

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**«Програмне забезпечення для підтримки вирішення
задачі удосконалення бізнес-процесів підприємства»**

Студента 2 курсу, 3м групи,
спеціальності 121 «Інженерія
програмного забезпечення»
освітньої програми «Інженерія
програмного забезпечення»

підпис студента

Литвиненка Сергія
Вікторовича

Науковий керівник
доктор технічних наук,
професор кафедри інженерії
програмного забезпечення та
кібербезпеки

підпис керівника

Цюцюра Микола
Ігорович

Гарант освітньої програми
кандидат педагогічних наук,
доцент кафедри інженерії
програмного забезпечення та
кібербезпеки

підпис гаранта

Котенко Наталія
Олексіївна

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення та кібербезпеки

Освітній ступінь магістр

Освітня програма 121 «Інженерія програмного забезпечення»

Затверджую

Зав. кафедри інженерії програмного
забезпечення та кібербезпеки

Криворучко О. В.

«27» листопада 2023 р.

**Завдання
на кваліфікаційну роботу студентів**

Литвиненку Сергію Вікторовичу

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи «Програмне забезпечення для підтримки
вирішення задачі удосконалення бізнес-процесів підприємства»

Затверджена наказом ректора від «27» листопада 2023 р. № 4193

2. Строк здачі студентом закінченої роботи 29 листопада 2024

3. Цільова установка та вихідні дані до роботи

Мета роботи аналіз наявних методів для отримання знань про бізнес-процеси компанії, та створення аналітично-інформаційного застосунок для вираховування економічних показників ефективності виробництва компанії.

Об'єкт дослідження процес створення застосунку, що призначений для моделювання процесу роботи компанії з виробництвом різних груп товарів, результатом якого є результати розрахунків економічних показників роботи підприємства.

Предмет дослідження аналіз методів одержання інформації про бізнес-процеси компанії для побудови програмного забезпечення, що базується на розрахунках точних економічних показників.

4. Консультанти роботи із зазначенням розділів, які консультують:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв
Розділ 1 «Аналіз предметної області «Програмне забезпечення для підтримки вирішення задачі удосконалення бізнес-процесів підприємства»»	Цюцюра Микола Ігорович	09.04.2024	09.04.2024
Розділ 2 «Проектування програмної реалізації «Програмне забезпечення для підтримки вирішення задачі удосконалення бізнес-процесів підприємства»»	Цюцюра Микола Ігорович	20.05.2024	20.05.2024
Розділ 3 «Програмна реалізація «Програмне забезпечення для підтримки вирішення задачі удосконалення бізнес-процесів підприємства»»	Цюцюра Микола Ігорович	04.06.2024	04.06.2024

5. Зміст кваліфікаційної роботи (перелік питань за кожним розділом)
ВСТУП

РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ «ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ ВИРІШЕННЯ ЗАДАЧІ УДОСКОНАЛЕННЯ БІЗНЕС-ПРОЦЕСІВ ПІДПРИЄМСТВА»

1.1. Коротка характеристика поняття бізнес-процесу

1.2. Огляд і аналіз підходів до застосування бізнес-процесів

1.3. Висновки до розділу 1

РОЗДІЛ 2 ПРОЕКТУВАННЯ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ «ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ ВИРІШЕННЯ ЗАДАЧІ УДОСКОНАЛЕННЯ БІЗНЕС-ПРОЦЕСІВ ПІДПРИЄМСТВА»

2.1. Опис концептуальної моделі

2.2. Аналіз обраної моделі

2.3. Обраний метод аналізу бізнес-процесів

2.4. Висновки до розділу 2

РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ «ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
ДЛЯ ПІДТРИМКИ ВИРІШЕННЯ ЗАДАЧІ УДОСКОНАЛЕННЯ БІЗНЕС-
ПРОЦЕСІВ ПІДПРИЄМСТВА»

3.1. Вибір засобу та мови програмування створення застосунку

3.2. Організація інтерфейсу з користувачем

3.3. Вхідні та вихідні дані дослідження

3.4. Приклад роботи та тестування

3.5. Висновок до розділу 3

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ДОДАТКИ

6. Календарний план виконання роботи

№ пор.	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	
		за планом	фактично
1	2	3	4
1.	<i>Вибір теми випускної кваліфікаційної роботи</i>	07.11.2023	07.11.2023
2.	<i>Розробка та затвердження завдання на роботу магістра (стац/заоч)</i>	13.12.2023	13.12.2023
3.	<i>Вступ та перелік літературних джерел</i>	22.02.2024	22.02.2024
4.	<i>Розробка технічного завдання</i>	14.03.2024	14.03.2024
5.	<i>Розділ 1. Аналіз предметної області «Програмне забезпечення для підтримки вирішення задачі удосконалення бізнес-процесів підприємства»</i>	10.04.2024	10.04.2024
6.	<i>Розділ 2. Проектування програмної реалізації «Програмне забезпечення для підтримки вирішення задачі удосконалення бізнес-процесів підприємства»</i>	23.05.2024	23.05.2024
7.	<i>Розділ 3. Програмна реалізація «Програмне забезпечення для підтримки вирішення задачі удосконалення бізнес-процесів підприємства»</i>	05.09.2024	05.09.2024
8.	<i>Розробка програми та методики тестування</i>	27.09.2024	27.09.2024
9.	<i>Написання наукової статті</i>	16.04.2024	27.09.2024
10.	<i>Керівництво користувача</i>	11.10.2024	11.10.2024
11.	<i>Висновки та пропозиції</i>	16.10.2024	16.10.2024
12.	<i>Задача випускної кваліфікаційної роботи на кафедрі (перша перевірка)</i>	18.10.2024	18.10.2024
13.	<i>Підготовка реферату та презентації доповіді</i>	28.10.2024	28.10.2024
14.	<i>Попередній захист випускної кваліфікаційної роботи</i>	29.10.2024 – 31.10.2024	29.10.2024 – 31.10.2024
15.	<i>Задача зброшурованої випускної кваліфікаційної роботи</i>	29.11.2024	29.11.2024
16.	<i>Зовнішнє рецензування випускної кваліфікаційної роботи</i>	28.10.2024	28.10.2024
17.	<i>Підготовка до публічного захисту випускної кваліфікаційної роботи</i>	02.12.2024-03.12.2024	02.12.2024-03.12.2024

7. Дата видачі завдання «13» листопада 2023 р.

8. Науковий керівник кваліфікаційної роботи

Цюцюра М.І.

(прізвище, ініціали, підпис)

9. Гарант освітньої програми

Котенко Н.О.

(прізвище, ініціали, підпис)

10. Завдання прийняв до виконання студент

Литвиненко С.В.

(прізвище, ініціали, підпис)

АНОТАЦІЯ

Відповідно до мети дослідження робота присвячена аналізу наявних методів для отримання знань про бізнес-процеси підприємства, та створенню аналітично-інформаційного застосунок для вираховування економічних показників ефективності виробництва підприємства. Випускна кваліфікаційна робота на тему «Програмне забезпечення для підтримки вирішення задачі удосконалення бізнес-процесів підприємства» містить 46 сторінок, 2 рисунки і 5 таблиць. Перелік використаних джерел налічує 11 найменувань.

Було проаналізовано різні методи отримання та аналізу даних про бізнес-процеси підприємства, обрано метод аналізу, який найбільше для побудови програмного забезпечення, що базується на розрахунках точних економічних показників.

В результаті здійснено аналіз наявних методів для отримання знань про бізнес-процеси підприємства, та створено аналітично-інформаційний застосунок для вираховування економічних показників ефективності виробництва підприємства.

Ключові слова: програмне забезпечення, бізнес-процеси, оптимізація, економічні показники, аналіз, моделювання, автоматизація, ефективність, підприємство, аналітичний застосунок.

ABSTRACT

According to the purpose of the study, the work is devoted to the analysis of existing methods for obtaining knowledge about the business processes of the enterprise, and the creation of an analytical and information application for calculating economic indicators of the efficiency of the enterprise's production. The final qualification work on the topic "Software to support the solution of the problem of improving the business processes of the enterprise" contains 40 pages, 2 figures and 5 tables. The list of sources used includes 11 items.

Various methods of obtaining and analyzing data on the business processes of the enterprise were analyzed, the analysis method was selected that is most suitable for building software based on the calculations of accurate economic indicators.

As a result, an analysis of existing methods for obtaining knowledge about the business processes of the enterprise was carried out, and an analytical and information application was created for calculating economic indicators of the efficiency of the enterprise's production.

