

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій  
(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки  
(повна назва)

## КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)

Розроблення системи автоматизації для управління програмою  
запуску та випуску продукції на виробничому підприємстві  
(тема)

Виконав:  
здобувач 2 року навчання,  
групи КІТПВМ-24-2  
Максим КОВАЛЕНКО  
(власне ім'я прізвище)

Спеціальність 174 Автоматизація,  
комп'ютерно-інтегровані технології та  
робототехніка  
(код і повна назва спеціальності)

Тип програми Освітньо-професійна  
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерно-  
інтегровані технологічні процеси і  
виробництва  
(повна назва освітньої програми)

Керівник доцент Рауф АЛЛАХВЕРАНОВ  
(посада, власне ім'я прізвище)

Допускається до захисту  
Зав. кафедри КІТАР

\_\_\_\_\_  
(підпис)

Ігор НЕВЛЮДОВ  
(власне ім'я прізвище)

2025 р.

Я, Коваленко Максим Григорович, 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.

«12» грудня 2025 р.

A handwritten signature in black ink, appearing to read 'MKob', is written over a horizontal line.

Максим КОВАЛЕНКО

# ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет	Автоматики і комп'ютеризованих технологій
Кафедра	Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки
Рівень вищої освіти	другий (магістерський)
Спеціальність	174 Автоматизація, комп'ютерно-інтегровані технології та робототехніка
Тип програми	освітньо-професійна
Освітня програма	Комп'ютерно-інтегровані технологічні процеси і виробництва

(код і повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри КІТАР \_\_\_\_\_  
(підпис)

«\_\_\_» \_\_\_\_\_ 2025р.

## ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві \_\_\_\_\_ *Коваленку Максиму Григоровичу*  
(прізвище, ім'я, по батькові)

1. Тема роботи *Розроблення системи автоматизації для управління програмою запуску та випуску продукції на виробничому підприємстві*  
затверджена наказом по університету від "10" листопада 2025р. № 1029 Ст.
2. Термін подання студентом роботи "24" грудня 2025р.
3. Вихідні дані до роботи 3.1 Випуск продукції на виробничому підприємстві;  
3.2 Система управління базами даних – PostgreSQL;  
3.3 Механізм взаємодії компонентів – REST API;  
3.4 Інтегроване середовище програмування – VS Code;  
3.5 Мова програмування – Python 3.11.
4. Перелік питань, що потрібно опрацювати в роботі  
4.2 Процес управління програмою запуску та випуску продукції на виробництві;  
4.3 Розроблення структури автоматизованої системи управління;  
4.4 Реалізація програмного комплексу автоматизованої системи управління; автоматизації контролю точності виготовлення деталей;  
4.5 Оцінка надійності та стійкості автоматизованої системи управління;  
4.6 Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій *Демонстраційний матеріал представлений у форматі презентації PowerPoint (\*.ppt) – 22 с. формату А4*

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	<i>Процес управління програмою запуску та випуску продукції на виробництві</i>	01.09 – 15.09.25	<i>виконано</i>
2	<i>Розроблення структури автоматизованої системи управління</i>	16.09 – 30.09.25	<i>виконано</i>
3	<i>Реалізація програмного комплексу автоматизованої системи управління</i>	01.11 – 21.11.25	<i>виконано</i>
4	<i>Оцінка надійності та стійкості автоматизованої системи управління</i>	22.11 – 07.12.25	<i>виконано</i>
5	<i>Оформлення пояснювальної записки</i>	08.12 – 13.12.25	<i>виконано</i>
6	<i>Подання роботи на перевірку Інтернет-системою StrikePlagiarism</i>	14.12 – 16.12.25	<i>виконано</i>
7	<i>Подання роботи на рецензію</i>	17.12 – 19.12.25	<i>виконано</i>
8	<i>Подання роботи на підпис зав. кафедри</i>	20.12 – 23.12.25	<i>виконано</i>
9	<i>Подання кваліфікаційної роботи в ЕК</i>	24.12.25	<i>виконано</i>

Дата видачі завдання 01.09.2025р.

Здобувач \_\_\_\_\_ Максим КОВАЛЕНКО  
(підпис)

Керівник роботи \_\_\_\_\_ доцент Рауф АЛЛАХВЕРАНОВ  
(підпис) (посада, власне ім'я прізвище)

## РЕФЕРАТ

Пояснювальна записка: 87 с., 11 табл., 15 рис., 3 дод., 36 джерел.

АВТОМАТИЗОВАНА СИСТЕМА, УПРАВЛІННЯ, ПРОДУКЦІЯ, ЗАПУСК, ВИПУСК, МЕТОД, МОДУЛЬ, БАЗА ДАНИХ, АЛГОРИТМ, ПРОГРАМНИЙ КОМПЛЕКС, ВИРОБНИЧЕ ПІДПРИЄМСТВО.

Мета роботи – підвищення ефективності управління виробництвом шляхом розроблення автоматизованої системи планування, моніторингу та аналізу виконання виробничих програм.

Об’єкт дослідження – процес управління програмою запуску та випуску продукції на виробничому підприємстві.

Предмет дослідження – методи, моделі та засоби автоматизації управління із використанням сучасних інформаційних технологій.

Кваліфікаційна робота містить повний цикл створення автоматизованої системи управління програмою запуску та випуску продукції – від аналізу проблеми до програмної реалізації та оцінки ефективності.

Створений програмний комплекс є інноваційним рішенням, що підвищує рівень цифровізації виробничих процесів, забезпечує надійне функціонування, стійкість до навантажень та значно покращує якість управління виробничою діяльністю підприємства.

Отримані результати роботи можна віднести до Цілі сталого розвитку 9 «Промисловість, інновації та інфраструктура», зокрема до пункту 9.4 «Розвиток високотехнологічного машинобудування».

## **ABSTRACT**

Explanatory note: 87 pp., 11 tab., 15 fig., 3 appendices, 36 sources.

**AUTOMATION SYSTEM, MANAGEMENT, PRODUCTION, LAUNCH, RELEASE, METHOD, MODULE, DATABASE, ALGORITHM, SOFTWARE COMPLEX, MANUFACTURING ENTERPRISE.**

The goal of the work is to improve production management efficiency by developing an automated system for planning, monitoring, and analyzing the implementation of production programs.

The object of the study is the process of managing the program for launching and releasing products at a manufacturing enterprise.

The subject of the study is methods, models, and means of automation management using modern information technologies.

The qualification work contains a complete cycle of creating an automated system for managing the program for launching and releasing products – from problem analysis to software implementation and effectiveness assessment.

The created software complex is an innovative solution that increases the level of digitalization of production processes, ensures reliable operation, resistance to loads, and significantly improves the quality of management of the enterprise's production activities.

The results of the work can be attributed to Sustainable Development Goal 9 “Industry, Innovation and Infrastructure,” in particular to paragraph 9.4 “Development of high-tech engineering”.

## ЗМІСТ

Перелік скорочень .....	9
Вступ .....	10
1 Процес управління програмою запуску та випуску продукції на виробництві .....	12
1.1 Сутність процесу управління програмою запуску та випуску продукції .....	12
1.2 Особливості традиційних систем управління виробництвом .....	17
1.3 Огляд сучасних автоматизованих систем управління виробництвом .....	21
1.4 Проблеми, що потребують вирішення .....	26
1.5 Висновки до розділу .....	30
2 Розроблення структури автоматизованої системи управління .....	31
2.1 Загальна концепція побудови системи .....	31
2.2 Модульна структура системи .....	35
2.3 Алгоритм роботи системи .....	40
2.4 Потоки даних між модулями в алгоритмі роботи системи .....	45
2.5 Висновки до розділу .....	46
3 Реалізація програмної частини автоматизованої системи управління .....	48
3.1 Архітектура програмного комплексу .....	48
3.2 Середовище розробки та мови для створення програмного комплексу ...	53
3.3 База даних і структура даних .....	56
3.4 Інтерфейс користувача програмного комплексу .....	64
3.5 Тестування та валідація системи .....	69
3.6 Висновки до розділу .....	72
4 Оцінка надійності та стійкості автоматизованої системи управління .....	73

4.1 Загальні принципи оцінки надійності та стійкості .....	73
4.2 Розрахунок надійності та стійкості системи .....	75
4.3 Заходи з безпечних умов праці у навчальній лабораторії .....	79
4.4 Висновки до розділу .....	81
Висновки .....	82
Перелік джерел посилання .....	84
Додаток А Структура програмного комплексу .....	86
Додаток Б Лістинг програми .....	87
Додаток В Публікація за темою досліджень .....	91
Додаток В Демонстраційний матеріал .....	97

## ПЕРЕЛІК СКОРОЧЕНЬ

БД – база даних;

КПО – коефіцієнт природної освітленості;

ПЗ – програмне забезпечення;

ПК – персональний комп'ютер;

СУБД – система управління базами даних;

ТЗ – технічне завдання.

## ВСТУП

Сучасність створює складні умови ринкової конкурентності на тлі розвику виробничих підприємств. Зокрема пріоритетними чинниками успіху стають гнучкість та оперативність, а також точність управління виробничими процесами. До того ж, виробнича програма, яка орієнтована на визначення обсягів, номенклатуру та строки випуску продукції, вимагає продуктивного планування, повсякчасного моніторингу та коригування з урахуванням реальних даних щодо стану ресурсів, завантаження обладнання та виконання замовлень.

Сталі методи ручного планування чи залучення окремих облікових програм здебільшого не можуть вирізнятися питомою швидкістю обробки інформації чи цілісності даних між підрозділами підприємства. Відтак усе це створює певні наслідки, а саме: затримки під час запуску виробництва, нераціональне послуговування ресурсами, а, отже, і зниження продуктивності.

За таких обставин автоматизація процесів управління програмою запуску та випуску продукції окреслюється як необхідна умова щодо підвищення ефективності діяльності підприємства, гарантування стабільності виробничих планів, а також своєчасного виконання замовлень.

Мета роботи – підвищення ефективності управління виробництвом шляхом розроблення автоматизованої системи планування, моніторингу та аналізу виконання виробничих програм.

Об'єкт дослідження – процес управління програмою запуску та випуску продукції на виробничому підприємстві.

Предмет дослідження – методи, моделі та засоби автоматизації управління із використанням сучасних інформаційних технологій.

Для досягнення окресленої мети необхідно розв'язати низку завдань:

- вивчити предметну область та нагальні інформаційні системи управління виробництвом (MES, ERP, APS);
- опрацювати структуру процесів формування та реалізації програми запуску продукції;
- розробити функціональну і модульну структуру автоматизованих систем управління;
- побудувати інформаційну модель бази даних для підтримки процесів планування та контролю;
- втілити програмний прототип системи і провести експериментальну перевірку на ефективність:
- виконати комплексне оцінювання щодо надійності, стійкості та безвідмовності розробленої системи.

Пояснювальна записка з кваліфікаційної роботи оформлена з урахуванням рекомендацій з підготовки й оформлення кваліфікаційної роботи здобувачами другого (магістерського) рівня вищої освіти [1, 2], а також за ДСТУ 3008:2015 [3]. Апробацію результатів досліджень за темою кваліфікаційної роботи проведено у [4].

# 1 ПРОЦЕС УПРАВЛІННЯ ПРОГРАМОЮ ЗАПУСКУ ТА ВИПУСКУ ПРОДУКЦІЇ НА ВИРОБНИЦТВІ

## 1.1 Сутність процесу управління програмою запуску та випуску продукції

Управління програмою запуску та випуску продукції визнано ваговою компонентою виробничої діяльності підприємства, адже спрямовується на узгоджене виконання замовлень, раціональне використання ресурсів, а також дотримання строків виготовлення продукції.

Окрім того, створення автоматизованої системи управління програмою запуску та випуску продукції обумовлено низкою чинників:

- перехід до цифрових моделей управління згідно з концепцією «Індустрія 4.0»;
- прагнення щодо зниження енергетичних і матеріальних витрат;
- зростання контролю виконання замовлень і точності планування;
- затребуваність в аналітичних системах, які є засобом для прогнозування ефективності виробничих рішень.

Власне процес зосереджено на етапах планування обсягів виробництва, контролі виконання плану, запровадження послідовності запуску замовлень, а також оперативному реагуванні на зміни у виробничому середовищі [6, 7].

Водночас механізм управління програмою запуску та випуску продукції на виробничому підприємстві цілком відповідає визначенню «центральна ланка системи виробничого менеджменту». Таке трактування пояснюється насамперед метою: забезпечення безперервності та ритмічності разом із своєчасним виготовленням продукції, що, зі свого боку, відповідає

затвердженим виробничому плану, обсягам замовлень, а також фактичним ресурсам.

Утім процес втілює комплекс взаємопов'язаних функцій, зокрема йдеться про підготовку та планування, координацію та контроль, моніторинг та аналіз результатів виробничої діяльності. Таким чином, утворюється замкнутий цикл управління. Як правило він реалізує співдію стратегічних, тактичних і оперативних цілей підприємства [6, 7].

### 1.1.1 Структура процесу управління програмою запуску

Насамперед управління програмою запуску відтворює циклічний процес відповідно до схеми «планування – виконання – контролю – аналізу – корекції», продемонстрованої на рисунку 1.1.



Рисунок 1.1 – Управління програмою запуску

Загалом процес управління програмою запуску продукції утворено шляхом взаємодії трьох рівнів:

- стратегічного – окреслює загальні цілі виробництва, номенклатуру продукції, прогнозує обсяги випуску, обмеження з урахуванням ресурсів і часу.

Зокрема на даному рівні розробляється виробнича програма підприємства як складова бізнес-плану;

- тактичного – конкретизує стратегічні цілі. На цьому рівні складаються щомісячні або квартальні плани запуску, встановлюється послідовність виконання замовлень, а також пріоритети та розподіл ресурсів між цехами;

- оперативного – відповідає за диспетчеризацію, моніторинг і контроль функціонування програми у реальному часі, реагує на відхилення (на кшталт збоїв, затримки постачання, дефектів), а також коригує плани.

### 1.1.2 Основні етапи процесу

Окреслимо головні етапи управління програмою запуску та випуску продукції:

- аналіз попиту, зокрема кошику замовлень. Проводиться оцінювання ринкових потреб і внутрішніх замовлень;
- формування виробничої програми. Встановлюються обсяги та терміни виготовлення кожного виду продукції;
- прорахунок потреб у ресурсах. Визначається кількість матеріалів, сировини, комплектування разом із трудовими й енергетичними ресурсами;
- розподіл завдань між цехами. Узгоджується календарний графік запуску вузлів і деталей;
- контроль функціонування програми. Моніторинг фактичного стану виробництва, відстежування появи відхилень;
- коригування планів і звітність. Оперативне перероблення графіків виробництва, а також аналіз ефективності виконання.

### 1.1.3 Основні інформаційні потоки

Здебільшого для реалізації окреслених етапів доцільно сформувати систему інформаційних потоків, за допомогою яких усі підрозділи підприємства можуть поєднуватися:

- потоки вхідної інформації охоплюють відомості щодо замовлення клієнтів, залишків на складах, даних про ресурси, технологічні карти;
- потоки вихідної інформації спрямовані на виробничі плани, графіки запуску, звіти про виконання;
- зворотні потоки інформації містять повідомлення про відхилення чи зміни, а також про аналітичні показники.

