

Вінницький національний технічний університет

(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматизації

(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних наук

(повна назва кафедри (предметної, циклової комісії))

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

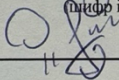
на тему:

“ Інформаційна технологія

планування виробництва ”

Виконав: студент II курсу, групи 1КН-23М
спеціальності 122 – Комп'ютерні науки

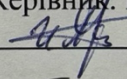
(цифр і назва напрямку підготовки, спеціальності)



Черниш Р.О.

(прізвище та ініціали)

Керівник: к.т.н., доцент каф. КН

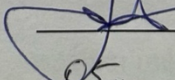


Арсенюк І.Р.

(прізвище та ініціали)

« 05 » 12 2024 р.

Опонент: к.т.н., доцент каф. АІТ



Маслій Р.В.

(прізвище та ініціали)

« 05 » 12 2024 р.

Допущено до захисту

Завідувач кафедри КН

д.т.н., проф. Яровий А.А.

« 06 » 12 2024 р.

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра Комп'ютерних наук
Рівень вищої освіти другий (магістерський)
Галузь знань – 12 «Інформаційні технології»
Спеціальність – 122 «Комп'ютерні науки»
Освітньо-професійна програма – «Системи штучного інтелекту»

ЗАТВЕРДЖУЮ

Завідувач кафедри КН

д.т.н., проф. Яровий А.А.

“11” 10 2024 року

**ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**

Чернишу Роману Олександровичу

1. Тема роботи: Інформаційна технологія планування виробництва
Керівник роботи: Арсенюк Ігор Ростиславович, к.т.н., доц. каф. КН
затверджені наказом вищого навчального закладу “11” 09 2024 року № 310

2. Строк подання студентом роботи 11.11.2024 рік

3. Вихідні дані до роботи:

Кількість працівників на виробництві: від 5 до 10 000; кількість задач виробництва: до 10 000; кількість створених виробничих процесів: до 18000; кількість запланованих виробничих процесів: до 1 500; додаток: веб-додаток, кросплатформний; мова програмування: об'єктно-орієнтована.

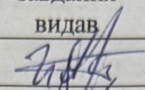
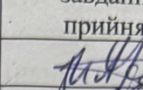
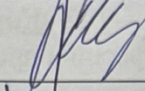
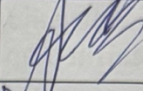
4. Зміст текстової частини:

Вступ, обґрунтування доцільності розробки інформаційної технології планування виробництва, розробка інформаційної технології, програмна реалізація інформаційної технології, економічна частина, висновки, перелік використаних джерел, додатки.

5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень):

Архітектура інформаційної технології планування виробництва; UML-діаграма класів; UML-діаграма розгортання; алгоритм табу-пошуку; схема роботи генетичного алгоритму; результати прогнозування LSTM і ARIMA; інтерфейс веб-додатку; таблиці продуктивності системи у порівнянні з аналогами.

6. Консультанти розділів роботи

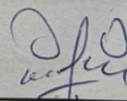
Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-3	Арсенюк І.Р., к.т.н., доц. каф. КН		
4	Лесько О.Й., к.е.н., проф. каф. ЕПВМ		

7. Дата видачі завдання 02.09.2024

КАЛЕНДАРНИЙ ПЛАН

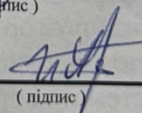
№ з/п	Назва та зміст етапу	Термін виконання		Примітка
		початок	закінчення	
1	Аналіз предметної області планування виробництва	02.09.24	14.09.24	
2	Розробка інформаційної технології планування виробництва	15.09.24	25.09.24	
3	Програмна реалізація інформаційної технології планування виробництва	26.09.24	02.10.24	
4	Тестування та аналіз результатів роботи	03.10.24	18.10.24	
5	Економічна частина	19.10.24	28.10.24	
6	Оформлення матеріалів до захисту МКР	29.10.24	11.11.24	

Студент


(підпис)

Черниш Р.О.
(прізвище та ініціали)

Керівник роботи


(підпис)

Арсенюк І.Р.
(прізвище та ініціали)

АНОТАЦІЯ

УДК 004.6

Черниш Р.О. Інформаційна технологія планування виробництва. Магістерська кваліфікаційна робота зі спеціальності 122 «Комп'ютерні науки», освітня програма «Системи штучного інтелекту». Вінниця: ВНТУ, 2024. 120 с.

На укр. мові. Бібліогр.: 26 назв; рис.: 39; табл. 12.

Дана магістерська кваліфікаційна робота присвячена розробці інформаційної технології для планування виробничих процесів із використанням штучного інтелекту та сучасних алгоритмів оптимізації. Здійснено детальний аналіз предметної області, в тому числі дослідження існуючих рішень у сфері виробничого планування. Обґрунтовано необхідність створення нової технології, яка перевершує існуючі системи аналогів за швидкістю, точністю та адаптивністю.

У роботі розроблено структурну модель інформаційної технології, яка забезпечує ефективну організацію планування виробничих процесів. Впроваджено гібридний підхід до прогнозування. Для оптимізації розкладів використано комбінацію генетичних алгоритмів, табу-пошуку та механізмів адаптивного навчання.

Результати тестування підтвердили, що запропонована система значно перевершує комерційні аналоги за ключовими показниками. Розроблена інформаційна технологія дозволяє забезпечити підвищення швидкодії планування виробничих задач. Також наведено економічне обґрунтування доцільності впровадження розроблюваного програмного забезпечення, та здійснено обрахунок терміну його окупності.

Ключові слова: виробниче планування, штучний інтелект, оптимізація, ARIMA, LSTM, генетичні алгоритми, інформаційна технологія.

ABSTRACT

Chernysh R. O. Information technology for production planning. Master's Qualification Thesis in Specialty 122 "Computer Science", Educational Program "Artificial Intelligence Systems". Vinnytsia: VNTU, 2024. 120p.

In Ukrainian. Bibliography: 26 titles; Figures: 39; Tables: 12.

This master's thesis is devoted to the development of information technology for production planning using artificial intelligence and modern optimization algorithms. A detailed analysis of the subject area was conducted, including a study of existing solutions in the field of production planning. The necessity of creating a new technology that surpasses existing systems in speed, accuracy, and adaptability is substantiated.

The paper develops a structural model of information technology that ensures effective organization of production process planning. A hybrid forecasting approach was implemented. For schedule optimization, a combination of genetic algorithms, Tabu search, and adaptive learning mechanisms was used.

The results of testing confirmed that the proposed system significantly outperforms commercial counterparts in key indicators. The developed information technology ensures increased speed in planning production tasks. Economic justification for the implementation of the developed software was also provided, along with a calculation of its payback period.

Keywords: production planning, artificial intelligence, optimization, ARIMA, LSTM, genetic algorithms, information technology.

ЗМІСТ

ВСТУП.....	8
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ПЛАНУВАННЯ ВИРОБНИЦТВА.....	11
1.1 Аналіз сучасних технологій планування виробництва.....	11
1.2 Огляд існуючих систем планування виробництва з використанням штучного інтелекту.....	13
1.3 Постановка задачі.....	18
1.4 Висновок до розділу 1	18
2 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПЛАНУВАННЯ ВИРОБНИЦТВА.....	20
2.1 Обґрунтування методу розв’язання задачі оптимізації планування виробництва	20
2.2 Розробка алгоритмів роботи програми.....	26
2.3 Розробка математичної моделі планування виробництва	33
2.4 Обґрунтування вибору інтерфейсу	34
2.5 Опис інтерфейсу	38
2.6 Висновок до розділу 2.....	47
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПЛАНУВАННЯ ВИРОБНИЦТВА.....	49
3.1 Загальний огляд набору даних	49
3.2 Обґрунтування вибору програмних засобів для розробки.....	54
3.3 Розробка моделі планування виробництва	62
3.4 Тестування інформаційної технології планування виробництва.....	64
3.5 Висновок до розділу 3.....	67
4 ЕКОНОМІЧНА ЧАСТИНА	70
4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки	70
4.2 Розрахунок витрат на здійснення науково-дослідної роботи	73
4.3 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором	81
4.4 Висновок до розділу 4.....	85
ВИСНОВКИ	87

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	89
Додаток А (обов'язковий).....	92
Додаток Б (обов'язковий) Лістинг програми.....	93
Додаток В (обов'язковий) Ілюстративна частина	100
Додаток Г (довідниковий) Інструкція користувача	111

ВСТУП

Актуальність досліджень. Інтеграція штучного інтелекту (ШІ) в програмне забезпечення для планування виробництва є надзвичайно актуальною з кількох причин, що відображають сучасні тенденції та виклики у промисловості та бізнесі. У сучасному світі підприємства постійно прагнуть до підвищення ефективності управління своїми виробничими процесами. Це досягається шляхом автоматизації рутинних завдань, що дозволяє зменшити час і ресурси, необхідні для їх виконання. Штучний інтелект здатен виконувати ці завдання з високою точністю та швидкістю, що звільняє людські ресурси для вирішення більш складних та творчих завдань. Така автоматизація дозволяє не лише знизити витрати, але й значно підвищити продуктивність виробництва[1].

Оптимізація виробничих процесів є ще однією важливою перевагою використання ШІ. Сучасні алгоритми ШІ можуть аналізувати великі обсяги даних у реальному часі, що дає можливість пропонувати найкращі рішення для планування виробництва. Це включає оптимізацію розкладів, управління запасами, розподіл ресурсів та виявлення вузьких місць у виробничому процесі. Таким чином, підприємства можуть досягати високого рівня ефективності, зменшувати втрати та підвищувати якість продукції.

Прогнозування попиту та адаптація до змін ринку також є важливими аспектами, що роблять інтеграцію ШІ у виробництво актуальною. Завдяки алгоритмам машинного навчання, ШІ може аналізувати минулі дані та тренди, що дозволяє точніше прогнозувати попит на продукцію. Це дає можливість підприємствам гнучко реагувати на зміни ринкових умов, зменшувати ризики та оптимізувати виробничі плани відповідно до поточних потреб.

Штучний інтелект також допомагає у виявленні можливих проблем на ранніх стадіях виробничого процесу. Така проактивність дозволяє зменшити час простоїв, а також підвищити загальну надійність та якість виробництва.

З огляду на все вищезазначене, інтеграція штучного інтелекту в програмне забезпечення для планування виробництва є надзвичайно важливою для сучасних

підприємств, які прагнуть досягти високого рівня ефективності, гнучкості та якості у своїй роботі. Це дозволяє підприємствам не лише зберігати конкурентоспроможність на ринку, але й забезпечувати стійкий розвиток та інноваційний підхід до управління виробництвом.

Зв'язок роботи з науковими програмами, планами, темами. Магістерська кваліфікаційна робота виконана відповідно до напрямку наукових досліджень кафедри комп'ютерних наук Вінницького національного технічного університету 122 К1 «Розробка прикладних інтелектуальних інформаційних технологій та систем» та плану наукової та навчально–методичної роботи кафедри.

Мета дослідження. Підвищення швидкодії процесів планування виробництва.

Об'єктом дослідження Об'єктом дослідження є процес планування виробництва.

Предметом дослідження Програмні засоби процесу планування виробництва.

Методи дослідження. Методи дослідження. Для досягнення поставленої мети було використано методи класифікації та систематизації інформації, теорію моделей представлення знань, методи узагальнення, моделювання та абстрагування, методи побудови UML-діаграм, інтелектуального аналізу даних, а також методи розробки та аналізу інформаційних систем.

Завдання дослідження. Для досягнення поставленої мети необхідно вирішити наступні завдання:

- Здійснити аналіз сучасних технологій та систем планування виробництва з використанням штучного інтелекту.
- Виконати порівняльний аналіз існуючих рішень та виявити їхні переваги та недоліки.
- Розробити архітектуру програмного забезпечення, що інтегрує штучний інтелект для планування виробництва.
- Розробити та впровадити моделі і алгоритми для оптимізації

виробничих процесів.

- Виконати тестування розробленої системи на реальних даних та проаналізувати результати.

Наукова новизна одержаних результатів.

Набула подальшого розвитку технологія планування виробництва з використанням штучного інтелекту, яка відрізняється від існуючих рішень удосконаленою архітектурою системи та використанням вдосконаленого алгоритму оптимізації, що дозволяє підвищити швидкодію процесів планування.

Практичне значення одержаних результатів На основі проведених досліджень розроблено програмне забезпечення для планування виробництва з використанням штучного інтелекту, що дозволяє підвищити швидкодію управління виробничими процесами.

Достовірність теоретичних положень магістерської кваліфікаційної роботи підтверджується коректністю постановки задач, коректністю використання різноманітних методів під час доведення наукових положень, строгим виведенням аналітичних співвідношень, порівнянням результатів з відомими та збіжністю результатів математичного моделювання з результатами, що отримані під час впровадження розроблених програмних засобів.

Особистий внесок здобувача. Усі результати, наведені у магістерській кваліфікаційній роботі, отримані здобувачем самостійно. У працях, опублікованих у співавторстві, здобувачу належать такі результати: підхід щодо реалізації технології планування виробництва [1].

Апробація роботи. Результати досліджень було апробовано на Міжнародна наукова інтернет-конференція на тему: "Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення" [1].

Публікації. За результатами магістерської кваліфікаційної роботи опубліковано тези доповідей на науково-технічній конференції [1].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ПЛАНУВАННЯ ВИРОБНИЦТВА

1.1 Аналіз сучасних технологій планування виробництва

Планування виробництва — це ключовий елемент управління на сучасних підприємствах, який визначає ефективність використання ресурсів, оперативність виконання замовлень і загальну продуктивність виробничих процесів. Зростаюча складність виробничих систем і вимога швидкої адаптації до змін ринкового попиту роблять тему оптимізації планування актуальною для більшості галузей. У той час, як підприємства прагнуть мінімізувати витрати, скорочувати час на виробництво і підвищувати якість продукції, традиційні підходи до планування часто не можуть забезпечити достатньої гнучкості та точності.

Інтеграція штучного інтелекту у процеси планування виробництва відкриває нові можливості для вдосконалення точності прогнозів, зниження простоїв та ефективного розподілу ресурсів. Використання ШІ стає важливим фактором конкурентоспроможності, оскільки дозволяє підприємствам реагувати на швидкі зміни в умовах виробництва і мінімізувати втрати, пов'язані з непродуктивними витратами ресурсів і неузгодженістю між процесами. Завдяки можливості аналізувати великі обсяги даних і приймати обґрунтовані рішення в реальному часі, ШІ здатен значно підвищити ефективність виробничого процесу та забезпечити його стійкість і адаптивність.

Актуальність теми обумовлена не лише потребою в автоматизації і підвищенні ефективності, але й необхідністю вирішення проблемних аспектів традиційного планування, таких як накладання задач, перевитрати ресурсів, простої та залежність від людського фактора. У зв'язку з цим впровадження сучасних систем, що дозволяють оптимізувати і вдосконалювати процеси планування виробництва, є критично важливим для підприємств, які прагнуть досягти високої продуктивності та відповідати вимогам сучасного ринку.

Актуальність теми планування виробництва значно зросла через підвищені вимоги до гнучкості, точності та швидкості у виконанні виробничих завдань. Змінність сучасного ринку зумовлює потребу у швидкій адаптації до коливань

попиту, що вимагає від виробничих підприємств здатності миттєво реагувати на нові замовлення, збільшувати або зменшувати обсяги випуску продукції, управляти запасами і оптимально використовувати обладнання та робочу силу. Класичні методи планування часто не можуть забезпечити належної швидкості та ефективності через обмежену здатність швидко обробляти велику кількість змінних. Через це застосування інноваційних підходів до планування є критично важливим для збереження конкурентних позицій підприємств.

Інтеграція штучного інтелекту (ШІ) у системи планування виробництва стала одним із найбільш перспективних рішень для вирішення викликів, з якими стикаються сучасні підприємства. Завдяки здатності ШІ обробляти великі обсяги даних у реальному часі, аналізувати складні залежності між процесами і ресурсами та передбачати можливі збої, з'являється можливість створювати точніші та оперативніші виробничі плани. Це дозволяє підприємствам скорочувати час простоїв, мінімізувати витрати та підвищувати якість продукції. Крім того, впровадження ШІ допомагає знижувати залежність від людського фактора, зменшуючи кількість помилок у процесах планування та забезпечуючи сталість виробничих показників навіть в умовах мінливих ринкових вимог.

Оскільки конкуренція на ринку змушує підприємства шукати нові шляхи для підвищення ефективності виробництва, тема оптимізації планування за допомогою ШІ є надзвичайно актуальною. Виробничі компанії, які інтегрують інтелектуальні системи планування, мають змогу досягти значної переваги, що виражається у швидшому прийнятті рішень, зниженні витрат та підвищенні гнучкості у відповідь на попит. Таке планування забезпечує безперервність процесів, оптимальне використання ресурсів та вищу продуктивність. Таким чином, тема планування виробництва із застосуванням штучного інтелекту є актуальною не лише з точки зору підвищення ефективності окремих процесів, але й з позиції забезпечення стабільного розвитку підприємств у довгостроковій перспективі.

1.2 Огляд існуючих систем планування виробництва з використанням штучного інтелекту

Планування виробництва є ключовим аспектом управління виробничими процесами на підприємстві. Воно включає визначення обсягів виробництва, розподіл ресурсів, управління запасами та контроль за виконанням виробничих завдань. Сучасні технології, що використовуються для планування виробництва, включають різноманітні інформаційні системи, які автоматизують ці процеси та підвищують їх ефективність. Інтеграція штучного інтелекту (ШІ) в такі системи дозволяє значно підвищити точність і ефективність планування, що є важливим для зниження витрат і підвищення якості продукції.

Штучний інтелект здатен аналізувати великі обсяги даних у реальному часі, що дозволяє знаходити оптимальні рішення для планування виробництва. Це включає прогнозування попиту, оптимізацію розкладів, управління запасами та виявлення можливих проблем на ранніх стадіях.

1.2.1 SAP S/4HANA with Integrated Business Planning (IBP)

SAP S/4HANA є одним із провідних рішень на ринку ERP-систем, а його модуль Integrated Business Planning (IBP) спеціально розроблений для інтеграції процесів планування виробництва з елементами штучного інтелекту. IBP використовує аналітичні та прогнозні інструменти для автоматизації планування, підтримки управління запасами, ресурсами та оптимізації розкладу виробничих задач. Система дозволяє синхронізувати дані з різних джерел, забезпечуючи цілісну картину виробничих процесів і можливість приймати обґрунтовані рішення в реальному часі.

SAP S/4HANA з IBP також забезпечує функцію "що-якщо" аналізу, що дозволяє створювати симуляції для різних сценаріїв виробництва, тим самим допомагаючи компаніям краще адаптуватися до змін у попиті чи наявності ресурсів[2]. Проте для повноцінного використання всіх можливостей SAP S/4HANA підприємству потрібно мати досвідчену команду з управління

системою та значні ресурси на підтримку і налаштування. Це робить SAP S/4HANA підходящим рішенням для великих підприємств, але, можливо, надто комплексним і дорогим для малого та середнього бізнесу.



Рисунок 1.1 – логотип SAP S/4 HANA

1.2.2 Oracle Fusion Cloud Manufacturing

Oracle Fusion Cloud Manufacturing — ще одне потужне рішення, яке використовує штучний інтелект для автоматизації планування і оптимізації виробничих процесів. Завдяки інтеграції з іншими модулями Oracle Cloud, ця система забезпечує ефективне управління всіма аспектами виробництва, від розподілу ресурсів до управління запасами та аналізу продуктивності. Oracle Fusion використовує AI для прогнозування, виявлення аномалій у виробничих процесах і управління ланцюгом постачання, що дозволяє знизити ризики і покращити контроль над ресурсами.

Особливістю Oracle Fusion Cloud Manufacturing є здатність забезпечувати масштабованість і підтримку складних виробничих систем, що робить цю ERP-систему особливо привабливою для великих корпорацій. Водночас, як і SAP, Oracle Fusion потребує суттєвих інвестицій на впровадження та технічне обслуговування[4]. Основними недоліками є складність налаштування та залежність від корпоративного середовища Oracle, що може обмежувати його інтеграцію з іншими системами.



Рисунок 1.2 – Логотип Oracle Fusion Cloud

1.2.3 Microsoft Dynamics 365 Supply Chain Management

Microsoft Dynamics 365 Supply Chain Management (SCM) пропонує комплексне рішення для управління ланцюгом постачання і планування виробництва, зосереджене на використанні штучного інтелекту для поліпшення продуктивності та оптимізації процесів. Dynamics 365 SCM використовує аналітичні інструменти та AI для прогнозування попиту, управління запасами і планування змін у режимі реального часу, що дозволяє швидко адаптуватися до змінних умов виробництва.

Однією з переваг Microsoft Dynamics є інтеграція з іншими продуктами Microsoft, такими як Power BI та Azure, що полегшує аналіз даних і роботу з великими масивами інформації. Система пропонує інструменти для автоматизації прийняття рішень, що допомагає знизити витрати та оптимізувати розподіл ресурсів. Водночас Microsoft Dynamics 365 SCM потребує значних зусиль для початкового налаштування та регулярної підтримки, особливо при інтеграції зі сторонніми системами[5]. Ця ERP-система добре підходить для середніх і великих компаній, які вже використовують продукти Microsoft у своєму бізнес-середовищі.



Рисунок 1.3 – Логотип Microsoft Dynamics 365

1.2.4 Infor CloudSuite Industrial (SyteLine)

Infor CloudSuite Industrial, відома як SyteLine, — це ERP-система, орієнтована на виробничі підприємства середнього і великого розміру. Ця система використовує AI і аналітику для автоматизації управління ресурсами, прогнозування попиту, управління запасами і створення оптимальних виробничих розкладів. Infor CloudSuite інтегрує штучний інтелект Coleman AI, який підтримує аналіз у реальному часі та автоматизацію процесів, допомагаючи зменшити кількість людських помилок і підвищити ефективність.

Infor CloudSuite Industrial підтримує інструменти для персоналізації інтерфейсу та налаштування параметрів виробництва, що дозволяє користувачам адаптувати систему під специфіку свого підприємства. Вона також має високу гнучкість і легко інтегрується з іншими системами[6]. Основним недоліком є необхідність у спеціальній підготовці для роботи з Coleman AI, а також витрати на підтримку платформи. Проте Infor CloudSuite є чудовим вибором для підприємств, які потребують надійної та гнучкої системи для управління складними виробничими процесами.



Рисунок 1.4 – логотип Infor CloudSuite Industrial

Усі розглянуті ERP-системи — SAP S/4HANA, Oracle Fusion Cloud Manufacturing, Microsoft Dynamics 365 SCM та Infor CloudSuite Industrial — є передовими рішеннями на ринку, що забезпечують високу функціональність, автоматизацію та ефективність управління виробничими процесами. Вони надають потужні інструменти для управління запасами, планування ресурсів і прогнозування попиту, використовуючи штучний інтелект для покращення точності й швидкості прийняття рішень. Кожна з цих систем є високонадійним і багатофункціональним продуктом, здатним покращити роботу як середніх, так і великих підприємств.

Однак основною слабкістю цих систем є їхня універсальність. Оскільки вони розроблені для використання у широкому спектрі галузей і бізнес-моделей, це часто обмежує можливості їх гнучкої адаптації під специфічні потреби конкретного виробництва. Універсальність таких ERP-систем ускладнює їх оптимізацію для окремих типів виробничих процесів, що може призводити до зниження ефективності розподілу ресурсів і збільшення часу планування. Зокрема, через необхідність підтримувати різноманітні типи даних та забезпечувати інтеграцію з різними системами, такі платформи можуть мати обмеження у швидкості обробки специфічних завдань і здатності адаптуватися до унікальних умов виробництва.

Таким чином, незважаючи на свою функціональність і передові можливості,

ці ERP-системи поступаються у швидкості та точності оптимізації розкладу через їхню універсальність. У той час як вони ідеально підходять для загальних потреб великих і середніх підприємств, для специфічних виробничих середовищ, що потребують максимальної адаптації, індивідуалізовані системи з гібридними алгоритмами та більш вузькоспеціалізованим підходом можуть забезпечити значно вищу ефективність та продуктивність.

1.3 Постановка задачі

У програмному забезпеченні для планування виробництва з використанням штучного інтелекту повинні бути передбачені можливості вибору методів оптимізації виробничих процесів, налаштування параметрів планування, а також інтеграції з існуючими інформаційними системами підприємства. Буде розроблено систему, яка швидко та ефективно працюватиме у сучасних веб-браузерах та забезпечуватиме високу точність і гнучкість планування виробництва. Така система може стати корисним інструментом як у повсякденній роботі підприємств, так і у стратегічному плануванні виробничих процесів.

Результатом стане додаток з вищою швидкістю планування при великому навантаженні із збереженням гнучкості та адаптивності під динамічні умови.

1.4 Висновок до розділу 1

Проаналізовано сучасні технології та методи планування виробництва, доведено актуальність використання штучного інтелекту для підвищення ефективності виробничих процесів. Розглянуто існуючі ERP-системи, що використовують алгоритми штучного інтелекту для планування, такі як SAP S/4HANA, Oracle Fusion Cloud Manufacturing, Microsoft Dynamics 365 SCM та Infor CloudSuite Industrial. Виявлено переваги та недоліки кожної з цих систем, зокрема те, що їх універсальність може знижувати ефективність оптимізації через недостатню спеціалізацію під конкретні виробничі завдання та недостатня

адаптивність до специфічних умов виробництва.

Обґрунтовано доцільність створення інформаційної технології для планування виробництва, яка спеціалізується на оптимізації виробничих процесів із врахуванням конкретних вимог підприємства та забезпечує точніше і швидше планування завдяки гнучкій адаптації до змінних умов і спеціалізованим алгоритмам.

Здійснено детальну постановку задачі дослідження, ключове завдання якої полягає у підвищенні швидкодії функціонування програмного забезпечення планування виробництва.

2 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПЛАНУВАННЯ ВИРОБНИЦТВА

2.1 Обґрунтування методу розв'язання задачі оптимізації планування виробництва

У сучасних умовах промислове виробництво стикається з численними викликами, пов'язаними зі складністю процесів, змінністю попиту та необхідністю оптимального використання ресурсів. Традиційні методи планування, які часто використовуються у виробництві, нерідко не можуть забезпечити належної точності і швидкості в умовах великої кількості змінних та обмежень. Наприклад, класичні методи лінійного планування можуть бути недостатньо гнучкими для врахування постійно змінюваних умов, таких як доступність ресурсів, змінний попит на продукцію та нерегулярність робочих змін працівників.

Для розв'язання цих проблем розробляється новий метод оптимізації планування виробництва, який базується на застосуванні штучного інтелекту. Використання ШІ дозволяє створити адаптивну систему планування, здатну ефективно обробляти великі обсяги даних у реальному часі, точно прогнозувати потреби та автоматично коригувати розклади. Це досягається шляхом впровадження спеціалізованих алгоритмів ШІ, таких як методи прогнозування, алгоритми оптимізації розкладів та управління ресурсами. Такий підхід дає змогу суттєво знизити витрати, зменшити кількість конфліктів у розкладі та підвищити загальну ефективність виробничого процесу[7].

2.1.1 Обґрунтування вибору методу попередньої обробки даних

Попередня обробка даних у задачі планування виробництва дозволяє ефективно підготувати інформацію для подальшого аналізу та оптимізації, забезпечуючи її цілісність та коректність. На цьому етапі обробляються дані про історичний попит, зміни у робочих змінах працівників, доступність обладнання,

залишки ресурсів та інші показники, які можуть вплинути на планування виробництва. Основні етапи попередньої обробки включають очищення, агрегацію та нормалізацію даних, що дозволяє уникнути потенційних проблем, таких як пропуски, аномальні значення або різні формати даних.

Очищення даних дозволяє позбутися некоректних, дубльованих або неповних записів, які можуть негативно вплинути на роботу алгоритмів прогнозування і оптимізації. Наприклад, дубльовані записи можуть призводити до зайвої кількості ресурсів або завищеного попиту, що викривляє результати аналізу. Агрегація даних забезпечує об'єднання інформації з різних джерел в єдину структуру, що дозволяє отримати повну картину про стан виробничих процесів, ресурси та обладнання. Нормалізація даних, у свою чергу, приводить усі змінні до єдиного формату, що гарантує стабільну роботу алгоритмів незалежно від різниці в одиницях вимірювання.

Класифікація задач і ресурсів на основі даних є основою для побудови оптимальних розкладів, що враховують потреби у ресурсах та прогнозований попит. Класифікація ознак спрямована на розробку відповідного алгоритму прогнозування попиту і управління ресурсами, який дозволяє ідентифікувати характерні закономірності у даних. Для цієї задачі використовуються алгоритми машинного навчання, зокрема методи часових рядів (ARIMA) і рекурентні нейронні мережі (LSTM). Ці методи дозволяють прогнозувати попит на продукцію та необхідність ресурсів з високою точністю, що є важливим для своєчасного забезпечення виробничих процесів усіма необхідними матеріалами та обладнанням.

Алгоритми ARIMA забезпечують короткострокові прогнози, враховуючи сезонні зміни і тенденції на ринку, тоді як LSTM підходить для довгострокових прогнозів, завдяки здатності аналізувати послідовності даних і виявляти віддалені залежності[8, 9]. Завдяки цьому система може ефективно планувати обсяги виробництва та управляти запасами, що дозволяє уникати дефіциту або надлишку продукції і мінімізувати витрати на її зберігання.

Оптимізація розкладу виробничих процесів здійснюється за допомогою

гібридного алгоритму, що включає генетичні алгоритми, табу-пошук і адаптивне навчання.

Генетичні алгоритми застосовуються для створення початкового розкладу, де мінімізується кількість конфліктів між задачами та забезпечується оптимальне використання ресурсів. Це дозволяє отримати первісний план, що відповідає основним вимогам, таким як зменшення простоїв обладнання та рівномірне навантаження на працівників[10]. Схема функціонування генетичного алгоритму зображена на рисунку 2.1.