Keywords: software, business processes, optimization, economic indicators, analysis, modeling, automation, efficiency, enterprise, analytical application.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Процес - послідовна зміна предметів і явищ, що відбувається закономірним порядком, сукупність ряду послідовних дій, спрямованих на досягнення певного результату, послідовна зміна станів об'єкту в часі.

Бізнес-процес - будь-яка діяльність, що має вхідний продукт, додає вартість до нього, та забезпечує вихідний продукт для внутрішнього або зовнішнього споживача.

Компанія - торговельне або промислове товариство (фірма), що об'єднує підприємців, спілка.

ARIS - Architecture Of Integrated Information Systems

DFD - Data Flow Diagrams

ERD - Entity Relationship Diagrams

FDD - Functional Dependency Diagram

GUI - Graphical User Interface

IDEF - I-CAM Definition Или Integrated Definition

SADT - Structured Analysis And Design Technique

UML - Unified Modeling Language

ЗМІСТ

ВСТУП	6
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ «ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ ВИРІШЕННЯ ЗАДАЧІ УДОСКОНАЛЕННЯ БІЗНЕС-ПРОЦЕСІВ ПІДПРИЄМСТВА»	9
1.1. Коротка характеристика поняття бізнес-процесу	9
1.2. Огляд і аналіз підходів до застосування бізнес-процесів	12
1.3. Висновки до розділу 1	14
РОЗДІЛ 2 ПРОЕКТУВАННЯ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ «ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ ВИРІШЕННЯ ЗАДАЧІ УДОСКОНАЛЕННЯ БІЗНЕС-ПРОЦЕСІВ ПІДПРИЄМСТВА»	16
2.1. Опис концептуальної моделі	16
2.2. Аналіз обраної моделі	17
2.3. Обраний метод аналізу бізнес-процесів	21
2.4. Висновки до розділу 2	24
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ «ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ ВИРІШЕННЯ ЗАДАЧІ УДОСКОНАЛЕННЯ БІЗНЕС-ПРОЦЕСІВ ПІДПРИЄМСТВА»	26
3.1. Вибір засобу та мови програмування створення застосунку	26
3.2. Організація інтерфейсу з користувачем	27
3.3. Вхідні та вихідні дані дослідження.....	31
3.4. Приклад роботи та тестування	35
3.5. Висновок до розділу 3	37
ВИСНОВКИ ТА ПРОПОЗИЦІЇ	38
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	40
ДОДАТКИ	41

ВСТУП

Актуальність роботи

Створення формальної моделі, яка зображує взаємопов'язані операції, відоме як моделювання бізнес-процесу, яке служить для візуального представлення суб'єктивного бачення робочого процесу підприємства. Моделювання служить для організації інформації про компанію та її бізнес-процеси у візуальному графічному та статистичному форматі, що полегшує аналітичну обробку даних про стан підприємства та виробництва.

Вибір методології та інструментів, що використовуються для моделювання бізнес-процесів, не є суттєвим фактором. Існують усталені методології та інструменти, які перевірені часом і можуть бути використані для моделювання підприємства та розробки його моделі. Основна перевага таких методів та інструментів полягає в їх легкості розуміння та доступності.

Мета магістерської роботи полягає в теоретичному обґрунтуванні та аналізі методів та концепцій підтримки вирішення задачі удосконалення бізнес-процесів підприємства, також створення програмного продукту, який призначений для підтримки вирішення задачі удосконалення процесу роботи підприємства.

Об'єктом дослідження є аналіз методів та процес створення програмного продукту для підтримки вирішення задачі удосконалення бізнес-процесів підприємства.

Предметом дослідження виступають особливості статистичних методів одержання знань про бізнес-процеси підприємства, та реалізація застосунку.

Предметна область цієї роботи охоплює підтримку вирішення задачі удосконалення бізнес-процесів підприємства для візуалізації та аналізу ефективності її діяльності. Основою є створення програмного продукту, який дозволяє моделювати процеси виробництва та економічні показники підприємства на основі зібраних статистичних даних. У цій області розглядаються методи одержання знань про бізнес-процеси, а також

інструменти та технології, що забезпечують автоматизацію та оптимізацію роботи підприємства.

Пошуки шляхів досягнення мети обумовили необхідність визначення **наступних завдань**:

- Здійснити огляд та стан аналізованої проблеми, визначити поняття бізнеспроцесу та підходи до їх застосування;
- Проаналізувати особливості концептуальної моделі та опис алгоритму роботи підтримки вирішення задачі удосконалення бізнес-процесів;
- Охарактеризувати методи, засоби та технології для підтримки вирішення задачі удосконалення бізнес-процесів підприємства;
- Програмна реалізація продукту для підтримки вирішення задачі удосконалення бізнес- процесів;
- Провести тестування ефективності програмної реалізації підтримки вирішення задачі удосконалення бізнес-процесів на тестових прикладах.

Наукова новизна дослідження полягає в тому, що концепція, описана при виконанні магістерської роботи для створення моделі забезпечує перевагу в своїй універсальності порівняно з іншими концепціями. Адже, підтримки вирішення задачі удосконалення бізнес процесів може відповідати практично всім запитам щодо підвищення ефективності діяльності компанії та її конкурентоспроможності. Крім того, впровадження такої програмної реалізації дозволяє керівнику або керівництву підприємства мати інформацію, яка дозволяє їм самостійно вдосконалювати свій бізнес і передбачати його майбутнє покладаючись на вираховані статистичні та аналітичні показники ефективності підприємства.

Методи дослідження у програмній реалізації застосунку підтримки вирішення задачі удосконалення бізнес-процесів використовується статистичний метод дослідження про бізнес-процеси, адже цей метод аналізу ефективності економічних показників допоможе розглянути такі показники

як рентабельність, ліквідність вкладених коштів, чистий прибуток, оборотність вкладених коштів та прибутковість продукту.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ «ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ ВИРІШЕННЯ ЗАДАЧІ УДОСКОНАЛЕННЯ БІЗНЕС-ПРОЦЕСІВ ПІДПРИЄМСТВА»

1.1. Коротка характеристика поняття бізнес-процесу

У сучасному світі бізнес-процеси є вирішальним елементом процвітання будь-якої установи. Вони служать засобом для розподілу завдань між різними секторами, підвищення продуктивності та мінімізації витрат. Залишається питання: що таке бізнес-процеси і як вони впливають на організацію?

Основна ідея концепції процесу полягає в прогресивній зміні об'єктів і явищ з часом. Послідовність дій виконується з метою досягнення певного результату. У результаті основні характеристики будь-якого процесу, в тому числі й у бізнесі, зосереджені навколо трансформації стану об'єкта та досягнення бажаної кінцевої мети.

Бізнес-процеси підприємства охоплюють величезну кількість процесів різноманітної природи, що пояснює відсутність загальноприйнятого визначення бізнес-процесу. В даний час існують численні тлумачення поняття бізнес-процесу, кожна з яких окреслює бізнес-процес з унікальної точки зору.

Теоретико-методологічним аспектам виявлення ядра бізнес-процесів на різних підприємствах присвячено декілька наукових досліджень. Слід зазначити, що ці дослідження були проведені видатними вченими як вітчизняного, так і міжнародного рівня, зокрема Хаммер М., Чампі Д., Портер М., Друккер П., Андерсен Б., Робсон М., Улла Ф., Чаадаєв В., Ареф'єва. О., Мельник І., Осовська Г., Денисенко Л., Шацька С. та ін.

Згідно з Б.Андерсеном, бізнес-процес відноситься до узгодженої послідовності взаємопов'язаних дій, які перетворюють входи на виходи або результати. Це пояснення бізнес-процесу впливає з формального поділу його основних компонентів, які складаються з «входу», «процесу», «виходу», «управління», «постачальника процесу» та «клієнта процесу» [1, с. 74].

Згідно з дослідженнями В.Горлачука та І.Яненкової, бізнес-процес – це систематичне та послідовне виконання взаємозалежних та логічно пов'язаних завдань, які використовують ресурси для виробництва споживчих товарів для клієнтів у сфері виробничої діяльності [2, с. 263].

Згідно Еріксону, бізнес-процес — це послідовність логічно пов'язаних повторюваних дій, які використовують ресурси компанії для фізичної або віртуальної обробки об'єкта, що в кінцевому підсумку призводить до вимірюваних результатів або продуктів, які задовольняють потреби як внутрішніх, так і зовнішніх споживачів [3, с. . 24].

Відповідно до М.Хаммера та Дж.Чампі, бізнес-процес включає вхід, ряд етапів діяльності та вихід. Вхід означає початок процесу, а вихід – результат виконаної роботи [5, с.66].