Однак управління цими потоками можна вважати ефективним лише у разі функціонування автоматизованої інформаційної системи, котра насамперед забезпечує актуальність, своєчасність і достовірність обміну даними.

### 1.1.4 Основні функції управління програмою запуску

Власне процес управління повинен характеризуватися низкою функцій:

- плануванням, зокрема встановлюються цілі, строки разом із ресурсним забезпеченням виробничої програми;
- організацією, тобто виконується розподілення завдань між підрозділами, а також оформлюються календарно-планові завдання;
- мотивацією, зокрема спонукати до виконання планів виробництва чи зниження витрат;
- контролем, тобто проводиться оцінювання фактичного виконання програми чи виявлення відхилень;
- регулюванням, отже, застосовується оперативне коригування програми за умови відхилень від плану;

– аналізу, тобто узагальнюються результати виконання, підготовка звітів для управлінських рішень.

#### 1.1.5 Чинники, що впливають на ефективність управління

Так, ефективність процесу управління програмою запуску може залежати від впливу таких чинників:

- структури виробництва, зокрема одиничного, серійного, масового;
- складності технологічних процесів;
- рівня інтеграції між підрозділами, на кшталт виробництво, склад, постачання чи збут;
- ступенем автоматизації виробничих і облікових систем;
- якістю інформаційного забезпечення, а саме: повноти, точності, актуальності даних;
- організаційної культури управління, зокрема затверджених регламентів, системи мотивації, прозорість звітності.

#### 1.1.6. Показники ефективності управління

З метою оцінювання ефективності програми запуску, зазвичай, орієнтуються на такі показники:

- коефіцієнт реалізації виробничого плану;
- середню тривалість виробничого циклу;
- рівень використання обладнання;
- рівень браку і простоїв;
- коефіцієнт завантаження персоналу;
- економічний ефект за результатами дії автоматизації (зменшення витрат, зростання продуктивності).

Підкреслимо, що сутність процесу управління програмою запуску та випуску продукції спрямована на оптимізацію взаємодії між усіма ланками виробничого процесу задля реалізації стабільного, економічно ефективного й своєчасного виконання замовлень. Водночас ефективне відтворення окресленого процесу неможливе без залучення сучасних інформаційно-аналітичних систем, які, зі свого боку, автоматизують планування, облік і моніторинг, а також аналіз виробничих процесів.

## 1.2 Особливості традиційних систем управління виробництвом

Традиційні системи управління виробництвом насамперед утворені шляхом залучення документного, табличного чи локально-обчислювального підходів до планування та контролю [8].

Значна кількість підприємств запроваджує процес управління як поєднання ручних методів (таблиць Excel, текстових документів, усних погоджень) із окремими автоматизованими підсистемами, які насправді не співдіють між собою.

Для зазначених систем характерним є власне лінійно-ієрархічний принцип управління, відповідно до якого кожний рівень (цех, відділ, ділянка) переважно працює автономно, надсилаючи інформацію «зверху вниз» або «знизу вгору» через накази, розпорядження, звіти.

Сталій системі управління також властиві нагальні підсистеми, зокрема:

- планово-економічна служба, котра розробляє виробничу програму, прораховує потреби в ресурсах, а також встановлює обсяги випуску продукції;
- бухгалтерія зазвичай веде фінансовий і ресурсний облік, а також складає звітність;
- відділ кадрів, зі свого боку, забезпечує трудові ресурси;

- виробничо-диспетчерський відділ, як правило, координує втілення планів, розподіляє завдання між цехами, веде облік простоїв і виконання;
- відділ матеріально-технічного забезпечення відповідає за постачання сировини, комплектування й енергоносіїв.

Окрім того, обмін інформацією між підрозділами реалізовано у паперовому або табличному вигляді, проте час від часу виникають затримки чи дублювання разом із можливими похибками.

Зауважимо, що незважаючи на простоту реалізації, загально прийняті системи вирізняються, зокрема, й суттєвими недоліками:

- відсутність інтегрованого інформаційного простору. Зазвичай відомості про запаси та ресурси, замовлення чи завантаження обладнання надходять до різних підрозділів, отже, не синхронізуються. За таких умов прийняття оперативних рішень або аналіз набувають ускладненого характеру;

- затримки під час передавання інформації. Загалом від моменту отримання первинних даних до процесу їхнього оброблення і використання втрачається багато часу. У зв'язку з цим значно знижується оперативність управління;

- обмежена аналітика. Так, не задіючи автоматичний збір статистичних показників і прогнозування, неможливо проводити об'єктивне оцінювання ефективності виробничих процесів;

- проблеми масштабування. Чим швидше підприємство розширюється, тим значно примножується обсяг інформації, проте ручним або частково автоматизованим системам не вистачає ресурсів для оброблення таких масивів даних;

- відсутність механізмів оптимізації. Розподілення завдань і ресурсів набуває інтуїтивного характеру, адже критерії ефективності, на кшталт мінімізації витрат або зменшення кількості пристроїв тощо, не враховуються;

– висока залежність від людського чинника. Більшість операцій супроводжується ручною працею. За таких обставин збільшується як ризик помилок і спотворення інформації, так і суб'єктивність у прийнятті рішень.

Зі свого боку, наявність тих чи інших недоліків у традиційних системах може спричинити такі наслідки:

- продуктивності праці знижується;
- час виробничого циклу збільшується;
- собівартість продукції підвищується;
- своєчасне виконання замовлень унеможлиблюється;
- прозорість управлінських процесів перебуває на низькому рівні.

З огляду на сказане вище, можемо підкреслити, що підприємство не зможе адаптуватися оперативнo до змін попиту чи порушень у ланцюгах постачання. Через це його конкурентоспроможність погіршується.

Утім сталим системам, попри численні обмеження, властиві й певні переваги, адже вони зумовлюють їхню поширеність, зокрема на малих і середніх підприємствах. Йдеться про:

- невисоку вартість реалізації;
- простоту експлуатації;
- мінімальні вимоги до комп'ютерного устаткування;
- можливість гнучкого та ручного коригування даних.

Однак наведені переваги стають дієвими лише під час стабільного виробництва, що супроводжується невеликим обсягом інформаційних потоків.

Таким чином, розвиток цифрових технологій, а також значне ускладнення виробничих процесів зумовлюють необхідність трансформації від традиційних систем до автоматизованих інтегрованих комплексів управління [9].



Рисунок 1.2 – Схема-порівняння «Традиційна система та автоматизована система»

Так, сучасні рішення – ERP, MES, APS – стають підґрунтям для формування суцільного інформаційного середовища, автоматичного оброблення даних, взаємодії між відділами, завдяки чому можливо позбавитись більшості недоліків загальноприйнятих систем [9, 10].

Водночас сталі системи управління виробництвом вирізняються фрагментарністю, низьким рівнем автоматизації, а також незначною взаємодією між підрозділами підприємства. Зважаючи на те, що вони можуть забезпечити базовий рівень контролю, проте їхня ефективність суттєво програє сучасним автоматизованим рішенням.

Загалом розвиток систем управління доцільно реалізувати на принципах централізованого збору й оброблення чи аналізу даних. Таким чином, утворюватиметься підґрунтя для переходу до інтегрованої системи автоматизації. Більш детально усе це розглянемо у наступних розділах.

### 1.3 Огляд сучасних автоматизованих систем управління виробництвом

Виробничі підприємства наразі поступово замінюють фрагментарні системи управління на інтегровані цифрові платформи, котрі, зі свого боку, об'єднують усі рівні виробничої діяльності, зокрема, від стратегічного планування до оперативного контролю технологічних процесів [9].

Зазначений перехід набуває актуальності з огляду на розвиток концепцій «Індустрія 4.0» та «Кіберфізичних виробничих систем» (CPS), адже основа управління зосереджується на інформаційно-аналітичній взаємодії фізичних процесів із цифровими моделями [10].

Окреслимо головні принципи сучасних автоматизованих систем:

- взаємодія даних між усіма підсистемами підприємства;
- гнучкість та адаптивність до змін виробничих умов;
- модульність із можливістю поетапного впровадження;
- аналітична підтримка процесів щодо прийняття рішень;
- послуговування єдиною базою даних для всіх управлінських рішень.

Загалом системи автоматизації управління виробництвом умовно можна розподілити на три основні класи:

- MES-системи (Manufacturing Execution Systems), тобто системи диспетчеризації й оперативного управління виробництвом;
- ERP-системи (Enterprise Resource Planning), власне системи управління ресурсами підприємства;
- APS-системи (Advanced Planning and Scheduling), якраз системи розширеного планування та календарного графіку виробничих операцій.

Для кожного із цих класів властиві специфічні функції. Однак за сучасних умов вони здебільшого суміщаються в єдину корпоративну платформу управління виробництвом (рисунок 1.3).

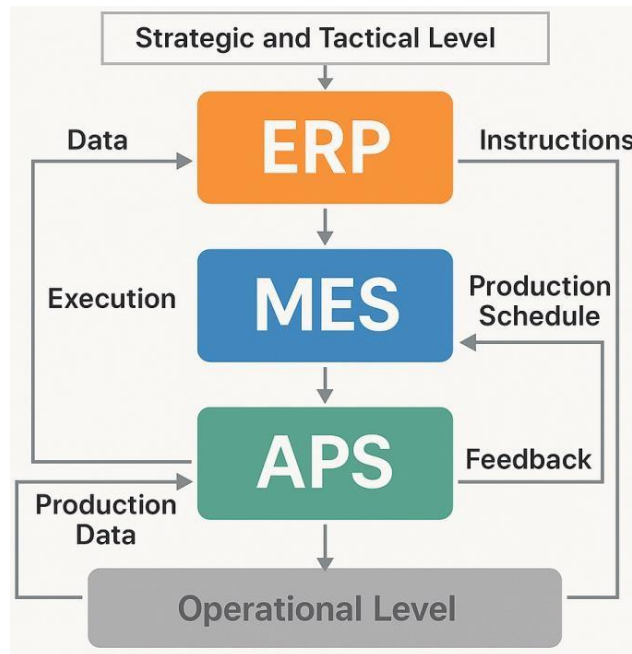


Рисунок 1.3 – Взаємозв’язок ERP-MES-APS систем у структурі управління виробництвом

### 1.3.1 Управління ресурсами підприємства

ERP-системи (управління ресурсами підприємства) визначено як основу корпоративного управління. Насамперед вони зорієнтовані на інтеграцію пріоритетних бізнес-процесів, а саме: виробництво, фінанси, закупівлі, складування, збут, управління персоналом.

Розглянемо ключові функції ERP-систем:

- ведення єдиної бази даних усіх ресурсів підприємства;
- облік матеріальних, трудових і фінансових ресурсів;
- автоматизація планово-економічних обчислень;
- розроблення виробничої програми, а також контроль її виконання;
- аналіз ефективності діяльності.

Наведемо приклади сучасних ERP-платформ:

- BAS ERP (Україна) – сучасна українська платформа, отже, розроблена відповідно до вітчизняного законодавства;

- Microsoft Dynamics 365 (США) – модульна ERP/CRM система, яка підтримує хмарні технології;
- Oracle E-Business Suite (США) – комплексна система для великих корпорацій, до того ж, орієнтована на багаторівневе управління;
- SAP ERP (SAP SE, Німеччина) – найпродуктивніше рішення у світі, інтегрує управління виробництвом і фінансами, а також логістикою та персоналом.

ERP-системи створюють інформаційне підґрунтя, проте їхнього рівня деталізації замало для ефективного управління оперативними виробничими процесами. Зважаючи на це, вони переважно доповнюються MES-рівнем.

### 1.3.2 Оперативне управління виробництвом

MES-системи (оперативне управління виробництвом) є проміжною ланкою між плануванням (ERP) і технологічним рівнем управління (PLC/SCADA). Ключову мету MES-систем можна окреслити як забезпечення оперативного контролю, координації й аналізу реалізації виробничих процесів у реальному часі.

Перелічимо головні функції MES-систем:

- відстеження партій – контроль щодо прямування матеріалів і продукції;
- диспетчеризація виробництва (розподілення завдань серед робочих місць, контроль за виконанням);
- збір і візуалізація показників ефективності (KPI, OEE);
- спостереження за обладнанням (визначення стану, завантаження, простоїв);
- облік виробничих операцій (час виконання, кількість продукції, відходи);
- план-факт аналіз і автоматичне оновлення графіків виробництва.

Розглянемо приклади MES-рішень:

- DELMIA Apriso (Dassault Systèmes, Франція) – підтримує інтеграцію з CAD/CAM системами;
- Ignition MES (Inductive Automation, США) – рішення з відкритою архітектурою для середніх підприємств;
- Siemens Opcenter (Німеччина) – інтегрована MES-платформа для складних виробничих систем;
- Wonderware MES (AVEVA, Велика Британія) – орієнтована на багаторівневі виробничі лінії.

Отже, MES-система втілює зворотний зв'язок між виробництвом і плануванням, утворюючи, зі свого боку, базу для аналітики та прогнозування.

### 1.3.3 Розширене планування

APS-системи (розширене планування) є системами для оптимізації графіків виробництва, що враховують обмеження ресурсів і потужностей, а також строки постачання та замовлення.

До основних можливостей APS-систем належать:

- вирахування оптимальної черговості щодо виконання замовлень;
- адаптація планів у разі зміни умов у реальному часі;
- синхронізація графіків запуску по цехах;
- візуалізація завантаження обладнання;
- інтеграція з ERP і MES.

APS-системами послуговуються там, де необхідно швидко оновити плани у разі виникнення нестандартних ситуацій, наприклад: під час перебоїв постачань або виходу з ладу обладнання.

### 1.3.4 Інтегровані системи нового покоління

Відповідно до концепції «Індустрія 4.0» сучасні автоматизовані системи наближаються до інтелектуальних кіберфізичних систем (CPS). Зокрема вони задіють:

- автоматизоване збирання даних (через IoT-сенсори);
- аналіз «великих даних» (Big Data);
- машинне навчання для прогнозування;
- хмарні платформи для спільного доступу до інформації.

Таким чином, системи спроможні як фіксувати факти, так і прогнозувати наслідки відхилень, водночас пропонують й оптимальні управлінські рішення. Як приклади таких систем наведемо Siemens MindSphere, ABB Ability, Schneider EcoStruxure, Honeywell Forge.

### 1.3.5 Порівняльна характеристика класів систем

Аналіз автоматизованих систем сучасності підтверджує, що ефективне управління виробництвом можна реалізувати лише шляхом інтеграції ERP, MES, APS із аналітичними модулями, тобто утворити єдине інформаційне середовище [9, 10].

У таблиці 1.1 відтворено порівняльну характеристику класів систем

Окреслені системи зорієнтовані на скороченні часу планування та перепланування, зменшенні втрати від простоїв, підвищенні точності виконання замовлень, а також забезпеченні прозорості й аналітичності всіх етапів виробництва.