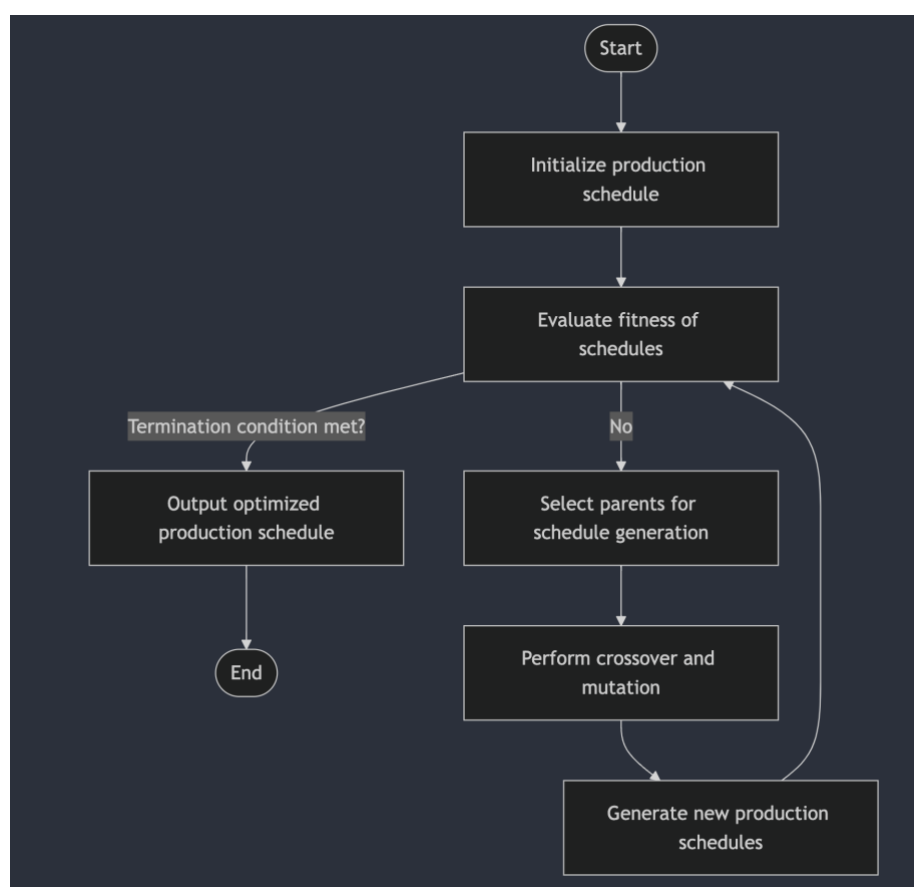


Рисунок 2.1 – Узагальнена схема функціонування генетичного алгоритму.

Табу-пошук забезпечує швидку адаптацію до змін у виробничих умовах, дозволяючи коригувати розклад без необхідності повного перерахунку. Наприклад, якщо певне обладнання тимчасово недоступне, алгоритм може швидко знайти новий варіант розподілу задач, який мінімізує затримки[11]. Схема функціонування табу-пошуку зображена на рисунку 2.2.

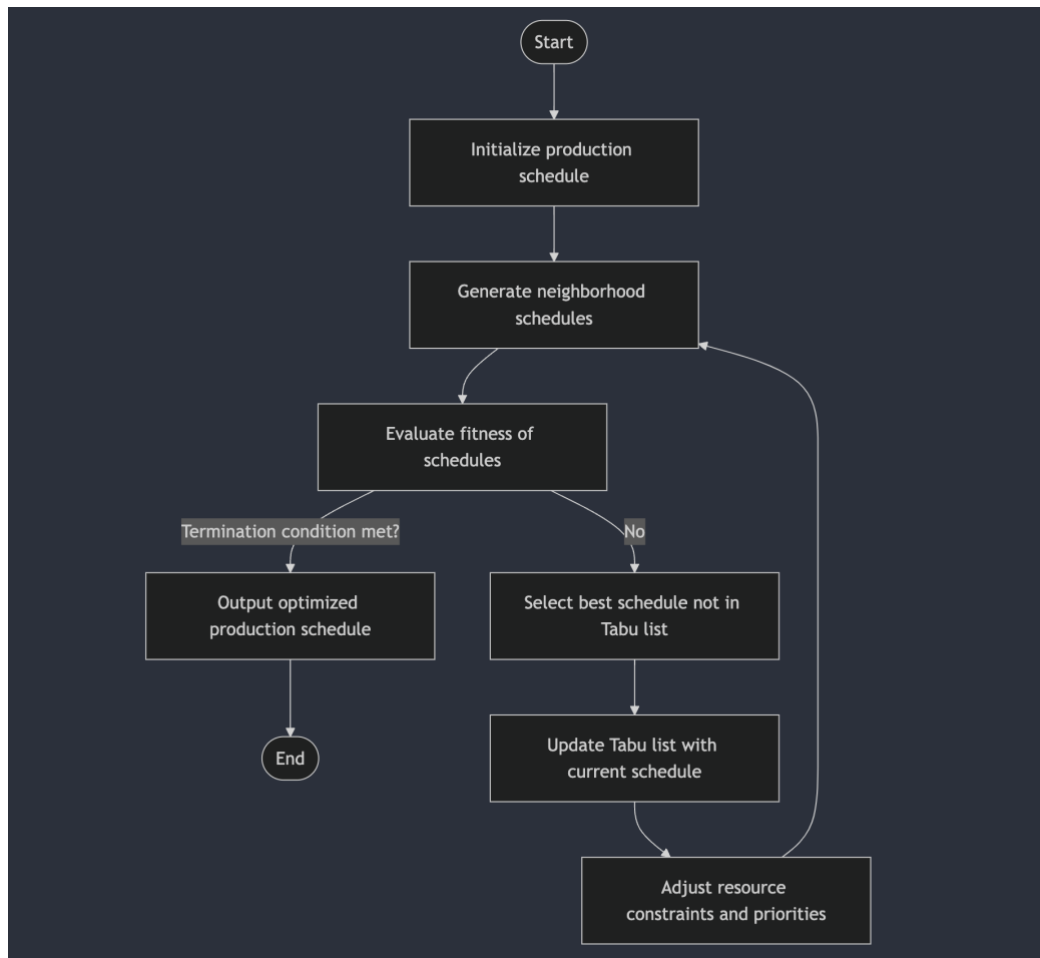


Рисунок 2.2 – Узагальнена схема функціонування табу-пошуку.

Адаптивне навчання забезпечує здатність системи до самонавчання і адаптації до змін у даних, що надходять із виробничих процесів, що дозволяє системі автоматично коригувати параметри оптимізації на основі нових даних.

Таким чином, обрані методи попередньої обробки даних, класифікації ознак та оптимізації розкладу дозволяють створити ефективну систему планування виробництва, яка здатна швидко реагувати на зміни у виробничих умовах, мінімізувати витрати на ресурси і забезпечити стабільну роботу підприємства. Впровадження цих методів дозволяє знизити залежність від людського фактора та підвищити точність планування, що є важливим для підтримання конкурентоспроможності підприємства в умовах динамічного ринку.

2.1.2 Обґрунтування вибору алгоритму прогнозування попиту та потреб у ресурсах

Наступним важливим етапом у створенні системи планування виробництва є вибір алгоритмів для прогнозування попиту та потреб у ресурсах. Цей етап має на меті обрати методи, які можуть точно передбачити кількість необхідних матеріалів, обсяг виробництва та кількість працівників на основі історичних даних та поточних тенденцій. Ефективність алгоритмів прогнозування прямо впливає на загальну продуктивність системи, оскільки дозволяє заздалегідь планувати виробничі процеси, уникати надмірних витрат на складування та зменшувати ймовірність дефіциту ресурсів.

Для задач прогнозування попиту та потреб у ресурсах обрано алгоритми, які можуть враховувати як короткострокові, так і довгострокові зміни в попиті. Вони поділяються на кілька основних типів, кожен з яких підходить для різних цілей прогнозування. Нижче детально розглянемо кожен із обраних методів прогнозування.

Методи на основі часових рядів дозволяють аналізувати історичні дані про попит, обсяги виробництва та інші показники, виявляючи тенденції та сезонні коливання. Серед методів часових рядів виділяється ARIMA (AutoRegressive Integrated Moving Average), який добре підходить для короткострокового прогнозування. Цей метод забезпечує точність у стабільних умовах, де можна передбачити повторювані цикли попиту. ARIMA дозволяє враховувати автокореляцію та сезонність, що робить його особливо корисним для підприємств з передбачуваними змінами попиту. До методів на основі часових рядів відносяться:

- ARIMA — метод, що прогнозує майбутні значення на основі історичних даних, враховуючи трендові і сезонні коливання.
- SARIMA (Seasonal ARIMA) — розширений варіант ARIMA, який краще враховує сезонність і повторювані цикли.

Методи на основі нейронних мереж дозволяють прогнозувати попит у довгостроковій перспективі, оскільки враховують складні залежності між подіями та можуть працювати з великими обсягами даних. Одним з найпоширеніших

методів є рекурентні нейронні мережі, зокрема LSTM (Long Short-Term Memory), які підходять для прогнозування попиту у випадках, коли дані мають тривалі та складні залежності. LSTM дозволяє враховувати динамічні зміни попиту і реагувати на нові тенденції в режимі реального часу. До методів нейронних мереж належать:

- LSTM — підходить для обробки послідовностей даних і врахування складних закономірностей у часі.
- GRU (Gated Recurrent Unit) — спрощений варіант LSTM, що забезпечує аналогічну точність при меншому обсязі обчислень.

Гібридні методи, що поєднують у собі характеристики методів часових рядів і нейронних мереж. Цей підхід дозволяє об'єднати точність короткострокових прогнозів ARIMA із довгостроковою здатністю прогнозування LSTM. Гібридні методи застосовуються, коли необхідно врахувати як сезонні коливання у попиті, так і довгострокові тенденції, забезпечуючи більш стабільні та точні результати.

Таблиця 2.1 – Порівняння характеристик алгоритмів прогнозування

Алгоритм	Точність	Стійкість до аномалій	Час обчислення	Оптимальне використання
ARIMA	Висока для короткострокових прогнозів	Низька	Швидкий час для невеликих обсягів даних	Короткострокові прогнози в умовах стабільного попиту
SARIMA	Висока для коротко- і середньострокових прогнозів	Середня	Середній	Прогноз із сезонністю та періодичністю змін
LSTM	Висока для довгострокових прогнозів	Висок	Високий	Довгострокові прогнози, складні залежності
GRU	Висока для довгострокових прогнозів	Середня	Середній	Довгострокові прогнози, обмежений обсяг даних
Гібрид ARIMA+LSTM	Дуже висока для складних прогнозів	Висока	Високий	Об'єднання коротко- і довгострокових прогнозів для високої точності

Методи ARIMA та LSTM були обрані для задач прогнозування попиту та потреб у ресурсах через їхню здатність забезпечувати точні прогнози у різних часових перспективах. ARIMA добре підходить для підприємств зі стабільним або передбачуваним попитом, оскільки дозволяє точно передбачити короткострокові

зміни на основі історичних даних. Використання цього методу дозволяє зменшити ризики дефіциту ресурсів або надлишку продукції, що є критично важливим для забезпечення ефективності виробничих процесів.

LSTM, у свою чергу, є оптимальним вибором для підприємств, що працюють у динамічних умовах, де попит на продукцію може різко змінюватися в залежності від ринкових умов. Завдяки здатності обробляти послідовності даних, LSTM враховує віддалені закономірності та дозволяє прогнозувати довгострокові тенденції, що допомагає планувати ресурси на кілька місяців або навіть років вперед.

Гібридні методи використовуються у випадках, коли необхідно об'єднати переваги обох підходів для отримання найбільш точних прогнозів. Наприклад, у короткостроковій перспективі використовується ARIMA для детального прогнозування щоденних обсягів виробництва, а для довгострокового планування використовується LSTM для аналізу загальних трендів на ринку. Такий підхід дозволяє оптимізувати планування, враховуючи як поточний попит, так і майбутні зміни, що є важливим для підтримки конкурентоспроможності підприємства.

Використання гібридного алгоритму для прогнозування попиту та потреб у ресурсах дозволяє забезпечити гнучке та точне планування виробництва. Завдяки поєднанню короткострокових і довгострокових методів прогнозування система планування здатна адаптуватися до змін у ринковому середовищі, зменшувати витрати та забезпечувати оптимальний розподіл ресурсів.

2.2 Розробка алгоритмів роботи програми

UML (Unified Modeling Language) діаграми – це стандартний спосіб візуалізації систем у сфері програмного забезпечення та інформаційних технологій. Вони використовуються для моделювання структури, поведінки та взаємодії різних компонентів системи. Використовуючи UML-діаграми можна візуально відобразити ключові процеси, зв'язки між об'єктами, а також алгоритми, що виконуються в системі. Це допомагає розробникам і користувачам системи

краще розуміти її функціонування[12].

UML-діаграми поділяються на кілька основних типів, кожен з яких має свою специфіку. Діаграми класів показують структуру системи, її компоненти та їх взаємозв'язки. Діаграми діяльності описують послідовність дій, які відбуваються в процесі роботи системи. Діаграми варіантів використання ілюструють, як користувачі взаємодіють із системою, а діаграми послідовності демонструють, як передаються повідомлення між різними об'єктами або модулями.

UML-діаграми широко використовуються на різних етапах розробки програмного забезпечення: від початкового проєктування до тестування та релізу продукту на ринок. Вони дозволяють створити зрозумілу та уніфіковану модель системи, яка може бути використана для полегшення комунікації між розробниками, замовниками та іншими учасниками.

Створення подібного проєкту потребує дуже глибокого розуміння всіх зв'язків між діючими елементами в системі. Саме тому для інформаційної технології планування виробництва першою діаграмою була обрана діаграма класів зображена на рисунку 2.3.

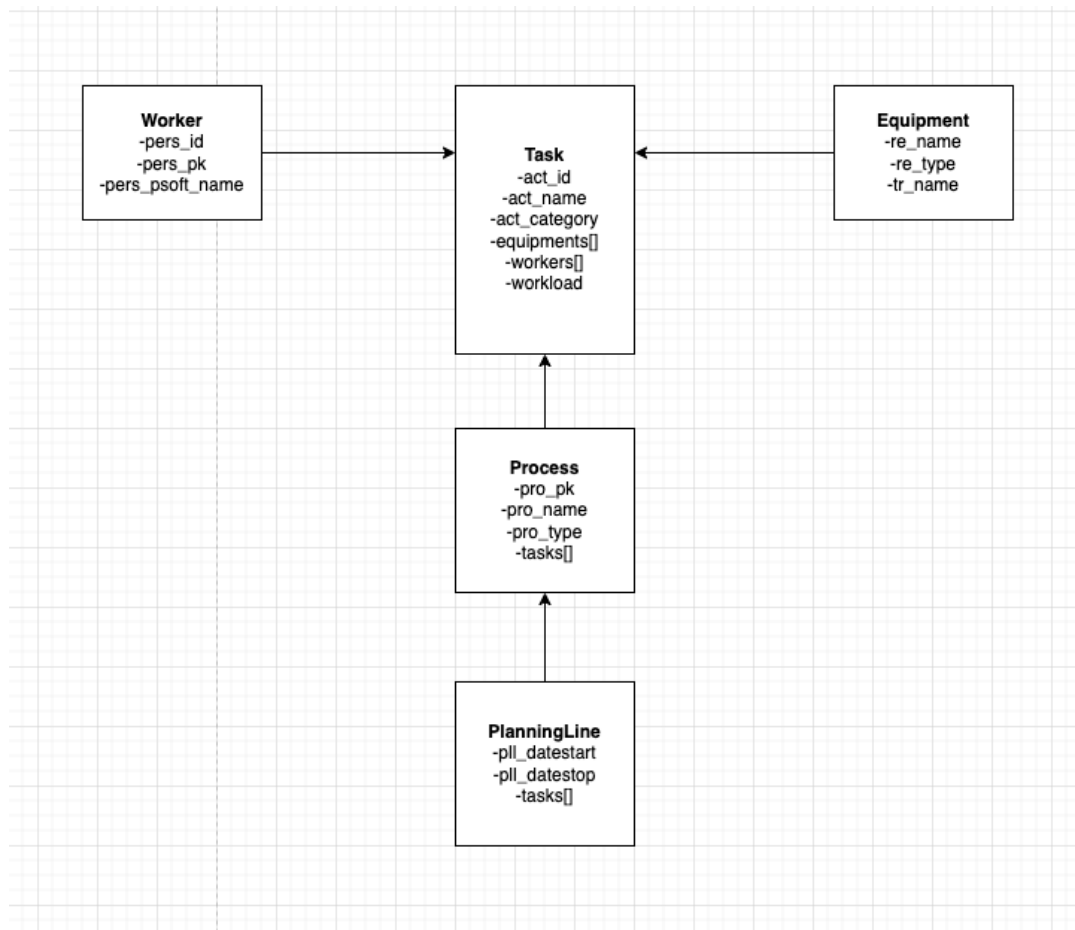


Рисунок 2.3 – Діаграма класів

Пропоную детально пройтись по кожному з елементі задля кращого розуміння.

Task (Задача)

Клас Task представляє окрему задачу, яка виконується на обладнанні.

Основні атрибути цього класу включають:

- **act_id**: унікальний ідентифікатор задачі
- **act_name**: назва задачі
- **act_category**: категорія задачі
- **equipments**: список обладнання, необхідного для виконання задачі
- **zones**: зони, в яких виконується задача
- **requiredWorkers**: робітники, необхідні для виконання задачі
- **workload**: робоче навантаження задачі

Взаємодії: Клас Task має зв'язок із класом Equipment і Worker, які

представляють обладнання та робітників, залучених до задачі.

Process (Процес)

Клас Process представляє набір задач, які виконуються в рамках одного виробничого процесу. Основні атрибути:

- pro_pk: унікальний ідентифікатор процесу
- pro_name: назва процесу
- pro_type: тип процесу
- tasks[]: список задач, що входять до цього процесу

Взаємодії: Process складається з кількох Task. Кожен процес може містити одну або більше задач, і ці задачі взаємодіють між собою в рамках процесу.

PlanningLine (Лінія планування)

Клас PlanningLine відповідає за заплановані задачі на певний час. Основні атрибути:

- pll_datestart: дата початку задачі
- pll_datestop: дата завершення задачі
- tasks[]: список запланованих задач

Взаємодії: PlanningLine містить кілька Task, які розміщуються у визначені часові інтервали. Також є зв'язок із класами Equipment та Worker, оскільки для виконання задач на лінії потрібні ресурси.

Equipment (Обладнання)

Клас Equipment представляє обладнання, яке використовується в процесі виконання задачі. Основні атрибути:

- RE_NAME: назва обладнання
- RE_TYPE: тип обладнання
- tr_name: тип обладнання/системи

Взаємодії: Equipment прив'язане до Task для виконання конкретних задач.

Worker (Робітник)

Клас Worker представляє робітників, залучених до виконання задачі. Основні атрибути:

- pers_psoft_name: ім'я робітника

- `pers_id`: ім'я робітника в системі
- `pers_pk`: унікальний ідентифікатор робітника

Таким чином ми можемо побачити що клас `Task` може бути пов'язаний з кількома класами `Equipment` та `Worker`. Клас `Process` може містити в собі довільну кількість елементів з класу `Task`. А клас `PlanningLine` може містити в собі необмежену кількість запланованих процесів або задач.

Для розробленої інформаційної технології планування виробництва було створено архітектуру, що відповідає сучасним вимогам масштабованості, продуктивності та інтеграції. Діаграма розгортання відображає ключові компоненти системи, їх зв'язки та роль у загальному процесі роботи.

Основний елемент архітектури — веб-інтерфейс, створений на основі `React`. Цей компонент працює на пристроях користувачів через веб-браузер і забезпечує інтерактивну взаємодію з системою. Інтерфейс дозволяє користувачам вводити дані про задачі та процеси, переглядати розклади та отримувати аналітичну інформацію.

Всі запити з фронтенду передаються на бекенд, реалізований за допомогою `Python` (наприклад, `Flask` або `Django`). Бекенд виконує роль основного посередника між користувачами, базою даних та модулями штучного інтелекту. Він забезпечує обробку запитів, запуск алгоритмів прогнозування (`LSTM`, `ARIMA`) і оптимізації (генетичні алгоритми, табу-пошук).

Для зберігання даних використовується база даних `Microsoft SQL`, яка забезпечує надійність, високу швидкість роботи та здатність обробляти великі обсяги даних. У ній зберігається інформація про задачі, процеси, доступність обладнання та історичні дані, необхідні для прогнозування.

Архітектура також включає модуль штучного інтелекту, що відповідає за виконання обчислювально складних завдань, таких як прогнозування попиту на ресурси та оптимізація розкладів. Цей модуль інтегрований з базою даних для отримання вхідних даних та з бекендом для передавання результатів користувачам.

Всі компоненти системи розміщуються на хмарній платформі `Microsoft`

Azure, яка забезпечує масштабованість, високу доступність та безпеку. Додатково система використовує хмарне сховище для зберігання звітів, журналів виконання задач та інших допоміжних даних.

Адміністратор має прямий доступ до бекенду для виконання управлінських завдань, таких як контроль доступності ресурсів, управління конфігурацією системи та моніторинг її роботи.

Таким чином, створена діаграма була візуалізована на рисунку 2.4 архітектура забезпечує високий рівень інтеграції між компонентами, швидкодію та адаптивність до змін у виробничому середовищі. Вона дозволяє ефективно вирішувати задачі планування виробництва та оптимізації ресурсів, залишаючись гнучкою для подальшого розширення.

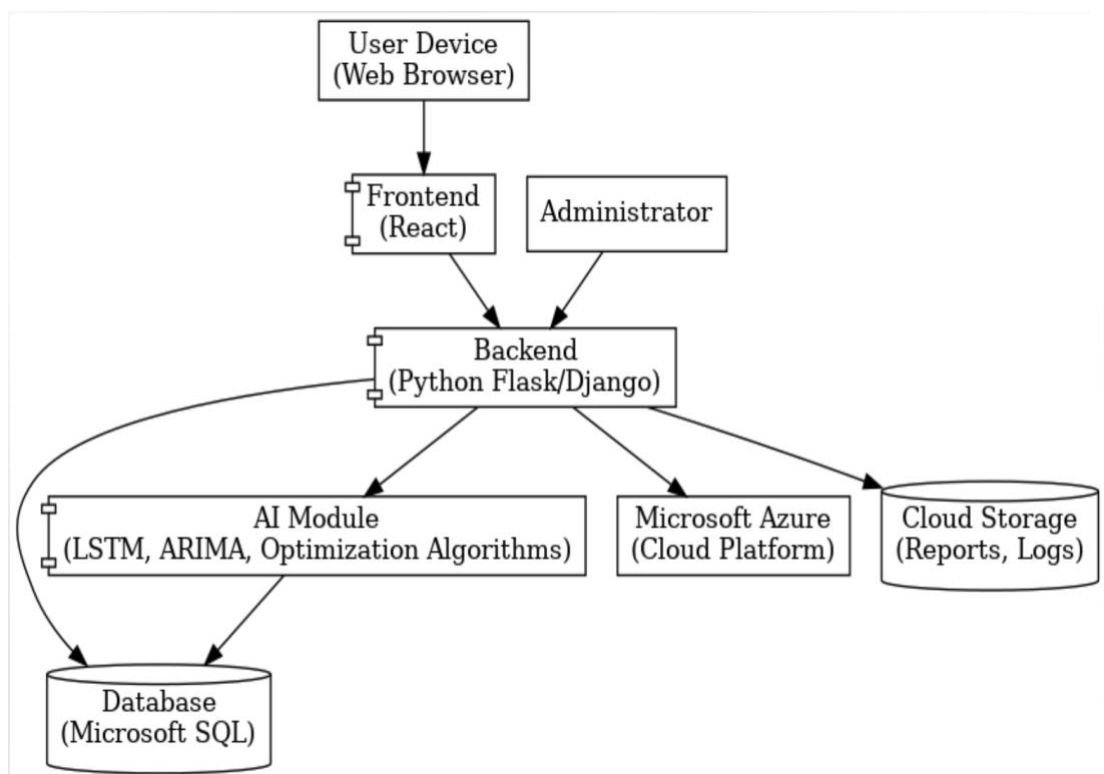


Рисунок 2.4 – UML-діаграма розгортання

На рисунку 2.5 зображена UML-діаграма діяльності (Activity Diagram). Вона використовується для моделювання робочого процесу або потоку дій в системі. У даному випадку вона ілюструє послідовність дій, що відбуваються під час планування процесу в системі управління виробництвом. Діаграма відображає

алгоритм, який система використовує для вибору дати планування процесу, враховуючи доступність обладнання, наявність інших запланованих задач і спеціальні правила.

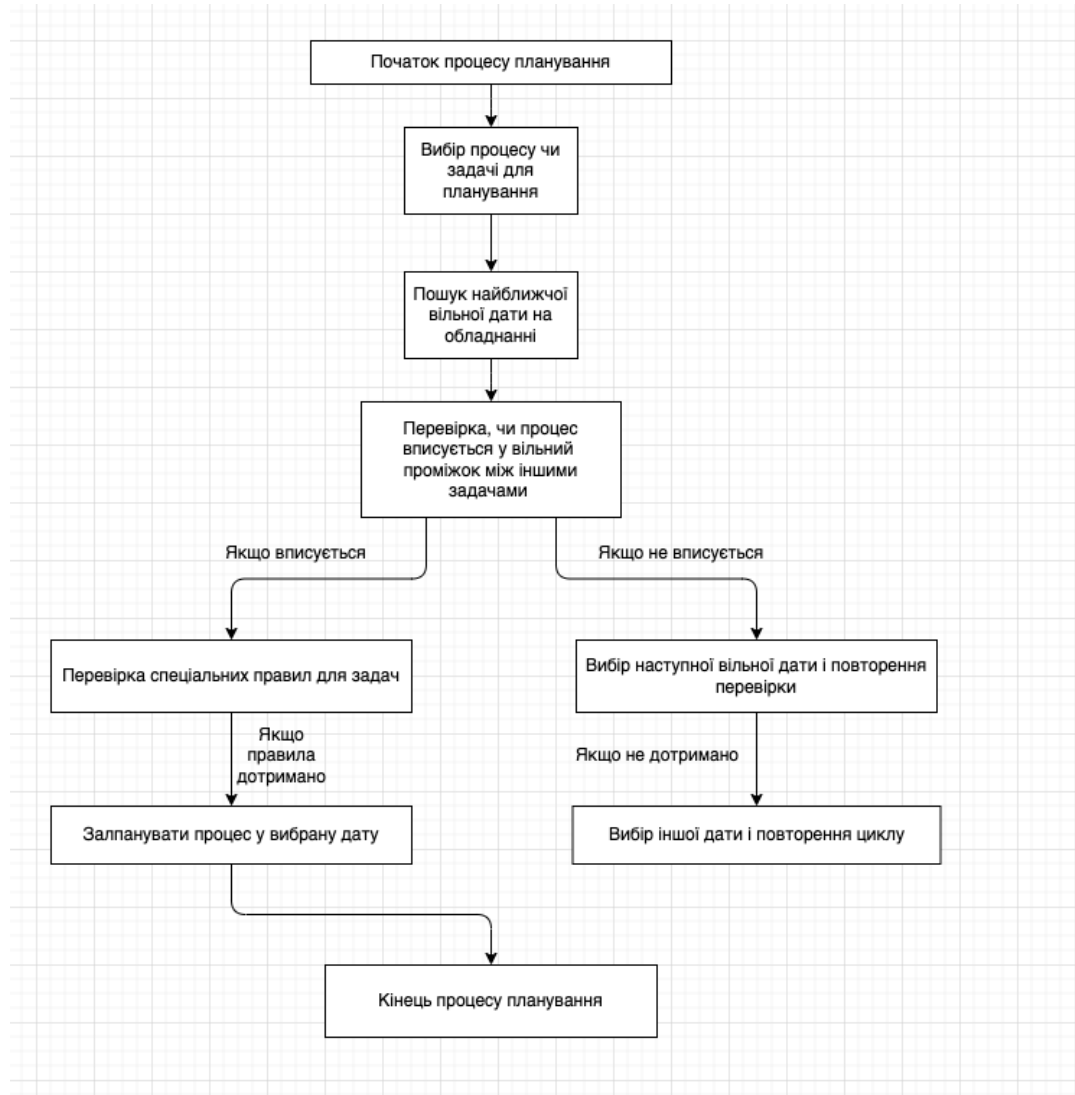


Рисунок 2.5 діаграма діяльності

Процес починається з вибору конкретного виробничого процесу, який потрібно запланувати. Далі система шукає найближчу вільну дату, коли необхідно обладнання буде доступне для виконання цього процесу. Якщо знайдено відповідний часовий проміжок між іншими запланованими задачами, система перевіряє, чи цей процес не порушує спеціальні правила. Наприклад, одне з правил може забороняти планування будь-яких задач за день до певної важливої операції на тому ж обладнанні.

Якщо процес задовольняє всі умови, він заплановується на знайдену дату. У випадку, якщо проміжок часу не підходить або порушує правила, система вибирає наступну вільну дату і повторює перевірки. Алгоритм завершується, коли процес успішно запланований, або система знаходить відповідне вікно для його виконання. Така діаграма дозволяє чітко візуалізувати логіку прийняття рішень системою під час планування задач і процесів на обладнанні.

2.3 Розробка математичної моделі планування виробництва

Мета моделі полягає в ефективному плануванні процесів із мінімізацією простоїв, уникненням колізій та помилок. Ця модель спрямована на оптимізацію використання ресурсів, часу та розподілу завдань для забезпечення безперервного і продуктивного виробництва[13].

Змінні моделі містять такі елементи, як час початку та завершення завдання, тривалість виконання, призначення працівників, а також ресурси, необхідні для виконання кожного завдання.

Час початку виконання завдання i позначається як $S_{\{i\}}$, а час завершення - як $E_{\{i\}}$. Тривалість виконання завдання позначається як $W_{\{i\}}$.

Бінарна змінна $A_{\{i, j\}}$ вказує, чи призначений працівник j на виконання завдання i ; ця змінна дорівнює 1, якщо працівник призначений, і 0, якщо ні.

Ресурс обладнання, необхідний для виконання завдання i , позначається як $R_{\{i\}}$, а доступний робочий графік працівника - як $P_{\{j\}}$.

Щоб забезпечити послідовність між завданнями, використовується бінарна змінна $G_{\{i, k\}}$, яка відображає залежність між завданнями.

Обмеження моделі включають декілька умов.