Згідно з М. Робсоном і Ф. Уллахом, бізнес-процес означає переміщення роботи від однієї особи до іншої, а в більших процесах, потенційно, між відділами. Ці процеси можна класифікувати на різних рівнях, але вони завжди мають початок, низку кроків посередині та чітке завершення [4, с. 27].

Підхід, запропонований М.Портером і В.Міллером, відомий як метод ланцюжка вартості, може бути використаний для ідентифікації фундаментальних компонентів бізнес-процесу в організації. Науковці стверджують, що точки входу та виходу, інтерфейси та структура організації, яка охоплює структуру споживача товарів і послуг, є ключовими детермінантами бізнес-процесу. Це включає вартість, додану до продукту чи послуги під час виробництва.

Бізнес-процеси – це «сукупність взаємопов'язаних і взаємодіючих заходів або завдань, спрямованих на створення певного продукту або послуги, які представляють цінність для споживачів» [6, с. 187]. Бізнес-процес є унікальним засобом, щоб «проаналізувати ефективність функціонування підприємства, побачити вузькі місця і слабкі сторони в процесі та поліпшити процес, оптимізувати його, якщо це буде необхідно» [7, с. 142].

Отже, в організації бізнес-процес означає серію дій, які виконуються для досягнення певної мети. Ці процеси мають різний рівень складності та тривалості, починаючи від елементарних процедур і закінчуючи широкими стратегічними ініціативами. Бізнес-процеси можуть виконуватися вручну або автоматизовано.

Організації використовують бізнес-процедури, щоб гарантувати ефективне виконання завдань і успішне досягнення бізнес-цілей. Крім того, ці процедури допомагають організаціям у вирішенні низки питань, таких як підвищення продуктивності, скорочення витрат і підвищення якості продуктів і послуг.

Бізнес-процеси можуть бути розглянуті з декількох різних точок зору. Наприклад, їх можна розглядати з точки зору їх структури, протоколів та правил, які визначають послідовність дій. Їх також можна досліджувати з точки зору використання технологій для автоматизації бізнес-процесів, таких як програмні системи та інші.

Шляхом ретельного аналізу часто можна зробити висновок, що здатність покращити бізнес-процедури полягає в їх оптимізації. Таким чином, послідовне виконання дій, які раніше виконувалися паралельно, або шляхом узагальнення та систематизації важливих даних, зібраних на вирішальних етапах бізнес-процесу, може збільшити його швидкість.

Відомо, що спрощення може бути реалізовано як у всьому бізнес-процесі, так і в окремих його підрозділах. Однак існують альтернативні методи вдосконалення бізнес-процесів, які виходять за рамки спрощення. Ці методи передбачають більш глибокі та радикальні втручання в структуру роботи та організацію бізнес-процесів. Злагоджена робота всіх бізнес-процесів – це те, що забезпечує ефективне функціонування будь-якого бізнесу. Значущість кожного бізнес-процесу оцінюється в процесі роботи підприємства.

Бізнес-процеси є важливою складовою діяльності будь-якої організації, оскільки вони дозволяють досягати бізнес-цілей, знижувати витрати та покращувати якість продуктів та послуг. Вони можуть бути довільної

складності та тривалості, від простих процедур до великих стратегічних проектів. Важливим аспектом бізнес-процесів є їх управління та автоматизація, що дозволяє забезпечувати відповідність виконання процесів стандартам та регуляторним вимогам. З метою досягнення максимальної ефективності бізнес-процесів, організації повинні визначати та моделювати свої бізнес-процеси, а також використовувати інструменти та підходи для їх управління та автоматизації.

1.2. Огляд і аналіз підходів до застосування бізнес-процесів

Як було зазначено в попередньому підрозділі, бізнес-процес – це логічний, послідовний, взаємопов'язаний набір заходів, який використовує ресурси, створює цінність і видає результат. Бізнес-процеси, пов'язані з отриманням додаткової вартості продукту чи послуги, поділяються на такі категорії, як базові бізнес-процеси; пропонувані бізнес-процеси, тобто ті, які не додають вартості продукту, а додають його вартість, і процеси управління бізнесом.

Бізнес-процеси є невід'ємною частиною будь-якої організації і можуть бути визначені як послідовність дій, які перетворюють вхідні дані в продукт або послугу для клієнта. Застосування бізнес-процесів дозволяє організації підвищити ефективність, знизити витрати та підвищити якість продукту або послуги.

У зв'язку зі складним ринковим середовищем, яке постійно змінюється, підприємствам необхідно постійно вдосконалювати свої системи управління та інформації та їхню підтримку. Сучасні тенденції розвитку інформаційних технологій призводять до зростання складності інформаційних систем у різних галузях економіки [8].

Огляд та аналіз методологічних підходів до застосування бізнес-процесів є важливим для розуміння того, як організації можуть підвищити ефективність та забезпечити успішність своєї діяльності.

Суттєвий аспект створення успішної системи управління підприємством полягає у використанні структурних або процесно-орієнтованих підходів до організації та управління діяльністю.

Структурний підхід використовує різноманітні організаційні структури, зазвичай ієрархічні, для управління та організації діяльності. У цьому підході управління діяльністю здійснюється через такі структурні компоненти, як департаменти, підрозділи, а взаємодії між цими компонентами сприяють такі посадові особи, як керівники департаментів та структурні підрозділи вищого рівня.

Процесний підхід зосереджений на створенні продуктів або послуг, які забезпечують цінність внутрішнім або зовнішнім клієнтам, а не на організаційній структурі підприємства, акцентуючи увагу на бізнес-процесах.

Підхід до управління процесами вимагає переходу до «бережливого виробництва», такої організаційної структури, яка зберігає ресурси. Відповідно до основними ознаками цієї реорганізації є [9, с.56]:

- виконавці наділяються широким колом повноважень і обов'язків;
- мінімізація рівнів, залучених до прийняття рішень;
- зростає увага до забезпечення якості як продуктів чи послуг, так і загальної діяльності підприємства;
- автоматизовано використання технологій для виконання бізнес-процесів.

Під процесним підходом до організації та управління діяльністю компанії передбачає узгодження діяльності компанії з її бізнес-процесами та впровадження системи управління, яка фокусується на кожному бізнес-процесі окремо.

Отже, бізнес-процеси прив'язані до показників ефективності компанії, які можна використовувати для розробки ієрархічного упорядкування показників. Ці метрики згодом побудують комплексну систему оцінки ефективності діяльності компанії та обраних ними бізнес-процесів.

Для оцінки ефективності бізнес-процесів стандартними показниками є [10, стр. 25]:

- кількість виробленої продукції або кількість наданих послуг;
- кількість споживачів, які купують продукцію або користуються послугами;

- кількість типових процедур, необхідних для виробництва продукції протягом певного періоду часу;
- собівартість виробничих витрат на продукцію або послугу;
- тривалість часу, необхідного для типових процедур.

Таким чином, початкова фаза етапу побудови процесного підходу включає в себе аналіз та класифікацію бізнес-процесів компанії. При виборі та класифікації учасників бізнес-процесу важливо визначити їх склад. Значну роль у цьому визначенні відіграє власник бізнес-процесу. Останнім кроком цього процесу є узгодження результатів класифікації як між власниками бізнес-процесів, так і між підприємством у цілому. Отже, класифікація бізнес-процесів зазвичай базується на чотирьох основних категоріях:

- основні бізнес-процеси – ті які орієнтуються на виробництво продукції або надання послуги, та забезпечують одереження доходу компанії;
- забезпечуючі бізнес-процеси – допоміжні процеси, які забезпечують ресурсами усі бізнес-процеси компанії;
- процеси, пов'язані з розвитком бізнесу – ті, що охоплюють весь комплекс функцій розвитку процесу;
- процеси управління бізнесом – ті, що охоплюють весь комплекс функцій управління процесу.

1.3. Висновки до розділу 1

Отже, поняття бізнес-процесів є фундаментальним для розуміння ефективної роботи будь-якої організації. Вони охоплюють взаємопов'язані дії, які перетворюють ресурси на продукти чи послуги, задовольняючи потреби клієнтів і забезпечуючи досягнення бізнес-цілей. У ході аналізу бізнес-процесів стає зрозумілим, що їх правильна організація та управління можуть суттєво підвищити продуктивність, скоротити витрати й поліпшити якість товарів і послуг. Різні наукові підходи, зокрема процесний та структурний, спрямовані на вдосконалення цих процесів, відображають важливість застосування сучасних технологій та інструментів для їх автоматизації. Оптимізація бізнес-процесів є важливим кроком до підвищення ефективності та конкурентоспроможності підприємства,

оскільки вона дозволяє уникнути вузьких місць і забезпечує послідовність у досягненні стратегічних цілей.