Зауважимо, що у подальшій роботі актуальності набуває створення модульної системи автоматизації управління програмою запуску та випуску продукції, яка поєднує переваги ERP- і MES-рішень, до того ж адаптується до умов конкретного підприємства.

Таблиця 1.1 – Порівняльна характеристика класів систем

Клас системи	Основна функція	Рівень управління	Основні користувачі
ERP	управління ресурсами, фінансами, закупівлями	стратегічний і тактичний	дирекція, планово-економічні служби
MES	виконання і контроль виробничих процесів	оперативний	диспетчери, майстри, технологи
APS	оптимізація календарних графіків	тактичний і оперативний	плановики, логісти
CPS / IoT-платформи	інтелектуальний аналіз і прогнозування	інтегрований	менеджери, аналітики, керівництво

#### 1.4 Проблеми, що потребують вирішення

З огляду на цифрову трансформацію промисловості для більшості українських виробничих підприємств набувають актуальності проблеми щодо низького рівня інтегрованості управлінських процесів, а також відсутності повного контролю за станом виробництва, а, отже, і неефективного використання ресурсів. Незважаючи на впровадження окремих елементів автоматизації, зокрема систем бухгалтерського обліку чи локальних баз даних (БД), програм для диспетчеризації, брак єдиного інформаційного простору, зі свого боку, призводить до втрати оперативності управління.

Водночас зростання конкуренції чи потреба у скороченні циклу виготовлення продукції, а також гнучке реагування на зміни замовлень – усе це потребує запровадження цілісної автоматизованої системи, котрій властива здатність координувати процеси планування, контролювати та аналізувати виробництво.

Класифікацію проблем управління запуском виробництва наочно продемонстровано на рисунку 1.4.



Рисунок 1.4 – Класифікація проблем управління запуском виробництва

#### 1.4.2 Технологічні проблеми

Більшість підприємств автоматизують тільки окремі підсистеми, наприклад: бухгалтерію, склад, плануванням закупівель. Оскільки дані між ними зазвичай не синхронізуються, то відтворення актуальної картини виробництва унеможлиблюється.

До того ж, моніторинг виконання плану у сталих системах здійснюється із запізненням (щозміни, щодоби або навіть щотижня). У зв'язку з цим можливість виявляти простои, дефекти або затримки виключається.

Окрім того, низка підприємств й досі послуговується застарілими технологічними лініями, для яких інтерфейси для передавання даних не є характерними. Таким чином, збір фактичної інформації про виконання операцій значно ускладнюється.

Тож відсутність автоматизованої аналітики позбавляє можливості щодо проведення поглибленого аналізу причин відхилень, а також прогнозувати ризики чи оптимізувати графіки запуску.

Підкреслимо, що зміни у замовленнях або непередбачені збої, насамперед потребують повного перепланування програми вручну, що, зі свого боку, забирає час, а також підвищує ризик помилок.

#### 1.4.3 Організаційні проблеми

Як правило, планово-економічні, диспетчерські, технологічні та складські служби функціонують автономно, проте за таких умов виникає дублювання функцій, неузгодженість даних або конфлікти між пріоритетами.

Окрім того, у багатьох підприємствах відсутній затверджений регламент дій у разі зміни виробничої ситуації, тож це ускладнює оперативне прийняття рішень.

Власне інформація про стан виконання планів надходить до керівництва із запізненням, отже, управлінські рішення здебільшого ґрунтуються на застарілих даних.

Відсутність належної фаховості працівників середньої ланки уповільнює впровадження сучасних систем, а також знижує їхню ефективність.

#### 1.4.4 Інформаційні проблеми

Виробничі та технологічні, фінансові та кадрові дані зберігаються в різних форматах. Це значно ускладнює їхнє орпацювання й унеможливорює автоматизований аналіз.

Через дублювання записів до різних систем обліку знижується і достовірність інформації, що негативно впливає на якість управлінських рішень.

Звітність подається у вигляді таблиць, отже, не містить графічних індикаторів (діаграм, графіків), що, зі свого боку, ускладнює швидке оцінювання стану виробництва.

Актуальні системи не устатковані механізмами для передбачення наслідків відхилень як-от: зупинка обладнання або затримка постачання. У зв'язку з цим рішення приймаються реактивно, а не проактивно.

#### 1.4.5 Економічні проблеми

Через прості та надлишкові запаси, а також неузгодженість дій між цехами збільшуються втрати матеріальних і трудових ресурсів.

Водночас брак оперативного управління примножують час проходження замовлення через усі етапи виробництва.

У зв'язку із нераціональним плануванням використання ресурсів, а також через неефективну логістику собівартість продукції перевищує оптимальні показники.

До того ж, рішення щодо оптимізації виробництва приймаються постфактум, спираючись на місячні звіти, тож це не дозволяє своєчасно реагувати на зміни.

Утім проблеми управління програмою запуску мають системний характер, тому зводяться до суперечності між складністю виробничих процесів і обмеженими можливостями сталих засобів управління. З одного боку, нагальною стає необхідність у точному плануванні, з іншого – підприємства не мають у своєму розпорядженні ефективних засобів для автоматизованого опрацювання великих масивів даних.

Отже, розв'язання окресленої суперечності можливе лише шляхом запровадження автоматизованої системи, котрій властиві всі етапи – від планування до аналітики.

## 1.5 Висновки до розділу

За результатами проведеного аналізу було встановлено, що ефективне управління програмою запуску та випуску продукції потребує розроблення інтегрованої автоматизованої системи, яка б, зі свого боку, реалізувала централізоване планування виробничих процесів, а також моніторинг стану виконання завдань у реальному часі чи аналітичну підтримку прийняття рішень разом із автоматичним обліком витрачених ресурсів.

Тож розроблення власної модульної автоматизованої системи для управління програмою запуску та випуску продукції є актуальним і водночас стратегічно важливим завданням для вітчизняних виробничих підприємств.

Під час аналізу було виявлено, що нагальними проблемами щодо управління програмою запуску визначено фрагментарність інформаційних систем, запізнення у моніторингу, а також відсутність інтеграції між підрозділами чи низька аналітична підтримка прийняття рішень.

Ці проблеми власне підтверджують необхідність у створенні єдиної автоматизованої системи, яка насамперед забезпечить централізоване планування та оперативний контроль виконання програм, інтеграцію з обліковими та ресурсними системами, до того ж, аналітичну підтримку керівництва.

## 2 РОЗРОБЛЕННЯ СТРУКТУРИ АВТОМАТИЗОВАНОЇ СИСТЕМИ УПРАВЛІННЯ

### 2.1 Загальна концепція побудови системи

Мета розроблення автоматизованої системи управління програмою запуску та випуску продукції полягає у створенні інтегрованого інструменту, що реалізує узгоджене функціонування всіх етапів виробничого процесу, зокрема, від формування замовлення до аналізу результатів виробництва.

Так, система повинна забезпечувати:

- оперативність прийняття рішень на основі достовірних даних;
- скорочення циклу планування та випуску продукції;
- зменшення людського чинника під час оброблення інформації;
- автоматичний моніторинг і контроль функціонування виробничих програм;
- оформлення аналітичних звітів у режимі реального часу.

Розроблення системи виконується відповідно до таких принципів:

- модульність – систему побудовано зі взаємопов'язаних, проте незалежних функціональних блоків (модулів), які дозволяють значно розширювати функціонал.
- інтегрованість – усі модулі співдіють з єдиною базою даних, тому дублювання або втрата інформації виключаються.
- відкритість архітектури – спрямована на інтеграцію з чинними ERP, MES або бухгалтерськими системами підприємства.
- масштабованість – система може функціонувати як на невеликих дільницях, так і великому підприємстві загалом.

– доступність інформації – користувачам різних рівнів (інженерам, диспетчерам, керівникам) надходять дані у відповідному для них форматі.

– надійність і захищеність – активовано контроль доступу, резервне копіювання, шифрування критичних даних.

Зі свого боку, система вирізняється трирівневою архітектурою управління, що відповідає сучасним стандартам промислової автоматизації (ISA-95) [11]:

– операційний рівень (Shop Floor) – збір первинних даних на виробництві (стан обладнання, виконання операцій, облік часу);

– тактичний рівень (MES-рівень) – планування та диспетчеризація, моніторинг та аналіз виробничих процесів;

– стратегічний рівень (ERP-рівень) – планування довгострокове, управління фінансами та ресурсами, а також закупівлями підприємства.

Отже, система функціонує як проміжна інтеграційна ланка між ERP і технологічним рівнем, проводячи обмін інформацією в обох напрямках. Наочне відтворення на рисунку 2.1.

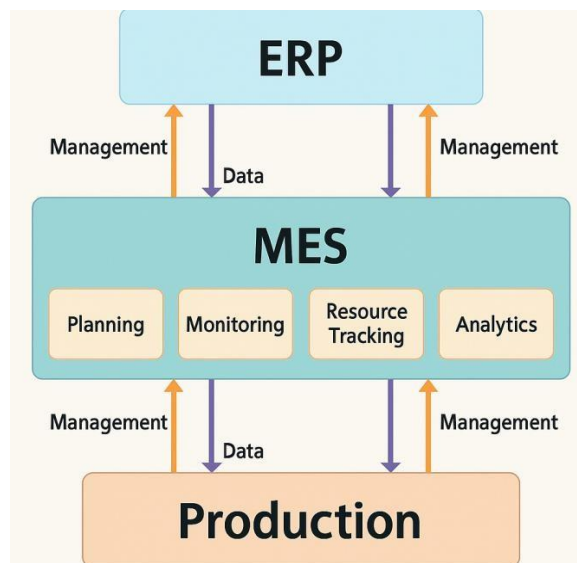


Рисунок 2.1 – Концептуальна архітектура автоматизованої системи

Функціонування системи розроблено за принципом замкненого інформаційного циклу, якому властиві такі етапи:

- отримання вхідних даних: з ERP, бази замовлень, складу ресурсів;
- складання виробничої програми: автоматичне обчислення графіків запуску;
- реалізація програми: скеровування завдань до виробничих підрозділів;
- коригування планів: оновлення програми випуску у разі відхилень;
- моніторинг: збір зворотної інформації щодо виконання у цехах;
- аналіз результатів: порівняння планових показників із фактичними.

Дана логіка подається як функціональний цикл управління «Планування → Виконання → Контроль → Аналіз → Корекція», що відтворює неперервність управління. Зокрема приклад схеми наведено на рисунку 2.2.

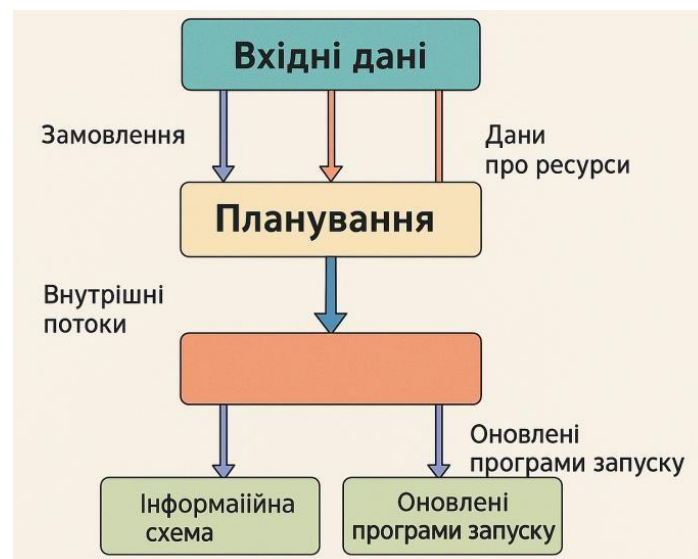


Рисунок 2.2 – Інформаційна схема реалізації автоматизованої системи управління запуском виробництва

Зауважимо, що для системи характерним є наявність трьох ключових груп інформаційних потоків [12, 13]:

- вхідні потоки: замовлення, дані про ресурси, планові показники;

- внутрішні потоки: взаємодія між модулями планування й обліку, а також моніторингу;
- вихідні потоки: аналітичні звіти, оновлені програми запуску, показники ефективності.

Отже, запровадження такої системи гарантуватиме:

- скорочення часу на розроблення виробничої програми до 30 – 50 %;
- забезпечення контролю функціонування програм у реальному часі;
- підвищення точність планування завдяки автоматичному аналізу даних;
- покращення комунікації між адміністрацією та цехами;
- зменшення витрати для ресурсного забезпечення.

Водночас система може стати підґрунтям для подальшого цифрового вдосконалення підприємства, зокрема шляхом упровадження елементів предиктивної аналітики чи машинного навчання.

Розроблена концепція побудови автоматизованої системи управління програмою запуску та випуску продукції, зі свого боку, спрямована на інтеграцію всіх управлінських процесів підприємства в єдиний інформаційний простір.

Утім система підпорядковується низці принципів, зокрема модульності, відкритості та масштабованості. Таким чином, вона стає придатною для ефективної адаптації відповідно до конкретних умов виробництва.

Тож отриману концепцію можна розглядати як методологічну основу задля подальшого проектування модульної структури й архітектури програмного забезпечення (ПЗ), що будуть описані у наступних підрозділах.

## 2.2 Модульна структура системи

Модульна структура, зазвичай, є базою для проектування автоматизованої системи управління програмою запуску та випуску продукції.

Побудова системи відповідно до модульного принципу реалізує гнучкість, розширюваність разом із можливістю адаптації до особливостей виробничих процесів підприємства.

Кожен модуль вирізняється чітко визначеними функціями, власними вхідними та вихідними даними, натомість обмін інформацією між ними втілюється через центральну БД. Окреслений принцип забезпечує єдину правду даних (single source of truth), тобто всі користувачі взаємодіють із актуальною інформацією.

Так, структура системи налічує п'ять основних функціональних модулів:

- модуль планування виробничої програми;
- модуль моніторингу виконання програми;
- модуль аналітики та звітності;
- модуль обліку ресурсів;
- модуль адміністрування системи.

Окрім того, між модулями активізовані логічні зв'язки, за допомогою яких реалізується обмін інформацією та синхронізація дій усіх підсистем.

На рисунку 2.3 відтворено модульну структуру системи автоматизації управління програмою запуску та випуску продукції.

Окреслимо переваги запропонованої модульної структури:

- можливість поетапного впровадження системи;
- простоту оновлення та супроводу окремих частин;
- розподілену відповідальність за функціональні завдання;
- масштабованість для підприємств різного рівня складності;
- гнучкість налаштування відповідно до специфіки конкретного виробництва.



Рисунок 2.3 – Модульна структура автоматизованої системи управління програмою запуску та випуску продукції

У таблиці 2.1 зібрано ключові функції, а також визначено призначення основних функціональних модулів.