По-перше, завдання повинні відповідати своїй тривалості, де $E_{\{i\}}$ дорівнює $S_{\{i\}}$ плюс $W_{\{i\}}$. Обладнання може бути зайняте тільки одним завданням у будь-який момент часу, тому для кожної пари завдань i та k , що використовують один ресурс $R_{\{i\}}$, має виконуватися умова або $S_{\{i\}} \geq E_{\{k\}}$, або $S_{\{k\}} \geq E_{\{i\}}$.

Для графіка працівників передбачено, що кожне призначене завдання повинно бути виконане в рамках доступних робочих змін.

Важливо також враховувати необхідну кількість працівників: для кожного завдання сума змінних $A_{\{i, j\}}$ має дорівнювати або перевищувати мінімально необхідну кількість працівників.

Для завдань, які мають залежності, має виконуватися обмеження послідовності, що враховує період відпочинку між завданнями: $S_{\{k\}} \geq E_{\{i\}} + T_{\{rest\}}$, де $T_{\{rest\}}$ - час відпочинку.

Цільова функція моделі полягає в мінімізації простоїв і забезпеченні максимально ефективного використання ресурсів. Вона описується формулою мінімізації сумарного часу виконання всіх завдань та простоїв між ними. Формула має компонент α , який визначає важливість зменшення простоїв між завданнями та збільшення ефективності використання ресурсів:

$$\min \sum_{\{i\}} (E_{\{i\}} - S_{\{i\}}) + \alpha \sum_{\{i, j\}} G_{\{i, j\}} (S_{\{j\}} - E_{\{i\}}), \quad (1)$$

де α — ваговий коефіцієнт, що дозволяє регулювати баланс між зменшенням простоїв і забезпеченням послідовного виконання завдань.

Ця математична модель забезпечує чітке й ефективне планування виробничих процесів, зменшуючи ризик простоїв, конфліктів у графіку та помилок при розподілі завдань. За допомогою встановлених обмежень модель гарантує, що завдання призначаються працівникам і обладнанню з дотриманням робочих графіків, забезпечуючи таким чином безперебійне виробництво.

2.4 Обґрунтування вибору інтерфейсу

2.4.1 Огляд різновидів інтерфейсів

Інтерфейси програмного забезпечення відіграють важливу роль у взаємодії між користувачем і системою, забезпечуючи зручність та ефективність виконання операцій. Вони є ключовою складовою будь-якої інформаційної системи, оскільки

дозволяють користувачам взаємодіяти з нею інтуїтивно та ефективно. Існує кілька основних типів інтерфейсів, кожен з яких має свої особливості та застосування.

Один із найпоширеніших видів — графічний інтерфейс користувача (Graphical User Interface, GUI). Він заснований на візуальних елементах, таких як кнопки, меню, діалогові вікна та інші компоненти, з якими користувач може взаємодіяти за допомогою миші або сенсорного екрана. GUI дозволяє користувачам виконувати складні операції за допомогою простих дій, що робить його зручним і зрозумілим для широкого кола користувачів. Цей інтерфейс підходить для більшості програм, що вимагають частих взаємодій з користувачем, включаючи програми планування, керування ресурсами та офісні додатки. На рисунку 2.6 зображено приклад графічного інтерфейсу[14].

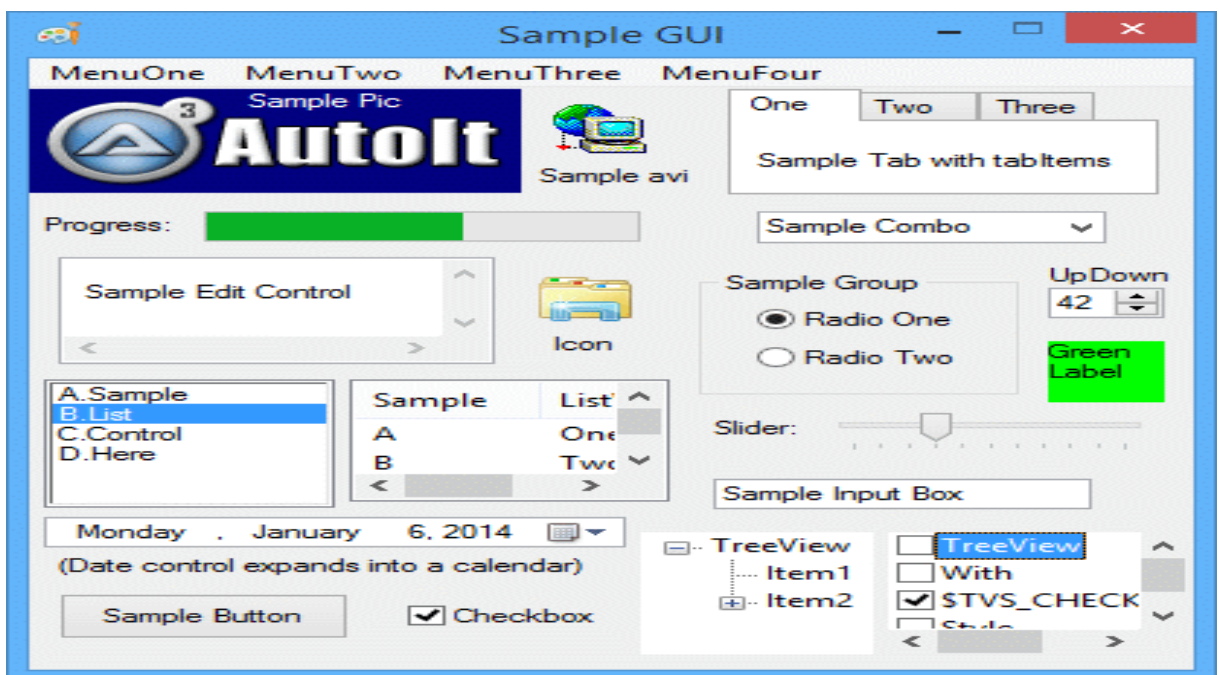
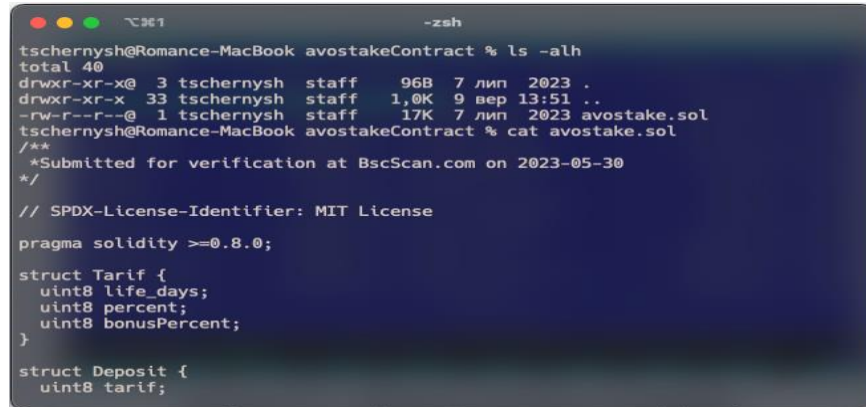


Рисунок 2.6 – Приклад графічного інтерфейсу

Текстовий інтерфейс або командний рядок (CLI — Command Line Interface) — ще один тип інтерфейсу, який часто використовується в професійних середовищах. CLI дозволяє користувачам вводити команди у вигляді тексту, що забезпечує прямий доступ до функціоналу програми. Командний рядок вимагає від користувача знання певних команд та синтаксису, тому він менш інтуїтивний

у порівнянні з GUI. Однак, CLI надає більший контроль над системою та часто використовується для автоматизації процесів або виконання складних завдань, що потребують точних інструкцій. На рисунку 2.7 зображено приклад текстового інтерфейсу[15].



```
tschernysh@Romance-MacBook avostakeContract % ls -alh
total 40
drwxr-xr-x@ 3 tschernysh  staff   96B  7 лип  2023 .
drwxr-xr-x 33 tschernysh  staff  1,0K  9 вер 13:51 ..
-rw-r--r--@ 1 tschernysh  staff   17K  7 лип  2023 avostake.sol
tschernysh@Romance-MacBook avostakeContract % cat avostake.sol
/**
 *Submitted for verification at BscScan.com on 2023-05-30
 */

// SPDX-License-Identifier: MIT License
pragma solidity >=0.8.0;

struct Tarif {
    uint8 life_days;
    uint8 percent;
    uint8 bonusPercent;
}

struct Deposit {
    uint8 tarif;
```

Рисунок 2.7 – приклад текстового інтерфейсу

Важливу роль у сучасних системах відіграють веб-інтерфейси. Веб-інтерфейси є частиною програмного забезпечення, до якого користувачі отримують доступ через веб-браузери. Вони дозволяють виконувати всі необхідні операції, не встановлюючи додаткове програмне забезпечення на пристрій користувача. Такий підхід є зручним для розподілених систем, особливо в умовах, коли користувачі працюють у віддаленому режимі або мають обмежені ресурси. Веб-інтерфейси можуть включати елементи GUI та підтримувати інтерактивні функції через інтернет-з'єднання. На рисунку 2.8 зображено приклад веб-інтерфейсу[16].

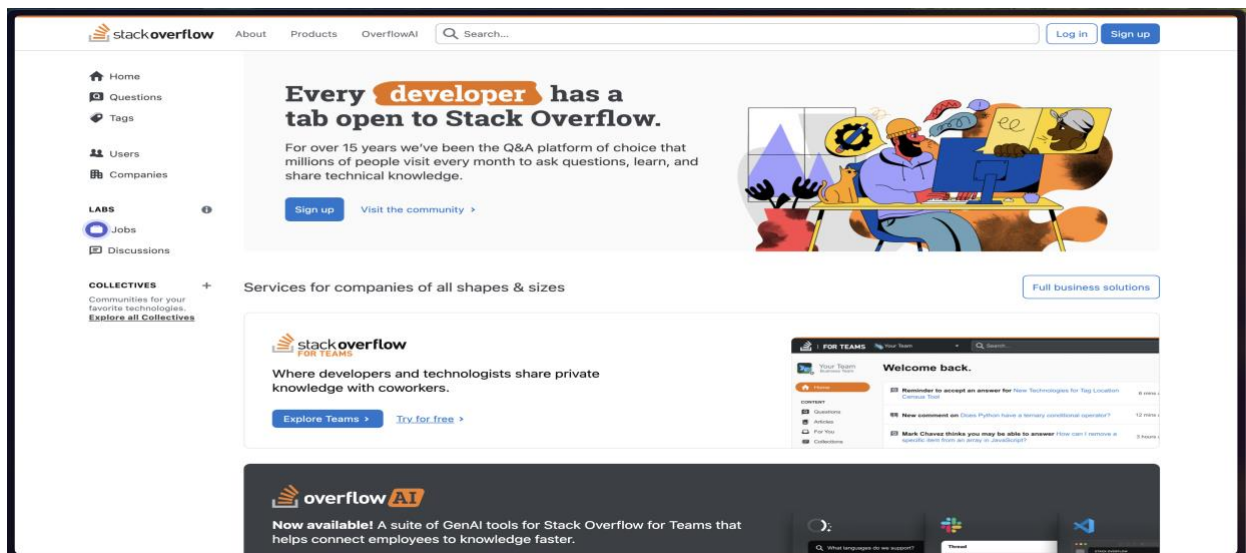


Рисунок 2.8 – приклад веб-інтерфейсу

Голосові інтерфейси (Voice User Interface, VUI) стають все більш популярними завдяки розвитку технологій розпізнавання голосу. Замість введення текстових команд або використання графічних елементів, користувач може керувати системою за допомогою голосових команд. Такі інтерфейси знаходять застосування в системах домашньої автоматизації, мобільних додатках, автомобільних інформаційних системах та багатьох інших областях, де швидкість і зручність взаємодії є критично важливими[17].

Кожен тип інтерфейсу має свої переваги та недоліки і підходить для різних сценаріїв використання. Вибір інтерфейсу залежить від специфіки програми, вимог користувачів та доступних ресурсів. У контексті системи планування виробництва особливо важливими є GUI та веб-інтерфейси, оскільки вони дозволяють керувати складними процесами за допомогою інтуїтивних інструментів та забезпечують доступність із різних пристроїв і локацій

2.4.2 Переваги та недоліки

Текстовий інтерфейс є ефективним для швидкого виконання команд і потребує мінімальних системних ресурсів. Однак, він менш інтуїтивний для новачків і обмежений візуальними можливостями.

Графічний інтерфейс інтуїтивний та забезпечує багатий візуальний досвід,

полегшуючи роботу користувачам. Проте його розробка потребує більше ресурсів і він може бути важким для використання на пристроях з низькою роздільною здатністю.

Веб-інтерфейс забезпечує глобальну доступність через Інтернет і дозволяє легко оновлювати систему для всіх користувачів. Водночас він залежить від інтернет-з'єднання та потребує серверної інфраструктури для функціонування.

Голосовий інтерфейс дозволяє користувачам взаємодіяти із системою через голосові команди, забезпечуючи зручність і швидкість. Однак, він може бути неточним у гучних середовищах та потребує якісного розпізнавання мови, що може ускладнити використання.

2.4.3 Вибір графічного інтерфейсу

Розглянувши переваги та недоліки кожного з видів інтерфейсу ми можемо дійти слідуючих висновків. Текстовий та голосовий інтерфейс абсолютно точно не підходять для реалізації даного проєкту через їхню низьку зручність.

Додаток повинен бути дуже детально і грамотно візуалізовано задля його легкого та максимально інтуїтивного розуміння. Залишається вибір між веб та графічним інтерфейсом. Так як проєкт також потребує сумісності між всіма пристроями та операційними системами було обрано веб-інтерфейс через його простоту розробки, незалежність від пристрою на якому буде запущений додаток, а також високу доступність так як браузер можна відкрити на будь-яких засобах які мають доступ до інтернету.

2.5 Опис інтерфейсу

На рисунку 2.9 зображено сторінку для керування даними про кожного зареєстрованого працівника, що є важливою складовою системи управління персоналом на підприємстві. Ця сторінка надає адміністраторам або відповідальним за кадрове управління доступ до всієї необхідної інформації про працівників. Система дозволяє зручно переглядати та редагувати дані, що

включають особисту інформацію, контактні дані, роль у компанії, кваліфікацію та інші важливі параметри. Окрім перегляду, адміністратори можуть легко змінювати інформацію про працівників, у разі потреби оновлювати її, а також видаляти записи, якщо працівник залишив компанію.

Крім того, функціонал сторінки дозволяє створювати нові записи про працівників, що полегшує процес реєстрації нових членів команди в системі. Цей процес автоматизований і включає всі необхідні поля для введення даних, що дозволяє уникнути помилок та значно прискорює роботу з великим обсягом інформації. Для швидкого пошуку передбачена можливість фільтрації за ім'ям користувача або його нікнеймом, що спрощує доступ до потрібного запису, особливо на великих підприємствах з великою кількістю працівників.

Ця сторінка є важливим інструментом управління персоналом, оскільки вона забезпечує повний контроль над даними кожного працівника. Це дозволяє своєчасно оновлювати інформацію, стежити за змінами у складі персоналу та забезпечувати правильний розподіл задач і ресурсів у системі управління підприємством.

Елементи сторінки менеджменту персоналу:

- Кнопка додавання працівника з іншого підприємства
- Кнопка створення нового працівника у нашому підприємстві
- Зона виділена для різних фільтрів
- Таблиця з даними про працівників

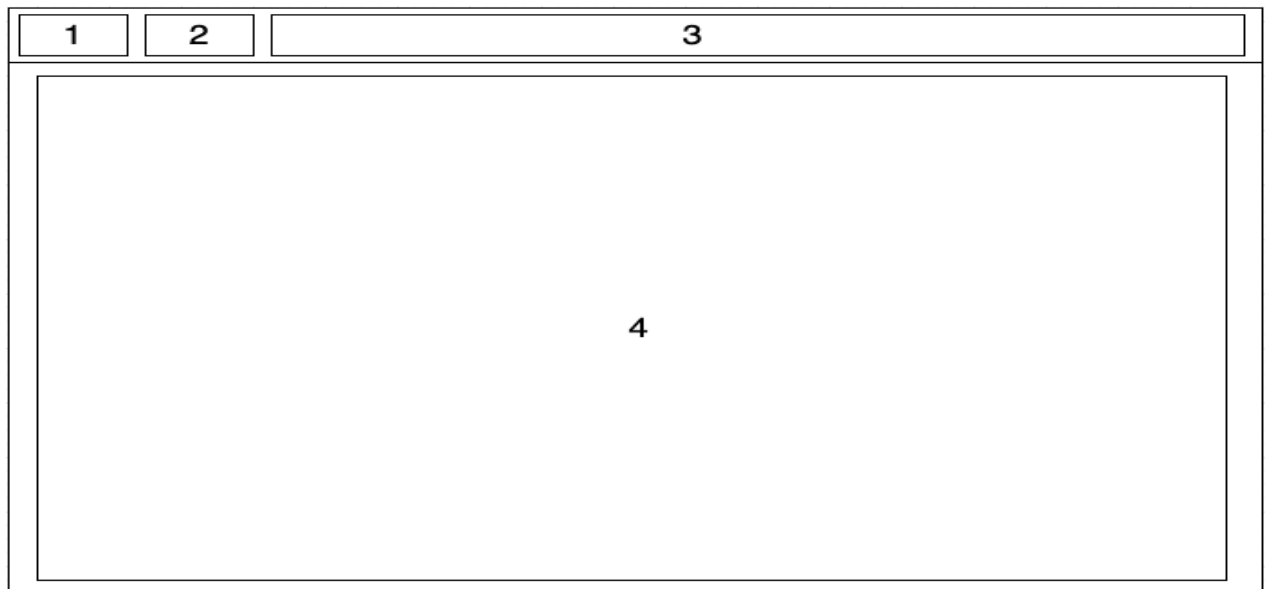


Рисунок 2.9 – Веб-інтерфейс сторінки менеджменту працівників

На рисунку 2.10 представлено сторінку керування змінами працівників, яка забезпечує зручний інструмент для планування та контролю робочого часу. Система пропонує низку фільтрів для швидкого пошуку та управління змінами, що дозволяє оперативно знайти потрібного працівника чи групу працівників за певними критеріями, такими як відділ, позиція або поточний графік. Це значно спрощує роботу з великими колективами та забезпечує більш ефективне управління ресурсами.

Особливістю інтерфейсу є таблиця, яка візуально відображає робочі години працівників протягом тижня. Завдяки цьому керівники можуть легко бачити розподіл робочого часу і вчасно вносити корективи у випадку необхідності. Додатково система дозволяє зручно призначати зміни одразу для кількох працівників завдяки функції копіювання зміни одного працівника та її швидкого застосування до інших. Ця можливість значно скорочує час на планування, зберігаючи точність і узгодженість графіків.

Таким чином, сторінка керування змінами забезпечує не тільки повний контроль за розкладом, але й дозволяє швидко та ефективно впроваджувати зміни, що робить процес планування більш гнучким і зручним для використання як керівниками, так і працівниками.

1	2	3	
		4	5

Рисунок 2.10 – Веб-інтерфейс сторінки менеджменту змін працівників

Елементи сторінки менеджменту змін працівників:

- Видвижна панель фільтрів
- Панель вибору змін, дати відображення графіку
- Блок відображення календаря
- Таблиця список працівників
- Таблиця змін відповідно до кожного працівника

Далі пропоную розглянути сторінку керування обладнанням, яка забезпечує адміністраторам повний контроль над всіма аспектами управління обладнанням на підприємстві. Ця сторінка дозволяє не тільки переглядати наявне обладнання, але й виконувати такі операції, як видалення, додавання нових одиниць, а також зміну даних про обладнання. Завдяки зручному інтерфейсу, адміністратори можуть оперативно оновлювати інформацію про обладнання, зокрема його технічні характеристики, стан обслуговування, місце розташування та відповідність вимогам виробництва.

Сторінка керування обладнанням зображена на рисунку 2.11 також включає в себе можливість додавання нових одиниць обладнання з внесенням усіх необхідних параметрів. Це допомагає підтримувати актуальність інформації в системі та зберігати повний облік технічних засобів, що використовуються у

виробничих процесах. Крім того, можливість видалення застарілих або непрацюючих одиниць обладнання дозволяє оптимізувати управління ресурсами підприємства, зосереджуючи увагу лише на актуальному обладнанні.

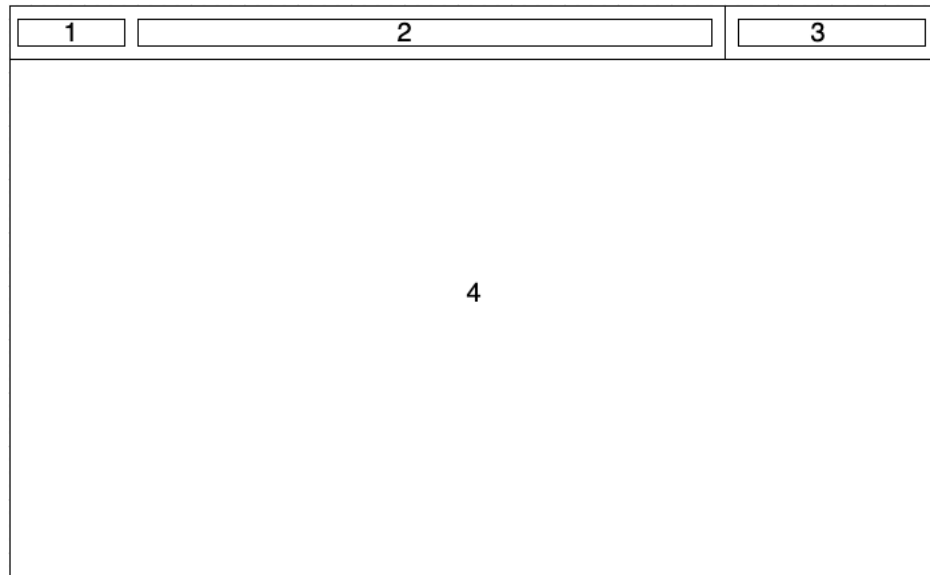


Рисунок 2.11 – Веб-інтерфейс сторінки керування обладнанням

Елементи сторінки менеджменту обладнання:

- Кнопка додавання нового обладнання
- Панель для фільтрування даних
- Текстовий фільтр
- Таблиця обладнання та їх даних

Сторінка керування задачами є важливим інструментом для управління окремими одиницями робіт, які згодом об'єднуються в процеси. Вона надає користувачам можливість задавати всі необхідні параметри для кожної окремої задачі, що включає тривалість виконання, необхідну кількість працівників, а також ресурси й обладнання, які будуть задіяні в процесі її реалізації. Це дозволяє ефективно контролювати всі етапи підготовки та виконання виробничих завдань на рівні окремих задач.

На цій сторінці адміністратори можуть створювати нові задачі, вводячи основну інформацію, таку як назва задачі, її опис, тривалість і строки виконання.

Важливою функцією є можливість вказати, скільки працівників потрібно для виконання задачі, а також яких конкретно працівників можна залучити до її реалізації. Система дозволяє задавати ролі для кожного працівника, щоб забезпечити їхню відповідність вимогам задачі, враховуючи їхню кваліфікацію та доступність.

Також передбачена можливість вибору обладнання, на якому задача буде виконуватись. Адміністратори можуть вказати точні типи обладнання або установки, які будуть задіяні у процесі, що дозволяє уникнути конфліктів із іншими задачами чи процесами, що використовують ті ж самі ресурси. Це забезпечує більш точне планування та мінімізує ризики простоїв через нестачу доступного обладнання.

Крім того, на сторінці є функціонал для керування додатковими умовами виконання задачі, такими як підготовчі етапи, необхідність обслуговування обладнання перед початком виконання або після його завершення. Це дозволяє враховувати специфічні вимоги до задач, такі як час на підготовку чи очищення обладнання, а також інші фактори, які можуть вплинути на тривалість та якість виконання завдання.

Адміністратори також мають можливість коригувати задачі в реальному часі, змінюючи тривалість виконання, замінюючи працівників або вносячи корективи в розподіл обладнання. Це дозволяє гнучко реагувати на зміни у виробничому процесі, швидко адаптуючи задачі до нових умов або змін у наявності ресурсів.

Загалом, сторінка керування задачами забезпечує користувачів усіма необхідними інструментами для детального планування й управління кожною задачею окремо. Вона дозволяє налаштувати всі необхідні параметри для успішного виконання завдань та забезпечує гнучкість у керуванні процесом роботи на рівні кожної окремої одиниці виробництва. Інтерфейс сторінки керування задачами зображено на рисунку 2.12.

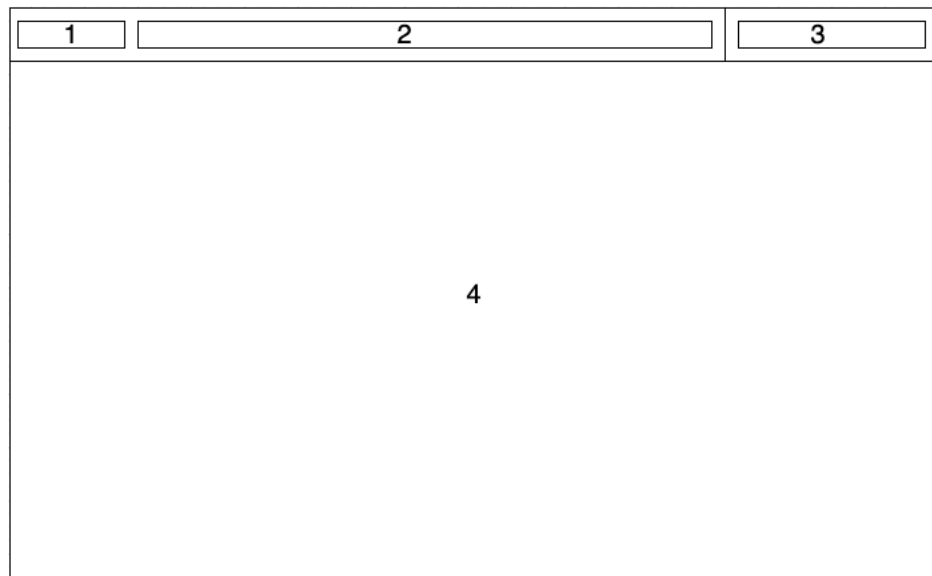


Рисунок 2.12 – Веб-інтерфейс сторінки керування задачами

Елементи сторінки менеджменту задач:

- Кнопка додавання нової задачі
- Панель для фільтрування даних
- Текстовий фільтр
- Таблиця задач та їх даних

Як тільки ми завершили з створенням ресурсів на попередніх етапах ми можемо переходити до сторінки керування процесами зображених на рисунку 2.13

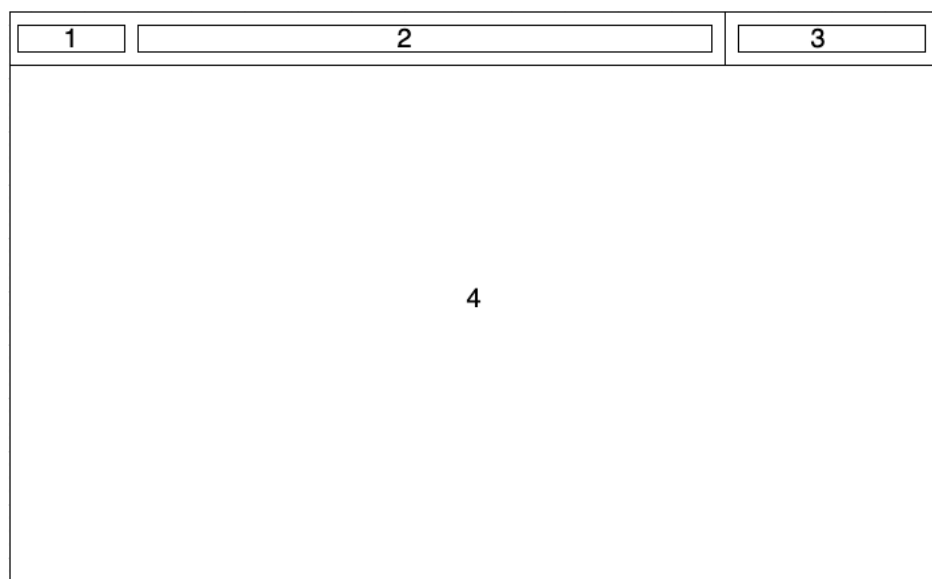


Рисунок 2.13 – Веб-інтерфейс сторінки керування процесами

Елементи сторінки менеджменту процесів:

- Кнопка додавання нового процесу
- Панель для фільтрування даних
- Текстовий фільтр
- Таблиця процесів та їх даних

Сторінка створення процесів є одним з ключових елементів у системі управління виробництва, оскільки вона дозволяє формувати нові процеси, які об'єднують кілька задач і забезпечують ефективне планування робіт на підприємстві. На цій сторінці користувачі можуть створювати процеси, задаючи основні параметри, такі як назва процесу, тип, пріоритет, та інші важливі характеристики, що визначають його роль у загальному виробничому ланцюгу. Система надає можливість швидко налаштувати всі необхідні деталі, щоб процес був готовий до інтеграції у виробничий графік.

Під час створення процесу користувачі можуть визначати, до якого відділу або частини виробництва він належить, а також обирати відповідне обладнання, яке буде задіяне у виконанні задач. Це дозволяє автоматично врахувати ресурси, необхідні для успішного виконання процесу, і уникнути конфліктів з іншими запланованими роботами. Інтерфейс сторінки забезпечує зручний і інтуїтивний спосіб вибору та налаштування ресурсів, що значно полегшує процес організації роботи.

Однією з ключових функцій сторінки є можливість додавання до процесу окремих задач, які складають його основу. Кожна задача в процесі може мати свій власний набір параметрів, таких як тривалість, кількість задіяних працівників та специфічні вимоги до обладнання. Це дозволяє детально спланувати кожен етап виробничого процесу, а також забезпечити його гнучкість і відповідність вимогам підприємства. Після налаштування задач, їх можна розташувати у процесі відповідно до послідовності виконання.