РОЗДІЛ 2

ПРОЕКТУВАННЯ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ «ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ ВИРШЕННЯ ЗАДАЧІ УДОСКОНАЛЕННЯ БІЗНЕС-ПРОЦЕСІВ ПІДПРИЄМСТВА»

2.1. Опис концептуальної моделі

При створенні моделі програми було проведено ретельний аналіз відповідної літератури. Це включало літературу з економіко-математичних та математичних економічних концепцій, а також літературу з моделювання процесів. Для забезпечення найточніших розрахунків і точних результатів було застосовано різноманітні методи, такі як системи отримання знань, економіко-математичне моделювання та використання імітаційного моделювання для отримання знань про бізнес-процеси. і сприяти зростанню підприємства. Для розрахунків у цьому методі використовуються балансові показники, які, як правило, оцінюють доцільність виробництва та реалізації продукції підприємством.

Отримана система забезпечує високоточні розрахунки та більш точну оцінку фінансового стану підприємства, одночасно полегшуючи впровадження бізнес-процесів, які можуть допомогти у прийнятті правильних управлінських рішень. Сучасний підхід до управління надає пріоритет організаційним факторам для досягнення успіху та все частіше використовує передові комп'ютерні інструменти для обробки та аналізу даних.

Дослідження економічних процесів передбачає використання як аналітичних, так і статистичних моделей. Кожен тип моделі має свій набір переваг і недоліків. Аналітичні моделі є менш детальними, оскільки вони враховують менше факторів і вимагають певних припущень і спрощень. Однак розрахунки, засновані на аналітичних моделях, легше перевірити та чітко відображають первинні закономірності, властиві цьому явищу. З іншого боку, статистичні моделі є більш точними та детальнішими порівняно з аналітичними моделями. Вони не вимагають грубих припущень і можуть враховувати більшу кількість факторів. Однак ці моделі також мають свої

недоліки, такі як їх громіздкість, високе споживання машинного часу та надзвичайні труднощі в пошуку оптимальних рішень, які потрібно шукати шляхом припущень, проб і помилок.

Метод статистичного моделювання є кращим рішенням для виконання завдань кваліфікаційного проєкту, та в більшості випадків, він є кращим. Але варто не забувати, про те, що проведення тестування за необхідних робочих умов може виявитися непрактичним для реальної системи.

2.2. Аналіз обраної моделі

Одним із найкорисніших ресурсів, доступних нам сьогодні, є комп'ютерне моделювання. Ця технологія є чудовим засобом підтримки та прояснення людської інтуїції. Хоча це не завжди може бути точним відображенням реальності, його можна використовувати для прийняття більш обґрунтованих рішень, ніж якщо б ви поклалися виключно на рішення прийняті людиною.

Метою кваліфікаційної роботи є моделювання виробничого процесу підприємства, яке створює товари різного призначення. Це моделювання можна застосувати до будь-якого типу виробничого продукту чи системи. Він призначений для отримання інформації про бізнес-процеси компанії шляхом обчислення економічних показників ефективності роботи виробництва підприємства.

Також результати цих обчислень візуально представлені у вигляді графіків і статистичних даних, що відображаються на головному екрані застосунку. А отже, алгоритм, який використовується в кваліфікаційному проєкті, базується на принципах імітаційного моделювання та оцінки економіко-математичних показників діяльності підприємства. Уточнюючи, модель заснована на уявному підприємстві, яке створює різноманітні продукти.

Адже, основною метою будь-якої організації є розширення її виробничих можливостей шляхом впровадження ефективних бізнес-процесів, які максимізують економічні вигоди. Для досягнення головної мети підприємство має доступ до наявних коштів, які зберігаються на

банківському рахунку. Перш ніж розпочати виробничий процес, підприємство має спочатку закупити сировину та матеріали, провести необхідні перевірки та випробування, а потім приступити до виробництва. На рисунку 2.1. схематично зображено процес визначення бізнес-процесів підприємства.

З огляду на схему, щоб досягти ефективного виробництва, компанія повинна провести дослідження ринку та встановити ефективні практики. Це передбачає зняття коштів з рахунку для покриття таких витрат, як сировина та закупівля матеріалів, заробітна плата працівникам і транспортні витрати.

Бізнес-процес починається із етапу закупівлі сировини та інших необхідних компонентів виробництва, які потім вводяться у виробничий процес. Придбання сировини відбувається через регулярні проміжки часу відповідно до заздалегідь визначеного графіку підприємства. Після завершення виробництва готова продукція передається на склад. Згодом ця продукція виставляється на ринок. Продаж цих товарів відбувається через певні проміжки часу, а кошти, отримані від продажу, повертаються на рахунок підприємства. А отже, під час визначення предметної області роботи підприємства, було обрано моделі, котрі розраховують різні економіко-математичні показники такі як: рентабельність виробництва, оборотність капіталу, оцінка продукції, обсяг продажів, обчислення собівартості, а також аналіз ліквідності. Та на основі параметрів моделювання та наданих показників роботи підприємства можна буде оцінити ефективність впровадженого бізнес-процесу на підприємстві та визначити очікуваний економічний результат від його реалізації.

Отже, така послідовність дій при визначенні бізнес-процесу допоможе спочатку сформулювати варіанти можливих моделей, провести аналіз даних варіантів моделі та допоможе затвердити більш релевантний варіант обраної моделі. Тому представлена модель допоможе управлінцям компанії прийняти сконструйовану та чітку думку стосовно процесів в підприємстві.

Адже, описаний процес проєктованого застосонку допоможе змусити особу, яка приймає рішення, створювати детальні описи основних причин проблеми. Крім того, спостереження за поведінкою моделі може призвести до появи нових гіпотез про досліджуваний об'єкт.

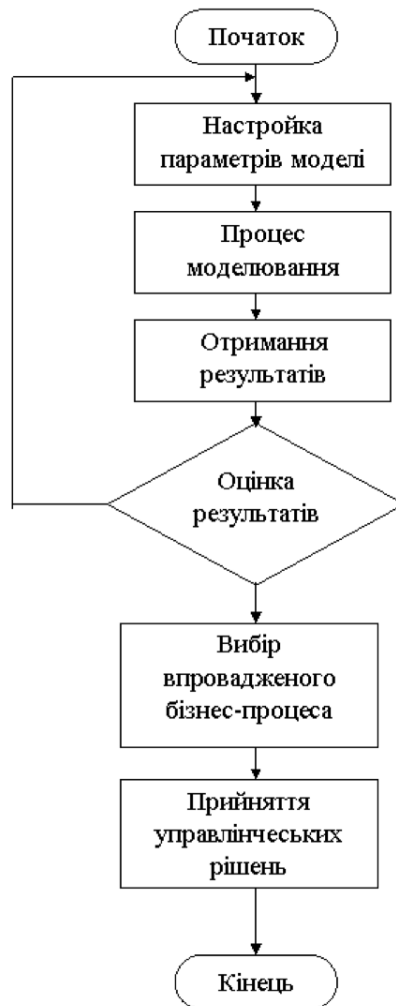


Рис.2.1. Процес визначення бізнес-процесів підприємства

Джерело: побудовано автором

Після досягнення прийняттого рівня надійності можна проводити офіційні експерименти з політикою управління, які забезпечують швидке розуміння ймовірних результатів різних варіантів управління. Хоча операційна модель завжди є формально завершеною, вона ніколи не є справді завершеною з точки зору змісту. Тим не менш, він служить інструментом комунікації, який дозволяє людям, які не брали участі в його створенні, зрозуміти проблему та можливі шляхи вирішення. Змінюючи політику управління та параметри моделі з подальшим аналізом отриманих змін, люди

отримують глибше розуміння того, як розвиваються та еволюціонують реальні системи.

За основу було взято економіко-математичне моделювання. Це є поєднанням складних економіко-математичних дисциплін, що базується на фундаментальних принципах економіки, теорії складних систем і математичних законів. Метою економіко-математичного моделювання є отримання комплексного розуміння економіки, створення економіко-математичної моделі та її оптимізація за допомогою програмного забезпечення [12, стр.24].

В даному методі об'єктом виступає саме економічна система. Тому математична модель економічної системи – це спростований образ, котрий представлений у вигляді математичних рівнянь, логічних співвідношень та графіків.