Таблиця 2.1 – Ключові функції та призначення основних функціональних модулів

Модуль	Основна функція	Вхідні дані	Вихідні дані
Планування	Розроблення виробничої програми, визначення обсягів і строків випуску	Замовлення, ресурси, графіки	Програма запуску
Моніторинг	Контроль виконання планів, фіксація стану обладнання	Дані з виробничих підрозділів	Статус виконання завдань
Облік ресурсів	Реєстрація використання матеріалів, енергії, персоналу	Виробничі операції	Зведення про витрати
Аналітика	Оформлення звітів, аналіз відхилень, прогнозування	Дані плану, моніторингу, обліку	Аналітичні звіти, КРІ
Адміністрування	Керування користувачами, правами доступу, налаштування системи	Дані користувачів	Конфігураційні файли, резервні копії

### 2.2.1 Модуль планування виробничої програми

Модуль планування вважається невід'ємним елементом системи, адже його функціональне призначення спрямовано на оформлення календарно-планових завдань з урахуванням замовлень, ресурсних можливостей, а також виробничих потужностей.

Утім даний модуль виконує низку інших функцій:

- створювати виробничі програми на конкретний період;
- автоматично прораховувати строки виконання;
- визначати обмежені місця у виробничому циклі;
- розподіляти замовлення між виробничими дільницями;
- моделювати варіанти виконання програми, зважаючи на обмеження.

Тож дані, що підлягають опрацюванню, насамперед містять замовлення клієнтів, довідники щодо виробів, матеріалів або ресурсів разом із календарно-плановими нормативами.

За результатами функціонування модуля втілюється виробнича програма запуску продукції, що передається до модулів моніторингу та обліку ресурсів.

### 2.2.2 Модуль моніторингу виконання програми

Модуль моніторингу є актуальним для оперативного контролю стану реалізації виробничої програми у реальному часі [14].

Перелічимо функціональні можливості модуля моніторингу виконання програми:

- фіксування етапів виконання завдань у цехах;
- відстеження простоїв обладнання;
- контролювання щодо завантаження персоналу;
- повідомлення про відхилення від графіка;

– наочне відтворення стану виробництва за допомогою таблиць, графіків чи індикаторів.

Окрім того, модуль взаємодіє з базою даних виробничих операцій, а також містить інтерфейс для відображення статусів («процес триває», «затримка», «виконано»).

Водночас функціонування модуля пов'язується з модулем аналітики, котрий послуговується зібраними даними для оцінювання ефективності виробництва.

### 2.2.3 Модуль обліку ресурсів

Зазначений модуль охоплює облік витрат матеріалів або енергії, праці або часу роботи обладнання.

Спершу він отримує інформацію від виробничих підрозділів, а відтак реалізує подання ресурсних звітів для економічного аналізу.

Так, функціональні завдання модуля обліку ресурсів спрямовані на:

- облік матеріалів, які використовуються на кожному етапі виробництва;
- фіксацію витрат енергоресурсів;
- контроль трудових витрат;
- прораховування фактичної собівартості продукції;
- порівняння планових ресурсів із фактичними.

Модуль власне поєднується з бухгалтерською чи ERP-системою підприємства, забезпечуючи, зі свого боку, узгодженість даних.

### 2.2.4 Модуль аналітики та звітності

Пріоритетною функцією даного модуля є оформлення аналітичних звітів, а також оцінювання ефективності реалізації виробничих програм.

За допомогою засобів статистичного та графічного аналізу модуль насамперед спроможний визначати відхилення та формувати рекомендації щодо оптимізації.

Перелічимо актуальні типи аналітичних звітів:

- план-фактний аналіз реалізації програми;
- коефіцієнт ефективності використання ресурсів;
- показники продуктивності обладнання (OEE);
- графічне відображення тенденцій (Gantt, Pareto, KPI).

Враховуючи інтеграцію з модулем планування, зазначений модуль може пропонувати автоматичні рекомендації щодо коригування планів за умов появи відхилень.

#### 2.2.5 Модуль адміністрування системи

Окреслений модуль зорієнтований на захисті даних, налаштуванні системи й управлінні користувачами.

Зокрема, до функцій модуля адміністрування системи належать:

- додавання та редагування користувачів;
- встановлення ролей і прав доступу;
- аудит дій користувачів;
- резервне копіювання разом із відновленням бази даних;
- налаштування інтерфейсу, звітів і довідників.

Тож модуль адміністрування, насамперед, відповідає за інформаційну безпеку та стабільність функціонування системи.

#### 2.3 Алгоритм роботи системи

Алгоритм функціонування автоматизованої системи управління програмою запуску та випуску продукції побудовано на послідовності реалізації

ключових процесів щодо планування, контролю й аналізу виробничої програми. Діє він за принципом замкнутого управлінського циклу, в якому акцент робиться на повсякчасному оновленні інформації та своєчасному коригуванні виробничих планів. Власне роботу зазначеного алгоритму наочно відтворено на рисунку 2.4.

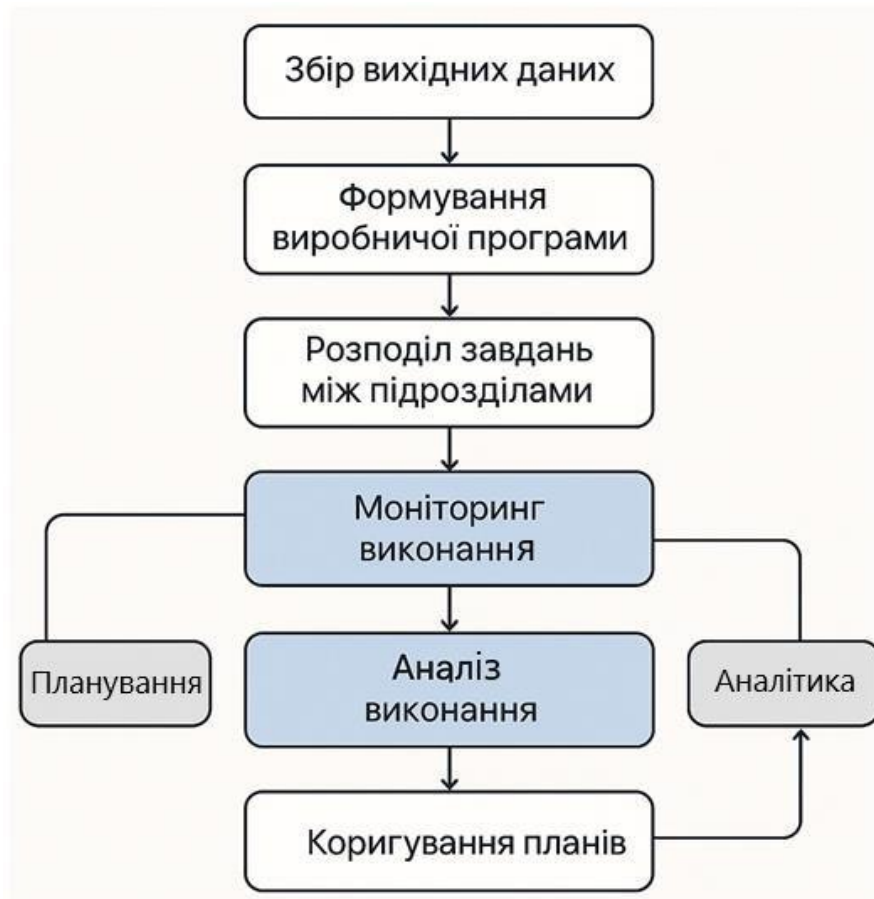


Рисунок 2.4 – Алгоритм роботи автоматизованої системи управління програмою запуску та випуску продукції

Мету алгоритму визначено як мінімізацію часу між надходженням замовлення і фактичним запуском продукції у виробництво шляхом автоматизованого оброблення даних, розроблення планів, а також моніторингу щодо їхньої реалізації.

Так, процес функціонування системи містить декілька етапів, які поєднуються між собою, тобто сумісно формують логічний цикл управління.

Зокрема, блок-схема алгоритму складається з семи ключових етапів:

- розроблення виробничої програми;
- розподілення завдань між підрозділами;
- ініціалізації та збору вхідних даних;
- моніторингу щодо виконання;
- аналізу результатів;
- коригування планів;
- архівації й аналітики.

### 2.3.1 Ініціалізація та збір вхідних даних

Як правило, такий етап охоплює низку процесів:

- імпортування даних про замовлення клієнтів, на кшталт найменування, кількості, термінів;
- отримання інформації щодо наявності матеріалів і комплектування;
- зчитування параметрів про виробничі потужності, а саме: обладнання, завантаження, графіки змін;
- оновлення довідників ресурсів і технологічних маршрутів.

Дані, перелік яких було наведено вище, зберігаються у централізованій БД з метою подальших обчислень і розроблення виробничої програми.

### 2.3.2 Розроблення виробничої програми

Цей етап є важливим для функціонування системи загалом.

Насамперед алгоритм реалізує такі процеси:

- аналізує замовлення за терміновістю, трудомісткістю, а також пріоритетом;

- обчислює потреби у матеріалах і ресурсах;
- утворює оптимальну черговість запуску замовлень із урахуванням обмежень;

- розробляє першопочаткову програму випуску продукції.

Отже, після автоматичного прораховування у користувача з'являється можливість редагувати план вручну, якщо це буде затребувано, зважаючи на специфіку технологічного процесу.

### 2.3.3 Розподіл завдань між виробничими підрозділами

Зазвичай план, який було попередньо розроблено, потім розподіляється між цехами або виробничими ділянками.

Утім кожному підрозділу система надсилає:

- список матеріалів і ресурсів;
- перелік виробів для виготовлення;
- календарний графік запуску;
- трудові нормативи.

Передавання даних втілюється за допомогою електронної мережі чи корпоративного порталу, тобто потреба у паперових розпорядженнях зникає.

### 2.3.4 Моніторинг реалізації програми

Після запуску виробництва система активує режим моніторингу в реальному часі.

За таких обставин, відповідальним необхідно відстежувати:

- фактичний стан виконання завдань (відсоток готовності);
- простої та відхилення від плану;
- витрату матеріалів;
- дотримання змінного графіка.

Так, дані автоматично виводяться на панелі керування (dashboard), зокрема, як таблиці чи графіки.

Окреслений етап загалом реалізує зворотний зв'язок між виробництвом і плануванням.

#### 2.3.5 Аналіз виконання й оформлення звітів

За результатами даних моніторингу система комплектує:

- план-фактний аналіз реалізації програми;
- коефіцієнти ефективності застосування обладнання (Кз, ОЕЕ);
- ресурсні звіти щодо витрат матеріалів, енергії, трудових ресурсів;
- аналітичні графіки відхилень.

Окрім того, модуль аналітики дозволяє визначити обмежені місця, вивчити причини затримок, а також сформувані рекомендації для оптимізації процесів.

#### 2.3.6 Коригування виробничої програми

Якщо під час виробництва система виявляє відхилення, то відразу переходить до етапу автоматичного перепланування, зокрема:

- перерозподіляє ресурси;
- надсилає повідомлення про зміни відповідальним особам у підрозділах;
- оновлює календарний графік.

За таких умов реалізується динамічна адаптація виробничої програми до реального стану виробництва без участі людини у рутинних розрахунках.

#### 2.3.7 Архівація й аналітична підтримка

Зазвичай після завершення циклу дані зберігаються в архіві БД з метою аналізу, побудови прогнозів і статистики у подальшому.

Насправді даний етап створює цифрову історію виробництва, якою можна послуговуватися задля:

- оцінювання ефективності управлінських рішень;
- аналізу завантаження обладнання;
- побудови перспектив для майбутніх періодів.

#### 2.4 Потоки даних між модулями в алгоритмі роботи системи

На рисунку 2.5 продемонстровано схему потоків даних між модулями в алгоритмі функціонування системи, котра, зі свого боку, втілює взаємодію між ключовими модулями:

- планування продукує дані для моніторингу, а також обліку ресурсів;
- моніторинг передає фактичні дані до аналітики;
- аналітика натомість надсилає результати аналізу назад до планування, аби коригувати програми.

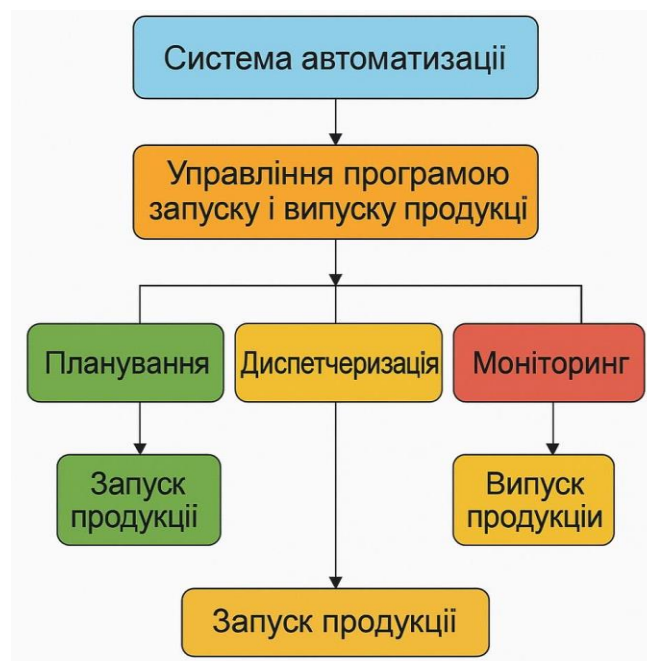


Рисунок 2.5 – Потоки даних між модулями в алгоритмі функціонування системи

Окреслена взаємодія утворює замкнутий цикл управління виробництвом, на кшталт «План → Виконання → Контроль → Аналіз → Корекція».

Тож визначимо переваги реалізованого алгоритму:

- автоматизація рутинних дій під час планування та звітності;
- можливість інтеграції з ERP і MES-системами;
- оперативне реагування на зміни у виробничих умовах;
- підвищення точності планування та контролю виконання;
- скорочення тривалості виробничого циклу;

Алгоритм власне побудовано на принципах адаптивного управління й аналітичного зворотного зв'язку. Таким чином, цілком можливо втілити концепцію «розумного виробництва».

## 2.5 Висновки до розділу

Даний розділ було присвячено розробленню структури та концепції автоматизованої системи управління програмою запуску та випуску продукції за допомогою принципів модульності й інтегрованості, відкритості та масштабованості. Разом з тим було детально обґрунтовано вибір архітектури, визначено базові функціональні модулі, алгоритми їхні взаємодії та нагальні інформаційні потоки.

Так, модульний підхід дозволяє адаптувати систему відповідно до специфіки підприємства, а застосування єдиної БД реалізує узгодженість і достовірність інформації. До того ж, модульна структура є підґрунтям для впровадження ефективної автоматизованої системи управління програмою запуску та випуску продукції. Розроблені модулі реалізують комплексне задіяння всіх етапів виробничого процесу: від планування до аналітичного контролю. Інтеграція між ними на тлі єдиної БД втілює узгодженість інформації

й оперативність управління, водночас можливість масштабування системи в майбутньому.

