software. – Boston : Addison-Wesley, 1994. – 395 p.

22. Silberschatz A., Galvin P. B., Gagne G. Operating System Concepts. – 10th ed. – Hoboken : Wiley, 2018. – 976 p.

23. Tanenbaum A. S., Bos H. Modern Operating Systems. – 4th ed. – Boston : Pearson, 2015. – 1136 p.

24. ISO/IEC 25010:2011. Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) – System and Software Quality Models. – Geneva : ISO, 2011. – 34 p.

25. Wiegers K., Beatty J. Software Requirements. – 3rd ed. – Redmond : Microsoft Press, 2013. – 672 p.

26. Red Hat Inc. REST API Design Guidelines [Електронний ресурс]. – Режим доступу: https://opensource.zalando.com/restful-api-guidelines/ (дата звернення: 21.04.2026).

27. Martin R. C. Clean Architecture: A Craftsman’s Guide to Software Structure and Design. – Boston : Prentice Hall, 2017. – 432 p.

28. Oracle Corporation. MySQL 8.0 Reference Manual [Електронний ресурс]. – Режим доступу: https://dev.mysql.com/doc/ (дата звернення: 21.04.2026).

29. Pressman R. S. Software Engineering: A Practitioner’s Approach. – 8th ed. – New York : McGraw-Hill Education, 2015. – 970 p.

30. Kendall K. E., Kendall J. E. Systems Analysis and Design. – 10th ed. – Boston : Pearson, 2019. – 768 p.

31. Silberschatz A., Galvin P. B., Gagne G. Operating System Concepts. – 10th ed. – Hoboken : Wiley, 2018. – 976 p.

32. Date C. J. An Introduction to Database Systems. – 8th ed. – Boston : Addison-Wesley, 2003. – 1024 p.

33. Object Management Group. OMG Unified Modeling Language (OMG UML), Version 2.5.1 [Електронний ресурс]. – Режим доступу: https://www.omg.org/spec/UML/2.5.1 (дата звернення: 21.04.2026).

34. Sommerville I. Software Engineering. – 10th ed. – Boston : Pearson, 2016. – 816 p.

35. Object Management Group. Business Process Model and Notation (BPMN), Version 2.0 [Електронний ресурс]. – Режим доступу: https://www.omg.org/spec/BPMN/2.0 (дата звернення: 21.04.2026).

36. Chen P. P.-S. The Entity-Relationship Model – Toward a Unified View of Data // ACM Transactions on Database Systems. – 1976. – Vol. 1, No. 1. – P. 9–36.

37. Laudon K. C., Laudon J. P. Management Information Systems: Managing the Digital Firm. – 16th ed. – Harlow : Pearson, 2020. – 669 p.

38. Shneiderman B., Plaisant C., Cohen M. et al. Designing the User Interface. – 6th ed. – Boston : Pearson, 2016. – 624 p.

39. Nielsen J. Usability Engineering. – San Francisco : Morgan Kaufmann, 1993. – 362 p.

40. ISO 9241-210:2019. Ergonomics of Human-System Interaction – Part 210: Human-Centred Design for Interactive Systems. – Geneva : ISO, 2019.

41. Gamma Technologies. Principles of Web Architecture [Електронний ресурс].

42. Google Developers. Web Performance Optimization Guide [Електронний ресурс].

43. OWASP Foundation. OWASP Top 10 Web Application Security Risks [Електронний ресурс]. – Режим доступу: https://owasp.org/www-project-top-ten/ (дата звернення: 21.04.2026).

44. ISO/IEC 27001:2022. Information Security, Cybersecurity and Privacy Protection – Information Security Management Systems – Requirements. – Geneva : ISO, 2022.

45. ДБН В.2.5-28:2018. Природне і штучне освітлення. – Київ : Мінрегіон України, 2018.

46. Комплекс навчально-методичного забезпечення навчальної дисципліни "Безпека життєдіяльності" підготовки освітнього рівня бакалавр усіх спеціальностей та усіх напрямів університету [Електронний ресурс] / ХНУРЕ; розроб.: Т. Є. Стиценко, Г. В. Пронюк. – Харків, 2024. – 463 с.

47. ISO 45001:2018. Occupational Health and Safety Management Systems – Requirements with Guidance for Use. – Geneva : ISO, 2018.