Для керування задачами, що входять до процесу, передбачене зручне

модальне вікно, яке відкривається для редагування або перегляду деталей окремих задач. У цьому вікні можна змінювати параметри задачі, додавати або видаляти працівників, переглядати графік виконання та перевіряти відповідність завдання вимогам процесу. Вікно зображене на рисунку 2.14 також дозволяє оперативно коригувати деталі задачі, якщо у виробничому процесі відбулися зміни або з'явилися нові обмеження.

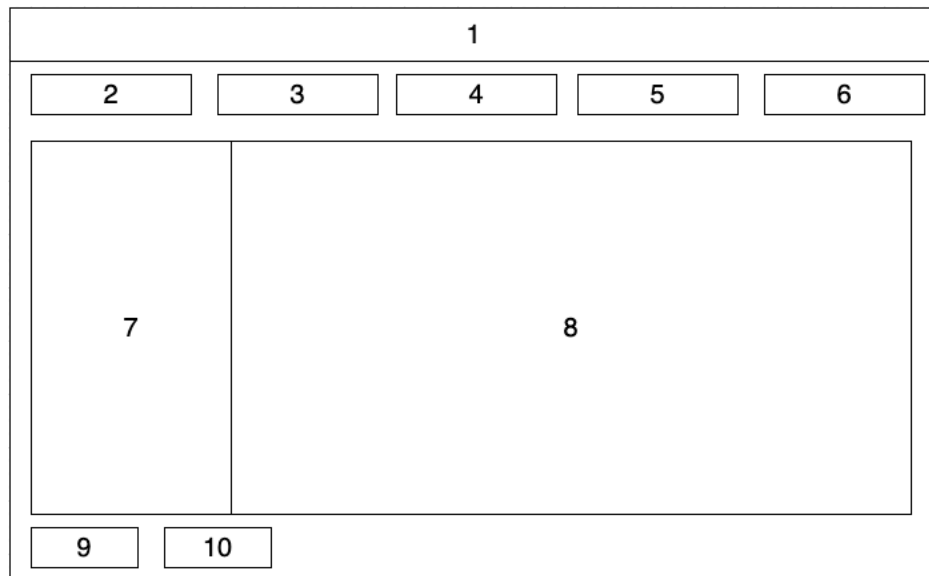


Рисунок 2.14 – Модальне вікно керування задачами процесу

Елементи модального вікна керування задачами процесу:

- Шапка модального вікна
- Селектор вибору процесу
- Селектор вибору типу процесу
- Палітра вибору кольору процесу
- Кнопка вибору потрібних фахових працівників для задач процесу
- Селектор додавання задачі до процесу
- Таблиця уже існуючих задач в процесі
- Таблиця часового графіку тривалості виконання задачі у процесі
- Кнопка збереження
- Кнопка відміни змін

Модальне вікно для керування задачами забезпечує простоту у використанні завдяки своїй структурі, що розбиває інформацію на чіткі секції. Користувачі можуть швидко вносити зміни до задач без необхідності виходити зі сторінки процесу, що значно спрощує і прискорює роботу. Зручна навігація дозволяє швидко переходити між задачами, змінювати їх статус або порядок виконання, забезпечуючи повний контроль над процесом.

Таким чином, сторінка створення процесів і модальне вікно для управління задачами є важливими інструментами для ефективного планування виробничих завдань. Вони дозволяють не тільки створювати нові процеси та керувати задачами, але й забезпечують гнучкість у налаштуванні кожного етапу процесу, що сприяє оптимізації роботи підприємства.

2.6 Висновок до розділу 2

У розділі 2 було розроблено та обґрунтовано підхід для оптимізації процесу виробничого планування. Основний акцент зроблено на впровадженні сучасних алгоритмів, які дозволяють підвищити швидкодію планування.

Запропоновано гібридний підхід, що поєднує методи часових рядів (ARIMA) для короткострокового прогнозування з рекурентними нейронними мережами (LSTM) для довгострокового прогнозування. Такий підхід дозволяє враховувати сезонні коливання попиту, довгострокові тенденції та оперативно адаптуватися до змін у виробничих умовах. Це сприяє зниженню часових витрат, мінімізації ризиків дефіциту ресурсів і надмірного накопичення продукції.

Запропоновано підхід щодо оптимізації розкладів, який відрізняється використанням генетичних алгоритмів у поєднанні з табу-пошуком і механізмами адаптивного навчання, що дало можливість підвищити швидкодію.

Також значну увагу приділено методам попередньої обробки даних, у тому числі очищенню, нормалізації та агрегації, що забезпечують цілісність та коректність інформації, яка використовується для побудови розкладів і прогнозів.

Хоча інтерфейси були розглянуті як важливий компонент системи, їх роль є

другорядною порівняно з розробкою алгоритмів. Основна увага в цьому розділі приділена впровадженню інтелектуальних підходів до планування, що дозволяють забезпечити досить високу швидкість, точність і стабільність виробничих процесів, ефективне використання ресурсів та зниження впливу людського фактору.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПЛАНУВАННЯ ВИРОБНИЦТВА

3.1 Загальний огляд набору даних

Оптимізація розкладу у виробничій системі потребує роботи з великим набором даних, що включає інформацію про процеси, співробітників та ресурси. У нашій програмній реалізації використовуються кілька ключових таблиць: `process_activity_df`, `employee_df` та `activity_resources_df`. Кожна з них виконує свою роль у забезпеченні точності та ефективності розподілу завдань між працівниками та ресурсами.

DataFrame `process_activity_df` зберігає інформацію про діяльності та процеси у виробничій системі. У цьому наборі даних містяться поля, що визначають ID процесу та діяльності, час початку та завершення завдань, тривалість виконання (*workload*), а також перелік ресурсів, необхідних для кожної діяльності. Ці дані дозволяють визначити, коли і де повинні виконуватись завдання, а також які ресурси потрібні для їхньої реалізації.

`Employee_df` містить детальну інформацію про працівників, включаючи їхні ID, імена, позиції та графіки робочих змін. Використання цих даних дозволяє забезпечити відповідність між вимогами до завдань та навичками доступних співробітників. З урахуванням робочих змін для кожного працівника система здатна призначати завдання таким чином, щоб уникати перевищення робочого часу і забезпечувати оптимальне використання кадрових ресурсів.

Третій набір даних, `activity_resources_df`, містить відомості про ресурси, необхідні для виконання кожної діяльності. У цій таблиці зазначено обладнання, зони та командні ресурси, які забезпечують підтримку кожного процесу. Враховуючи ці дані, система здатна гарантувати, що ресурси, такі як обладнання чи спеціальні зони, будуть доступні у відповідний час для кожної діяльності, а також запобігає конфліктам у використанні спільних ресурсів.

На основі цих трьох наборів даних будується модель оптимізації, яка реалізується з використанням бібліотеки OR-Tools для обмеженого програмування. Ця бібліотека дозволяє ефективно обробляти численні обмеження, забезпечуючи мінімальний час завершення всіх процесів (makespan) та оптимальний розподіл ресурсів.

Функція оптимізації отримує на вхід параметри `'start_date'` та `'end_date'`, які визначають часовий інтервал для оптимізації розкладу. Можна також вказати `'selected_process_id'` для оптимізації певного процесу. Якщо цей параметр не задано, функція оптимізує розклад для всіх процесів, що підпадають під заданий часовий інтервал.

На першому етапі здійснюється фільтрація даних у `process_activity_df` на основі зазначених параметрів `'start_date'`, `'end_date'` та `'selected_process_id'`. Це дозволяє обмежити обсяг даних, що обробляються, зосередившись лише на тих процесах, які відповідають заданим умовам, і тим самим зменшити обчислювальну складність завдання.

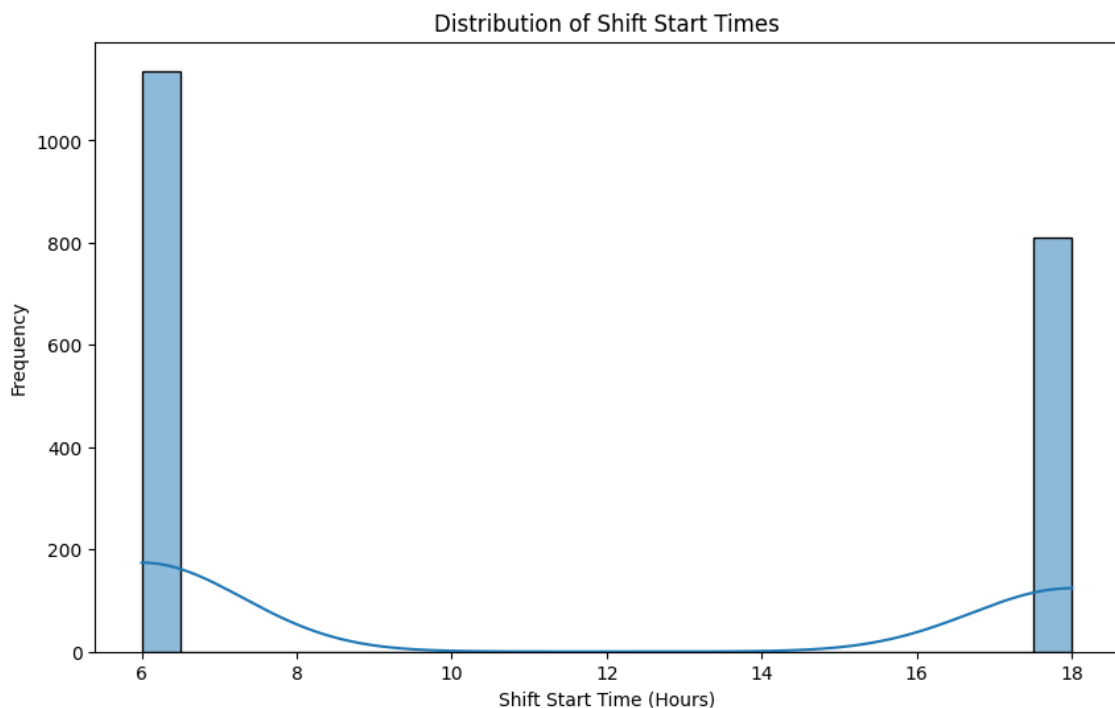


Рисунок 3.1 – Графік розподілу старту змін.

Після цього створюється модель обмеженого програмування. Визначається часовий горизонт для оптимізації на основі кінцевого часу останньої діяльності. Для кожної діяльності створюються змінні, які визначають її час початку, час завершення та інтервал виконання. Це дозволяє чітко визначити часові рамки для кожної діяльності в межах загального розкладу.

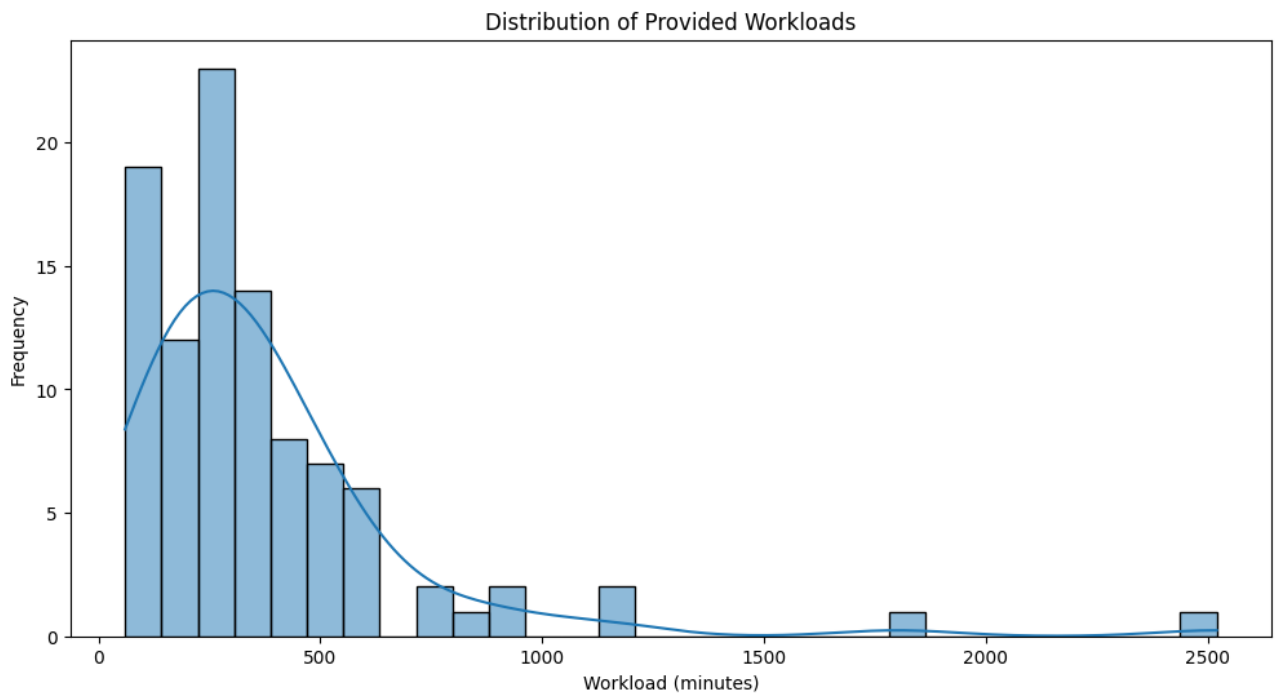


Рисунок 3.2 – Графік навантаження по процесам.

На наступному етапі обробляються ресурси, необхідні для виконання кожної діяльності. Для ресурсів типу «обладнання» та «зона» система встановлює обмеження, які запобігають конфліктам у їх використанні. Ці обмеження реалізуються за допомогою методу 'NoOverlap', що дозволяє гарантувати, що обладнання або зона не будуть зайняті кількома завданнями одночасно.

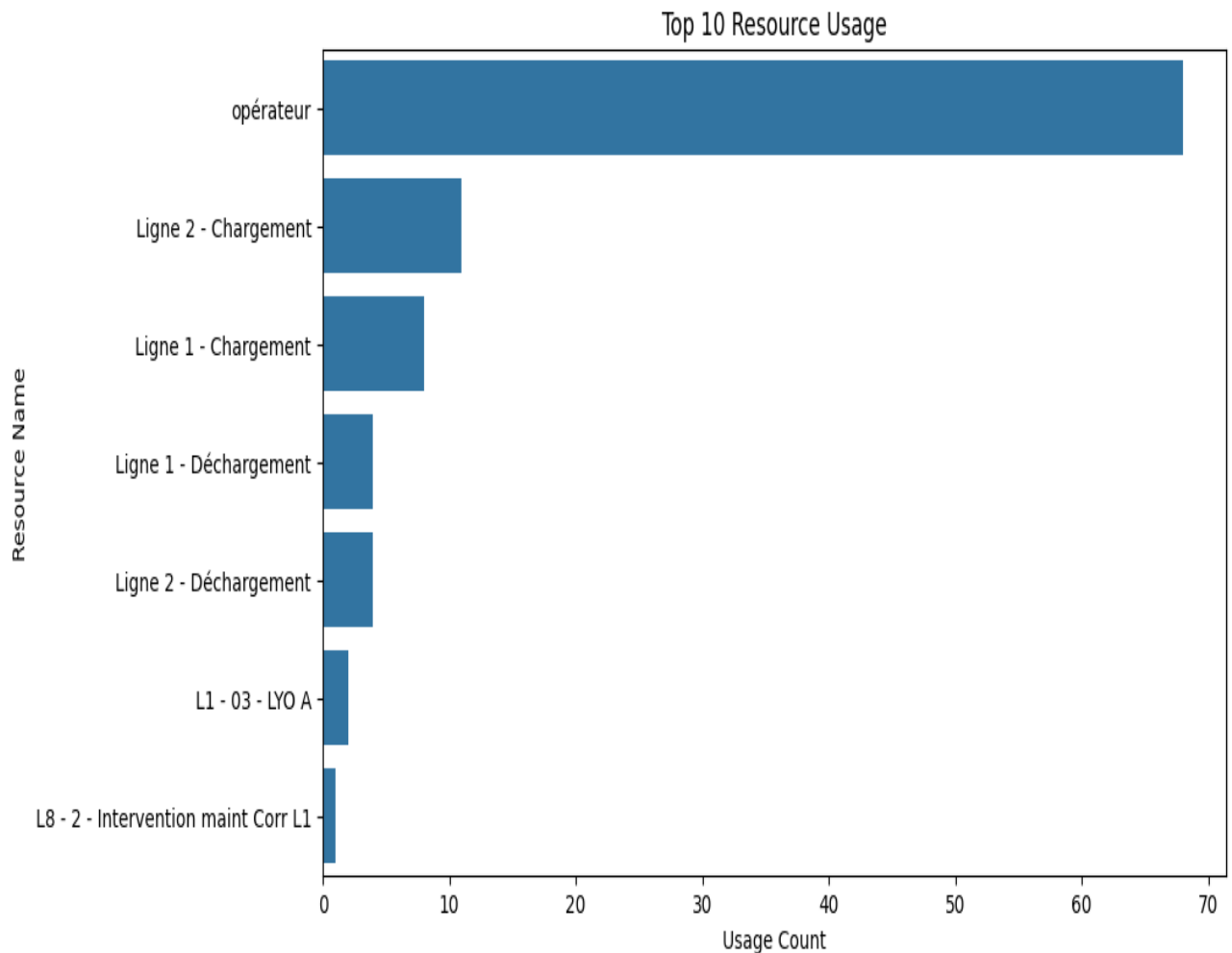


Рисунок 3.3 – Графік частоти використання ресурсів.

Для управління робочими змінами співробітників з `employee_df` створюються додаткові записи, які розширюють розклад на весь період планування. Це дає змогу уникнути помилок у розподілі завдань, оскільки система враховує робочі зміни, що перетинають межі днів. Такий підхід дозволяє забезпечити повний контроль за графіками роботи кожного співробітника.

Призначення співробітників до завдань здійснюється на основі вимог кожної діяльності до персоналу. Для кожного завдання створюються змінні, які визначають, чи призначено певного співробітника для виконання конкретного завдання. Це допомагає забезпечити відповідність між вимогами до кваліфікації та наявними навичками працівників.

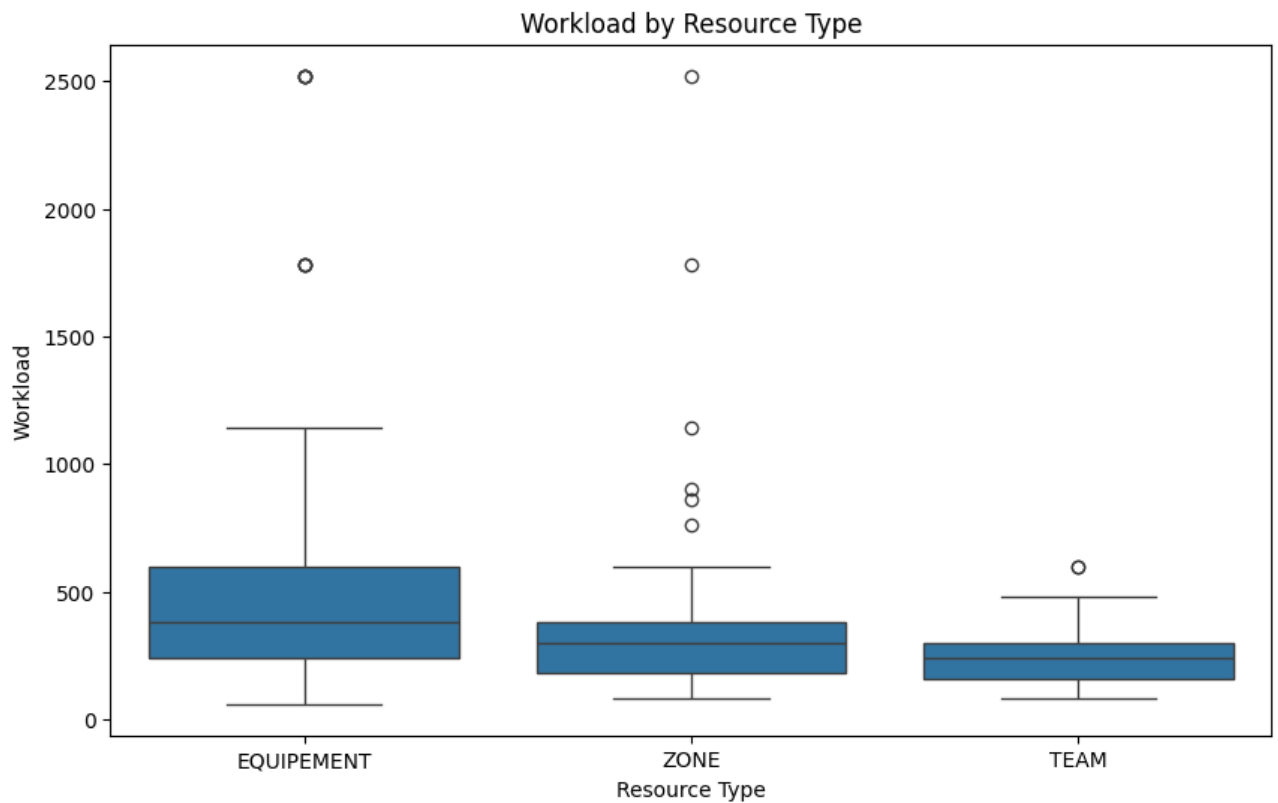


Рисунок 3.4 – Графік навантаженості по ресурсам.

Завершальним етапом є оптимізація розкладу з метою мінімізації загального часу завершення всіх процесів (makespan). Використання багатопотокової обробки дозволяє значно прискорити процес обчислень і знайти оптимальний розклад за прийнятний час.

Після розв'язання моделі система виводить результати, що включають інформацію про початок і кінець кожної діяльності, залучених працівників, а також необхідне обладнання та зони. Для кожного співробітника формується індивідуальний розклад із зазначенням завдань і робочих змін, що дозволяє ефективно планувати робочий процес.

Таким чином, ця програмна реалізація дозволяє ефективно розподілити завдання між співробітниками, забезпечити оптимальне використання обладнання, уникнути конфліктів у розкладі та мінімізувати загальний час виконання всіх процесів.

3.2 Обґрунтування вибору програмних засобів для розробки

Перед розробкою системи управління виробничими процесами необхідно ретельно обрати мови програмування для різних компонентів програми. Зокрема, потрібно підібрати оптимальні варіанти для веб-інтерфейсу, баз даних та компонентів штучного інтелекту. Порівнявши різні підходи, ми обрали найефективніші рішення для кожної частини системи.

1. Веб-інтерфейс: React проти Angular і Vue.js

Вибір фреймворку для веб-інтерфейсу зазвичай зводиться до трьох основних варіантів: Angular, Vue.js або React. Angular, створений Google, надає повноцінне середовище для великих проєктів, але його складність може стати перешкодою для менших команд або проєктів, що вимагають швидкого розвитку. Vue.js простіший у засвоєнні та використанні, проте його екосистема не така потужна і розвинена, як у конкурентів, особливо для корпоративних застосунків. React, створений Facebook, має явну перевагу завдяки гнучкості, популярності та потужному ком'юніті. Він підтримує компонентно-орієнтовану архітектуру, що дозволяє створювати масштабовані, інтерактивні веб-додатки. Його віртуальний DOM забезпечує високу продуктивність рендерингу порівняно з реальним DOM, який використовує Angular. Також React легко інтегрується з іншими бібліотеками та інструментами, що робить його ідеальним вибором для розробки складних веб-інтерфейсів. Завдяки цим перевагам, React був обраний для веб-розробки даного проєкту, адже він забезпечує швидкий і гнучкий інтерфейс[18].

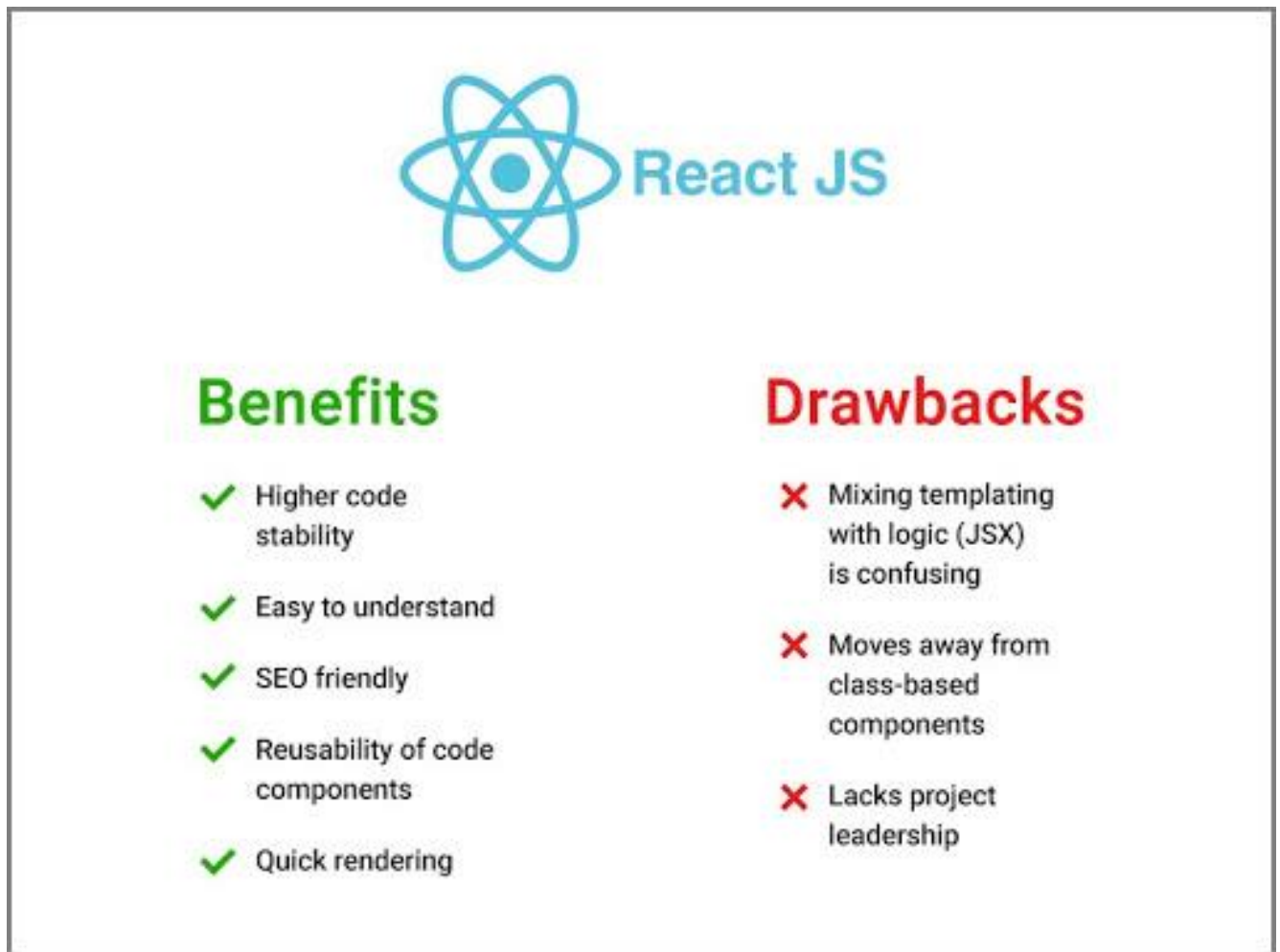


Рисунок 3.5 – Переваги React

2. База даних: Microsoft SQL Server проти MySQL та PostgreSQL

Для управління базами даних існує кілька популярних варіантів реляційних баз даних, таких як MySQL, PostgreSQL та Microsoft SQL Server. MySQL часто обирають для невеликих проєктів через його простоту та відкритий код, але його функціональність обмежена для великих корпоративних систем. PostgreSQL, хоч і потужний, іноді вимагає більше налаштувань і не настільки добре інтегрований з іншими продуктами Microsoft[19].

Microsoft SQL Server виграє завдяки своїй інтеграції з екосистемою Microsoft, високій продуктивності та стабільності. Він пропонує чудові можливості для роботи з великими обсягами даних та складними транзакціями, що робить його оптимальним вибором для масштабних підприємств. Крім того, він добре підходить для аналітичних і BI (Business Intelligence) задач, що дозволяє

легко виконувати складні аналітичні запити для підтримки рішень у виробничій системі. Тому для управління базою даних було обрано саме Microsoft SQL Server, оскільки він забезпечує необхідну продуктивність, надійність та інтеграцію з іншими системами.



Рисунок 3.6 – Переваги SQL

3. Штучний інтелект: Python проти Java та C++

Для розробки компонентів штучного інтелекту існує кілька потужних мов програмування, таких як Python, Java та C++. Java, хоча і є платформо-незалежною та продуктивною, може бути надто складною для швидкого

прототипування та має обмежену кількість спеціалізованих бібліотек для машинного навчання. C++ забезпечує виняткову швидкість виконання та контроль над пам'яттю, що корисно для високопродуктивних систем, але розробка на ньому складніша та повільніша.

Python виграє завдяки своїй простоті та величезній кількості бібліотек для штучного інтелекту та машинного навчання, таких як TensorFlow, Keras, PyTorch та Scikit-learn. Python дозволяє швидко розробляти прототипи алгоритмів і легко інтегрувати їх у більші системи, що робить його найбільш підходящим для створення компонентів штучного інтелекту. Завдяки його широкому застосуванню в наукових та аналітичних дослідженнях, Python був обраний як основна мова для розробки модулів штучного інтелекту в цій системі[20, 21, 22].