Розроблений алгоритм функціонування системи є основою роботи ПЗ, реалізуючи безперервний контроль щодо виконання виробничих програм, автоматичне оновлення графіків виробництва, аналітичну підтримку управлінських рішень, а також ефективну координацію між усіма модулями системи. Утім запропонована логіка сприяє зменшенню навантаження на персонал, підвищенню продуктивності, а отже, і забезпеченню стабільного функціонування підприємства.

Тож окреслена автоматизована система управління може бути реалізована як веборієнтований програмний комплекс із розподіленим доступом користувачів, а також взаємодією з системами обліку матеріалів, постачання і фінансів. Запропонований підхід цілком відповідає вимогам цифрової трансформації виробництва сучасності та концепції «Індустрія 4.0».

## 3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО КОМПЛЕКСУ АВТОМАТИЗОВАНОЇ СИСТЕМИ УПРАВЛІННЯ

### 3.1 Архітектура програмного комплексу

Архітектура програмного комплексу невід’ємна складова автоматизованої системи управління програмою запуску та випуску продукції, адже окреслює як логіку взаємодії компонентів, так і принципи оброблення інформації, способи забезпечення надійності та масштабованості, а отже, і можливості інтеграції з іншими корпоративними інформаційними системами [15-18].

У кваліфікаційній роботі розробляється багаторівнева клієнт-серверна архітектура, що відповідає тенденціям цифровізації промисловості сучасності та вимогам концепції «Індустрія 4.0».

На рисунку 3.1 продемонстровано архітектуру програмного комплексу автоматизованої системи управління програмою запуску та випуску продукції, котра, як видно з наочного прикладу, побудована за трирівневою моделлю [17, 18].

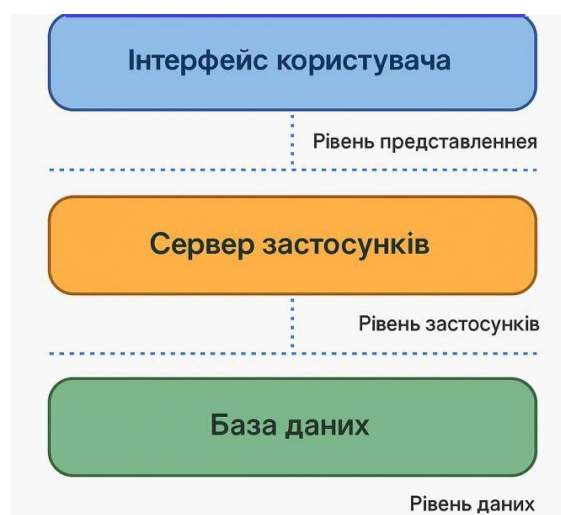


Рисунок 3.1 – Архітектура програмного комплексу

Зокрема, тривимірна модель (Three-Tier Architecture) програмного комплексу містить:

- рівень представлення (Presentation Layer) реалізує взаємодію користувача з системою за допомогою браузера чи іншого інтерфейсу;
- логічний рівень (Application Layer) залучає бізнес-логіку, працює з даними, реалізує обчислення та алгоритми;
- рівень даних (Data Layer) охоплює зберігання та структурування, індексацію та доступ до інформації у БД;

Окрім того, зазначена структурна модель є засобом для розподілу відповідальності, підвищення безпеки, а також полегшення масштабування системи за умови збільшення кількості користувачів і обсягів інформації.

Утім вибір трирівневої архітектури насамперед зумовлено низкою як технічних, так і експлуатаційних переваг [18]:

а) масштабованість. У разі збільшення кількості робочих місць або примноження обсягу даних можна масштабувати незалежно:

- 1) сервер БД – для прискореного оброблення запитів;
- 2) сервер застосунків – для збільшення обчислювальної потужності;
- 3) клієнтський інтерфейс – через балансувальники навантаження;

б) підвищена безпека. Як відомо, рівень даних ізольований від користувача, тож доступ до нього реалізується лише через сервер застосунків, який, зі свого боку, зорієнтовано на:

- 1) контроль прав доступу;
- 2) шифрування даних, які передаються;
- 3) фільтрацію запитів;
- 4) захист від SQL-ін'єкцій;

в) гнучка підтримка та модернізація. Окремі модулі доступні для змін або доповнень без впливу на інші. Так, оновлення інтерфейсу не відображається на

логіці роботи чи БД;

г) висока продуктивність. Зазвичай розподіл навантаження сприяє оптимізації взаємодії між компонентами. Водночас сервер застосунків виконує попереднє оброблення та кешування, зменшуючи, таким чином, навантаження на БД.

### 3.1.1 Опис рівнів архітектури системи

Рівень представлення втілено як веб-інтерфейс, який активує доступ до функцій системи з будь-якого пристрою [18].

Власне даний рівень вирізняється:

- адаптивним дизайном (Bootstrap 5);
- інтуїтивним меню модулів;
- інтерактивними графіками (Chart.js) ;
- таблицями з фільтрами та сортуванням;
- панелями стану з КРІ та індикаторами виконання.

До того ж, інтерфейс підтримує виокремлення категорій, а саме: адміністратор, диспетчер, плановик, керівник підрозділу й аналітик.

Логічний рівень (сервер застосунків) утворено за допомогою фреймворку Django, який, насамперед реалізує:

- автоматичну генерацію API;
- модульність структури програмного коду;
- опрацювання GET/POST/PUT/DELETE запитів;
- чітке розрізнення моделей, контролерів і шаблонів;

До ключових компонентів логічного рівня належать:

- модуль планування, як правило, розробляє виробничі програми, визначає терміни, а також перевіряє наявність ресурсів;
- модуль моніторингу спрямовано на отримання даних про виконання від

різних підрозділів, оновлює індикатори та графіки в режимі реального часу;

- модуль аналітики, насамперед виконує статистичне оброблення даних, будує аналітичні графіки та формує звіти у форматах PDF та Excel;
- модуль обліку ресурсів, зі свого боку, фіксує витрати, формує ресурсні картки, аналізує фактичні показники;
- модуль адміністрування, зокрема, керує користувачами, проводить аудит дій, виконує резервування та відновлення даних.

Підкреслимо, що рівень даних також послуговується реляційною системою управління БД (СУБД) PostgreSQL, оскільки вона характеризується високою надійністю, транзакційною цілісністю (ACID), підтримкою складних запитів, а також безпечним зберіганням даних [19, 20].

Окрім того, у БД наявні такі таблиці: Orders, Execution, Resources, ProductionProgram, Reports та Users.

З метою оптимізації роботи задіяно індексацію полів, зовнішні ключі, нормалізацію до третьої нормальної форми.

### 3.1.2 Особливості та вимоги архітектури системи

Зазвичай механізм взаємодії компонентів (між клієнтом, сервером і БД) реалізується через REST API [21].

Відтворимо загальноприйнятую послідовність взаємодії компонентів:

- користувач уводить дані;
- запит надходить до сервера;
- сервер перевіряє права;
- опрацьовує дані;
- виконує запити до бази;
- повертає результати клієнту.

Утім задля забезпечення надійності та захисту інформації доцільно:

- проводити аутентифікацію й авторизацію користувачів;
- задіювати хешування паролів (bcrypt);
- використовувати захищений протокол HTTPS;
- активувати захист від XSS, CSRF, SQL-ін'єкцій;
- записувати до журналів події та помилки;
- реалізовувати автоматичне резервне копіювання БД.

Тож для коректного функціонування системи чинне серверне обладнання повинно відповідати мінімальним вимогам. Зокрема :

- 4-ядерний процесор (Intel i5 / Ryzen 5);
- оперативна пам'ять на 8 – 16 ГБ;
- DDR або SSD до 120 – 240 ГБ;
- ОС: Linux Ubuntu Server 20.04 або вище.

За умови збільшення навантаження можна запроваджувати горизонтальне (кілька серверів застосунків) та вертикальне (збільшення ресурсів сервера БД) масштабування.

Виокремимо переваги побудованої архітектури системи, зокрема:

- високу продуктивність та надійність;
- можливість інтеграції з ERP, MES, SCADA у подальшому;
- легку підтримку та розширення;
- адаптивний інтерфейс для користувачів з урахуванням їхніх уподобань;
- уніфіковані механізми обміну даними;
- підтримку цифрової трансформації виробництва.

Отже, представлена архітектура програмного комплексу втілює стабільне й ефективне функціонування системи автоматизації, підтримку оброблення великих обсягів даних, безпеку інформації, здатність до масштабування й інтеграції з іншими цифровими платформами підприємства. Також вона цілком відповідає сучасним вимогам виробничих підприємств, водночас може

застосовуватися як фундамент для подальшого розвитку системи [22].

### 3.2 Середовище розробки та мови для створення програмного комплексу

Середовище розроблення програмної частини системи набуває вагомого значення на тлі реалізації її працездатності, масштабованості та надійності. Утім правильний вибір відповідних технологій та інструментів сприяє оптимізації процесу створення ПЗ, зменшенню ризиків, а також підвищенню продуктивності разом із забезпеченням цифрової трансформації виробничих підприємств, які, зі свого боку, відповідають сучасним вимогам.

У даному підрозділі докладно описані технологічні засоби та мови програмування, інструменти тестування та платформи, на яких ґрунтується функціонування системи.

#### 3.2.1 Програмне середовище розробки

Розроблення системи було визначено втілювати в інтегрованому середовищі програмування Visual Studio Code (VS Code), адже це сучасний та легкий, а водночас і високопродуктивний редактор, який підтримує розширення для Python, JavaScript, HTML, CSS та API-інструментів [23].

Більш детально окреслимо переваги VS Code:

- системи керування підтримується версіями Git;
- значна кількість плагінів (Python, Django Snippets, REST Client);
- вбудована підтримка терміналу;
- спроможність працювати у контейнерах і віртуальних середовищах;
- навігація за проектом доволі зручна.

Окрім того, вибір пав на такі додаткові середовища:

- Postman – для тестування REST API;

- PgAdmin 4 – для адміністрування бази даних PostgreSQL;
- Docker Desktop – для контейнеризації окремих модулів системи.

### 3.2.2 Мови програмування та фреймворки

Для серверної логіки (Backend), з огляду на властивості, доцільно застосувати мову програмування Python 3.11, адже вона характеризується [23]:

- високою читабельністю коду;
- значною кількістю бібліотек для оброблення даних;
- швидким розробленням застосунків;
- підтримкою масштабованих веб-фреймворків.

Окрім того, задля реалізації серверної частини також залучено Django 4.0, оскільки йому притаманні [24]:

- модельно-орієнтований підхід до проєктування даних (ORM);
- автоматична генерація SQL-запитів;
- вбудовані засоби захисту (CSRF, XSS);
- чітке розмежування логіки, відтворення та моделі.

Серед додаткових бібліотек Python виокремили декілька затребуваних:

- Django Rest Framework – розроблення REST-інтерфейсу;
- pandas, NumPy – оброблення статистичних даних;
- ReportLab, openpyxl – генерація PDF- і Excel-звітів;
- gunicorn – WSGI-сервер для продакшн-розгортання.

Однак для розроблення клієнтського інтерфейсу (Frontend) було задіяно [25]:

- AJAX / Fetch API – динамічне завантаження даних без перезавантаження сторінки;
- CSS3 (Bootstrap 5) – стильове оформлення й адаптивність;
- HTML5 – розмітка сторінок;

- JavaScript – інтерактивна логіка.

Також послуговувалися і додатковими інструментами, зокрема:

- Chart.js / Plotly – побудова графіків, діаграм та KPI-індикаторів;
- jQuery (частково) – для прискорення роботи з DOM;
- DataTables – розширена робота з таблицями (фільтрація, сортування, експорт).

Зауважимо, що у системі управління БД, з метою як зберігання, так і структуризації інформації використано PostgreSQL 15, тобто одну з найнадійніших реляційних СУБД класу enterprise [20].

Окреслимо переваги PostgreSQL для даної роботи:

- підтримка ACID-транзакцій;
- висока продуктивність за умови великих обсягів даних;
- надійність і можливість резервування;
- підтримка складних типів даних і тригерів;
- гнучкі механізми масштабування.

СУБД, як правило, функціонує у кластерному режимі з можливістю реплікації. Тож для адміністрування було обрано PgAdmin 4, з огляду на можливість щодо:

- створення таблиць та індексів;
- відслідковування активних з'єднань;
- виконання SQL-запитів;
- проведення резервного копіювання.

У системі управління версіями та командної роботи задля підтримки управління проектом задіяно Git, що, зі свого боку, реалізує:

- історію про зміни коду;
- підтримку паралельного розроблення у різних гілках;
- контроль конфліктів під час злиття;

– можливість командного доступу.

Власне репозиторій розміщено на GitHub. Таким чином, забезпечується зберігання коду у хмарі, автоматизація (GitHub Actions) та контроль доступу для різних категорій розробників.

Зважаючи на безпеку та надійність розроблення програмного комплексу, необхідно виконувати:

- захист від CSRF-атак;
- контроль журналу подій (audit log);
- обмеження користувачів відповідно до їхньої специфікації;
- регулярне резервування БД (cron);
- хешування паролів bcrypt / PBKDF2;
- шифрування каналу зв'язку HTTPS.

Окреслені засоби утворюють надійну, продуктивну та масштабовану платформу задля реалізації автоматизованої системи управління запуском і випуском продукції.

Задіявши під час роботи Python, Django та PostgreSQL, вдалося доволі швидко побудувати модульну архітектуру з високим рівнем безпеки. Водночас сучасні інструменти забезпечують стабільність та коректність функціонування програмного комплексу на всіх робочих режимах.

Власне структуру програмного комплексу відтворено у Додатку А, а лістинг програми – у Додатку Б.

### 3.3 База даних і структура даних

БД є одним із вагомих компонентів програмного комплексу, адже забезпечує зберігання, оброблення разом із передаванням виробничої інформації між усіма модулями системи. Зокрема, її структура й ефективність

функціонування впливають на продуктивність усього ПЗ, а також швидкість виконання операцій, точність планування й аналітичних обчислень [19, 26].

У цьому підрозділі відтворено детальний опис логічної та фізичної моделі БД, структури таблиць, системи зв'язків, а також механізмів, які забезпечують цілісність даних.

На рисунку 3.2 продемонстровано структурну схему БД автоматизованої системи управління програмою запуску та випуску продукції.

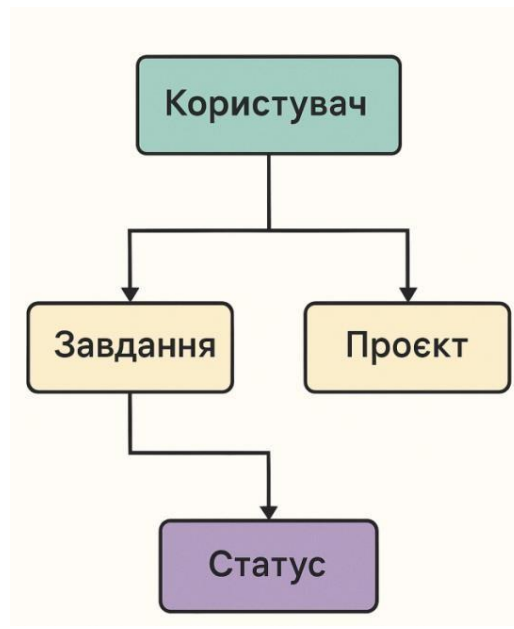


Рисунок 3.2 – Структурна схема бази даних автоматизованої системи управління програмою запуску та випуску продукції

### 3.3.1 Загальні принципи проектування бази даних

Розроблення БД було виконано із дотриманням низки принципів [19, 26]:

а) модульності та розподілення відповідальності. Насамперед йдеться про спрощення масштабування, розширення та модифікації структури. Разом з тим кожна таблиця відповідає окремому напрямку виробничої діяльності:

- 1) замовлення;
- 2) виробнича програма;

- 3) ресурси;
- 4) виконання операцій;
- 5) результати та звітність;
- 6) користувачі та права доступу.