Оскільки економічні моделі за своєю суттю абстрактні, вони неповні за визначенням. Виокремлюючи характерні моделі, він ігнорує інші змінні, які, незважаючи на їхню, здавалося б, незначну природу, можуть у сукупності чи за конкретних обставин впливати не лише на поведінку суб'єкта дослідження, але й на саму поведінку. Тим не менш, при використанні цього методу розуміння єдиним виходом є припущення, що невивчені фактори мають незначний вплив на суб'єкта, або включити їх у модель і включити в аналіз, якщо це можливо.

Наприклад, у найпростішій моделі попиту передбачається, що визначення попиту на продукт є простим процесом. Вплив на попит не залежить виключно від співвідношення між ціною продукту та доходом його споживачів. Насправді існують інші сприяючі фактори, які мають значний вплив на попит споживачів, такі як їхні переваги та очікування, ціни на інші конкуруючі продукти, реклама та поточні тенденції. Бувають випадки, коли роль цих факторів відіграє більшу роль у визначенні рівня попиту на продукт.

Тому, економічна система є лише одним з аспектів більшої, складнішої соціально-економічної системи, яка сама по собі є динамічною та вірогідною

системою, яка охоплює взаємопов'язані процеси виробництва матеріальних благ, обліку-розподілу та споживання. Та через багаторівневу природу економічних систем будь-яка невизначеність або випадковість у вхідних параметрах на нижчих рівнях може призвести до невизначеності та варіацій у вихідних параметрах підсистем вищого порядку, що зрештою впливає на систему в цілому [13, стр.35].

2.3. Обраний метод аналізу бізнес-процесів

При виконанні кваліфікаційної роботи бізнес-процес в цілому були розглянуті через призму фінансового менеджменту. Адже програмна реалізація будується на визначенні тих фінансово-економічних показників, які визначають ступінь ефективності бізнес-процесу. Тому, за основу програмної реалізації був обраний метод економіко-математичного моделювання.

З метою застосування теоретичних аспектів даної роботи на практиці було обрано метод отримання знань про бізнес-процеси через аналіз ефективності використання коштів та оцінку балансу підприємства. Даний метод передбачає врахування кількох показників, таких як рентабельність продукції, що випускається підприємством, ліквідність інвестованих коштів у продукт, оборотність інвестованих коштів, рентабельність проектів, що виконуються на підприємстві [10, стр.147].

Основний капітал компанії складається з матеріальних активів, які виробляються за допомогою праці та залишаються у своїй натуральній формі протягом тривалого періоду, беручи участь у виробничому процесі та передаючи свою вартість готовій продукції. Основою будь-якого підприємства є основні кошти, якими воно володіє. Ці кошти є рушійною силою виробничих можливостей підприємства і є прямим відображенням його технічної оснащеності. Основні фонди також тісно пов'язані з продуктивністю праці, а також ступенем механізації та автоматизації, що здійснюється у виробничому процесі. Зрештою, собівартість продукції, рівень прибутку та загальна прибутковість компанії визначаються кількістю та якістю наявного основного капіталу компанії.

Концепція прибутку та відповідної йому рентабельності ґрунтується на розподілі витрат, понесених у виробництві та збуті, на постійні та змінні категорії, а також на маржинальний дохід. Цей метод широко впроваджується в країнах зі сформованими ринковими відносинами. Це дає змогу ретельно досліджувати взаємозв'язок між прибутком і вибраними ключовими факторами, тим самим полегшуючи стратегічне управління. Тому варто використовувати систему, що пропонує більш комплексний підхід, який враховує кореляції між показниками та дозволяє точніше вимірювати вплив важливих факторів.

При виконанні завдань кваліфікаційного проєкту та її програмної реалізації було обрано вхідні дані за допомогою яких було виконано аналіз ефективності використання коштів та оцінку балансу підприємства:

- Ціна за одиницю матеріалу;
- Кількість закуповуваних матеріалів для виробництва;
- Кредитні кошти, які взяті в кредит;
- Інші витрати на з/п робітникам, маркетинг, логістику та інше;
- Кількість одиниць реалізованої продукції за 1 виробничий цикл;
- Кількість матеріалу на одиницю реалізованої продукції;
- Ціна одиниці реалізованої продукції;
- Кількість продажу за 1 раз;
- Банк-кошти, які розташовує банк для кредиту;
- Кількість одиниць матеріалу, яка зберігається на складі;
- Склад реалізованої продукції, що не продана ще;
- Власні кошти на рахунку компанії;
- Час основний;
- Час відхилення;
- Межа надмірності – максимально;
- Межа мінімальності – мінімально.

Результати введених даних визначались за допомогою формул, які допомогли визначити показники ефективності використаних коштів та оцінки балансу підприємства. Формули визначення для вихідних даних наведено в таблиці 2.1.

Таблиця 2.1

Формули визначення показників

Назва вихідних даних	Формула визначення
Витрати за період	$Costs = RrchRes \cdot CostPerUnit \cdot Pr R eslter,$ <p>де RrchRes – це кількість придбаних ресурсів, CostPerUnit – вартість за одиницю, – закупівля ресурсів ітерацій.</p>
Витрати на виробництво	$Costs Pr o duct = zp Pr o duct \cdot 12,$ <p>де zpProduct – з/п виробництво, 12 – кількість місяців в році;</p>
Витрати на реалізацію	$CostsRealization = zpRealization \cdot 12 ,$ <p>де zpRealization – з/п реалізації, 12 – кількість місяців в році;</p>
Інші витрати	$OtherCosts = OtherExp \cdot FundsExpI ter ,$ <p>де OtherExp – це інші витрати, FundsExpIter – це кошти на інші витрати ітерацій.</p>
Собівартість одиниці реалізованого продукту	$UnitCost = \frac{(Costs+OtherCosts+Costs Pr o duct+CostRealization)}{made},$ <p>де Costs – це витрати за період, OtherCosts – це інші витрати, CostsProduct – це витрати на виробництво, CostRealization – це витрати на реалізацію, made – кількість виготовлених одиниць.</p>
Загальна собівартість реалізованої продукції	$Cost = UnitCost \cdot made ,$ <p>де UnitCost – це собівартість одиниці, made – кількість виготовлених одиниць.</p>
Виторг, загальна кількість коштів, отриманої при реалізації	$Pr o ceeds = ScopeImpl * Unit Pr i ce,$ <p>де ScopeImpl – це обсяг реалізації, UnitPrice – це собівартість одиниці.</p>

продукції	
Чистий прибуток	$Net Profit = Proceeds - (Costs + OtherCosts + Costs Product + CostsRealization),$ <p>де Proceeds – це виторг, Costs – це витрати за період, OtherCosts – це інші витрати, CostsProduct – це витрати на виробництво, CostRealization – це витрати на реалізацію.</p>
Рентабельність виробництва	$Profitability = \frac{Net Profit}{(Costs+OtherCosts+Costs Product+CostRealization)} \cdot 100,$ <p>де NetProfit– це чистий прибуток, Costs – це витрати за період, OtherCosts – це інші витрати, CostsProduct – це витрати на виробництво, CostRealization – це витрати на реалізацію.</p>
Власні кошти на рахунку компанії	$CompMoney = Money + Net Profit,$ <p>де NetProfit – це чистий прибуток, Money – це власні кошти на рахунку компанії.</p>

Джерело: побудовано автором

2.4. Висновки до розділу 2

Отже, до розробки та аналізу програмної моделі застосовано комплексний підхід: економіко-математичне та імітаційне моделювання бізнес-процесів підприємства дозволили вивчити та оцінити різні аспекти функціонування підприємства. Особливо це стосується впровадженої системи, яка дозволяє правильно розраховувати економічну ефективність бізнесу та приймати більш правильні управлінські рішення, підвищувати якість та швидкість аналізу, полегшувати процес управління виробничими ресурсами, давати глибше розуміння ринкових умов.

Розроблені алгоритм і методика дозволяють визначити низку важливих показників ефективності, в тому числі ліквідність, оборотність капіталу, рентабельність інвестицій, які дають підприємству реальні можливості для більш ефективного управління ресурсами в оптимізації виробничих процесів і забезпеченні більшої економічної стабільності. . Крім того, моделювання

таких процесів як закупівля сировини, виготовлення продукції та її реалізація дозволяє прогнозувати фінансові показники та забезпечує ефективність бізнесу на тривалий період.

Звідси випливає, що такий підхід дозволяє не тільки автоматизувати управлінські рішення, а й забезпечує ефективність використаних коштів та оцінки балансу підприємства, а тому такі показники роботи підприємства є такими за якими можна робити висновок про бізнес-процеси компанії, які впроваджуються на підприємстві або плануються для впровадження.