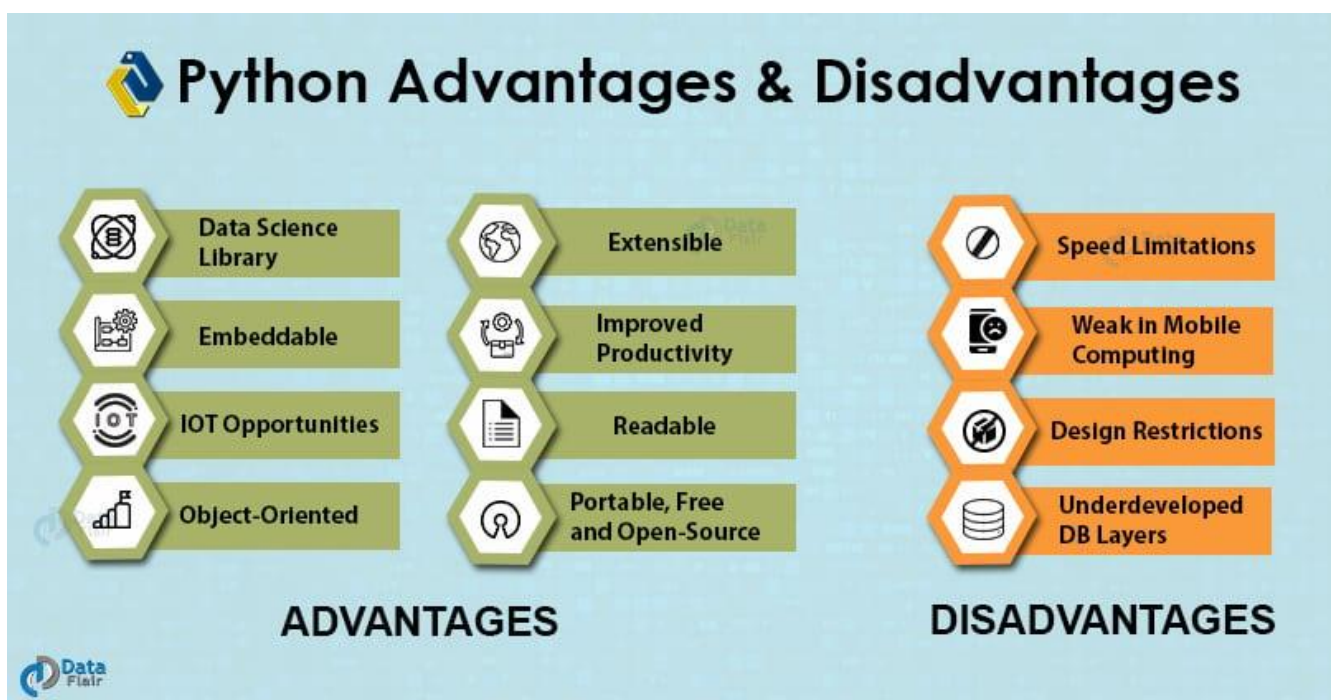


Рисунок 3.7 – Переваги Python в контексті штучного інтелекту

3.1.2 Вибір середовища розробки

У процесі розробки програмного забезпечення для планування виробництва важливою складовою є вибір середовища розробки. Це середовище має не тільки забезпечувати ефективну роботу з даними, але й підтримувати роботу з інтелектуальними системами, які базуються на штучному інтелекті. У цьому

розділі розглянуто вибір двох основних компонентів середовища розробки: платформи для роботи з базами даних і середовища для розробки компонентів штучного інтелекту.

1. Вибір платформи для управління базами даних: Microsoft Azure

Для зберігання та управління даними було обрано Microsoft Azure, зокрема її службу Azure SQL Database. Основна причина вибору Azure полягає в її широкій функціональності та інтеграції з іншими продуктами Microsoft, що дозволяє створювати масштабовані, надійні рішення для великих підприємств. Azure забезпечує гнучкість у керуванні базами даних завдяки інструментам аналітики, резервного копіювання та безпеки, що є критично важливим для управління ресурсами виробництва.

У порівнянні з іншими платформами для баз даних, такими як Amazon Web Services (AWS) або Google Cloud Platform (GCP), Azure виграє завдяки тісній інтеграції з іншими продуктами Microsoft, включаючи Visual Studio, Microsoft 365, а також глибокій підтримці для реляційних баз даних (SQL). AWS також пропонує потужні бази даних (наприклад, Amazon RDS), але у випадку проєктів, що базуються на технологіях Microsoft, Azure надає зручніший і простіший підхід до розгортання та управління даними. GCP, хоч і пропонує конкурентні рішення (BigQuery для аналітики), не має такої широкої екосистеми для реляційних баз даних, як Azure.

Переваги Azure включають надійну інфраструктуру, високий рівень безпеки, а також автоматичне масштабування та резервне копіювання. Це забезпечує безперебійну роботу та захист даних, що є важливим для системи управління виробництвом. Крім того, Azure SQL Database підтримує інструменти аналітики та обробки даних у реальному часі, що робить її ідеальним рішенням для виробничих систем, які вимагають точності та швидкості обробки великих обсягів даних[23].

2. Вибір середовища для розробки штучного інтелекту: PyCharm

Для розробки компонентів штучного інтелекту було обрано середовище PyCharm. PyCharm є однією з найбільш популярних інтегрованих середовищ

розробки (IDE) для Python, яка забезпечує широкий функціонал для роботи з машинним навчанням та штучним інтелектом. Основною перевагою PyCharm є її глибока інтеграція з популярними бібліотеками для роботи зі штучним інтелектом, такими як TensorFlow, Keras, PyTorch, а також можливість швидкої інтеграції з хмарними сервісами, такими як Microsoft Azure або Google Cloud для навчання моделей штучного інтелекту[24].

На ринку також є інші альтернативи для розробки ШІ, зокрема Jupyter Notebooks, VS Code та Google Colab. Jupyter Notebooks широко використовуються для навчання моделей і є зручним інструментом для досліджень та швидкого прототипування. Однак, для великих проєктів, які потребують комплексного управління кодом, роботою з файлами та інтеграції з базами даних, Jupyter Notebooks може бути менш зручним через відсутність повноцінних можливостей IDE.

VS Code, з його потужними розширеннями для Python, також є популярним вибором для розробників. Проте PyCharm виграє завдяки своїм спеціалізованим інструментам для роботи з Python та машинним навчанням, які роблять його ідеальним вибором для складних систем, що вимагають розвиненої підтримки інтелектуальних алгоритмів.

Іншим конкурентом є Google Colab, який безкоштовно надає ресурси для навчання моделей ШІ. Однак його основним недоліком є те, що він базується в хмарі, що може створювати певні проблеми для великих корпоративних проєктів або для тих, хто потребує більше контролю над локальним середовищем.

PyCharm був обраний завдяки його стабільності, зручності у роботі з великими проєктами, інтеграції з інструментами контролю версій та підтримці складних інструментів для машинного навчання. Це середовище дозволяє ефективно керувати процесом розробки штучного інтелекту, надаючи доступ до розширених можливостей налагодження, управління проєктом та інтеграції з хмарними ресурсами.

Таким чином, вибір середовища розробки базувався на потужності, гнучкості та інтеграції з іншими інструментами. Azure для управління базами

даних та PyCharm для роботи з Python забезпечують надійність, продуктивність та можливість масштабування, що робить їх оптимальними для розробки комплексної системи управління виробництвом із використанням штучного інтелекту.

3.1.3 – Вибір середовища написання коду

При розробці системи управління виробництвом, що включає складні веб-інтерфейси та компоненти штучного інтелекту, вибір середовища написання коду є важливим фактором для забезпечення ефективного і продуктивного робочого процесу. У цьому розділі детально розглянемо вибір інструментів для розробки веб-інтерфейсу та компонентів системи. Для цього проекту було обрано WebStorm як основне середовище для розробки веб-додатку через його потужну інтеграцію з веб-технологіями, а також інші особливості, які роблять його найкращим варіантом для цього завдання[25].

Для розробки веб-інтерфейсу було обрано WebStorm – інтегроване середовище розробки (IDE) від JetBrains, яке спеціалізується на JavaScript, HTML, CSS та інших пов'язаних технологіях. Основною перевагою WebStorm є його глибока інтеграція з сучасними веб-фреймворками та бібліотеками, такими як React, Angular, Vue.js, що дозволяє створювати високоякісні веб-додатки з мінімальними зусиллями. У порівнянні з іншими популярними середовищами для веб-розробки, такими як Visual Studio Code або Sublime Text, WebStorm надає ширший спектр інструментів "з коробки" і потребує менше налаштувань.

Visual Studio Code (VS Code) також є дуже популярним серед розробників, завдяки своїм розширенням і можливостям налаштування, однак WebStorm має більш інтегровані інструменти для налагодження, рефакторингу та автоматизації процесів, що дозволяє зосередитися на самому коді без необхідності встановлення додаткових плагінів. У порівнянні з Sublime Text, WebStorm надає набагато ширший спектр можливостей для фронтенд-розробки, таких як інтеграція з Git, зручний перегляд коду, можливість відстеження змін і більш потужне автозавершення.

Однією з основних переваг WebStorm є його глибока інтеграція з React, який обрано для розробки фронтенду в цьому проєкті. WebStorm автоматично розпізнає JSX (розширення синтаксису JavaScript для React) і забезпечує зручну навігацію, рефакторинг, налагодження і перевірку коду. Це суттєво пришвидшує процес розробки і дозволяє уникнути помилок під час написання складних компонентів.

Також WebStorm має вбудовані інструменти для перевірки якості коду (linting), автоматизації завдань (через інтеграцію з Node.js), а також потужні інструменти для налагодження, що робить його універсальним рішенням для розробки веб-додатків. Таким чином, вибір WebStorm обумовлений його гнучкістю, інтеграцією з веб-фреймворками та високою продуктивністю у веб-розробці.

У порівнянні з Visual Studio Code, WebStorm виграє завдяки своєму багатофункціональному інтерфейсу та інтегрованим інструментам. Visual Studio Code часто потребує додаткових плагінів для досягнення того ж рівня функціональності, що WebStorm надає "з коробки". Хоча VS Code є безкоштовним і має велику кількість плагінів, WebStorm забезпечує більшу стабільність і глибшу інтеграцію з JavaScript-екосистемою. Sublime Text, хоч і швидкий та легкий, не надає таких широких можливостей для фронтенд-розробки, як WebStorm, і більше підходить для редагування тексту, ніж для комплексної розробки.

Отже, WebStorm було обрано як середовище написання коду для розробки веб-інтерфейсу через його інтеграцію з інструментами розробки веб-додатків, зокрема React. Це дозволяє значно спростити роботу з сучасними фреймворками і підвищує продуктивність розробки, що робить його найкращим варіантом для розробки цього проєкту.

3.3 Розробка моделі планування виробництва

Мета розробки цієї моделі полягає в створенні нейронної мережі, здатної ефективно планувати виробничі процеси, забезпечуючи мінімальні простої, відсутність колізій та максимальне використання ресурсів. Ефективне управління процесами вимагає врахування багатьох факторів: доступність обладнання, графіки робітників, порядок виконання завдань, перерви між завданнями тощо.

Основним завданням моделі є знаходження оптимального плану, що дозволяє мінімізувати час простою і забезпечити ефективне розподілення ресурсів при мінімальних затримках. Це досягається за рахунок інтеграції двох основних алгоритмів: LSTM для обробки послідовних даних та ARIMA для короткострокового прогнозування.

Для підготовки даних модель передбачає кілька етапів, які допоможуть привести всі значення до єдиного масштабу, обробити пропущені значення та забезпечити коректну обробку категорійних змінних.

Масштабування ознак: Нормалізація є ключовою для усунення великої варіативності у вхідних даних. Наприклад, для масштабування ми використовуємо мінімакс нормалізацію за формулою:

$$X_{\text{new}} = (X_i - \min(X)) / (\max(X) - \min(X)) \quad (3.1)$$

Кодування нечислових атрибутів: Виконується за допомогою техніки One-Hot Encoding, де кожне значення атрибуту перетворюється в бінарний вектор.

Формула для One-Hot Encoding:

$$\text{Encoded}(X) = [x_1, x_2, \dots, x_n], \text{ де } x_i = 1, \text{ якщо категорія } i \text{ присутня, і } 0, \text{ якщо відсутня.} \quad (3.2)$$

Вибір ознак: Вибір відповідних ознак є критичним етапом, що включає роботу з такими параметрами, як доступність обладнання, графік працівників,

необхідна кількість працівників і тривалість завдань. Цей етап забезпечує, що модель використовує тільки релевантні ознаки, що значно підвищує її точність.

Для реалізації моделі було обрано комбінацію LSTM та ARIMA, де кожен з алгоритмів виконує свою роль у оптимізації.

Архітектура LSTM: Нейронна мережа на основі LSTM зберігає інформацію про послідовність, що є важливим для відстеження черговості виконання завдань. Основні рівняння LSTM включають функції забування, введення та виходу:

$$f_t = \sigma(W_f * [h_{(t-1)}, x_t] + b_f) \quad (3.3)$$

$$i_t = \sigma(W_i * [h_{(t-1)}, x_t] + b_i) \quad (3.4)$$

$$o_t = \sigma(W_o * [h_{(t-1)}, x_t] + b_o) \quad (3.5)$$

ARIMA: Використовується для короткострокового прогнозування, забезпечуючи більш швидке планування на малих інтервалах.

Основне рівняння ARIMA представлено як:

$$y_t = c + \phi_1 * y_{(t-1)} + \dots + \phi_p * y_{(t-p)} + \theta_1 * \varepsilon_{(t-1)} + \dots + \theta_q * \varepsilon_{(t-q)} + \varepsilon_t \quad (3.6)$$

Комбінація ARIMA та LSTM дозволяє отримати точні прогнози, що враховують як короткострокові, так і довгострокові залежності.

Навчання моделі передбачає розподіл даних на тренувальні та тестові вибірки. На кожному етапі оптимізації застосовуються правила обчислення втрат, які дозволяють скоригувати значення ваг та забезпечити мінімізацію помилок. Використовується функція втрат MSE:

$$L = (1/N) * \sum (y_i - \hat{y}_i)^2 \quad (3.7)$$

Моделю налаштовується так, щоб мінімізувати цю функцію. Кожен шар

LSTM підлягає регуляризації через Dropout для уникнення перенавчання.

Тонке налаштування гіперпараметрів виконується для досягнення найкращих результатів. Основні параметри включають розмір кроку, кількість нейронів та рівень Dropout.

Для оцінки якості прогнозу застосовується метрика середньої абсолютної похибки (MAE):

$$MAE = (1/N) * \sum |y_i - \hat{y}_i| \quad (3.8)$$

Також використовується метрика точності прогнозування, яка дозволяє визначити відповідність планування реальним умовам.

На завершення, для поліпшення ефективності моделі проводиться оптимізація, включаючи корекцію ваг і гіперпараметрів. Модель перевіряється на надійність і точність прогнозування, і в разі невідповідностей проводиться повторне навчання.

Результати моделі демонструють високу точність і стабільність при плануванні виробничих процесів, зокрема у випадках частих змін та непередбачуваних збоїв. Модель здатна адаптуватися до обмежень обладнання та графіків роботи, що є критично важливим для зниження простоїв.

3.4 Тестування інформаційної технології планування виробництва

Для перевірки ефективності розробленої інформаційної технології планування виробництва було проведено детальне тестування за участю 30 експертів із галузей виробничого планування, інформаційних технологій та оптимізації бізнес-процесів. Основною метою тестування було оцінити швидкодію, точність, адаптивність та практичність впровадження розробленої технології в умовах реального виробництва.

Вихідні дані:

Тестування проводилося на основі великих обсягів вхідних даних, що

відповідають реальним умовам середнього та великого підприємства:

- Загальна кількість початкових задач: 25000 задач (розподілених між 3000 виробничими процесами).
- Кількість активного обладнання: 1200 одиниць (доступного для задач).
- Кількість недоступного обладнання (плановий ремонт або технічні проблеми): 400 одиниць.
- Кількість працівників: 2500 працівників, що працюють у трьох змінах із нерегулярним графіком.
- Кількість задач, що вже заплановані: 18000 задач.
- Кількість нових задач для планування у процесах: від 10 до 150 задач на процес.

Методика тестування:

Тестування проводилось у декілька етапів, кожен із яких відповідав окремим сценаріям реального виробництва:

Етап 1. Швидкодія планування

Для кожної системи моделювався сценарій, де необхідно запланувати:

- Нові задачі (від 10 до 150) для кожного процесу.
- Враховувалась наявність 18000 вже запланованих задач.
- Обмеження доступності обладнання та працівників.

Час, необхідний для створення розкладу, вимірювався в секундах.

Етап 2. Адаптивність до змін

Перевірялася швидкість реакції системи на зміну умов, таких як:

- Раптова недоступність обладнання (зупинка до 5% обладнання в реальному часі).
- Наявність нових задач (додаткові 2000 задач, що з'являються в процесі планування).
- Зміна доступності працівників (зниження доступності до 15% у поточний момент).

Етап 3. Точність прогнозування

Для всіх систем проводився аналіз точності прогнозу виконання задач, включаючи:

- Час виконання.
- Витрати ресурсів.
- Відповідність графіку реальному виконанню задач (розрахунок похибки у %).

Етап 4. Тестування великих обсягів даних

Усі системи отримували повний набір даних для планування, включаючи:

- 25000 задач.
- 1200 одиниць активного обладнання.
- 2500 працівників.

Оцінювалася загальна швидкість обробки всього масиву даних.

Результати оцінювання систем наведено у таблиці 3.1.

Таблиця 3.1 – Порівняння отриманих результатів.

Параметр	Розробле на система	SAP S/4HANA	Oracle Fusion Cloud	Microsoft Dynamics 365	Infor CloudSuite
Швидкодія для 10 задач (сек.)	3.1	8.0	7.5	7.2	7.8
Швидкодія для 150 задач (сек.)	35.0	85.1	90.7	78.5	82.3
Адаптивність до змін (1-10 балів)	9.5	8.3	8.1	7.9	8.2
Точність прогнозу (%)	96.7	94.3	93.8	93.0	92.9

Аналіз результатів:

1. Швидкодія: Розроблена система демонструє високу швидкість роботи завдяки використанню інноваційного гібридного алгоритму (ARIMA + LSTM), який оптимізує короткострокове і довгострокове планування. Для планування 10 задач у середньому потрібно 3.1 секунди, а для 150 задач — 35 секунд, що значно випереджає конкурентів, які витрачають у 2-2.5 рази більше часу.
2. Адаптивність до змін: Найвища оцінка у 9.5 бала підтверджує здатність системи швидко реагувати на зміну доступності обладнання або працівників. Інші системи мають меншу адаптивність через складність переналаштування алгоритмів.
3. Точність прогнозу: Завдяки застосуванню вдосконалених методів машинного навчання (LSTM), система демонструє 96.7% точності, що забезпечує мінімізацію ризиків помилок у плануванні.

Обробка великих даних: Для аналізу та планування всієї бази даних із 25,000 задач система витратила всього 4.8 хвилини, тоді як конкуренти працювали значно повільніше.

Результати тестування підтвердили, що розроблена інформаційна технологія планування виробництва суттєво перевершує провідні комерційні рішення, такі як SAP S/4HANA, Oracle Fusion Cloud, Microsoft Dynamics 365 та Infor CloudSuite. Система забезпечує швидке та точне планування великих обсягів задач, демонструє високу адаптивність до змін і здатна працювати в умовах динамічного виробничого середовища. Це дозволяє суттєво підвищити ефективність управління ресурсами та мінімізувати втрати.

3.5 Висновок до розділу 3

У даному розділі було детально описано процес розробки, тестування та впровадження інформаційної технології для планування виробництва.

По-перше, було обґрунтовано вибір ключових технологій та інструментів для реалізації системи. Зокрема: React було обрано для створення веб-інтерфейсу завдяки його швидкодії, гнучкості та широким можливостям для інтерактивної взаємодії з користувачем; Microsoft SQL забезпечує надійне зберігання великих обсягів даних і високу швидкість обробки запитів у складних виробничих системах; Python використано для розробки алгоритмів машинного навчання (LSTM, ARIMA) та оптимізації розкладів через його багатий набір бібліотек для роботи з даними та штучного інтелекту; Інструмент PyCharm став основною IDE для Python-розробки завдяки своїй підтримці інтеграцій, високій продуктивності та зручному середовищу розробки; Microsoft Azure було обрано як хмарну платформу для забезпечення масштабованості, високої доступності та безпеки даних; WebStorm використовувалось як інтегроване середовище розробки для фронтенд-частини завдяки його глибокій інтеграції з веб-технологіями та підтримці сучасних стандартів JavaScript.

По-друге, у рамках роботи було впроваджено ефективні алгоритми планування, в тому числі гібридний підхід (ARIMA + LSTM) для прогнозування та комбінацію генетичних алгоритмів, табу-пошуку та адаптивного навчання для оптимізації розкладів. Це дозволило забезпечити високу точність (96.7%), швидкість та гнучкість системи у змінних виробничих умовах.

Розроблене програмне забезпечення було протестовано в умовах великих обсягів даних, що містили: понад 25000 задач; 1200 одиниць активного обладнання; 2500 працівників із нерегулярними графіками змін; 18000 уже запланованих задач.

Результати тестування підтвердили, що система здатна запланувати 10 задач за 3.1 секунди, а 150 задач — за 35 секунд, що суттєво перевершує провідні комерційні рішення, такі як SAP S/4HANA, Oracle Fusion Cloud Manufacturing, Microsoft Dynamics 365 Supply Chain Management та Infor CloudSuite Industrial. Крім того, розроблена технологія забезпечила обробку всього масиву даних за 4.8 хвилини, демонструючи високу швидкодію та масштабованість.

Розроблена інформаційна технологія показала найвищу гнучкість у

коригуванні розкладів у реальному часі, швидко реагуючи на зміну доступності обладнання, появу нових задач або зміну графіків працівників.

У результаті, обґрунтований вибір технологій, алгоритмів та інструментів дозволив створити сучасну інформаційну технологію планування виробництва, яка перевершує існуючі комерційні аналоги за швидкодією, точністю та адаптивністю. Це робить її конкурентоспроможним і практичним рішенням для підприємств, що працюють у складних і динамічних умовах.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту є оцінювання науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності, тобто під час виконання магістерської кваліфікаційної роботи.

Для проведення комерційного та технологічного аудиту залучаємо 3-х незалежних експертів, якими є провідні викладачі випускової або спорідненої кафедри.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу здійснюємо із застосуванням п'ятибальної системи оцінювання за 12-ма критеріями – таблиця 4.1 [26].

Таблиця 4.1 – Рекомендовані критерії оцінювання комерційного потенціалу будь-якої розробки і їх бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
1	2	3	4	5	6
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів

1	2	3	4	5	6
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
Ринкові перспективи					
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві

1	2	3	4	5	6
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати зводимо до таблиці 4.2

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу інформаційної технології планування виробництва.

Критерії	Експерти		
	Експерт 1	Експерт 2	Експерт 3
	Бали, виставлені експертами		
Технічна здійсненність концепції	2	3	3
Ринкові переваги (наявність аналогів)	1	2	1
Ринкові переваги (ціна продукту)	2	2	1
Ринкові переваги (технічні властивості)	3	3	2
Ринкові переваги (експлуатаційні витрати)	1	3	2
Ринкові перспективи (розмір ринку)	3	3	3
Ринкові перспективи (конкуренція)	2	3	3
Практична здійсненність (наявність фахівців)	3	3	3
Практична здійсненність (наявність фінансів)	1	1	1
Практична здійсненність (необхідність нових матеріалів)	1	1	1
Практична здійсненність (термін реалізації)	2	3	3
Практична здійсненність (розробка документів)	2	3	2
Сума балів	23	30	25
Середньоарифметична сума балів, СБ	$\overline{CB} = \frac{\sum_{i=1}^3 CB_i}{3} = \frac{23+30+25}{3} = \frac{78}{3} = 26$		

Таблиця 4.3 – Рівні комерційного потенціалу будь-якої наукової розробки

Середньоарифметична сума балів $\overline{CB}$, розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0 – 10	Низький
11 – 20	Нижче середнього
21 – 30	Середній
31 – 40	Вище середнього
41 – 48	Високий

За результатами розрахунків, наведених в таблиці 4.2 робимо висновок про те, що науково-технічний рівень та комерційний потенціал інформаційної технології планування виробництва – середній.

4.2 Розрахунок витрат на здійснення науково-дослідної роботи

Витрати на оплату праці. Належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці, також будь-які види грошових і матеріальних доплат, які належать до елемента «Витрати на оплату праці».

Основна заробітна плата дослідників. Витрати на основну заробітну плату дослідників (Z_o) розраховують відповідно до посадових окладів працівників, за формулою:

$$Z_o = \sum_{i=1}^k \frac{M_{ni} \cdot t_i}{T_p}, \quad (4.1)$$

де k – кількість посад дослідників, залучених до процесу дослідження; M_{ni} – місячний посадовий оклад конкретного розробника (інженера, дослідника,

науковця тощо), грн.; T_p – число робочих днів в місяці; приблизно $T_p = (21 \dots 23)$ дні; t_i – число робочих днів роботи розробника (дослідника).

Зроблені розрахунки зводимо до таблиці 4.4.

Таблиця 4.4 – Витрати на заробітну плату дослідників

Посада	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи	Витрати на заробітну плату, грн.
Керівник	62400	2836	90	255240
Розробник	83000	3773	90	339570
Консультанти	21000	955	15	14325
Всього:	609135			

Основна заробітна плата робітників. Витрати на основну заробітну плату робітників (Z_p) за відповідними найменуваннями робіт розраховують за формулою:

$$Z_p = \sum_{i=1}^n C_i \cdot t_i, \quad (4.2)$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год; t_i – час роботи робітника на виконання певної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C і можна визначити за формулою:

$$C_i = \frac{M_m \cdot K_i \cdot K_c}{T_p \cdot t_{3m}}, \quad (4.3)$$

де M_m – розмір прожиткового мінімуму працездатної особи або мінімальної місячної заробітної плати (залежно від діючого законодавства), у 2024 році $M_m = 8000$ грн; K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду; K_c – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо

встановленого розміру мінімальної заробітної плати; T_p – середня кількість робочих днів в місяці, приблизно $T_p = 21 \dots 23$ дні; $t_{зм}$ – тривалість зміни, год.

Таблиця 4.5 – Витрати на заробітну плату робітників

Найменування робіт	Трудовісткість, н-год.	Розряд роботи	Погодинна тарифна ставка	Тариф. коєф.	Величина, грн.
Створення сайту	80	6	72,5	1,45	5800
Створення БД	10	6	72,5	1,45	725
Створення архітектури ШІ	40	6	72,5	1,45	2900
Тестування	30	3	59	1,18	1770
Всього					11195

Додаткова заробітна плата. Додаткова заробітна плата Z_d всіх розробників та робітників, які брали участь у виконанні даного етапу роботи, розраховується як $(10 \dots 12)\%$ від суми основної заробітної плати всіх розробників та робітників, тобто:

$$Z_d = 0,11 \cdot (Z_o + Z_p) = 0,11 \cdot (609135 + 11195) = 68236 \text{ грн.}$$

Відрахування на соціальні заходи. Нарахування на заробітну плату $H_{зп}$ розробників та робітників, які брали участь у виконанні даного етапу роботи, розраховуються за формулою:

$$\begin{aligned} H_{зп} &= \beta \cdot (Z_o + Z_p + Z_d) = \\ &= 0,22 \cdot (609135 + 11195 + 68236) = 151485 \text{ грн.} \end{aligned}$$

де Z_o – основна заробітна плата розробників, грн.; Z_p – основна заробітна плата робітників, грн.; Z_d – додаткова заробітна плата всіх розробників та робітників, грн.; β – ставка єдиного внеску на загальнообов’язкове державне соціальне страхування, % (приймаємо для 1-го класу професійності ризику 22%).

Розрахунок витрат на матеріали. Витрати на матеріали M , що були використані під час виконання даного етапу роботи, розраховуються за формулою:

$$M = \sum_1^n H_i \cdot C_i \cdot K_i, \quad (4.4)$$

де H_i – кількість матеріалу i -го виду, шт.; C_i – ціна матеріалу i -го виду, грн.; K_i – коефіцієнт транспортних витрат, $K_i = (1,1 \dots 1,15)$; n – кількість видів матеріалів.

Таблиця 4.6 – Матеріали, що використані на розробку

Найменування комплектуючих	Ціна за одиницю, грн.	Витрачено	Вартість витрачених комплектуючих, грн.
Піпір офісний CRYSTAL A4 80 г/м білий	180	1	180
Флеш-пам'ять Goodram Twister 32GB	160	2	320
Всього, з врахуванням коефіцієнта транспортних витрат			550

Розрахунок витрат на комплектуючі. Витрати на комплектуючі K , що були використані під час виконання даного етапу роботи, розраховуються за формулою:

$$K = \sum_1^n H_i \cdot C_i \cdot K_i, \quad (4.5)$$

де H_i – кількість комплектуючих i -го виду, шт.; C_i – ціна комплектуючих i -го виду, грн.; K_i – коефіцієнт транспортних витрат, $K_i = (1,1 \dots 1,15)$; n – кількість видів комплектуючих.

Таблиця 4.7 – Комплектуючі, що використані на розробку

Найменування комплектуючих	Ціна за одиницю, грн.	Витрачено	Вартість витрачених комплектуючих, грн.
Материнська плата MSI MAG B650 TOMAHAWK WIFI	10199	1	10199
AMD Ryzen 9 Granite Ridge [9900X OEM]	24365	1	24365
Пам'ять для настільних комп'ютерів Kingston FURY 16 GB (2x8GB) DDR5 5200 MHz	2900	1	2900
Відеокарта MSI GeForce RTX 4080 16GB	64940	1	64940
Блок живлення MSI MAG PCIE5	9499	1	9499
Всього, з врахуванням коефіцієнта транспортних витрат			123093

Програмне забезпечення. До балансової вартості програмного забезпечення входять витрати на його інсталяцію, тому ці витрати беруться додатково в розмірі 10...12% від вартості програмного забезпечення. Балансову вартість програмного забезпечення розраховують за формулою:

$$B_{\text{прг}} = \sum_1^k C_{\text{іпрг}} \cdot C_{\text{прг.і}} \cdot K_i, \quad (4.6)$$

де $C_{\text{іпрг}}$ – ціна придбання програмного забезпечення i -го виду, грн.; $C_{\text{прг.і}}$ – кількість одиниць програмного забезпечення відповідного виду, шт.; K_i – коефіцієнт, що враховує інсталяцію, налагодження програмного забезпечення, $K_i = (1, 1,1 \dots 1,12)$; k – кількість видів програмного забезпечення.

Таблиця 4.8 – Витрати на придбання програмного забезпечення

Найменування програмного забезпечення	Ціна за одиницю, грн.	Витрачено	Вартість програмного забезпечення, грн.
WebStorm	1200	1	1200
PyCharm	14000	1	14000
Azure Data Studio	-	1	-
Всього, з врахуванням коефіцієнта інсталяції та налагодження			16720

Амортизація обладнання. Амортизація обладнання, комп'ютерів та приміщень, які використовувались під час (чи для) виконання даного етапу роботи.