б) нормалізації. Всі таблиці упорядковані щонайменше до третьої нормальної форми (ЗНФ). Зі свого боку, це гарантує:

- 1) уникнення дублювання даних;
- 2) зменшення ймовірності помилок під час оновлення;
- 3) прискорення пошуку та фільтрації;
- 4) підвищення цілісності.

в) забезпечення цілісності даних. Зокрема, були встановлені:

- 1) первинні ключі (PRIMARY KEY);
- 2) зовнішні ключі (FOREIGN KEY);
- 3) обмеження NOT NULL, UNIQUE;
- 4) каскадне оновлення та видалення.

### 3.3.2 Концептуальна модель бази даних

У концептуальній моделі БД виокремлено сім пріоритетних компонент [19, 26]:

- Execution – фактичні показники щодо виконання операцій;
- LogActions – журнал дій користувачів.
- Orders – замовлення на виготовлення продукції;
- ProductionProgram – планові дані щодо запуску та випуску продукції;
- Reports – статистичні та аналітичні показники;
- Resources – матеріальні, технічні та трудові ресурси;
- Users – адміністрування доступу;

Усі компоненти взаємодіють між собою за допомогою логічних зв'язків, на кшталт:

- 1:N (один-до-багатьох), а саме: одне замовлення → багато етапів виконання;
- N:M (багато-до-багатьох) використовуються проміжні таблиці, а саме: ресурси ↔ виробничі програми.

### 3.3.3 Логічна модель бази даних

На рисунку 3.3 продемонстровано логічну модель (ER-діаграма) БД.

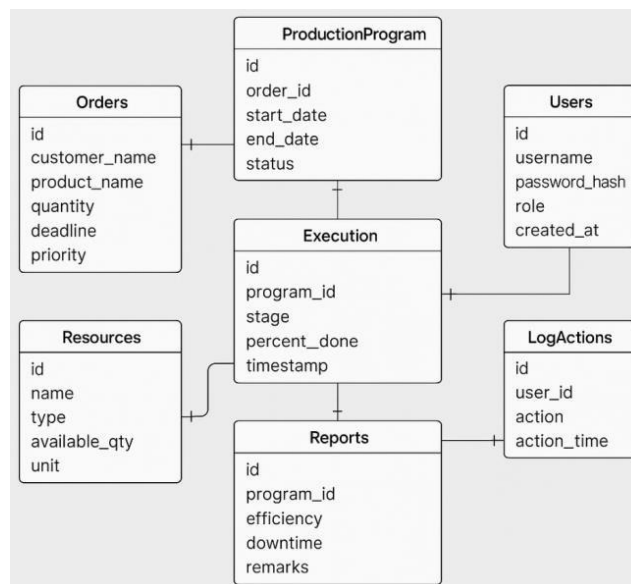


Рисунок 3.3 – ER-діаграма бази даних

Нижче оформлено типові таблиці із повним переліком затребуваних полів.

Так, у таблиці 3.1. зібрано інформацію про замовників, продукцію, а також про параметри виготовлення.

Таблиця 3.1 – Заовлення (Orders)

Поле	Тип	Опис
id	SERIAL PK	Унікальний ідентифікатор
customer_name	VARCHAR(100)	Назва клієнта
product_name	VARCHAR(100)	Номенклатура продукції
quantity	INTEGER	Обсяг замовлення
deadline	DATE	Термін виконання
priority	INTEGER	Пріоритет замовлення

Інформацію про збереженні дані разом із виробничими планами зібрано у таблиці 3.2.

Таблиця 3.2 – План виробництва (ProductionProgram)

Поле	Тип	Опис
id	SERIAL PK	Ідентифікатор
order_id	INT FK → Orders	Замовлення
start_date	DATE	Дата запуску
end_date	DATE	Дата завершення
status	VARCHAR(20)	Статус (заплановано/в роботі/виконано)

У таблиці 3.3. зазначено опис матеріалів, обладнання чи персонала.

Таблиця 3.3 – Ресурси (Resources)

Поле	Тип	Опис
id	SERIAL PK	Унікальний номер ресурсу
name	VARCHAR(100)	Назва ресурсу
type	VARCHAR(50)	Тип (матеріал/обладнання/персонал)

Продовження таблиці 3.3

Поле	Тип	Опис
available_qty	INTEGER	Доступна кількість
unit	VARCHAR(20)	Одиниця вимірювання

Інформацію про фактичні дані виробництва зібрано у таблиці 3.4.

Таблиця 3.4 – Виконання (Execution)

Поле	Тип	Опис
id	SERIAL PK	Ідентифікатор
program_id	INT FK	Програма виробництва
stage	VARCHAR(100)	Етап виробництва
percent_done	NUMERIC	Відсоток готовності
timestamp	TIMESTAMP	Час фіксації

У таблиці 3.5. відображено інформацію за результатами аналітичних обчислень.

Таблиця 3.5 – Звіти (Reports)

Поле	Тип	Опис
id	SERIAL PK	Ідентифікатор
program_id	FK → ProductionProgram	Програма
efficiency	NUMERIC	Показник ефективності
downtime	NUMERIC	Простію (хвилини)
remarks	TEXT	Примітки

Інформацію щодо забезпечення контролю доступу подано у таблиці 3.6.

Таблиця 3.6 – Користувачі (Users)

Поле	Тип	Опис
id	SERIAL PK	Ідентифікатор
username	VARCHAR(80)	Логін
password_hash	VARCHAR(200)	Захешований пароль
role	VARCHAR(50)	Роль користувача
created_at	TIMESTAMP	Дата створення

Інформацію про аудит зібрано у таблиці 3.7.

Таблиця 3.7 – Журнал дій (LogActions)

Поле	Тип	Опис
id	SERIAL PK	Ідентифікатор
user_id	FK → Users	Користувач
action	TEXT	Дія
action_time	TIMESTAMP	Час виконання

Зауважимо, що між таблицями бази даних задіяні такі взаємозв'язки [19, 26]:

- «Orders → ProductionProgram (1:N)», тобто одне замовлення може стосуватися кількох планів;
- «ProductionProgram → Execution (1:N)», де кожен план втілює низку етапів виконання;
- «ProductionProgram ↔ Resources (M:N)», коли один план може залучати багато ресурсів чи навпаки;
- «ProductionProgram → Reports (1:N)», зокрема, кожна програма може оформлювати безліч звітів;

– «Users → LogActions (1:N)», наприклад, один користувач може виконувати багато дій.

### 3.3.4 Механізми забезпечення продуктивності бази даних

З метою підвищення швидкодії БД доцільно послуговуватися такими механізмами для забезпечення продуктивності:

– індексацією. Проводиться індексування полів (order\_id, program\_id, timestamp), а також використовуються комбіновані індекси задля прискорення пошуку;

– оптимізацією SQL-запитів. Оминаються вкладені підзапити, а також для відображення складних даних використовується VIEW;

– кешуванням. Застосовується кешування результатів важких аналітичних запитів у Redis (опціонально);

– резервуванням. Задіюється щоденне автоматичне резервне копіювання і механізм відновлення бази після аварій.

### 3.3.5 Переваги окресленої структури бази даних

Серед переваг окресленої структури БД виокремимо:

– високу гнучкість розширення (відкритість для додавання нових модулів і таблиць без зміни основи);

– високу цілісність разом із достовірністю даних;

– підтримку великих обсягів інформації;

– оптимальну структуру для аналітичних запитів;

– підтримку інтеграції з ERP/MES-системами.

Так, структура БД, яку було запропоновано, забезпечує насамперед надійне та продуктивне, водночас масштабоване зберігання інформації, необхідної для функціонування автоматизованої системи управління запуском і

випуском продукції. Зі свого боку, вона підтримує всі функціональні потреби програмного комплексу, а також дозволяє легко інтегруватися з іншими інформаційними системами підприємства, отже, забезпечує підґрунтя для розширення аналітичних можливостей [19, 26].

### 3.4 Інтерфейс користувача програмного комплексу

Інтерфейс користувача відіграє невід’ємну роль серед складових програмного комплексу, адже він реалізує взаємодію між оператором та програмною системою. Втім від якості його розроблення залежить як ефективність роботи персоналу, так і швидкість виконання виробничих операцій. Йдеться також про зручність доступу до інформації разом із мінімізацією ймовірності помилкових дій.

У даному підрозділі зібрано детальний опис структури та функціональних блоків і принципів проектування інтерфейсу користувача автоматизованої системи управління програмою запуску та випуску продукції на виробничому підприємстві.

Під час розроблення інтерфейсу було зроблено акцент на сучасні підходи UX/UI-дизайну, адже вони містять такі принципи [27, 28]:

- простоту й інтуїтивність. Як правило, всі елементи інтерфейсу упорядковані логічно та структуровано, тому додаткове навчання користувачеві не потрібно, оскільки базових навичок роботи з веб-інтерфейсами цілком достатньо.

- адаптивність. Загалом на стаціонарних комп’ютерах або ноутбуках, планшетах або смартфонах інтерфейс відображається коректно. Така властивість є особливо важливою для диспетчерів, адже тоді вони можуть контролювати стан виробництва поза робочим місцем;

– узгодженість візуальних елементів. Власне система застосовує у стилі єдність оформлення, тобто кольорова гама та типографіка, структура таблиць, а також кнопок відповідають корпоративному дизайну, отже, відтворюють однакове сприйняття всіх модулів;

– мінімізація кількості дій. Будь-яку операцію в системі можна виконати, зазвичай, не більше, ніж за 2 чи 3 кліки. Так, відстежувати стан виконання замовлення цілком реально безпосередньо з панелі моніторингу;

– захист від помилкових дій. Усі критичні операції на кшталт видалення чи редагування, навіть коригування плану супроводжуються діалоговими вікнами, що відображають опції підтвердження чи скасування.

Зазвичай основна частина інтерфейсу розподілена на кілька логічних зон, як продемонстровано на рисунку 3.4.

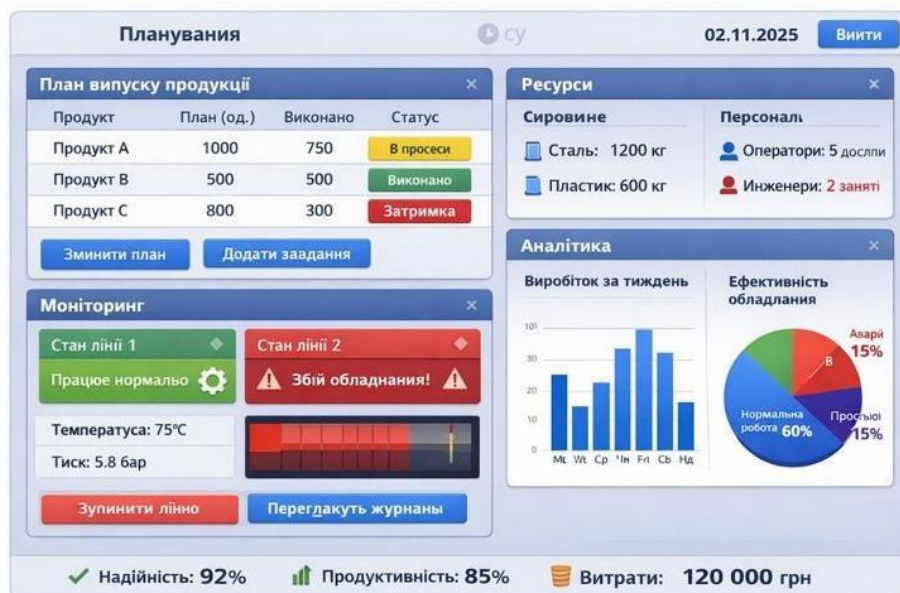


Рисунок 3.4 – Головне вікно інтерфейсу користувача

Головна панель навігації розташована у лівій частині екрану (SideBar), де розташовуються такі розділи, зокрема, планування та моніторинг виробництва, ресурси й аналітика, звітність та адміністрування, а також налаштування

користувача. До того ж, панель залишається видимою навіть під час перемикання між модулями, тож реалізує швидку навігацію.

Верхня службова панель (Header), зі свого боку, відображає ім'я користувача, поточну роль, кнопку швидкого доступу до журналу подій, а також кнопку виходу із системи разом із індикатором сповіщень.

Основна робоча область розташована у центральній частині інтерфейсу, в якій зазвичай заповнюються таблиці замовлень, форми введення даних, створюються діаграми виконання програм або графіки КРІ, а також звіти і дашборди. Хоча кожному з перелічених форматів і властиве власне робоче середовище, втім структура відображення буде однаковою для всієї системи, завдяки чому знижується когнітивне навантаження на користувача.

Панель швидкого доступу функціонує праворуч. Як правило, вона відображає статус ресурсів та індикатори виконання, перелік активних програм разом із швидкими діями на кшталт нового замовлення чи звіту, оновлення даних.

Контекстні меню, зі свого боку, з'являються автоматично під час роботи з таблицями: «відкрити програму» або «переглянути виконання», «додати етап виробництва» або «створити звіт» тощо.

Форми введення даних відповідають принципу мінімального введення, зокрема:

- усі поля групуються за категоріями: замовлення, виробнича програма, ресурси, коригування;
- форми активують автоматичну перевірку правильності введення, а саме, вказують на поля для обов'язкового заповнення, формати дат або числові діапазони.

Усі таблиці підтримують сортування, пошук у реальному часі, пагінацію, експорт даних у формати Excel або PDF, а також кольорове позначення статусів.

Інтерактивні графіки побудовані за допомогою бібліотек Plotly та Chart.js: план-факт аналіз, завантаження обладнання, використання ресурсів і динаміка виконання програми.

До того ж, вони динамічно оновлюються у разі зміни даних у БД.

Інтерфейс адаптується під категорію користувача, що насамперед визначає рівень доступу. Кожна роль, яка надається певному користувачу, містить свій набір дозволених елементів у головному меню, зокрема на прикладі таблиці 3.8.

Таблиця 3.8 – Рівень доступу та доступні функції

Роль	Доступні функції
Адміністратор	налаштування користувачів, ролей, прав доступу, аудит
Плановик	створення та редагування виробничих програм
Диспетчер	моніторинг виконання в режимі реального часу
Аналітик	формування звітів, аналіз відхилень
Керівник	перегляд зведених показників, прийняття рішень

Інтерфейс безпосередньо взаємодіє з сервером через REST API, що реалізує незалежність клієнта та сервера, а також можливість використання мобільних застосунків у подальшому, водночас гарантує підвищену безпеку передавання даних.