РОЗДІЛ 3

ПРОГРАМНА РЕАЛІЗАЦІЯ «ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДТРИМКИ ВИРІШЕННЯ ЗАДАЧІ УДОСКОНАЛЕННЯ БІЗНЕС-ПРОЦЕСІВ ПІДПРИЄМСТВА»

3.1. Вибір засобу та мови програмування створення застосунку

Вибір засобів і мов програмування для створення системи «Програмне забезпечення для підтримки вирішення задачі удосконалення бізнес-процесів підприємства» є критичним етапом розробки, оскільки від нього вирішальною мірою залежить успіх і ефективність виконання завдань. Для оптимального вибору інструментів і мов програмування враховуються такі фактори: широкі можливості для реалізації функціональності системи, простота використання, підтримка конкретних бібліотек і модулів, а також наявність документації та підтримка спільноти розробників для вирішення можливих проблем.

Під час аналізу варіантів була обрана мова програмування Python, яка найкраще відповідає вимогам проєкту. Python — це потужна та гнучка мова, яка надає широкий спектр функціональних можливостей і дозволяє швидко розробляти програми. Він також має багатий вибір сторонніх бібліотек, що дозволяє легко реалізувати необхідні функції для обробки даних і візуалізації результатів.

Для реалізації функціональності системи використовуються додаткові бібліотеки, які забезпечують додатковий функціонал та зручний інтерфейс користувача. NumPy є однією з таких бібліотек, яка підтримує маніпулювання багатовимірними масивами даних і математичні операції. Це дозволяє ефективно працювати з числовими даними та виконувати обчислення.

Ще однією важливою бібліотекою є Matplotlib, яка надає зручні інструменти для візуалізації даних у вигляді графіків і діаграм. За допомогою Matplotlib можна відобразити результати обчислень у зручному та легкому для розуміння вигляді, що зручно для користувачів для аналізу та розуміння даних.

Крім того, для реалізації інтерфейсу користувача використовується бібліотека CustomTkinter. Ця бібліотека надає можливість створювати графічні інтерфейси користувача за допомогою Tkinter – стандартної бібліотеки Python для роботи з графічними інтерфейсами. CustomTkinter дозволяє зручно розміщувати елементи у вікні програми та надає можливість взаємодії з користувачем.

Python надає широкі можливості розробки та простоту використання, тоді як бібліотеки надають додаткові функції для маніпулювання даними та візуалізації результатів. Таким чином, програма стає потужним інструментом для аналізу бізнес-процесів, оптимізації витрат і прийняття вирішальних рішень.

3.2. Організація інтерфейсу з користувачем

Організація інтерфейсу користувача є важливою частиною системи реалізації програмного забезпечення бізнес-процесів компанії. Добре організований і зручний інтерфейс дозволяє користувачам легко взаємодіяти з програмою, забезпечуючи легке введення даних, відображення результатів і контроль робочого процесу.

Для досягнення мети ефективної взаємодії інтерфейсу використовується бібліотека CustomTkinter, яка надає можливість створювати графічні інтерфейси користувача. Завдяки цій бібліотеці досягається зручне розміщення та організація елементів у вікні програми. Основними менеджерами розташування елементів у вікні в бібліотеці CustomTkinter є grid та pack. При програмній реалізації продукту було обрано саме grid, бо він має багато переваг у порівнянні з pack. Grid дозволяє розташовувати елементи у вигляді сітки, в якій визначаються рядки та стовпці. Основною перевагою grid є можливість точного контролю над розташуванням елементів. Ви можете вказати, у якому рядку та стовпці має знаходитися кожен елемент, а також визначити його розмір і розміри області, яку він займає. Це дозволяє створювати більш складні та гнучкі макети.

Основні переваги використання grid включають:

- гнучкість: grid дозволяє розміщувати елементи в будь-якому рядку та стовпці, забезпечуючи більш гнучкий макет;
- точний контроль: ви можете вказати точне розташування кожного елемента за допомогою координат рядків та стовпців;
- складні макети: grid дозволяє створювати складні макети з вкладеними рядками та стовпцями, що дозволяє створювати більш структуровані інтерфейси;
- автоматичне налаштування розміру: grid може автоматично налаштовувати розмір елементів залежно від вмісту або вказаних параметрів;
- зручність для праці зі значеннями "sticky": grid надає можливість задавати значення "sticky" для кожного елемента, що визначає спосіб розтягування елемента у вибраному напрямку.

У порівнянні з grid, pack працює на основі концепції розташування елементів у пакеті. Він автоматично розташовує елементи відповідно до їх порядку додавання до контейнера. Pack простий у використанні, але є менш гнучким і не дає такого точного контролю над розміщенням елементів, як grid.

Також для реалізації інтерфейсу користувача були проаналізовані і обрані найбільш підходящі контейнери для організації зручного інтерфейса. Для програмної реалізації продукту було обрано 7 основних контейнерів: Frame, Canvas, LabelFrame, ScrollbarFrame, Text, Entry та Button.

Інтерфейс програми зображено на рисунку 3.1, інтерфейс складається з трьох розділів, які візуально розділяють різні частини функціональності програми та забезпечують зручну організацію робочого простору за допомогою контейнера Frame і організації сітки (grid) 3x2.

У лівій частині інтерфейсу є поля для введення даних користувачем ці поля зроблені за допомогою контейнера Entry, LabelFrame і знаходяться усі ці дані у контейнері ScrollbarFrame. Користувачі можуть заповнювати ці поля відповідно до потреб компанії та вводити такі значення, як ціна одиниці ресурсу, кількість придбаного ресурсу, інші витрати та кількість виробничого продукту. Ці поля дозволяють користувачам налаштовувати вхідні дані відповідно до поточної ситуації в компанії. У цьому розділі також є кнопка «Змоделювати», вона знаходиться у контейнері Button, для дозволу

натискання на елемент на реалізації `event.listener`. Коли користувач натискає її, вона приймає значення для кожного поля введення даних і передає їх у модуль для обчислення даних, після обчислень дані будуть виведені в іншій частині інтерфейсу.

У середній частині інтерфейсу розміщені графіки, що відображають результати розрахунків і надають візуальне представлення даних, сприяючи аналізу змін в діяльності компанії. Цей розділ інтерфейсу поділений на чотири підрозділи за допомогою контейнера `Frame` і сітки (`grid`) `2x2`, кожен з яких включає конкретну діаграму, що відображає певний аспект роботи компанії. Ці діаграми знаходяться у контейнерах `Canvas`, цей контейнер дозволяє малювати графічні об'єкти, такі як лінії, прямокутники, кола та текст. Він надає можливість створювати власні графічні ефекти.

– У верхньому лівому куті інтерфейсу розташована діаграма «Частка ринку», яка надає користувачам можливість візуально оцінити частку ринку, яку компанія займає в порівнянні зі своїми конкурентами. Ця діаграма дозволяє швидко отримати уявлення про позицію компанії на ринку та визначити її конкурентоспроможність або потенційні слабкі сторони. Через графічне відображення даних, користувач може зрозуміти, яким чином компанія співвідноситься з іншими учасниками ринку і виявити можливі можливості для покращення своєї позиції;

– У верхньому правому куті інтерфейсу знаходиться блок, що складається з трьох графіків, які наочно відображають результати розрахунків. Цей блок надає користувачеві можливість швидко оцінити ключові показники, такі як прибуток, вартість і випуск, а також проаналізувати їх взаємозв'язок і вплив на ефективність компанії. Графіки дозволяють візуально відстежувати зміни в цих показниках та сприяють кращому розумінню фінансового стану та продуктивності компанії. Користувач може аналізувати тенденції, виявляти залежності та приймати обґрунтовані рішення, спираючись на інформацію, що відображена на цих графіках;

– У нижньому лівому куті середньої частини інтерфейсу є графік запасів та ресурсів, яка надає користувачеві візуалізацію використання ресурсів і складу товарів, вироблених під час моделювання. Програма має важливі наслідки для управління запасами та оптимізації виробничих процесів компанії. Він допомагає контролювати рівень забезпеченості ресурсами, виявляти можливу нестачу або надлишок запасів, а також надає

інформацію про структуру та склад готової продукції. Це дозволяє керівництву компанії ефективно планувати закупівлі, управляти виробництвом і забезпечувати оптимальне використання ресурсів, тим самим знижуючи витрати і підвищуючи продуктивність;

– У середній частині програми, в нижньому правому куті інтерфейсу, було розміщено графік під назвою «Розподіл витрат за типом». Ця діаграма є важливим інструментом для аналізу витрат, що дозволяє детально зрозуміти напрямок найбільших витрат у бізнес-процесах компанії. Аналізуючи розподіл витрат за типами, компанії можуть визначити області, де витрати є надмірними або неоптимальними, і зосередити зусилля на зниженні цих витрат. Це може бути переогляд контрактів з постачальниками, вдосконалення виробничих процесів, впровадження енергозберігаючих технологій та інші заходи.