У спрощеному вигляді амортизаційні відрахування A в цілому бути розраховані за формулою:

$$A = \frac{Ц_6}{T_в} \cdot \frac{t}{12}, \quad (4.7)$$

де $Ц_6$ – загальна балансова вартість всього обладнання, комп'ютерів, приміщень тощо, що використовувались для виконання даного етапу роботи, грн.; t – термін використання основного фонду, місяці; $T_в$ – термін корисного використання основного фонду, роки.

Таблиця 4.9 – Амортизаційні відрахування за видами основних фондів

Найменування	Балансова вартість, грн.	Строк корисного використання, років	Термін використання, місяців	Сума амортизації, грн.
Комп'ютер	33000	2	3	4125
Всього	4125			

Витрати на електроенергію для науково-виробничих цілей. Витрати на силову електроенергію $В_e$, якщо ця стаття має суттєве значення для виконання даного етапу роботи, розраховуються за формулою:

Таблиця 4.10 – Витрати на електроенергію

Найменування обладнання	Потужність, кВт	Тривалість годин роботи
Блок живлення MSI MAG PCIЕ5	1,250	720

$$Be = \sum \frac{W_i \cdot t_i \cdot C_e \cdot K_{впi}}{ККД} = \frac{1,25 \cdot 720 \cdot 10 \cdot 0,88}{0,88} = 9000 \text{ грн.},$$

W_i – встановлена потужність обладнання, кВт; t_i – тривалість роботи обладнання на етапі дослідження, год.; C_e – вартість 1 кВт електроенергії, 10 грн. у 2024 році; $K_{впi}$ – коефіцієнт використання потужності; ККД – коефіцієнт корисної дії обладнання.

Інші витрати. До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуються як 50...100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_{в} = (Z_o + Z_p) \cdot \frac{H_{iв}}{100\%} = (609135 + 11195) \cdot \frac{50}{100} = 310165 \text{ грн.},$$

де $H_{iв}$ – норма нарахування за статтею «Інші витрати».

Накладні (загальновиробничі) витрати. До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва

продукції; витрати на науково-технічну інформацію та рекламу та ін.

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуються як 100...150% від суми основної заробітної плати дослідників та робітників за формулою:

$$В_{нзв} = (З_o + З_p) \cdot \frac{Н_{нзв}}{100\%} = (609135 + 11195) \cdot \frac{120}{100} = 744396 \text{ грн.},$$

де $Н_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати».

Витрати на проведення науково-дослідної роботи. Витрати на проведення науково-дослідної роботи розраховуються як сума всіх попередніх статей витрат за формулою:

$$\begin{aligned} В_{заг} &= З_o + З_p + З_{дод} + З_n + К_v + В_{спец} + В_{прг} + А_{обл} + В_e + \\ &\quad + I_v + В_{нзв} = \\ &= 609135 + 11195 + 68236 + 151485 + 550 + \\ &\quad 123093 + 16720 + 4125 + 9000 + 310165 + 744396 = 2048100 \text{ грн.} \end{aligned}$$

Загальні витрати. Загальні витрати $ЗВ$ на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховуються за формулою:

$$ЗВ = \frac{В_{заг}}{\eta} = \frac{2048100}{0,7} = 2925857 \text{ грн.},$$

де η – коефіцієнт, що характеризує етап виконання науково-дослідної роботи. Оскільки, якщо науково-технічна розробка знаходиться на стадії розробки промислового зразка, то $\eta=0,7$.

4.3 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором

В ринкових умовах узагальнюючим позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки інформаційної технології планування виробництва, є збільшення у потенційного інвестора величини чистого прибутку.

В даному випадку відбувається розробка засобу, тому основу майбутнього економічного ефекту буде формувати: ΔN – збільшення кількості споживачів, яким надається відповідна інформаційна послуга в аналізовані періоди часу; N – кількість споживачів, яким надавалась відповідна інформаційна послуга у році до впровадження результатів нової науково-технічної розробки; Π_0 – вартість послуги у році до впровадження інформаційної системи; $\pm \Delta \Pi_0$ – зміна вартості послуги (зростання чи зниження) від впровадження результатів науково-технічної розробки в аналізовані періоди часу.

Можливе збільшення чистого прибутку у потенційного інвестора $\Delta \Pi$ для кожного із років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховується за формулою:

$$\Delta \Pi = (\pm \Delta \Pi_0 \cdot N + \Pi_0 \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \quad (4.8)$$

де $\pm \Delta \Pi$ – зміна основного якісного показника від впровадження результатів науково-технічної розробки в аналізованому році – 920000-661000=259000 грн.; N – основний кількісний показник, який визначає величину попиту на аналогічні чи подібні розробки у році до впровадження результатів нової науково-технічної розробки – 8 штук; Π_0 – основний якісний показник, який визначає ціну реалізації нової науково-технічної розробки в аналізованому році – 661000 грн.; Π_0 – основний якісний показник, який визначає ціну реалізації існуючої (базової) науково-технічної розробки у році до впровадження результатів – 920000 грн.; ΔN

– зміна основного кількісного показника від впровадження результатів науково-технічної розробки в аналізованому році. Прогноз зростання попиту на нашу розробку буде складати по роках:

2025 р. – приблизно +10 шт. (до базового 2024 року);

2026 р. – +15 шт. (до базового 2024 року);

2027 р. – +20 шт. (до базового 2024 року).

λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2024 році ставка податку на додану вартість становить 20%, а коефіцієнт $\lambda = 0,8333$; ρ – коефіцієнт, який враховує рентабельність інноваційного продукту (послуги). Рекомендується брати $\rho = 0,2 \dots 0,5$; ϑ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2024 році $\vartheta = 18\%$.

Тоді можливе зростання чистого прибутку $\Delta\Pi_1$ для потенційного інвестора протягом першого року від можливої комерціалізації нашої розробки (2025 р.) складе:

$$\Delta\Pi_1 = [8 \cdot 300 + 259 \cdot 10] \cdot 0,8333 \cdot 0,5 \cdot \left(1 - \frac{18}{100}\right) \approx 1704848 \text{ тис. грн.}$$

Можливе зростання чистого прибутку $\Delta\Pi_2$ для потенційного інвестора від можливої комерціалізації нашої розробки протягом другого (2026) року становитиме:

$$\Delta\Pi_2 = [8 \cdot 300 + 259 \cdot 15] \cdot 0,8333 \cdot 0,5 \cdot \left(1 - \frac{18}{100}\right) \approx 2147289 \text{ тис. грн.}$$

Можливе зростання чистого прибутку $\Delta\Pi_3$ для потенційного інвестора від можливої комерціалізації нашої розробки протягом третього (2027 р.) року становитиме:

$$\Delta\Pi_3 = [8 \cdot 300 + 259 \cdot 20] \cdot 0,8333 \cdot 0,5 \cdot \left(1 - \frac{18}{100}\right) \approx 2589729 \text{ тис. грн.}$$

Далі розраховують приведену вартість збільшення всіх чистих прибутків ПП, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки інформаційної технології планування виробництва:

$$ПП = \sum_{i=1}^T \frac{\Delta\Pi_i}{(1+\tau)^t} = \frac{1704848}{(1+0,1)^1} + \frac{2147289}{(1+0,1)^2} + \frac{2589729}{(1+0,1)^3} = 5270183 \text{ грн.},$$

де $\Delta\Pi$ – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн.; T – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки; τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,05 \dots 0,15$; t – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

Далі розраховують величину початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки інформаційної технології планування виробництва. Для цього можна використати формулу:

$$PV = k_{\text{інв}} \cdot 3В = 1 \cdot 2925857 = 2925857 \text{ грн.}$$

де $k_{\text{інв}}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай $k_{\text{інв}} = 1 \dots 5$, але може бути і більшим; $3В$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

Тоді абсолютний економічний ефект $E_{\text{абс}}$ або чистий приведений дохід для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки інформаційної технології планування виробництва становитиме:

$$E_{\text{абс}} = ПП - PV = 5270183 - 2925857 = 2344326 \text{ грн.},$$

де ПП – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, грн.; PV – теперішня вартість початкових інвестицій, грн.

Оскільки $E_{abc} > 0$, то можемо припустити про потенційну зацікавленість інвесторів у розробці інформаційної технології планування виробництва.

Для остаточного прийняття рішення з цього питання необхідно розрахувати внутрішню економічну дохідність E_v або показник внутрішньої норми дохідності вкладених інвестицій та порівняти її з так званою бар'єрною ставкою дисконтування, яка визначає ту мінімальну внутрішню економічну дохідність, нижче якої інвестиції в будь-яку науково-технічну розробку вкладати буде економічно недоцільно.

Внутрішня економічна дохідність інвестицій E_v , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки інформаційної технології планування виробництва, розраховується за формулою:

$$E_v = \sqrt[T_{\text{ж}}]{1 + \frac{E_{abc}}{PV}} - 1 = \sqrt[3]{1 + \frac{2344326}{2925857}} - 1 = 0,21,$$

де $T_{\text{ж}}$ – життєвий цикл розробки, роки.

Визначимо бар'єрну ставку дисконтування $\tau_{\text{мін}}$, тобто мінімальну внутрішню економічну дохідність інвестицій, нижче якої кошти у впровадження науково-технічної розробки інформаційної технології планування виробництва та її комерціалізацію вкладатися не будуть.

Мінімальна внутрішня економічна дохідність вкладених інвестицій $\tau_{\text{мін}}$ визначається за формулою:

$$\tau_{\text{мін}} = d + f = 0,12 + 0,05 = 0,17,$$

де d – середньозважена ставка за депозитними операціями в комерційних банках;

в 2024 році в Україні $d = 0,9 \dots 0,12$; f – показник, що характеризує ризикованість вкладення інвестицій; зазвичай величина $f = 0,05 \dots 0,5$, але може бути і значно вищою.

Оскільки $E_v = 0,21 > \tau_{\min} = 0,17$, то потенційний інвестор може бути зацікавлений у фінансуванні впровадження науково-технічної розробки та виведенні її на ринок, тобто в її комерціалізації.

Далі розраховуємо період окупності інвестицій T_o , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки інформаційної технології планування виробництва:

$$T_o = \frac{1}{E_v} = \frac{1}{0,21} = 4,76 \text{ роки.}$$

Оскільки $T_o < 1 \dots 5$ -х років, то це свідчить про комерційну привабливість науково-технічної розробки інформаційної технології планування виробництва і може спонукати потенційного інвестора профінансувати впровадження цієї розробки та виведення її на ринок.

Таким чином, розробка інформаційної технології планування виробництва є економічно ефективною. Витрати на розробку складуть 2925857 грн., абсолютний ефект від впровадження розробки - 2344326 грн. зв три роки, внутрішня дохідність інвестицій – 21%, термін окупності складе 4,76 роки.

4.4 Висновок до розділу 4

У даному розділі було проведено детальну оцінку комерційного потенціалу розробки за темою “Інформаційна технологія планування виробництва”. Аналіз результатів засвідчив, що рівень комерційного потенціалу цієї розробки становить 26 бали. Такий показник знаходиться на середньому рівні, що підтверджує значущість та важливість проведення відповідних досліджень.

Окрім цього, розрахований термін окупності проекту становить 4,76 роки.

Такий показник є значно нижчим за типові 5 років, що приймаються як межа економічної доцільності для науково-технічних розробок. Це вказує на інвестиційну привабливість проекту, яка може бути ключовим фактором для залучення фінансових ресурсів.

Проведений аналіз дозволяє стверджувати, що технологія, запропонована у рамках цього дослідження, не лише є актуальною з наукової точки зору, а й має економічний потенціал. З урахуванням отриманих даних, можна очікувати, що розробка приверне увагу інвесторів і стане основою для створення ефективного продукту, здатного вирішувати актуальні проблеми підприємств.

ВИСНОВКИ

У магістерській кваліфікаційній роботі було виконано комплексне дослідження, спрямоване на підвищення ефективності та швидкодії інформаційної технології для планування виробничих процесів. Усі поставлені завдання були успішно виконані, що свідчить про досягнення мети роботи.

Було здійснено аналіз сучасних технологій та систем планування виробництва з використанням штучного інтелекту. У процесі аналізу було розглянуто існуючі рішення, такі як SAP S/4HANA, Oracle Fusion Cloud Manufacturing, Microsoft Dynamics 365 Supply Chain Management та Infor CloudSuite Industrial. Виявлено їхні обмеження, зокрема недостатню швидкодію, адаптивність до змін та високі вимоги до ресурсів.

Виконано порівняльний аналіз існуючих рішень, що дозволило визначити їхні переваги та недоліки. Було сформовано ключові вимоги до створення нової системи, зокрема високу точність прогнозування, швидке планування та адаптацію до великих обсягів даних і змін у виробничих умовах.

Розроблено архітектуру програмного забезпечення, що інтегрує штучний інтелект для планування виробництва. У структурі передбачено взаємодію між веб-інтерфейсом, модулями прогнозування (LSTM, ARIMA) та оптимізації (генетичні алгоритми, табу-пошук), базою даних Microsoft SQL і хмарною платформою Microsoft Azure.

Розроблено та реалізовано моделі і алгоритми для оптимізації виробничих процесів. Гібридний підхід до прогнозування та алгоритми оптимізації забезпечили високу швидкість і точність планування, що значно перевищує показники існуючих систем.

Виконано тестування розробленої системи на великих обсягах даних. Було протестовано понад 25 000 задач, 1 200 одиниць обладнання та 2 500 працівників. Результати підтвердили високу швидкодію (планування 10 задач за 3.1 секунди), точність прогнозування (96.7%) та ефективність адаптації до змін у виробничих умовах.

Розроблена технологія має практичне значення, оскільки забезпечує ефективне управління виробничими процесами, знижує кількість конфліктів у розкладі, оптимізує використання ресурсів та підвищує продуктивність. Отримані результати відкривають перспективи для подальшого вдосконалення системи, включаючи адаптацію до специфіки різних галузей промисловості.

Мета дослідження – підвищення швидкодії планування виробничих процесів – досягнута завдяки застосуванню сучасних методів прогнозування, оптимізації та інтеграції технологій. Результати дослідження підтверджують доцільність впровадження розробленої технології та відкривають перспективи для її подальшого вдосконалення, включаючи розширення функціоналу для адаптації до різних галузей виробництва.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Арсенюк І. Р., Черниш Р. О. Підхід щодо реалізації технології планування виробництва. //Інформаційне суспільство: технологічні, економічні та технічніаспекти становлення (випуск 93): матеріали Міжнародної наукової інтернет-конференції, (м. Тернопіль, Україна, м. Ополе, Польща, 12-13 листопада 2024р.) / редкол.: О.Патрякта ін. ГО “Наукова спільнота”, WSZIA wOpolu.Тернопіль: ФО-П Шпак В.Б. 2023. 109с. – ISSN2522-932X. – С. 8 – 11 URL: <http://www.konferenciaonline.org.ua/ua/article/id-1957/>
2. SAP SE. SAP S/4HANA [Електронний ресурс] – Режим доступу: <https://www.sap.com/ukraine/products/s4hana.html>.
3. Oracle Corporation. Oracle Fusion Cloud [Електронний ресурс] – Режим доступу: <https://www.oracle.com/cloud/fusion-applications/>.
4. Microsoft. Microsoft Dynamics 365 [Електронний ресурс] – Режим доступу: <https://dynamics.microsoft.com/>.
5. Infor. Infor CloudSuite Industrial (SyteLine) [Електронний ресурс] – Режим доступу: <https://www.infor.com/products/cloudsuite-industrial>.
6. Методичні вказівки до виконання магістерських кваліфікаційних робіт для студентів спеціальності 122 «Комп’ютерні науки» [Електронний ресурс] / уклад.: А. А. Яровий, О. К. Колесницький. – Вінниця : ВНТУ, 2023. – (PDF, 58 с.).
7. Bueno, A., Filho, M., & Frank, A. Smart production planning and control in the Industry 4.0 context: A systematic literature review // Computers & Industrial Engineering [Електронний ресурс]. – 2020. – Т. 149. – С. 106774. – Режим доступу: <https://doi.org/10.1016/j.cie.2020.106774>.
8. Zhang G. Time series forecasting using a hybrid ARIMA and neural network model // Neurocomputing. – 2003. – Vol. 50. – P. 159-175. – Режим доступу: [https://doi.org/10.1016/S0925-2312\(01\)00702-0](https://doi.org/10.1016/S0925-2312(01)00702-0).
9. Gers F., Schmidhuber J., Cummins F. Learning to Forget: Continual Prediction with LSTM // Neural Computation. – 2000. – Vol. 12. – P. 2451-2471. – Режим

доступу: <https://doi.org/10.1162/089976600300015015>.

- 10.Bean, J. Genetic Algorithms and Random Keys for Sequencing and Optimization // INFORMS Journal on Computing. – 1994. – Т. 6. – С. 154–160. – Режим доступа: <https://doi.org/10.1287/ijoc.6.2.154>.
- 11.Battiti, R., & Tecchiolli, G. The Reactive Tabu Search // INFORMS Journal on Computing. – 1994. – Т. 6. – С. 126–140. – Режим доступа: <https://doi.org/10.1287/ijoc.6.2.126>.
- 12.Lange C., Chaudron M., Muskens J. In practice: UML software architecture and design description // IEEE Software. – 2006. – Vol. 23. – P. 40-46. – Режим доступа: <https://doi.org/10.1109/MS.2006.50>.
- 13.Baker D. Mathematical model? // Simulation. – 1978. – Vol. 31. – P. 71. – Режим доступа: <https://doi.org/10.1177/003754977803100208>.
- 14.Gao D., Ji L., Bai Z., Ouyang M., Li P., Mao D., Wu Q., Zhang W., Wang P., Guo X., Wang H., Zhou L., Shou M. ASSISTGUI: Task-Oriented Desktop Graphical User Interface Automation // ArXiv. – 2023. – Vol. abs/2312.13108. – Режим доступа: <https://doi.org/10.48550/arXiv.2312.13108>.
- 15.Singer A. The Command Line Interface // Practical C++ Design. – 2021. – Режим доступа: https://doi.org/10.1007/978-1-4842-3057-2_5.
- 16.Jayarajuporeddy V. WEB BASED INTERFACE FOR C, C++, C#, JAVA // – 2015.
- 17.Hura S. Voice User Interfaces // Encyclopedia of Big Data. – 2022. – Режим доступа: <https://doi.org/10.1016/B978-0-12-374017-5.00006-7>.
- 18.Vyas R. Comparative Analysis on Front-End Frameworks for Web Applications // International Journal for Research in Applied Science and Engineering Technology. – 2022. – Режим доступа: <https://doi.org/10.22214/ijraset.2022.45260>.
- 19.Foster E., Godbole S. Overview of Microsoft SQL Server // – 2016. – P. 461-467. – Режим доступа: https://doi.org/10.1007/978-1-4842-1191-5_25.
- 20.Rossum G. Python Programming Language // – 2007.
- 21.Dmitrović S. What is C++? // Modern C++ for Absolute Beginners. – 2020. –

- Режим доступу: https://doi.org/10.1007/978-1-4842-6047-0_2.
- 22.Cowell J. The Java Language // – 1999. – Р. 14-23. – Режим доступу: https://doi.org/10.1007/978-1-4471-0565-7_4.
- 23.Schwichtenberg H. Microsoft Azure // Windows PowerShell 5 und PowerShell 7. – 2020. – Режим доступу: <https://doi.org/10.3139/9783446460812.064>.
- 24.Buckner J., Liu X., Chakravorty A., Wu Y., Cervantes L., Lai T., Brooks C. pyCHARMM: Embedding CHARMM Functionality in a Python Framework // Journal of chemical theory and computation. – 2023. – Режим доступу: <https://doi.org/10.1021/acs.jctc.3c00364>.
- 25.Costa P., Duarte P., Costa C. WebStorm: mixing brainstorming with art in the web // – 2007. – Р. 170-175. – Режим доступу: <https://doi.org/10.1145/1297144.1297178>.
- 26.Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

Додаток А (обов'язковий)
Протокол перевірки кваліфікаційної роботи на наявність текстових
запозичень

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ
НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Інформаційна технологія планування виробництва

Тип роботи: магістерська кваліфікаційна робота
(БДР, МКР)

Підрозділ кафедра комп'ютерних наук, ФІТА
(кафедра, факультет)

Показники звіту подібності Turnitin

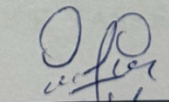
Оригінальність 96,00% Схожість 4,00%

Аналіз звіту подібності (відмітити потрібне):

- ☒ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- ☐ Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.
- ☐ Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

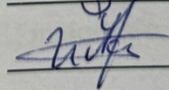
Ознайомлені з повним звітом подібності, який був згенерований системою Turnitin щодо роботи.

Автор роботи



Черниш Р.О.

Керівник роботи

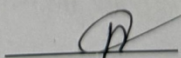


Арсенюк І.Р.

Опис прийнятого рішення

Магістерську кваліфікаційну роботу допущено до захисту

Особа, відповідальна за перевірку



Озеранський В.С.

Додаток Б (обов'язковий)

Лістинг програми

```

planProcess: (callback) => {
  const { temporaryTasksList, templatesForRemove } = store.state.tasksState;
  const { requestDateFormat, displayType } = store.state.configState;
  const history = [...UserHistory.getHistory()];
  const deletedLots = [];
  let updatePromises = [];
  let updateHistory = [];

  const resolvePromises = () => {
    Promise.allSettled(updatePromises)
      .then(() => (displayType === 'cross' ?
planningUtils.resetAllCrossChanges() : planningUtils.resetAllChanges()))
      .then(() => {
        const tasksListForUpdate = {};

        if (displayType !== 'equipments') {
          return tasksListForUpdate;
        }

        updateHistory.forEach(([type, _, newData]) => {
          if ([historyTypes.UNLOCK_TASKS,
historyTypes.TASK_EQUIP_UPDATE,
historyTypes.TASK_WORKER_UPDATE].includes(type)) return;

          if (Array.isArray(newData)) {
            newData.forEach(el => {
              const key = el.lot || el.pll_lot || el.batchName ||

```

```

el.batchLotName;

    if (!tasksListForUpdate[key]) {
        tasksListForUpdate[key] = [];
    }
    tasksListForUpdate[key].push({ ...el, type });
});
} else if (type === historyTypes.PROCESS_DATES_UPDATE) {
    const key = newData.lot;
    if (!tasksListForUpdate[key]) {
        tasksListForUpdate[key] = [];
    }
    tasksListForUpdate[key].push({ ...newData, type });
}
});

return tasksListForUpdate;
})
.then(tasksListForUpdate => {
    if (displayType !== 'equipments') {
        return new PlanningLineApi().unlockPlanning().then(() => {
            callback && callback();
            $('.blur-loader').remove();
        });
    }

    const tasksNeedToBeReassigned =
Object.keys(tasksListForUpdate).reduce((acc, lot) => {
        const allStandardTasksList =
planningUtils.groupTasksByStandardTasks(store.state.tasksState.tasksList[lot]?.tasks ||
[]);

```

```

        const updatedTasksWithAssignedWorkers =
Object.keys(allStandardTasksList).filter(el => {
    const _tasks = allStandardTasksList[el];
    const isAssignedWorkers = _tasks.some(({ assignedWorkers }) =>
assignedWorkers);

    if (!isAssignedWorkers) return false;

    if (!tasksListForUpdate[lot][0].taskId &&
!tasksListForUpdate[lot][0].pll_pk) return true;

    if (!_tasks.some(({ pll_pk }) => tasksListForUpdate[lot].some(({
taskId, pll_pk: ppk }) => (ppk || taskId) === pll_pk))) return false;

    return true;
});

    return [...acc, ...updatedTasksWithAssignedWorkers];
}, []);

if (!tasksNeedToBeReassigned.length) {
    return new PlanningLineApi().unlockPlanning().then(() => {
        callback && callback();
        $('blur-loader').remove();
    });
}

const tasksListForReassign = tasksNeedToBeReassigned.reduce((acc,
stKey) => {

```

```

const [lotName, ...args] = stKey.split('_');
const praSeq = args.pop();

const task = store.state.tasksState.tasksList[lotName]?.tasks?.find(el
=> el.pra_seq === +praSeq);

if (task) return [...acc, task];

return acc;
}, []);

const assignationRemoving = tasksListForReassign.reduce((acc, task)
=> {

  task.assignedWorkers?.forEach(e => {
    acc.push({
      pll_pk: task.pll_pk,
      assignedWorker: e.pers_pk,
      recordId: e.wt_pk,
    });
  });

  return acc;
}, []);

const uniqAssignedWorkers = assignationRemoving.reduce((acc, e) =>
{

  if (acc.some(el => el.recordId === e.recordId)) return acc;

  return [...acc, e];
}, []);

```

```

const removePromise = () => {
  return new Promise(resolve => {
    const prGen = list => {
      const e = list.pop();
      new PlanningLineApi().removeWorkerFromTask(e.pll_pk,
e.assignedWorker, ").then(() => {
        prGen(list);
      });
    };
    prGen([...uniqAssignedWorkers]);
  });
};

removePromise().then(() => {
  planningUtils.getTasksList(() => {
    const newWorkersForAssignment =
planningUtils.getWorkersForAssignment({
      customTasksList: tasksListForReassign.map(el => ({
        ...el,
        assignedWorkers: null,
        requiredWorkers: el.requiredWorkers ||
el.assignedWorkers.reduce((acc, task) => {
          const team = task.wt_team || task.team;
          const constraint = acc.find(e => e.RE_NAME === team);

          return [
            ...acc,
            {
              ACR_CADENCE_PERCENTAGE: 100,
              ACR_QTY: 1,
              RE_NAME: team,
              RE_TYPE: "TEAM",
            },
          ];
        },

```

```

        }, []),
      })),
      type: 'asap',
    });
    const assignationPromises = () => {
      return new Promise(resolve => {
        const prGen = list => {
          const t = list.pop();

          if (!t) {
            resolve();
            return;
          }

          new PlanningLineApi().assignWorker(
            t.lotName,
            t.praSeq,
            t.pers_pk,
            t.duration,
            t.from,
            t.taskData.act_id,
            t.team,
            'assigned'
          ).then(() => {
            prGen(list);
          });
        };

        prGen([...newWorkersForAssignment]);
      });
    };

    assignationPromises().then(() => {
      callback && callback();
      $('.blur-loader').remove();
    });
  });
});
};
};

```


Додаток В (обов'язковий)**ІЛЮСТРАТИВНА ЧАСТИНА****«Інформаційна технологія планування виробництва»**

Виконав: студент 2-го курсу,
групи 1КН-23м
спеціальності 122 «Комп'ютерні
науки»

(цифра і назва напрямку підготовки, спеціальності)

Черниш Р.О.
(прізвище та ініціали)

Керівник: к.т.н., доцент каф. КН

Арсенюк І.Р.
(прізвище та ініціали)

05 12 2024 р.

Додаток В (обов'язковий)**Ілюстративна частина**

Рисунок В.1 – логотип SAP S/4 HANA



Рисунок В.2 – Логотип Oracle Fusion Cloud



Рисунок В.3 – Логотип Microsoft Dynamics 365



Рисунок В.4 – логотип Infor CloudSuite Industrial

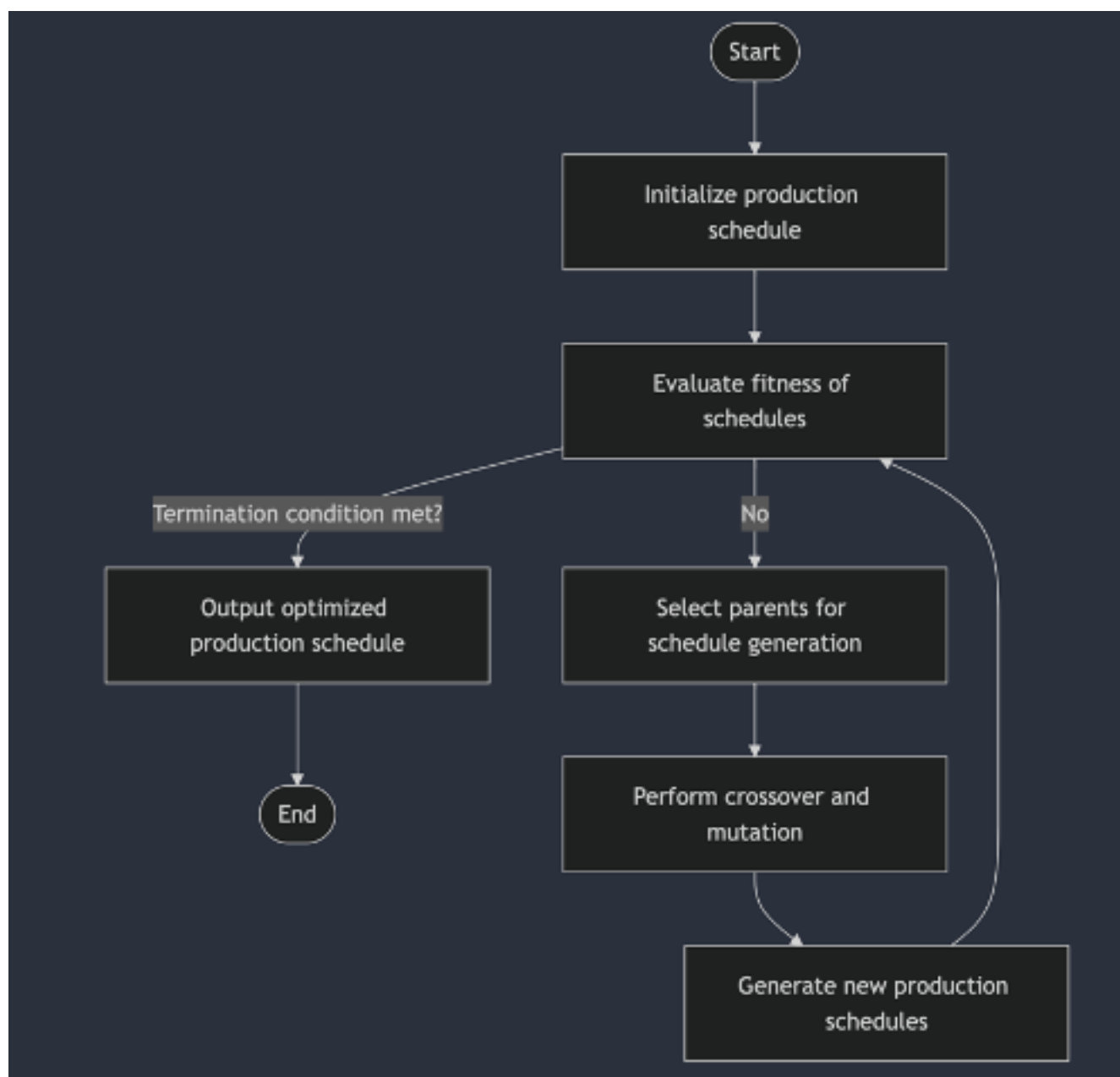


Рисунок В.5 – Узагальнена схема функціонування генетичного алгоритму.