Наведемо приклад типової схеми взаємодії: користувач натискає кнопку → JS-скрипт створює HTTP-запит → сервер обробляє запит → дані вносяться у базу → клієнту повертається відповідь → інтерфейс оновлює вміст без перезавантаження сторінки (AJAX).

Тож створений інтерфейс користувача насамперед є гнучким, інтуїтивно зрозумілим і функціонально повним. Він вирізняється зручним доступом до всіх

модулів системи, сприяє ефективному управлінню виробничими програмами та ресурсами, а також допомагає зменшити час виконання операцій та появу ймовірних помилок.

Таким чином, залучення багаторівневого підходу до організації інтерфейсу, підтримка різних ролей користувачів, а також послуговування інтерактивними елементами характеризують систему як сучасний та ефективний інструмент цифрового управління виробництвом [27, 28].

### 3.5 Тестування та валідація системи

Тестування та валідація програмного комплексу – неодмінні етапи розроблення програмного забезпечення, котрі, зі свого боку, підтверджують його коректність, надійність та безпеку, а також підкреслюють його відповідність вимогам технічного завдання. У даному підрозділі опрацьовані методи, сценарії та результати тестування [29-31].

Власне проведення тестування програмного комплексу є обов'язковим. Тож за його допомогою:

- перевіряють правильність функціонування модулів;
- виявляють помилки та недоліки;
- оцінюють стабільність роботи відповідно до навантаження;
- перевіряють зручність інтерфейсу;
- підтверджують, чи відповідає функціонал проєктній документації.

З метою комплексного оцінювання системи були проведені певні види тестування, які є нагальними для нашої роботи :

- модульне тестування (Unit Testing). Окремі функції та методи були перевірені за допомогою модулів Django, зокрема, розроблення графіка виробництва, перевірка ресурсів, генерація звітів та API-запити;

- інтеграційне тестування. Власне було проведено тестування щодо взаємодії між: модулем планування та БД, модулем моніторингу та API, модулем аналітики та модулем виконання. До того ж виконано перевірку щодо коректності передавання та збереження даних;

- функціональне тестування. Спрямовано на перевірку щодо виконання вимог технічного завдання: формування виробничої програми, оброблення відхилень, побудова аналітичних графіків, а також відображення стану виконання. Окрім того, було розроблено тестові сценарії для кожної функції;

- навантажувальне тестування. Було реалізовано за допомогою JMeter, а також імітації на кшталт 50 користувачів одночасно, 10 000 записів у таблицях, інтенсивність API-запитів. Утім тест визначив: середній час відповіді сервера ~ 0,8 с, відсутність збоїв у разі пікових навантажень;

- тестування інтерфейсу (UI Testing). Задіявши Selenium, вдалось перевірити: відповідність відображення елементів, функціонування кнопок і меню, коректність форм та повідомлень;

- тестування безпеки. Насамперед було перевірено авторизацію та стійкість до SQL-ін'єкцій, поведінку системи у разі некоректних запитів, а також CSRF-захист.

Результати тестування зведені у таблиці 3.9.

Отже, можемо підсумувати, що система успішно пройшла всі перевірки та продемонструвала такі вагомні результати:

- стабільну роботу за умови різних навантажень;
- коректність усіх обчислень;
- підвищену продуктивність (на 35 – 40 % швидше за очікуване значення);
- зниження на 60 % помилок введення завдяки вбудованій валідації;
- швидке формування звітів (~ 6 с);

– повну працездатність API.

Таблиця 3.9 – Результати тестування програмного комплексу

Тестований параметр	Метод тестування	Очікуваний результат	Фактичний результат	Відповідн. очікуванням
Час розроблення виробничої програми	Навантажув. тестування	$\leq 20$ с	14 с	Відповідає
Коректність оброблення даних замовлень	Модульне тестування	Система приймає всі допустимі формати	Усі формати оброблено успішно	Відповідає
Реакція системи на відсутність даних	Перевірка виключень	Формування повідомлення про помилку	Повідомлення сформовано	Відповідає
Час оновлення статусу викон.	Інтеграційне тестування	$\leq 3$ с	1,8 с	Відповідає
Опрацювання великих масивів даних	Навантажувал. тест (10 000 записів)	Система не повинна зависати	Працює стабільно	Відповідає
Формування аналітичних звітів	Функціональне тестування	Формування $\leq 10$ с	6 с	Відповідає
Коригування виробничої програми	Ручне + автоматичне тестування	План оновлюється без помилок	Оновлюється коректно	Відповідає
Робота авторизації користувачів	Тестування безпеки	Успішний вхід/відмова у разі помилки	Працює правильно	Відповідає
Стійкість до одночасного доступу	Тест навантаження	Відсутність критичних затримок	Максимальна затримка 0,5 с	Відповідає
Відновлення після збоїв	Тест аварійного завершення	База даних відновлюється автоматично	Успішне відновлення	Відповідає

Тож результати тестування підтвердили, що розроблений програмний комплекс характеризується високою якістю. Окрім того, система відповідає

вимогам технічної специфікації, стабільно функціонує, швидко обробляє дані, а також реалізує високий рівень надійності і безпеки. Комплексне тестування, зі свого боку, засвідчило готовність системи до експлуатації на виробничому підприємстві.

### 3.6 Висновки до розділу

У третьому розділі було втілено комплексне програмно-технічне обґрунтування розробленої автоматизованої системи управління програмою запуску та випуску продукції на виробничому підприємстві.

Результати дослідження підтвердили ефективність утворених моделей, алгоритмів і програмних засобів, а також підкреслили можливість практичного застосування системи в умовах реального виробництва.

Таким чином, розроблена автоматизована система спроможна:

- підвищити ефективність планування виробництва;
- скоротити час формування виробничих програм;
- зменшити ризики помилок;
- забезпечити прозорість процесів;
- покращує оперативність управлінських рішень;
- стимулює зменшення простоїв та оптимізацію ресурсів.

Окрім того, було підтверджено наукову та практичну цінність розробленого програмного комплексу. Відтак програмний комплекс можна інтегрувати до цифрової інфраструктури підприємства, адже він адаптований до умов різних видів виробництва.

## 4 ОЦІНКА НАДІЙНОСТІ ТА СТІЙКОСТІ АВТОМАТИЗОВАНОЇ СИСТЕМИ

У даному розділі увагу приділено проведенню комплексного оцінювання щодо надійності та стійкості. Окрім того було вивчено безвідмовність розробленої автоматизованої системи управління програмою запуску та випуску продукції.

Загалом система містить декілька логічних і технічних підсистем, зокрема, серверну частину, БД, клієнтський інтерфейс, а також мережеву інфраструктуру. Тож актуальною потребою стає визначення її здатності щодо стабільного функціонування в умовах зовнішніх та внутрішніх впливів, у разі відмов окремих компонентів або надмірного навантаження.

### 4.1 Загальні принципи оцінювання надійності та стійкості

Насамперед автоматизована система управління повинна вирізнятися [32]:

- стійкістю, тобто спроможністю зберігати працездатність, незважаючи на певні чинники, на кшталт збоїв в мережі, пікових навантажень або дефіциту ресурсів;
- надійністю, йдеться про забезпечення безвідмовної роботи протягом визначеного інтервалу часу;
- відмовостійкістю, що окреслюється здатністю зберігати працездатність навіть у разі відмови окремих елементів;
- витривалістю, що полягає у спроможності відновлюватися після збоїв і аварій.

Власне структурну схему стійкості автоматизованої системи управління програмою запуску та випуску продукції продемонстровано на рисунку 4.1.

Натомість підкреслимо, що стійкість системи в інформаційно-керуючих комплексах окреслюється і технічними компонентами, й алгоритмами оброблення даних, і структурою БД, і якістю взаємодії між модулями [32, 33].



Рисунок 4.1 – Структурна схема стійкості автоматизованої системи

Задля того, щоб краще оцінити стійкість системи доречно виконати її структурний аналіз. Як відомо, система складається з низки ключових компонентів, таких як:

- серверу застосунків (Application Server);
- БД PostgreSQL;
- клієнтської частини (Web UI);
- мережевої інфраструктури (локальної мережі, Internet VPN);
- системи автентифікації та безпеки;
- служби резервного копіювання.

Кожен із зазначених компонентів цілком може спричинити появу відмов.

Тож з метою оцінювання стійкості необхідно проаналізувати такі параметри, як:

- показники безвідмовності;
- можливості дублювання та резервування;

- час реакції системи;
- час, який необхідний для відновлення після відмови (MTTR);
- вірогідність відмови (Pf).

## 4.2 Обчислення надійності та стійкості системи

### 4.2.1 Математична модель оцінювання надійності

Для системи, в якій елементи дотримуються послідовного зв'язку, загальну вірогідність безвідмовного функціонування можна відобразити як [34]:

$$P(t) = e^{-\sum \lambda_i t}, \quad (4.1)$$

де  $\lambda_i$  – інтенсивність відмови  $i$ -го компонента;

$t$  – час, протягом якого система власне працює.

З огляду на зазначений випадок доцільно врахувати три ключові компоненти:

- сервер –  $\lambda_1 = 1,0 \cdot 10^{-5}$  (1/год);
- БД PostgreSQL –  $\lambda_2 = 0,8 \cdot 10^{-5}$  (1/год);
- мережеву інфраструктуру –  $\lambda_3 = 1,2 \cdot 10^{-5}$  (1/год);

Відтак загальна інтенсивність відмов може дорівнювати:

$$\lambda_{\Sigma} = \lambda_1 + \lambda_2 + \lambda_3, \quad (4.2)$$

$$\lambda_{\Sigma} = 1,0 \cdot 10^{-5} + 0,8 \cdot 10^{-5} + 1,2 \cdot 10^{-5} = 3,0 \cdot 10^{-5} \text{ (1/год).}$$

Проведемо обчислення надійності системи (4.1) з урахуванням інтервалу 1000 год:

$$P(1000)=e^{-(3,0 \cdot 10^{-5}) \cdot 1000}=e^{-0,03}=0,9704.$$

Так, вірогідність безвідмовної роботи системи протягом 1000 год може досягати 97,04 %.

Підкреслимо, що це безсумнівно високий показник надійності для систем управління середнього рівня критичності.

Отже, за допомогою отриманих результатів утворимо графік залежності надійності від часу, приклад якого продемонстровано на рисунку 4.3.

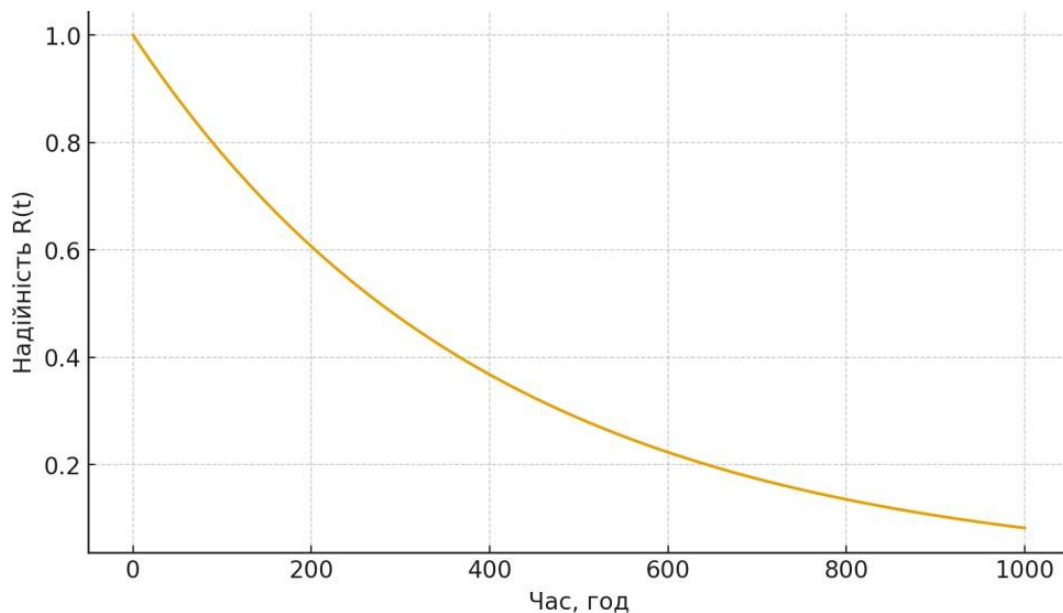


Рисунок 4.2 – Графік залежності надійності від часу

#### 4.2.2 Оцінювання стійкості системи до впливів

Насамперед необхідно оцінити стійкість системи щодо:

а) пікових навантажень. Так, навантажувальні випробування підтвердили, що:

- 1) за умови 50 користувачів разом середній час відповіді становитиме 0,8 с;
- 2) система зберігає стабільність до 90 % завантаження CPU;

3) погіршення продуктивності БД за наявності 10 000 записів відсутнє;

б) збоїв у БД. Навіть за умов аварійних ситуацій система зазвичай зберігає цілісність даних. Така властивість реалізується шляхом залучення:

- 1) механізмів rollback;
- 2) точок збереження (savepoints);
- 3) занотовування транзакцій;
- 4) автоматичного резервного копіювання.

в) затримок у мережі. Як правило, комплекс може функціонувати стабільно, якщо утворюється ймовірність щодо:

- 1) затримок до 120 – 150 мс;
- 2) втрати пакетів до 2 %, натомість у разі 5 % втрат система може

уповільнюватися, проте продовжує функціонувати й надалі;

г) нестачі ресурсів. Таким чином, реалізується алгоритмічна стійкість, адже система спроможна діяти коректно :

- 1) надсилати сповіщення щодо дефіциту ресурсів;
- 2) пропонувати коригування плану;
- 3) переналаштовувати графік.

#### 4.2.3 Встановлення показників функціональної стійкості

Так, для оцінювання стійкості застосовують показник [32, 34]:

$$K_{ст} = T_{ф}/T_{ну}, \quad (4.3)$$

де  $T_{ф}$  – час, протягом якого система продовжує функціонувати, незважаючи на впливи;

$T_{ну}$  – тривалість дії несприятливих умов.

Згідно з результатами проведених експериментів встановили, що:

- система починає втрачати продуктивність після 240 с пікового навантаження;

- тривалість впливу може досягати 300 с.

З огляду на викладене вище, доцільно вирахувати  $K_{ст}$ :

$$K_{ст} = 240/300 = 0,8.$$

Так, коефіцієнт стійкості системи набуває значення 0,8, а для інформаційних систем встановлений показник відповідає рівню «висока стійкість».

#### 4.2.4 Оцінювання живучості та відмовостійкості

Як правило, система забезпечує високий рівень живучості послуговуючись [32, 35]:

- спроможністю до швидкого відновлення (MTTR $\approx$ 3 хв);
- автоматичним перезапуском сервісів;
- резервуванням даних;
- дублюванням критичних функцій;
- класифікуванню ролей.