У правій частині інтерфейсу користувачу доступні детальні результати розрахунків, результати знаходяться у контейнері TextVox, це дозволяє отримати повну картину фінансового стану компанії та провести глибокий аналіз її діяльності.

Вихідні дані, які видає програма представлені в таблиці 3.2 пункту 3.3 Вхідні та вихідні дані проєкту.

У правій частині інтерфейсу користувачу доступні детальні результати розрахунків, що дозволяють отримати повну картину фінансового стану компанії та провести глибокий аналіз її діяльності.

Крім того, у правій частині інтерфейсу розміщені кнопки для збереження введених даних користувача або відновлення раніше збережених даних. Це надає зручну можливість порівняти різні моделі бізнес-процесів, зберегти розрахунки для подальшого використання або вибрати найоптимальніші варіанти для компанії. Це сприяє зручності та ефективності в роботі з програмою та допомагає користувачеві приймати обґрунтовані управлінські рішення для покращення фінансової стабільності та прибутковості компанії.

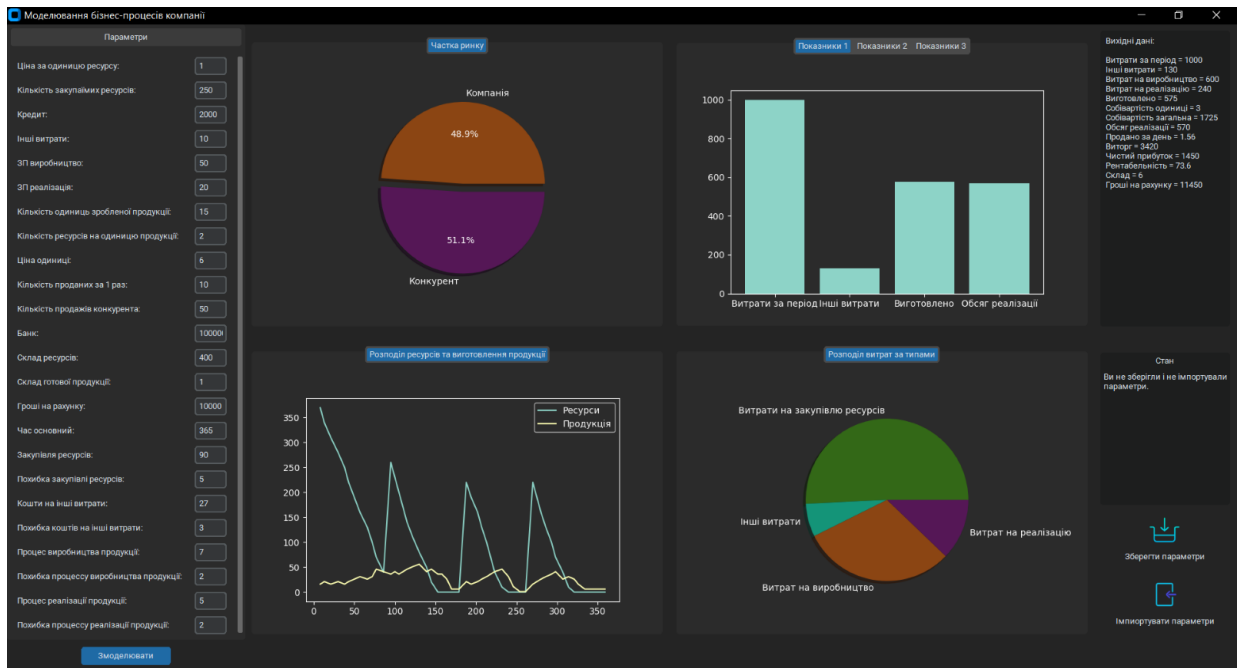


Рис.3.1 Інтерфейс програми для підтримки вирішення задачі удосконалення бізнес-процесів підприємства
Джерело: побудовано автором

3.3. Вхідні та вихідні дані дослідження

Вхідні та вихідні дані є ключовими компонентами процесу аналізу. Вони надають необхідну інформацію для розрахунку та аналізу бізнес-процесів компанії. У таблиці «Введення даних» можна ввести параметри, які впливають на розрахунок. Ці дані можна використовувати як відправну точку для аналізу та визначення найкращої стратегії. Цей підрозділ присвячено вхідним і вихідним даним та інформації, яка надходить в систему для обробки.

Отже, вхідні дані наведено нижче в таблиці 3.1.

Таблиця 3.1

Вхідні дані

Назва	Опис
Ціна за одиницю ресурсу	Ціна за одиницю закупаемого ресурсу
Кількість закупаємих ресурсів	Загальні кількість ресурсів для виробництва товару, яка закупается за один цикл закупівлі ресурсів.

Кредит	Кількість грошей, яку потрібно брати в кредит
Інші витрати	Витрати грошей не рекламу, маркетингові дослідження, дрібні ремонти та покупки і непередбачувані витрати
Зарплатня виробництво	Витрати грошей на зарплатню робітникам, які працюють на виробництві товару, за один місяць
Зарплатня реалізація	Витрати грошей на зарплатню робітникам, які працюють на продажі товару, за один місяць
Кількість одиниць зробленої продукції	Кількість одиниць зробленої продукції за виробничий цикл
Кількість ресурсів на одиницю продукції	Кількість матеріалів, яку потрібно витратити для виготовлення однієї одиниці продукції
Ціна одиниці	Ціна за якою буде продана одна одиниця продукції
Кількість проданих за 1 раз	Кількість проданої продукції за один цикл
Кількість продажів конкурента	Кількість, яку продає конкурент за 1 місяць
Банк	Кошти які знаходяться в банку та сума, на яку можна позичити кошти
Склад ресурсів	Кількість ресурсів, яка на даний момент зберігається на складі
Склад готової продукції	Кількість готової продукції, що зберігається на складі в перший виробничий цикл
Гроші на рахунку	Кількість коштів, що є у підприємства на даний момент в банку
Час основний	Час настання події генерації змінної
Закупівля ресурсів	Час настання події закупівлі ресурсів

Кошти на інші витрати	Час наступання події нарахування коштів на інші витрати
Процес виробництва продукції	Час наступання події закінчення виробництва продукції
Процес реалізації продукції	Час наступання події повної реалізації продукту

Джерело: побудовано автором

Після обчислення та обробки вхідних даних програма видасть вихідні дані, які зображені в таблиці 3.2. Результати роботи програми виводяться в графічній інтерпретації (у вигляді графіків) і у вигляді вихідних даних, які користувач може спостерігати на екрані. Вихідні дані дають повне уявлення про фінансові та операційні показники компанії.

Таблиця 3.2

Вихідні дані

Назва	Опис
Витрати за період	Кількість грошей, яка була на закупівлю сировини і матеріалів у процесі моделювання
Інші витрати	Кількість грошей, яка була витрачена на рекламу, маркетингові дослідження, дрібні ремонти та покупки і непередбачувані витрати
Витрат на виробництво	Кількість грошей, яку було витрачено на зарплатню робітникам, які працюють на виробництві товару
Витрат на реалізацію	Сума коштів, витрачених на процес реалізації продукції, включаючи витрати на маркетинг, збут, рекламу, дистрибуцію та інші витрати, пов'язані зі збутом товарів або послуг.

Виготовлено	Кількість виготовленої продукції за час роботи моделі
Собівартість одиниці	Вартість виготовлення або виробництва однієї одиниці продукції, включаючи всі прямі та операційні витрати.
Собівартість загальна	Загальна собівартість продукції
Обсяг реалізації	Кількість реалізованої продукції, за час роботи моделі
Виторг	Сума грошей, отриманих від продажу продукції, включаючи всі доходи, отримані від реалізації.
Чистий прибуток	Різниця між виторгом та загальними витратами, яка вказує на фінансовий результат після врахування всіх витрат та доходів.
Рентабельність	Відношення чистого прибутку до виторгу, що вказує на ефективність та прибутковість діяльності компанії.
Склад	Кількість товарів, яка залишилась на складі на кінець моделювання.
Гроші на рахунку	Кількість ресурсів, яка на даний момент зберігається на складі

Джерело: побудовано автором

Як видно з таблиці, значну частину вихідних даних складають основні показники діяльності підприємств. Ці показники дають змогу зрозуміти бізнес-процеси, які впроваджує підприємство. І зробити висновки щодо подальших дій компанії.