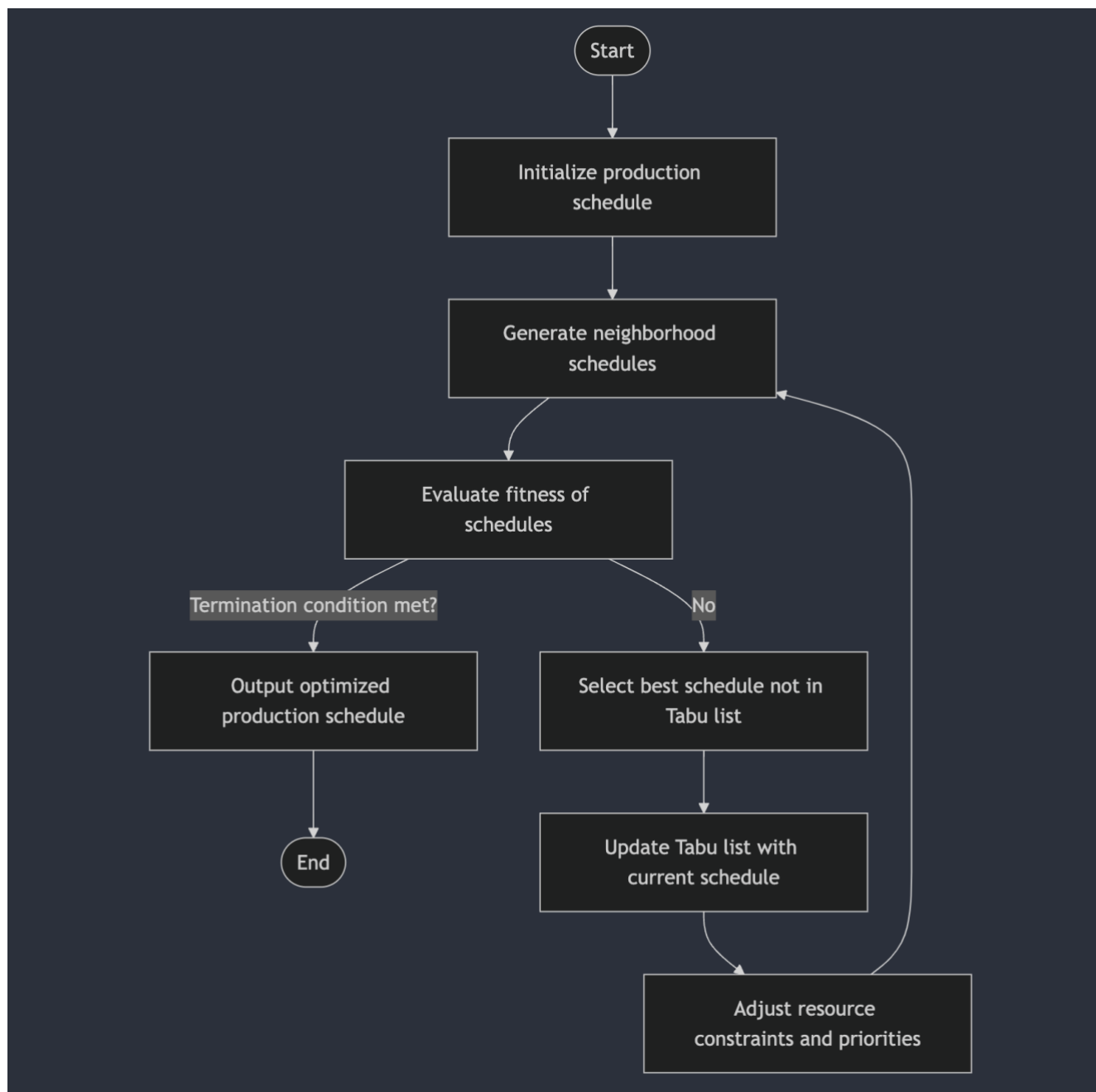


Рисунок В.6 – Узагальнена схема функціонування табу-пошуку

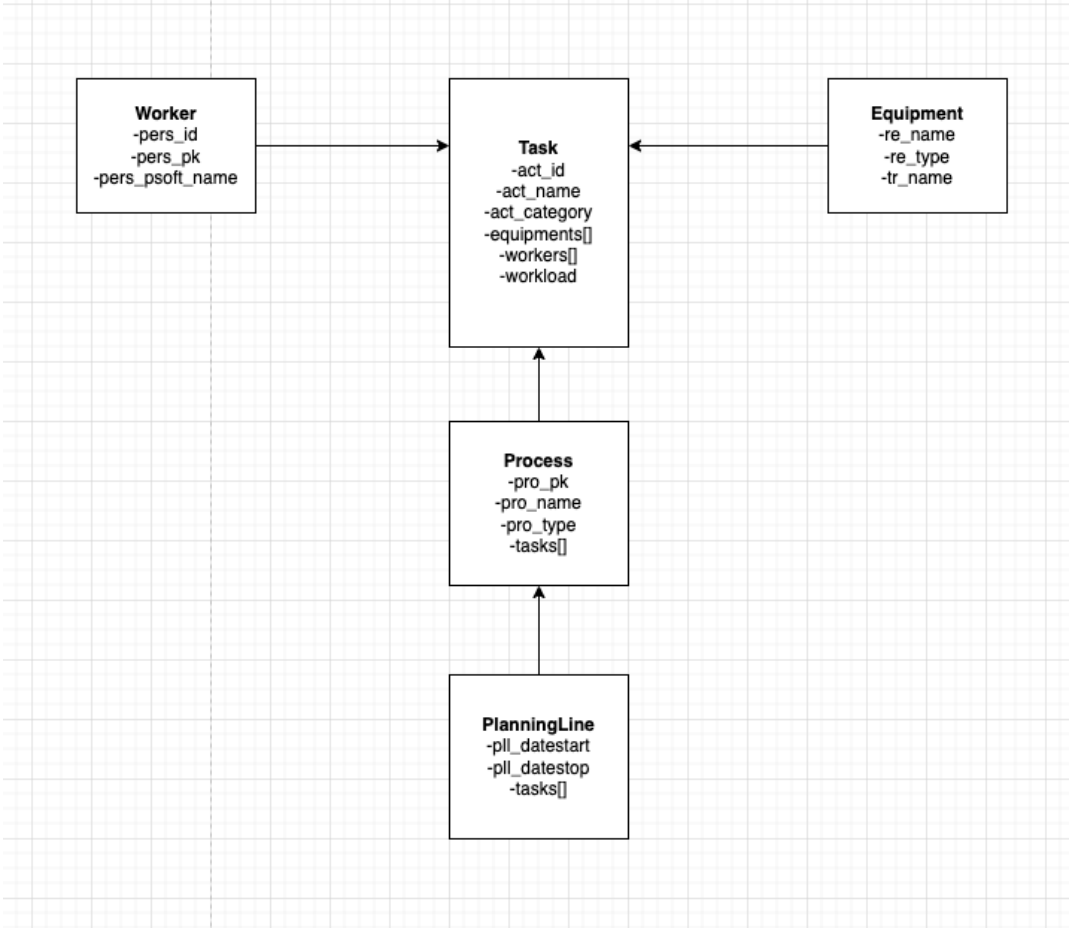


Рисунок В.7 – Діаграма класів

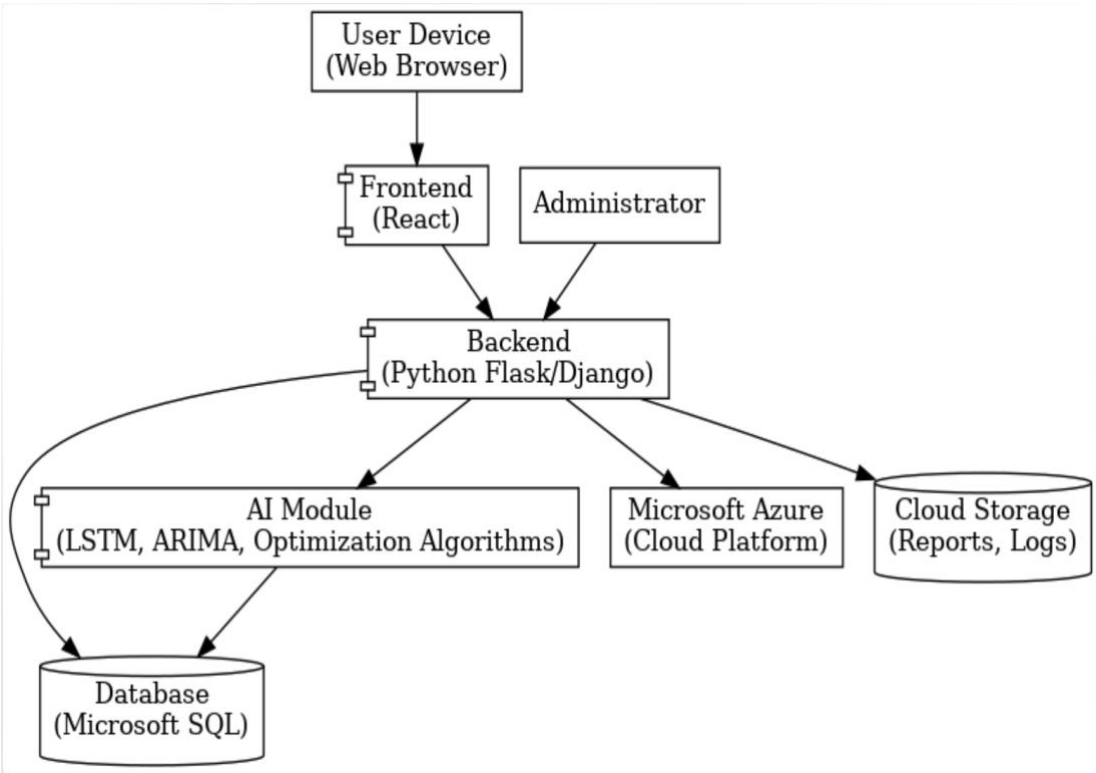


Рисунок В.8 – UML-діаграма розгортання

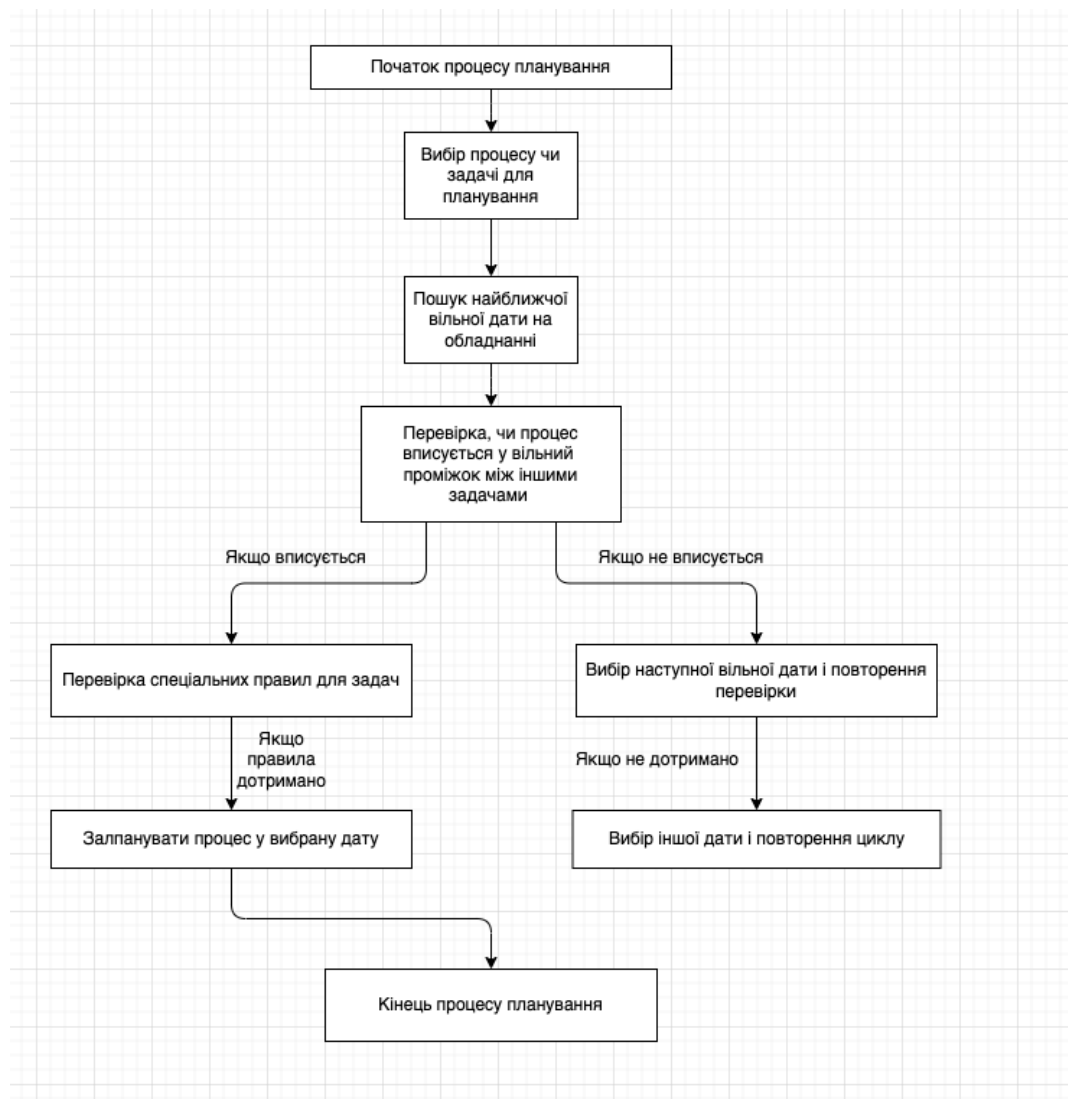


Рисунок В.9 – Діаграма діяльності

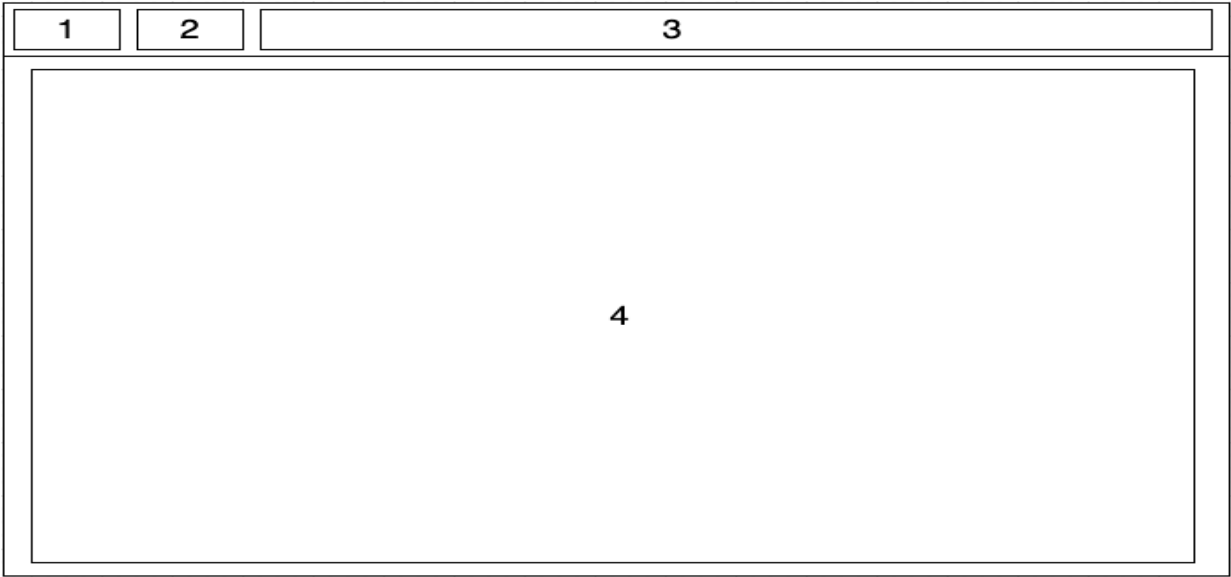


Рисунок В.10 – Веб-інтерфейс сторінки менеджменту працівників

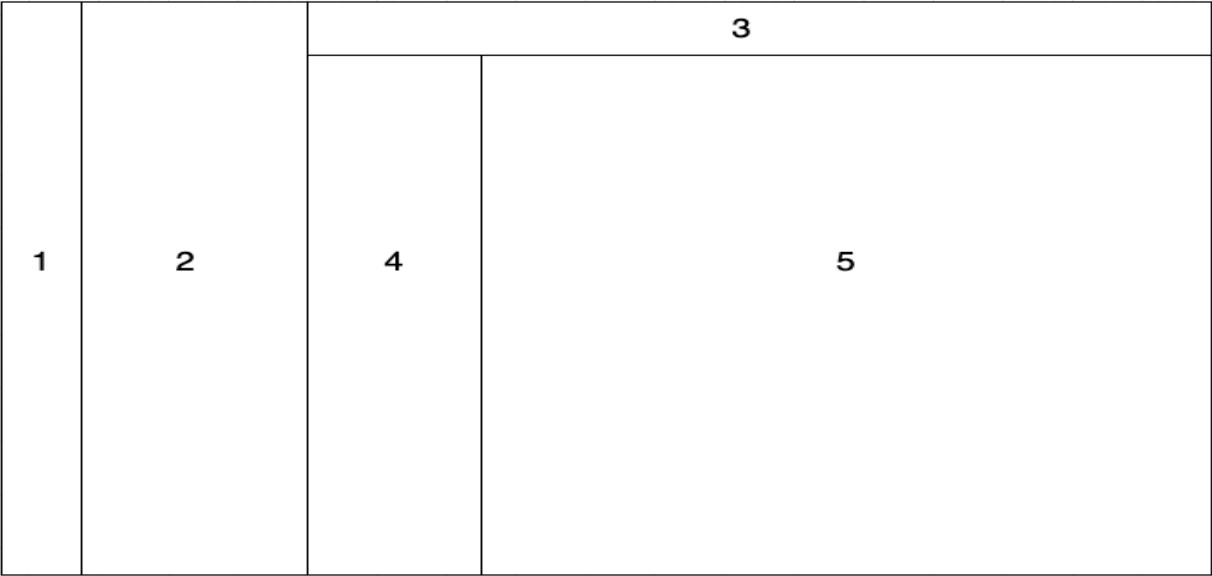


Рисунок В.11 – Веб-інтерфейс сторінки менеджменту змін працівників

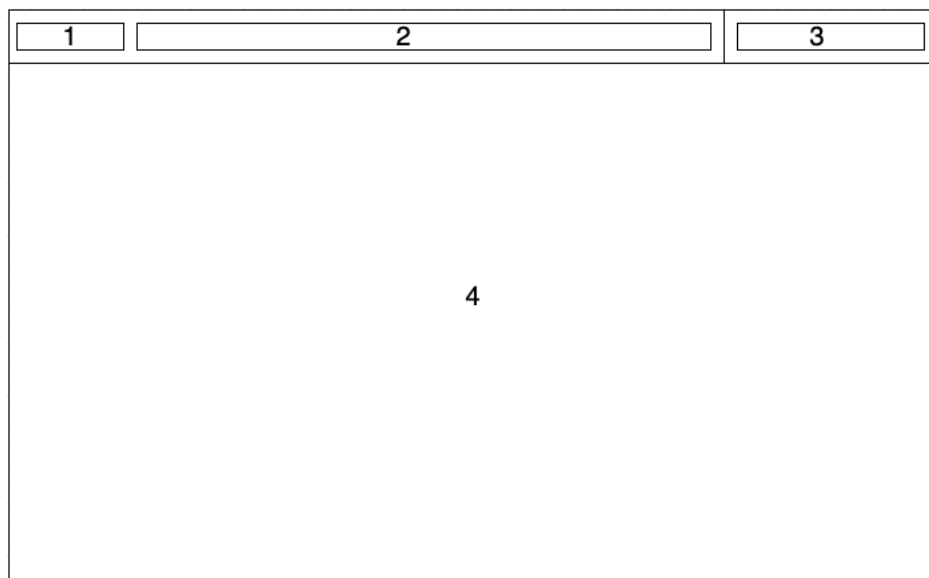


Рисунок В.12 – Веб-інтерфейс сторінки керування обладнанням

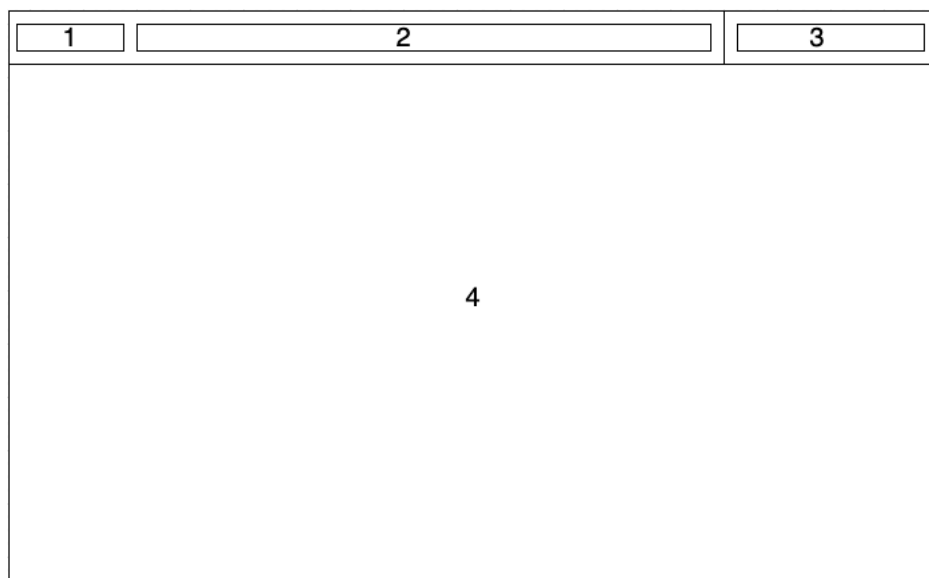


Рисунок В.13 – Веб-інтерфейс сторінки керування задачами

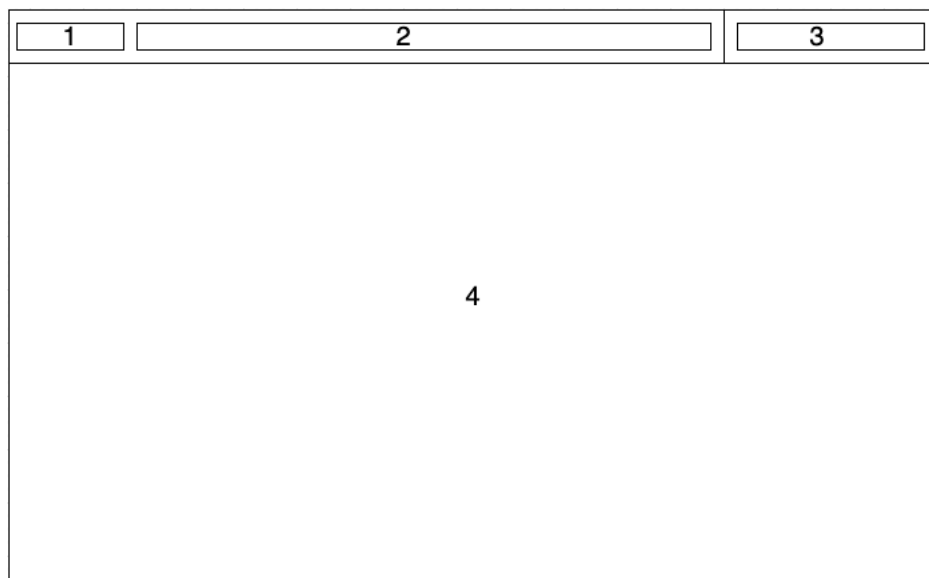


Рисунок В.14 – Веб-інтерфейс сторінки керування процесами

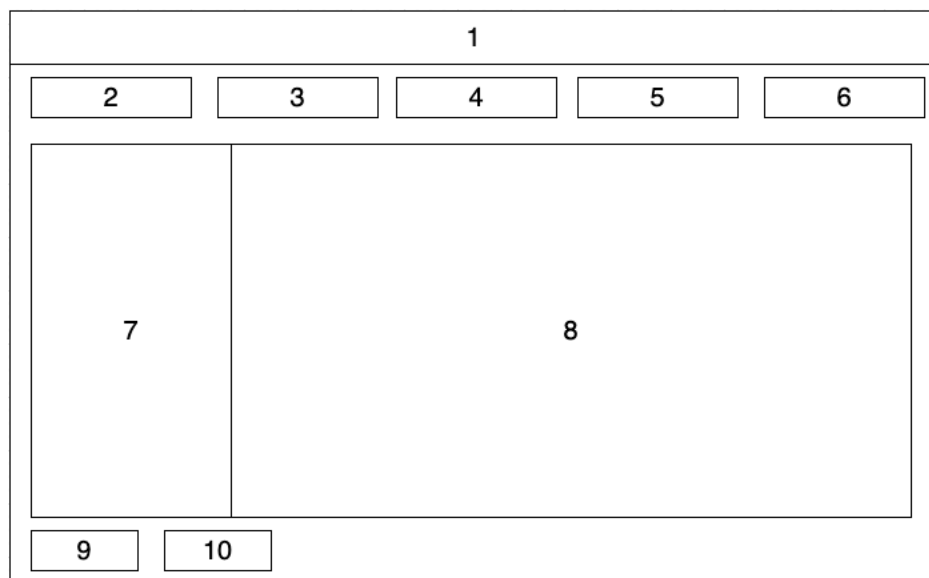


Рисунок В.15 – Модальне вікно керування задачами процесу

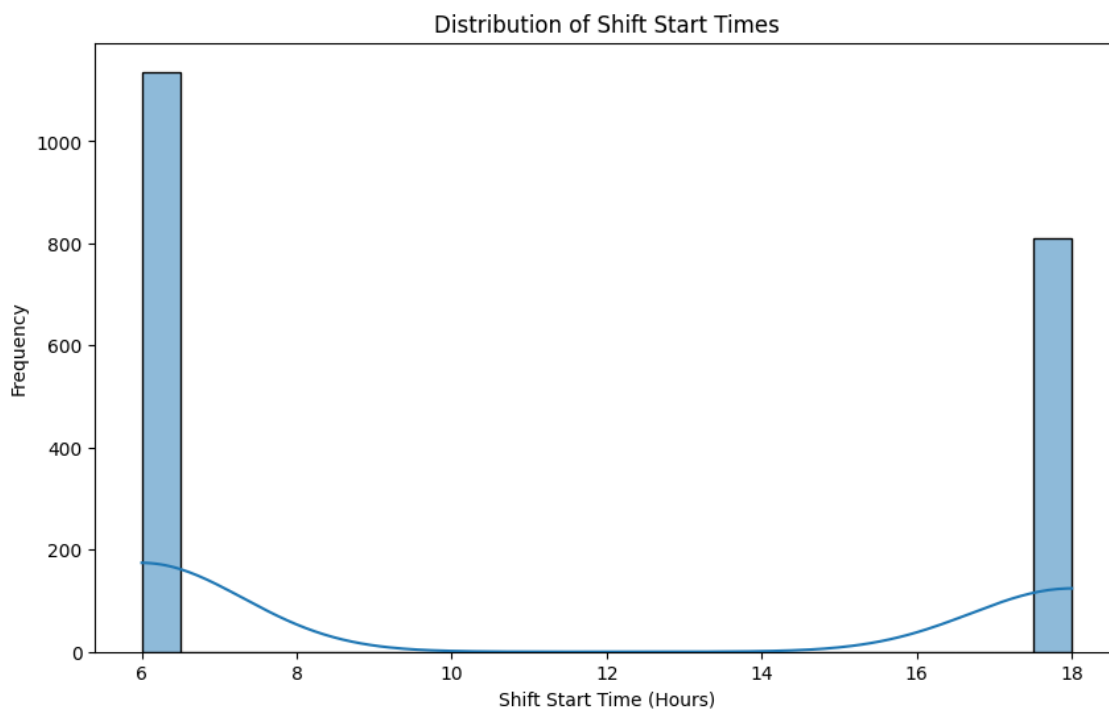


Рисунок В.16 – Графік розподілу старту змін.