Утім вірогідність повної відмови системи можемо вирахувати шляхом [32, 36]:

$$P_f = 1 - P(t), \tag{4.4}$$

$$P_f = 1 - 0,9704 = 0,0296 \approx 3 \text{ \%}.$$

Таким чином, вдалось довести, що за 1000 год роботи ймовірність, що збій буде критичним, становитиме лише 3%, отже, це є прийнятним для підприємницьких систем.

#### 4.3 Заходи з безпечних умов праці у навчальній лабораторії

Розміри навчальної лабораторії, в якій були проведені дослідження відповідно до теми кваліфікаційної роботи, складає  $8,3 \times 6 \times 3,1$  м.

Робоче місце устатковано столом і персональним комп'ютером (ПК), потужністю у 600 Вт, а також монітором LCD – 24", оновлення екрану – 60 Гц.

У приміщенні працює 2 людини. Площа приміщення  $49,8 \text{ м}^2$ , а об'єм –  $154,38 \text{ м}^4$ . Згідно з НПАОП 0.00-1.28-10 площа на одне робоче місце має становити не менше  $6 \text{ м}^2$ , а об'єм –  $20 \text{ м}^4$ .

Згідно з НПАОП 40.1-1.21-98 приміщення можна віднести до категорії без підвищеної небезпеки (сухе не запилене приміщення з ізолювальною підлогою) [36].

У приміщенні використовується система живлячих провідників, трифазна, чотирипровідна з глухо заземленою нейтраллю напругою до 1000 В, отже, згідно з НПАОП 40.1-1.32-01 встановлено систему заземлення TN-C-S типу. Окрім того, додатково застосовується повторне заземлення нульового проводу задля зниження потенціалу корпусів і напруги дотику у випадках обриву нульового проводу [36].

Робота в приміщенні проводиться сидячи, отже, не вимагає фізичного навантаження, тому належить до категорії Ia (легкі фізичні роботи, енерговитрати до 120 ккал/ч). З метою забезпечення комфортних умов для працівників, а також відповідно до ДСН 4.4.6.042-99 у приміщенні встановлені такі оптимальні показники мікроклімату [36]:

– для холодного періоду: температура повітря 22 – 24 °С; вологість повітря 40 – 60 %; швидкість руху повітря оптимальна 0,1 м/с;

– для теплого періоду року: температура повітря 23 – 25 °С; вологість повітря 40 – 60 %; швидкість руху повітря оптимальна 0,1 м/с, допустима 0,2 м/с.

У приміщенні лабораторії залучено природне та штучне освітлення відповідно до ДБН Ст. 2.5-28-2006 [36].

Природне світло проникає через бічні світлопроєми, які направлені на північ, отже, забезпечують коефіцієнт природної освітленості (КПО) не нижче 1,2 %.

Необхідна освітленість приміщення 200 – 500 лк [36].

Згідно з ДСН 4.4.6.037-99 рівень шуму в лабораторії не перевищує 50 дБА.

Категорія приміщення П – Па згідно з ПУЕ-2011. Лабораторія розташована в будівлі II ступеня вогнестійкості згідно з ДБН 1.1.7-2002. Зважаючи на пожежо- і вибухонебезпечності належить до категорії В відповідно до НАПБ Б.04.002-2007 (адже у приміщенні лабораторії знаходяться тверді горючі матеріали) [36].

У приміщенні підготовлено 3 робочі місця, устатковані ПК. У разі підвищення температури окремих вузлів ймовірні ситуації щодо оплавлення ізоляції сполучних проводів, що може спричинити замикання.

Пожежна безпека в лабораторії забезпечується відповідно до ДСТУ 12.1.004-91, системою запобігання пожежі, протипожежного захисту й організаційно-технічними заходами: встановлено 1 вогнегасник ВВК-1,4 (зокрема 1 вогнегасник на 3 ПК); приміщення лабораторії обладнано системою автоматичної пожежної сигналізації з застосуванням димових датчиків типу ДИП-1 (1 шт.) згідно з ДБН Ст. 2.5-56-2010 (1 на 86 м<sup>2</sup>); навчання персоналу

правилам пожежної безпеки, регулярні інструктажі з пожежної безпеки; розміщені інструкції та план евакуації персоналу у випадку пожежі [36].

#### 4.4 Висновок до розділу

Даний розділ було присвячено комплексному оцінюванню надійності та стійкості розробленої автоматизованої системи управління. Тож отримані результати визначили, що системі властиві:

- висока надійність – 97,04 % безвідмовної роботи на 1000 год;
- висока стійкість до зовнішніх впливів, пікових навантажень, дефіциту ресурсів, а також до затримок у мережі;
- висока живучість і відмовостійкість, завдяки резервуванню та механізмам відновлення;
- низька ймовірність критичного збою – 3%;
- стабільне опрацювання великих масивів даних;
- спроможність відновлюватися після аварій за 2 – 3 хв.

Отже, система характеризується високим рівнем технічної надійності, алгоритмічної стійкості, тож цілком підготовлена для функціонування на виробництві.

Окрім того, були проведені заходи щодо безпечних умов праці у навчальній лабораторії, де проводились дослідження за темою кваліфікаційної роботи.

## ВИСНОВКИ

Кваліфікаційну роботу було присвячено проведенню комплексного дослідження, що зорієнтовано на розробленні, програмній реалізації, а також тестуванні автоматизованої системи управління програмою запуску та випуску продукції на виробничому підприємстві.

Були опрацьовані як сучасні підходи до управління виробничими процесами, так і методики планування, обліку та контролю виконання програм. Окрім того, було виявлено нагальні проблеми підприємств.

Проведений аналіз підтвердив, що розроблення єдиної автоматизованої системи, здатної забезпечити оперативність, точність і повноту керування програмами запуску та випуску продукції наразі є актуальним.

Побудовано концепцію й архітектуру автоматизованої системи. Утворено концептуальну структуру системи. До того ж, проєкт системи вирізняється трирівневою архітектурою, зокрема, рівнями представлення та застосунків, а також рівнем даних. Підкреслимо, що зазначений підхід забезпечив модульність та масштабованість на тлі можливої інтеграції з ERP- і MES-системами підприємства.

Відтак було реалізовано БД і розроблено ПЗ. Так, задіюючи побудовану інформаційну модель, було створено реляційну БД у СУБД PostgreSQL. Окрім того, серверну частину було розроблено на Python/Django, а клієнтську – послуговуючись сучасними веб-технологіями. Втім автоматизована система підтримує резервування, контроль доступу, журналювання подій разом із опрацюванням великих масивів даних.

Також увагу було приділено питанням тестування та валідації функціоналу. Система витримала модульне й інтеграційне, функціональне та

навантажувальне, а також UX-тестування, за результатами яких було доведено відповідність розробленого комплексу технічним та функціональним вимогам.

Окрім того, було виконано оцінювання стійкості та надійності системи. У процесі роботи побудовано графік залежності надійності від часу та розглянуто основні компоненти стійкості, зокрема, йдеться про надійність і відмовостійкість, живучість та відновлення. Однак математичні моделі довели, що система відповідає нормативам функціональної стійкості, а також характеризується високим рівнем відмовостійкості, забезпечуючи водночас безперервність критично важливих процесів.

Як підсумок, можемо зазначити, що розроблений програмний комплекс є цілком придатним для впровадження на виробничому підприємстві. За його допомогою реалізується:

- покращення оперативності управління;
- автоматизація планування;
- скорочення простоїв;
- оптимізація ресурсів;
- оформлення точних аналітичних звітів;
- підвищення продуктивності персоналу.

Підкреслимо, що система легко адаптується до виробничих підприємств різних типів і галузей.

За темою досліджень кваліфікаційної роботи було написано та опубліковано тези доповіді “Automated hardware and software measuring complex”, до III Міжнародної науково-практичної конференції “Modern science: Exploring theories, innovations and practical solutions”, Одеса, Україна, 2024р. (Додаток В).

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Методичні вказівки з підготовки та захисту кваліфікаційної роботи здобувачами другого (магістерського) рівня вищої освіти спеціальності 174 Автоматизація, комп'ютерно-інтегровані технології та робототехніка, освітньо-професійних програм: «Комп'ютерно-інтегровані технологічні процеси і виробництва», «Комп'ютеризовані та робототехнічні системи» /Упоряд. І. Ш. Невлюдов, Р. В. Артюх, В. В. Безкоровайний, Н. П. Демська, В. В. Євсєєв, О. І. Филипенко, О. М. Цимбал. Харків: ХНУРЕ, 2023. – 49 с.

2. Положення про кваліфікаційну роботу здобувача вищої освіти на другому (магістерському) рівні [Електронний ресурс]: Наказ ХНУРЕ від 06 травня 2021 р. № 143. – Режим доступу: [https://nure.ua/wpcontent/uploads/Main\\_Docs\\_NURE/143-vid-06.05.2021-pro-vvedennja-v-dijurishennja-vchenoi-radi-universitetu.pdf](https://nure.ua/wpcontent/uploads/Main_Docs_NURE/143-vid-06.05.2021-pro-vvedennja-v-dijurishennja-vchenoi-radi-universitetu.pdf).

3. ДСТУ 3008:2015 Інформація та документація «Звіти у сфері науки і техніки». Структура та правила оформлювання. / В. Земцева; Ю. Поліщук, канд. фіз.-мат. наук; Р. Санченко, канд. техн. наук; Л. Шрамко; А. Ямчук (науковий керівник) ДП «УкрНДНЦ» від 22 червня 2015р. № 61 з 2017-07-01.

4. Andrii Frolov, Maksym Kovalenko, Oleksandr Shcherbyna. Automated hardware and software measuring complex // III International scientific and practical conference “Modern science: Exploring theories, innovations and practical solutions” – (December 30-31, 2024. Odessa, Ukraine). European Open Science Space, 2024. pp. 14-15. Режим доступу: [https://www.eoss-conf.com/wp-content/uploads/2024/12/Odesa\\_Ukraine\\_30\\_12\\_24.pdf](https://www.eoss-conf.com/wp-content/uploads/2024/12/Odesa_Ukraine_30_12_24.pdf).

5. Невлюдов І. Ш. Основи виробництва електронних апаратів / І. Ш. Невлюдов. – Харків: Компанія СМІТ, 2005. – 592 с.
6. Невлюдов І. Ш. Технічні засоби автоматизації: Підручник / І. Ш. Невлюдов, А. О. Андрусевич, О. І. Филипенко, Н. П. Демська, С. П. Новоселов. – Кривий Ріг : Криворізький коледж НАУ, 2019. – 366 с.
7. Невлюдов І. Ш. Людино-машинний інтерфейс в технічних засобах автоматизації: Навчальний посібник / І. Ш. Невлюдов, О. І. Филипенко, Б. О. Шостак. – Харків : «ХТМТ», 2019. – 244 с.
8. Автоматизація виробничих процесів: підручник / І. В. Ельперін, О. М. Пупена, В. М. Сідлецький, С. М. Швед. – Київ: Ліра-К, 2015 – 300 с.
9. Автоматизація технологічних процесів: навчальний посібник / В. М. Синеглазов, І. Ю. Сергеев. – К.: НАУ, 2015. – 444 с.
10. Діагностика та контроль робочих процесів: навч. посібник для студентів спеціальності «Прикладна механіка» денної та дистанційної форм навчання / В. М. Доля – Харків: НТУ «ХП», 2019. – 129 с.
11. Грінченко І.В. Основи промислової автоматизації: SCADA, PLC, IoT-технології. – Львів: Новий Світ-2000, 2019. – 412 с.
12. Жалдак М. І., Касьянов В. М. Інформаційні технології управління виробничими процесами. – Київ: Ліра-К, 2018. – 260 с.
13. Літвінов С. М. Бази даних і знань: теорія та проектування інформаційних систем. – Київ: Видавництво КНТ, 2019. – 344 с.
14. ISA-95 / IEC 62264. Enterprise-Control System Integration. Overview of the ISA-95 standard. International Society of Automation, 2024.
15. Шило В. П. Проектування інформаційних систем : навч. посіб. – К.: НУХТ, 2019. – 368 с.
16. Федоренко В. Г., Пустовая Г. І. Інформаційні системи в управлінні організаціями. – К.: КНЕУ, 2021. – 352 с.

17. Роберт Мартін. Чиста архітектура. – К.: Стартап Ленд, 2021. – 384 с.
18. IEEE Standard 1471-2000. Recommended Practice for Architectural Description of Software-Intensive Systems. – IEEE Press, 2020.
19. Мартинюк О. О. Бази даних: моделювання, проєктування та застосування : навч. посібник. – Львів: ЛНУ, 2022. – 404 с.
20. PostgreSQL Documentation. Version 15. – 2023. Режим доступу: <https://www.postgresql.org/docs/>.
21. Кузьменко Д. С. Архітектура програмного забезпечення: методи та моделі. – Харків: ХНЕУ, 2020. – 280 с.
22. Bass L., Clements P., Kazman R. Software Architecture in Practice. – 4th ed. – Addison-Wesley, 2021. – 640 p.
23. Python Software Foundation. Python 3 Documentation. – 2024. – Режим доступу: <https://docs.python.org/>.
24. Django Software Foundation. Django Documentation. – 2024. Режим доступу: <https://docs.djangoproject.com/>.
25. Date C. J. Database Systems: Introduction to Database Systems. – 8th ed. – Addison Wesley, 2019. – 1024 p.
26. Лунін В. В., Шкурко Н. Г. Автоматизовані системи управління. – К.: Кондор, 2018. – 412 с.
27. Pressman R. S., Maxim B. Software Engineering: A Practitioner's Approach. – 9th ed. – McGraw-Hill, 2020. – 936 p.
28. Фігурнов В. Б. Сучасні інформаційні технології. – К.: Фенікс, 2020. – 480 с.
29. Котенко А. І., Кохан В. П. Програмування та тестування інформаційних систем. – К.: Вид-во НТУУ «КПІ», 2022. – 318 с.
30. ISO/IEC/IEEE 29119 Software Testing Standards. – International Organization for Standardization, 2021.

31. McDermid J. Software Verification and Validation. – Springer, 2021. – 342 p.
32. Левитський В. І. Надійність і стійкість автоматизованих систем управління. – К.: Кондор, 2018. – 268 с.
33. IEC 61508. Functional safety of electrical/electronic/programmable electronic safety-related systems. International Electrotechnical Commission, 2010.
34. Герасимов Ю. М. Теорія надійності та діагностика технічних систем. – Київ: НАУ, 2020. – 284 с.
35. Хоменко В. Д., Мірошник А. В. Системи керування: надійність, відмовостійкість, живучість. – Харків: «Старий Лев», 2017. – 344 с.
36. Комплекс навчально-методичного забезпечення навчальної дисципліни «Організація керування умовами праці» підготовки освітнього рівня бакалавр усіх спеціальностей та усіх напрямів університету [Електронний ресурс] / ХНУРЕ; розроб.: Т. Є. Стищенко, Г. В. Пронюк, Н. М. Сердюк. – Харків, 2017. – 108 с.