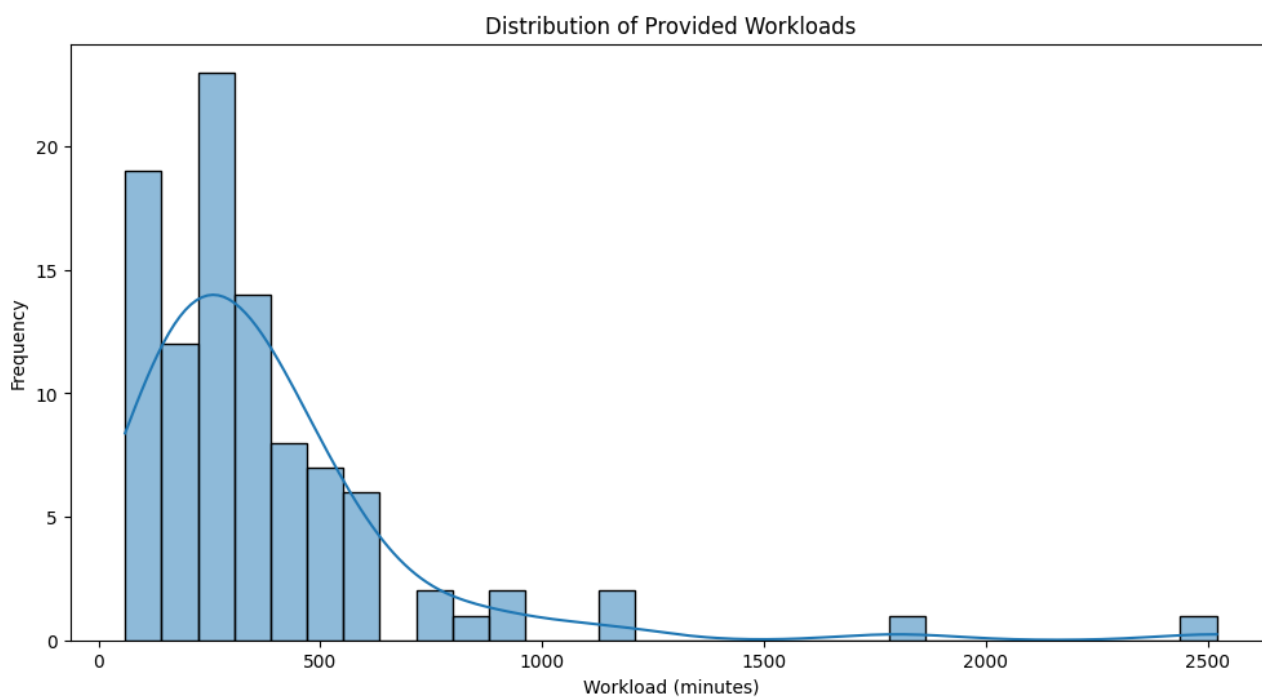


Рисунок В.17 – Графік навантаження по процесам.

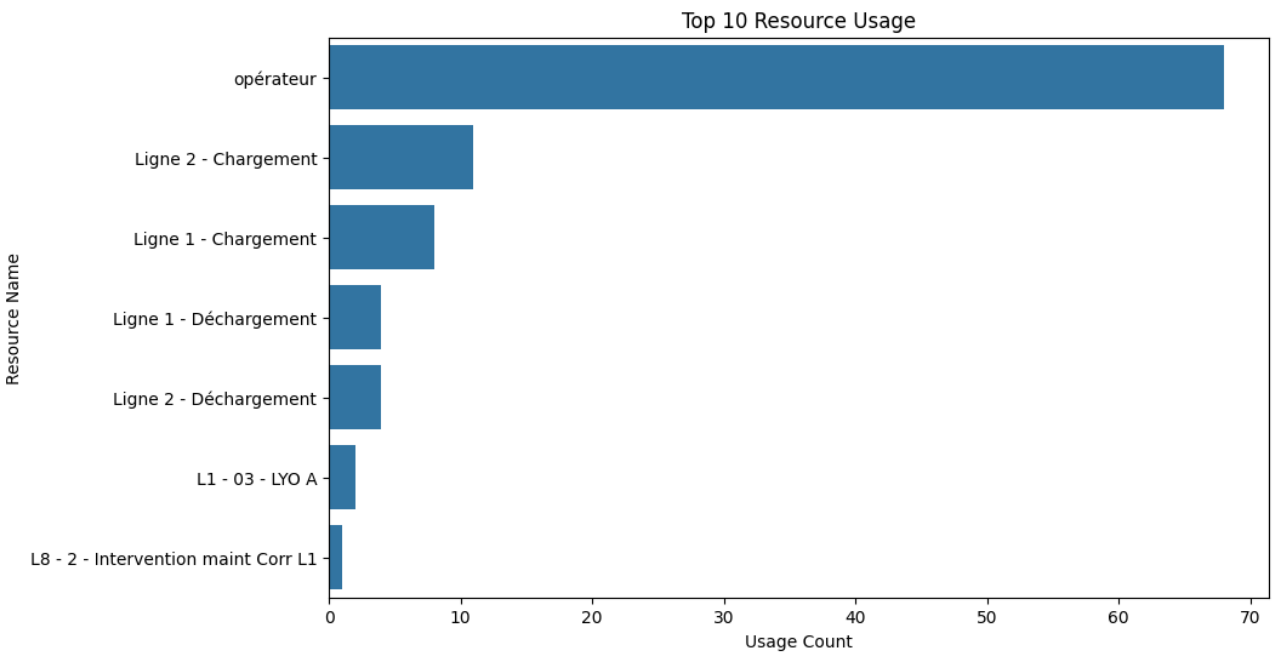


Рисунок В.18 – Графік частоти використання ресурсів.

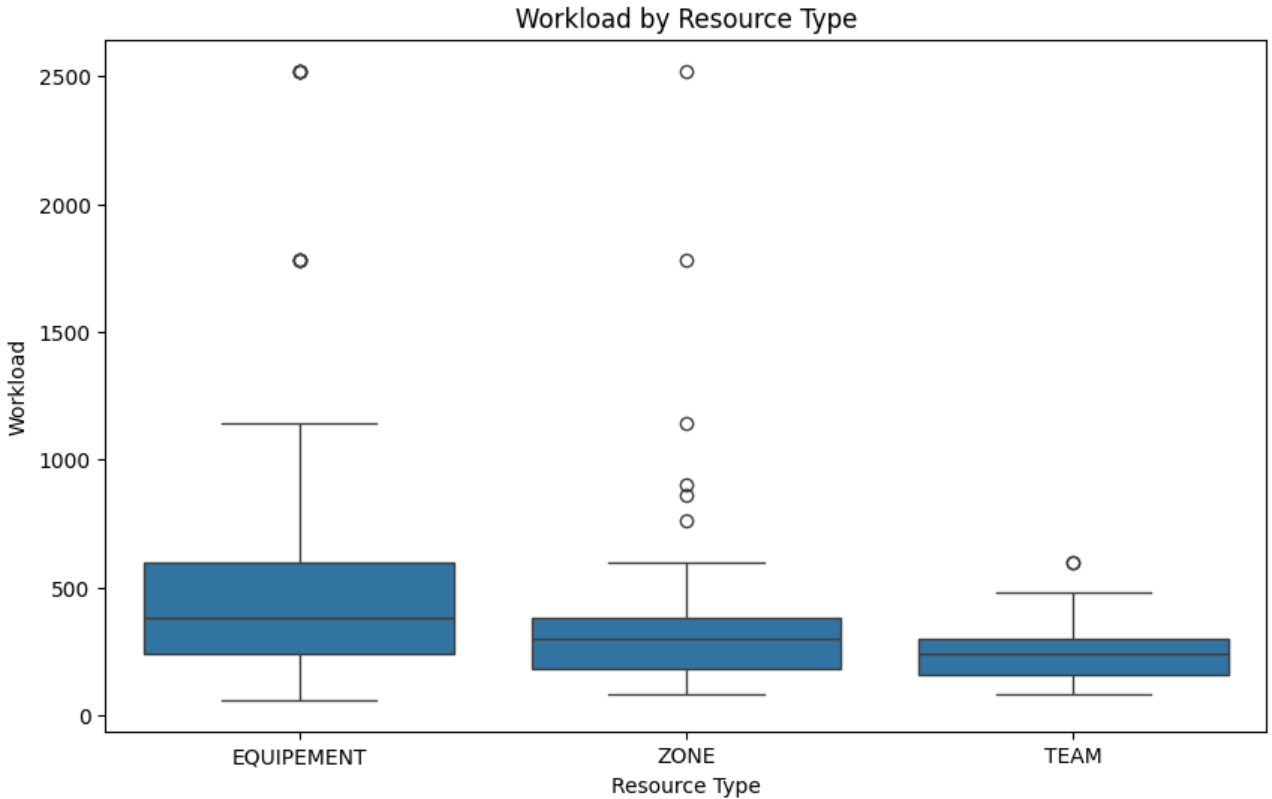


Рисунок В.19 – Графік навантаженості по ресурсам.

Додаток Г (довідниковий)

Інструкція користувача

Для того щоб можна було зайнятися плануванням процесів потрібно розглянути усі передуючі веб-інтерфейси. Далі буде розглянуто сторінки менеджменту працівників, їх зміни, задачі, процесу та обладнання. Згодом буде розглянуто основну сторінку де уся попередньо налаштована інформація буде мати сенс та своє відображення в єдиному місці.

User name	User Id	Cadence	Scoring	Contract	Team	Subteam	Role	Supervisor	Contract Start	Contract End
Abby Halvorson	User36	View/Edit	View/Edit	CDI	AQL					
Adah Torphy	User38	View/Edit	View/Edit	CDI	AQL					
Agostino, Davide	agostd00	View/Edit	View/Edit		AQL					
Agustina Weber	User62	View/Edit	View/Edit		AQL					
Akar, Antoine	aha87331	View/Edit	View/Edit		AQL				23-12-2023	
Alivia Harris	user257	View/Edit	View/Edit	Interim	Classificateur			Janae Schmidt		
Anderson McCullough	User50	View/Edit	View/Edit	CDD	Classificateur			Janae Schmidt		
Arous, Aurore	aa384893	View/Edit	View/Edit		Classificateur			Abbeloos, Antoine		
artem	artemterno	View/Edit	View/Edit		OP VI					
artem ternopilski	artem	View/Edit	View/Edit		OP				01-01-1900	01-01-1900
artem ternopilski	asdasf	View/Edit	View/Edit		OP			Albina Jenkins	01-01-1900	01-01-1900
Artem Ternopilski	artemTerno	View/Edit	View/Edit		OP VI					
artem ternopilski	artemterno	View/Edit	View/Edit		OP VI					
artem ternopilski	artemterno	View/Edit	View/Edit		OP VI					
Artem Ternopolskiy	artemono	View/Edit	View/Edit		Classificateur			Romano Test		

Рисунок Г.1 – Сторінка менеджменту працівників.

На рисунку Г.2 ми можемо побачити описаний раніше інтерфейс із таблицею усіх працівників закріплених за конкретним виробництвом. В цій частині ми можемо змінити усі параметри окремо взятого працівника.

Перелік параметрів:

- Scoring – відповідає за оцінку наскільки працівник підходить під виконання задач на конкретному обладнанні.
- Contract – визначає тип контракту для конкретного працівника.
- Team – визначає до якої команди належить працівник.

- Subteam – визначає до якої під команди належить працівник.
- Role – визначає яку роль виконує в команді працівник.
- Supervisor – визначає який працівник є супервізором для цього працівника.
- Contract start – Дата початку дії контракту із працівником.
- Contract end – Дата кінця дії контракту із працівником.

Для того щоб додати до системи нового працівника потрібно натиснути в шапці кнопку «Create user».

The screenshot shows a 'Create person' modal window. The header is purple with the text 'Create person' and a close button. The form contains the following fields:

- User name ***: A text input field.
- User Id ***: A text input field.
- Team**: A dropdown menu.
- Subteam**: A dropdown menu showing 'null'.
- Role**: A dropdown menu.
- Supervisor**: A button labeled 'Choose'.
- Contract**: A dropdown menu showing 'CDI'.
- Contract start date**: A text input field.
- Contract end date**: A text input field.

At the bottom of the form, there are two buttons: 'Save' (purple) and 'Cancel' (white).

Рисунок Г.2 – Модальне вікно створення нового працівника.

Тут задаються початкові параметри для створення нового працівника які будуть подалі використані в системі для розраховувань і планування процесів. Після того як всі параметри будуть введені в поля вводу потрібно натиснути кнопку «Save» і тоді в базі буде записано всі дані нового робітника.

Task Name	Category	Workload	Cadence	Workers	% Knowledge	Curricula	Zones	Equipments	Splitting
a5	az22	1h 0m	N/A	4	N/A	0	2	12	☐
aaa1	21	0h 0m	N/A	2	N/A	0	3	0	☐
absent	Absence	8h 0m	N/A	0	N/A	0	0	0	☒
Activité Libre	FREETASK	0h 0m	N/A	0	N/A	0	0	0	☐
ADV CLB J1 - Film	Fermentation	5h 0m	N/A	5	N/A	0	1	5	☐
ADV Correction	Preparation	2h 0m	N/A	1	N/A	0	0	0	☐
ADV Preparation	Cycle Lyo	2h 0m	N/A	2	N/A	0	1	4	☐
ADV Preprise Production	Finalisation	2h 0m	N/A	3	N/A	0	1	2	☐
ADV Redaction	Maintenance	3h 0m	N/A	1	N/A	0	0	0	☐
AdvPlanningToEquipTask	NoEquip	0h 0m	N/A	0	N/A	0	1	1	☐
AM TIME OFF	Absence	0h 0m	N/A	0	N/A	0	1	0	☒
AQL	Absence	0h 0m	N/A	0	N/A	0	1	0	☒
AQL Full-Time	Maintenance	1h 0m	248	2	N/A	0	0	1	☐
asdfa	21	0h 0m	N/A	0	N/A	0	0	0	☐
BLOCK RESOURCE	BLOCKTASK	0h 0m	N/A	0	N/A	0	0	0	☐
Bondage Creation	az22	0h 30m	248	3	N/A	0	1	1	☐
Box Creation	autoAssignmentTest	2h 0m	248	3	N/A	0	1	1	☐

Рисунок Г.3 – Сторінка менеджменту задач.

На цій сторінці презентовано інтерфейс редагування, додавання та видалення існуючих задач. Таблиця відображає параметри і якщо натиснути на назву, можна побачити модальне вікно редагування.

Task management

General Zones Equipments Knowledge Workers Sharing Advanced Subtasks

Task name: *

absent

Short name:

null

Category *

Select category New Category

Workload *

8H 0M

Close

Рисунок Г.4 – Модальне вікно редагування параметрів задачі.

Якщо натиснути кнопку «New Task», ми можемо створити нову задачу, яку потім можна використовувати для огорнення в процес і використання на Planning Line.

New task FCP

Task name: *

Short name:

Category *

Select category ▼ New Category

Workload *

0H ▼ 0M ▼

Close

Рисунок Г.5 – Модальне вікно створення задачі.

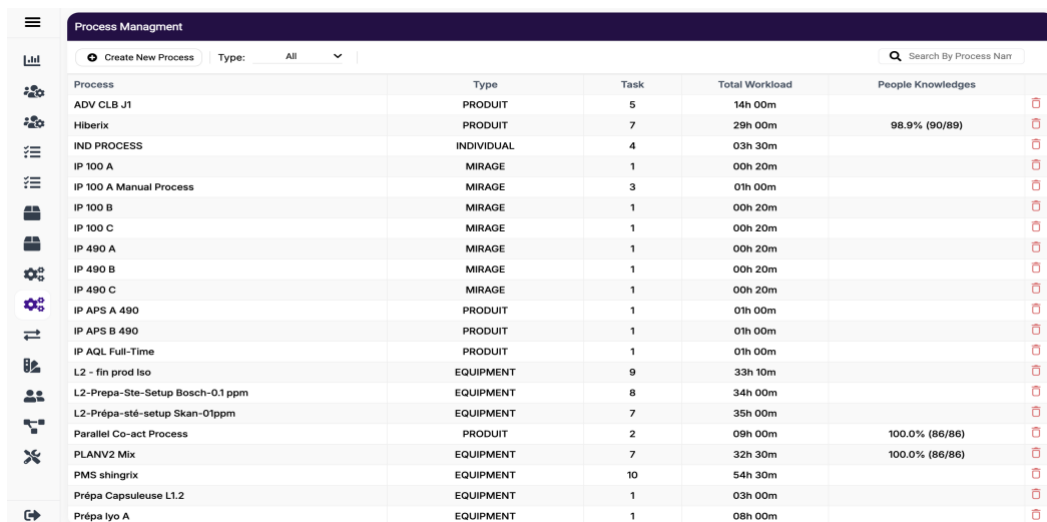
Розглянемо параметри які можна задати.

- Task name – Назва задачі.
- Short name – Короткий тег задачі.
- Category – Категорія задачі.
- Workload – Час за який задача має бути виконана.

Також на сторінці для зручності є велика кількість основних фільтрів таких як:

- Фільтр по підприємствам.
- Фільтр по категоріям.
- Фільтр по зонам.
- Фільтр по назві.

Далі ми розглянемо інтерфейс сторінки менеджменту процесів. Це є однією з найважливіших деталей усієї програми, так як тут збирається до купи вся важлива інформація для того щоб на виході отримати готовий для планування процес і подальшого функціонування підприємства адаптований під їх потреби.



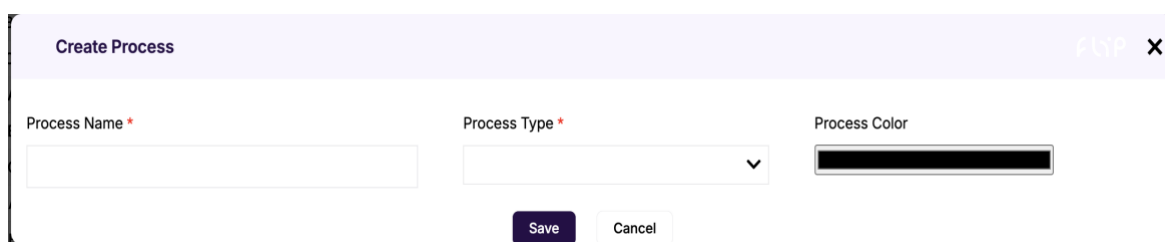
Process	Type	Task	Total Workload	People Knowledge
ADV CLB J1	PRODUIT	5	14h 00m	
Hiberix	PRODUIT	7	29h 00m	98.9% (90/89)
IND PROCESS	INDIVIDUAL	4	03h 30m	
IP 100 A	MIRAGE	1	00h 20m	
IP 100 A Manual Process	MIRAGE	3	01h 00m	
IP 100 B	MIRAGE	1	00h 20m	
IP 100 C	MIRAGE	1	00h 20m	
IP 490 A	MIRAGE	1	00h 20m	
IP 490 B	MIRAGE	1	00h 20m	
IP 490 C	MIRAGE	1	00h 20m	
IP APS A 490	PRODUIT	1	01h 00m	
IP APS B 490	PRODUIT	1	01h 00m	
IP AQL Full-Time	PRODUIT	1	01h 00m	
L2 - fin prod Iso	EQUIPMENT	9	33h 10m	
L2-Prepa-Ste-Setup Bosch-0.1 ppm	EQUIPMENT	8	34h 00m	
L2-Prépa-sté-setup Skan-0.1ppm	EQUIPMENT	7	35h 00m	
Parallel Co-act Process	PRODUIT	2	09h 00m	100.0% (86/86)
PLANV2 Mix	EQUIPMENT	7	32h 30m	100.0% (86/86)
PMS shingrix	EQUIPMENT	10	54h 30m	
Prépa Capsuleuse L1.2	EQUIPMENT	1	03h 00m	
Prépa Iyo A	EQUIPMENT	1	08h 00m	

Рисунок Г.6 – Сторінку менеджменту процесів

На цій сторінці ми по аналогії з попередніми ми бачим таблицю уже існуючих процесів з їх наступними параметрами.

- Process – Назва процесу.
- Type – Тип процесу.
- Task – Кількість задач у процесі.
- Total Workload – Загальний час виконання процесу.

Для того щоб почати працювати потрібно спочатку створити процес, для цього натискаєм на кнопку «Create New Process».



Create Process

Process Name *

Process Type *

Process Color

Save

Cancel

Рисунок Г.7 – Модальне вікно створення процесу.

Тут потрібно задати початкові параметри такі як назва процесу, його тип та колір.

Після того як ми створили бажаний процес, ми повинні повернутись до таблиці і натиснути на назву, для того щоб побачити більш детальне модальне вікно налаштування.

Modify Process

Process Name * Process Type * Process Color Worker constraint Add process task

#	Task name	Timeline
1	▼ Hiberix Preparation	[Blue bar]
2	▼ L1 - Chargement	[Green bar]
3	▼ L1 - Cycle Lyo - Hiberix	[Purple bar]
4	▼ L1- Déchargement	[Cyan bar]
5	▼ L1 - Interlot déchargement	[Yellow bar]
6	▼ Prepa Lyo	[Dark Blue bar]
7	▼ L1 - auto prod chgt	[Light Green bar]

Рисунок Г.8 – Модальне вікно налаштування процесу.

Тут ми можемо змінювати попередні налаштування а також зайнятись найголовнішим етапом. Надається можливість додати задачі до процесу, налаштувати в якому порядку вони будуть виконуватись, а також побачити візуально змодельований результат налаштування в режимі реального часу.

Важливим моментом також є визначення тип працівників якого має бути задіяно на тій чи іншій задачі. Для того щоб виконати це налаштування нам потрібно натиснути на кнопку «Worker constraint».

Seq	Task	Worker Rule
1	Hiberix Preparation	All Workers
2	L1 - Chargement	Custom Workers ▼ Excluded With ▼ L1 - Interlot déchargement ▼ Selected (16) ▼
3	L1 - Cycle Lyo - Hiberix	All Workers
4	L1- Déchargement	All Workers
5	L1 - Interlot déchargement	All Workers
6	Prepa Lyo	All Workers
7	L1 - auto prod chgt	All Workers

Рисунок Г.9 – Модальне вікно налаштування потрібних працівників для задач.

Найголовніше тут це те що програма надає гнучку кастомізацію з можливістю створювати власну логіку алгоритму підбору найкращого кандидата для виконання певної задачі.

Остання частина налаштування всіх так званих Constraints – є визначення годин праці кожного робітника. Це є дуже важливим, так як це напряду впливає на правильність того як штучний інтелект у подальшому повинен підбирати працівника на виконання задачі. Адже правильна стиковка усіх цих даних повинна дати у результаті оптимізований та високоефективний графік для усіх сторін підприємства. Не рідкість коли одна задача повинна виконуватись для прикладу більше 8 годин і в години коли більшість підходящих працівників уже давно закінчили свої зміни. В такому випадку потрібно розділити виконання на дві частини, коли одразу обидві людини буде вести до кінця поставлену перед ним мету.

Саме з таких цілей існує сторінка менеджменту змін працівників.

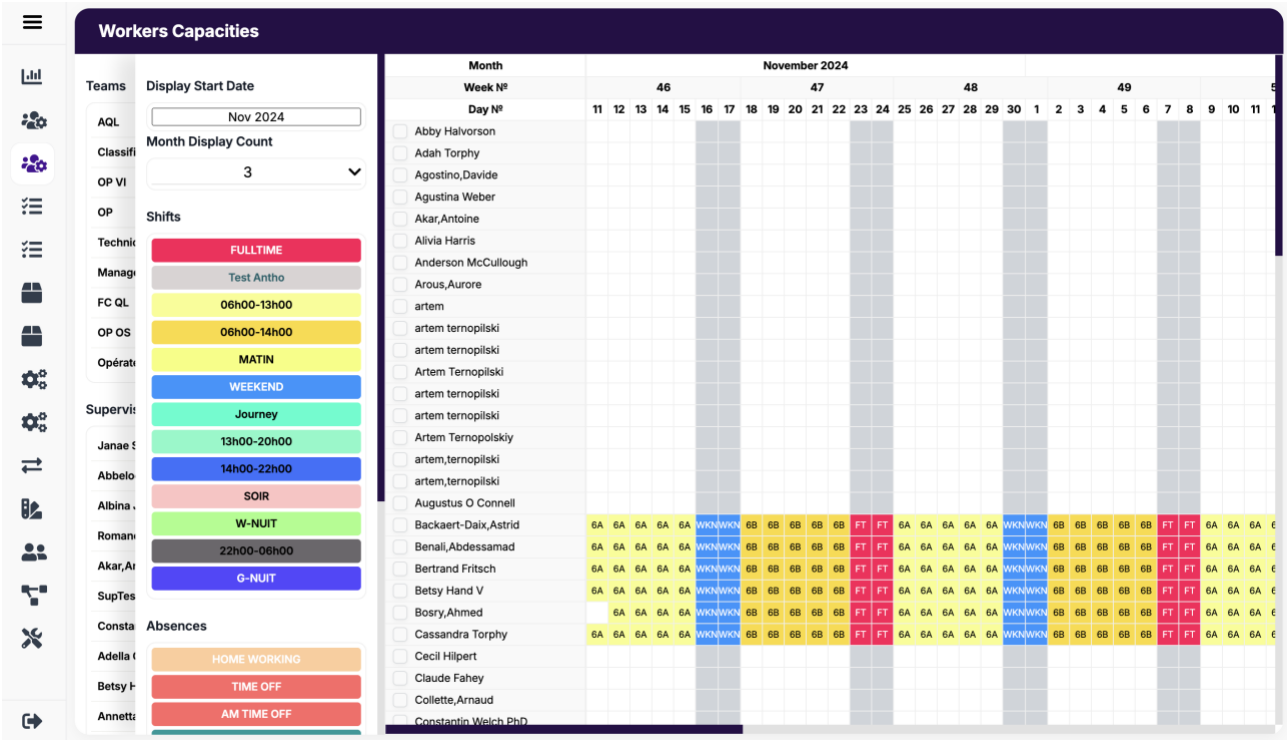


Рисунок Г.10 – Стоірнка менеджменту змін працівників.

Як можна побачити на рисунку Г.10, стоїрнка представляє з себе велику таблицю розділену по тижням, місяцям – рокам. Що надає високу інформативність і зручність у зчитуванні інформації. Для того щоб назначати конкретному працівнику його зміни, потрібно просто натиснути на початковий відповідний день у таблиці, і потім тиснути на кінцевий день його зміни, і у лівій частині екрану обрати потрібну зміну, наприклад «06h-14h00», таким чином система запише його новий робочий час і буде його враховувати при назначенні працівників на задачі. Модуль також надає високу гнучкість дозволяючи видаляти зміни, копіювати їх від одного працівника і назначенні таких самх змін іншому, а також створення паттерну за яким будуть назначенні зміни всім обраним працівникам.

Поєднанням усіх попередніх кроків, їх сплетінням в один цілісний живий організм – є лінія планування, точка перетину і перлина програми. Саме тут набуває візуалізації усі дані які були зазначені раніше.

Equipments	Wednesday, 06.11.2024	Thursday, 07.11.2024	Friday, 08.11.2024
01_SAO4_LM1 [LM]	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
01_SAO4_LM2 [CIP 1]			
01_SAO4-dota [Back]			
01_SAO4-net [true]			
02_SAO4_WAM 1 [WAM 1]			
03_SAO4_WAM 2 [WAM 2]			
04_SAO4_CIP 1 [CIP 1]			
05_SAO4_CIP 2 [CIP 2]			
06_SAO4_Montage 1 [Table 1]			
06_SAO4_Montage 1 / ... [06_SAO4_Montage 2 / ...]			
07_SAO4_Montage 2 [Table 2]			
08_SAO4_Autoclave 1 [AU_6901]			
09_SAO4_Autoclave 2 [AU_6902]			
10_SAO4_Autoclave 3 [AU_6903]			
11_SAO4_Autoclave 4 [AU_6904]			
11_SAO4_Autoclave 5 [AU_6905]			
21_SAO3_CIP 1 [CIP 1]			
22_SAO3_Montage 1 [Table 1]			
22_SAO3_Montage 1 [Table 2]			
24_SAO3_Autoclave 1 [AU_6901]			
25_SAO3_Autoclave 2 [AU_6902]			

Рисунок Г.11 – Сторінка лінії планування

Зустрічає нас інтерфейс з різноманіттю різних функцій, фільтрів і можливостей. Загострим увагу саме на процесі планування і як це зробити. Для того щоб запланувати процес, створений і зібраний до купи у попередніх кроках, потрібно двічі натиснути на обладнання до якого прикріплений процес і далі можна побачити додаткову вкладку у якій ми задаємо параметри для планування.

Process/Tasks Creation	
Create once	▼
Type	▼
Process	▼
Hiberix [PRODUIT]	▼
Batch Name	
Start from	
06-11-2024	07:00
Create	

Рисунок Г.12 – Вкладка параметрів планування.

Тут можна обрати у селекторі який саме процес ми хочем запланувати. У полі «Batch name» потрібно задати унікальний ідентифікатор процесу, адже їх може бути незчисленна кількість, тому потрібно їх відрізнити за точною ознакою. Далі потрібно обрати дату коли процес має початись і натиснути кнопку «Create».

Після того як усі вищепересислені дії було виконано, на лінії планування можна побачити задачі з процесу які ми запланували.

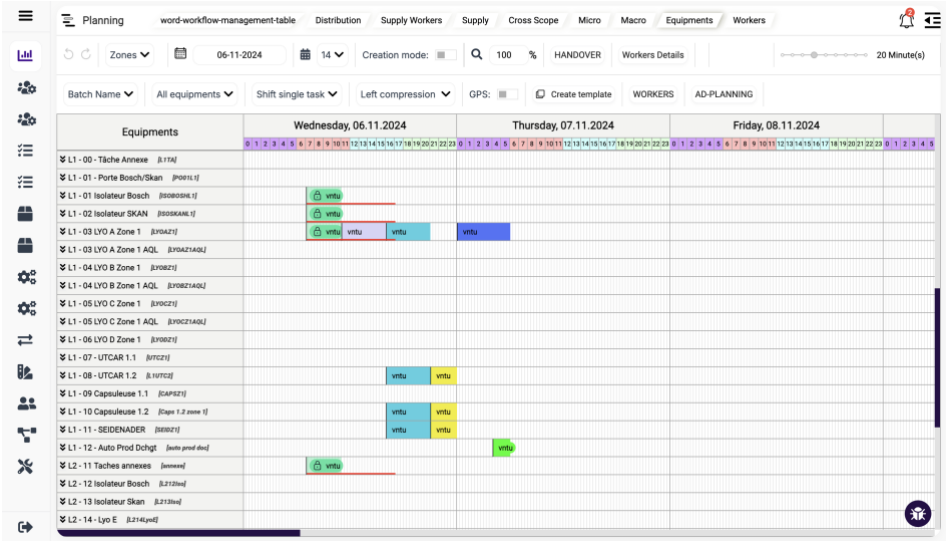


Рисунок Г.13 – Запланований процес на лінії планування.

Останнє що залишається це назначити працівників на задачі, для цього в шапці стоїрнки натискаєм на «Workers» і тут нас зустрічає ось таке модальне вікно.



Рисунок Г.14 – Модальне вікно автоматичного назначення працівників на задачі.

Тут все що потрібно це натиснути на «Prepare workers» і за допомогою штучного інтелекту, буде знайдено оптимальна кількість робітників, зважаючи на їх робочий час, назначеність на інші задачі та інші парметри і призначить виконавців.