3.4. Приклад роботи та тестування

Щоб візуалізувати і протестувати роботу програми на конкретному прикладі, розглянемо виробничу модель гіпотетичного продукту, наприклад мікросхем Arduino nano. Основні вхідні дані представлені у таблиці 3.3:

Таблиця 3.3

Вхідні тестові дані

Назва	Початкові дані			
	Екперимент 1	Екперимент 2	Екперимент 3	Екперимент 4
Ціна за одиницю ресурсу	2	3	1	2
Кількість закупаємих ресурсів	100	120	200	100
Кредит	2000	2000	2000	2000
Інші витрати	10	20	30	25
Зарплатня виробництва	40	60	100	70
Зарплатня реалізація	20	40	60	40
Кількість одиниць зробленої продукції	10	14	22	22
Кількість ресурсів на одиницю продукції	2	4	3	2
Ціна одиниці	6	8	9	15
Кількість проданих за 1 раз	8	17	22	24
Кількість продажів конкурента	31	46	52	51
Банк	1000000			
Склад ресурсів	400	400	400	400
Склад готової продукції	1	1	1	1
Гроші на рахунку	100000	100000	100000	100000
Час основний	365	365	365	365
Закупівля ресурсів	90	90	90	90

Кошти на інші витрати	30	30	30	30
Процес виробництва продукції	7	7	7	7
Процес реалізації продукції	5	5	5	5

Джерело: побудовано автором

Також хотів би зазначити, що закупівля ресурсів відбувається з інтервалом +/- 5 днів, кошти на інші витрати з інтервалом у +/- 3 дні, процес виробництва продукції має інтервал у +/- 2 дні і процес реалізації продукції також має інтервал у +/- 2 дні.

За період моделювання, а саме 365 календарних днів програма надала результати, які представлені в таблиці 3.4:

Таблиця 3.4

Вихідні тестові дані

Назва	Початкові дані			
	Екперимен т 1	Екперимен т 2	Екперимен т 3	Екперимен т 4
Витрати за період	800	1080	600	800
Інші витрати	130	260	390	325
Витрат на виробництво	480	720	1200	840
Витрат на реалізацію	240	480	720	480
Виготовлено	350	182	330	330
Собівартість одиниці	4	13	8	7
Собівартість загальна	1400	2366	2640	2310
Обсяг реалізації	344	170	330	312
Виторг	2064	1360	2970	4368
Чистий прибуток	414	-1180	60	1923
Рентабельність	25.09	-46.46	2.06	78.65
Склад	7	13	1	19
Гроші на рахунку	10414	8820	10060	11923

Таким чином ми отримали дані які можуть бути проаналізовані експертами. Наприклад, якщо порівняти експеримент 4 з експериментом 1, виявиться, що за рахунок оптимізації кількості закупаємих ресурсів і збільшення вартості одиниці продукції, інших витрат, заробітної плати виробництва, заробітної плати реалізації, кількості виробництва, кількості придбаних ресурсів та інших показників призведуть до збільшення витрат на продукт, але прибутки зросте більше ніж у чотири з половиною рази, а рентабельність зросте втричі.

3.5. Висновок до розділу 3

Визначивши основні критерії вибору інструментів і мов програмування для реалізації системи підтримки бізнес-процесів, Python був обраний через його гнучкість, широкий функціонал і багатий набір бібліотек. Використання таких бібліотек, як NumPy, Matplotlib або CustomTkinter, забезпечує ефективну обробку даних, візуалізацію результатів і зручний інтерфейс.

Якщо проаналізувати таблицю вихідних даних, можна визначити динаміку змін витрат, кількості виготовленої продукції, виторгу та рентабельності. Після детального аналізу накопичених даних експерти чи високопосадовці зможуть зробити важливі висновки та вибрати один із бізнес-процесів як наступний крок для впровадження на підприємстві. Однак для отримання оптимальної виробничої моделі необхідні десятки ітерацій моделювання, щоб визначити максимальний економічний ефект від впровадження бізнес-процесу на підприємстві.

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

Підприємства виробничого сектора України відіграють визначальну роль у формуванні фінансового потенціалу країни. Криза в Україні спричинила серйозні труднощі для багатьох підприємств, оскільки вони стикаються з нестачею фінансових ресурсів, що паралізує їх діяльність. Змінити скрутне фінансове становище бізнесу можна кількома способами: залучення зовнішніх інвестицій, оптимізація використання власних ресурсів, впроваджувати нові інформаційні та виробничі технології, оптимізувати виробничий процес, ефективніше використовувати наявні ресурси підприємства.

Організації працюють на основі своїх бізнес-процесів, які визначаються цілями та завданнями організації. Ці процеси забезпечують реалізацію всіх аспектів діяльності, пов'язаної з виробництвом товарів та наданням послуг, які організація виробляє, продає та постачає, або виконує всі ці функції разом.

Процес виконання виділення та класифікації бізнес-процесів та визначення їх параметрів є окремою задачею високого рівня при переході до процесної організації та управління підприємством. Тому заключним етапом у цій роботі є узгодження результатів класифікації між власниками бізнес-процесів і керівниками підприємства.

Основні етапи процесу моделювання включають аналіз системи, яку ми моделюємо, створення відповідної математичної моделі, розробку імітаційної моделі, оцінку адекватності імітаційної моделі та контроль результатів для екстремальних значень, а також перевірку моделі. Після цього результати моделювання аналізуються для досягнення заданої точності.

Програмний продукт був розроблений мовою програмування Python. Крім того, під час виконання завдання і написання програмного продукту використовуються власноруч розроблені і універсальні бібліотеки, які широко використовуються розробниками. Використавши цей набір бібліотек для створення програмного продукту, який буде корисним для різної

категорії користувачів, таких як керівники вищої та середньої ланки, бізнес-експерти, які займаються впровадженням інновацій та нових інформаційних технологій, аналітики тощо. Розроблений програмний продукт реалізації бізнес-процесів підприємства має значний потенціал для поліпшення ефективності управління та прийняття обґрунтованих рішень. Завдяки можливостям програми, користувач може оперативно аналізувати ключові показники, виявляти тенденції та проблемні моменти, що дозволяє приймати вчасні заходи для оптимізації бізнес-процесів. Результати обчислень та виведені показники надають зрозумілу інформацію, що сприяє прийняттю обґрунтованих рішень та досягненню бажаних результатів у діяльності підприємства.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Business Process Improvement Toolbox.
2. Горлачук В. В. Економіка підприємства.
3. Hammer M., Champy J. Corporation reengineering: a manifesto of revolution in business.
4. Онищенко В.П. Особливості міжнародного маркетингу ХХІ ст. Зовнішня торгівля: економіка, фінанси, право.
5. Schimansky T., UI-library Tkinter, Github, March 2021.
6. GitHub, Beautiful-tkinter-examples.
7. PyCharm, Python IDE Documentation.
8. Matplotlib, Matplotlib 3.9.2 documentation.
9. Хазанова Л. Математичне моделювання в економіці.
10. Harmon P., Tregear R., Business Process Change: A Business Process Management Guide for Managers and Process Professionals.
11. Ситник В.Ф. Каратодава Е.А. Математичні моделі в плануванні й керуванні підприємствами.

ДОДАТКИ

Додаток А

Лістинг прграмного коду

```
main.py
import random

import math

cost_per_unit = 1 #Ціна за одиницю ресурсу

number_of_purchased_resources = 250 #Кількість закупаємих ресурсів

credit = 2000 #Кредит

other_expenses = 10 #Інші витрати

zr_product = 50 #ЗП виробництво

zr_realization = 20 #ЗП реалізація

made_products = 15 #Кількість одиниць зробленої продукції

resources_per_unit = 2 #Кількість ресурсів на одиницю продукції

unit_price = 6 #Ціна одиниці

sold_per_iteration = 10 #Кількість проданих за 1 раз

bank = 1000000 #Банк

composition_of_resources = 400 #Склад ресурсів

composition = 1 #Склад готової продукції

money = 10000 #Гроші на рахунку

time = 365 #Час основний

procurement_of_resources = 90 #Закупівля ресурсів

procurement_of_resources_margin = 5 #Похибка закупівлі ресурсів
```

```

funds_for_other_expenses = 27 #Кошти на інші витрати

funds_for_other_expenses_margin = 3 #Похибка коштів на інші витрати

production_process = 7 #Процес виробництва продукції

production_process_margin = 2 #Похибка процесу виробництва продукції

process_of_selling_products = 5 #Процес реалізації продукції

process_of_selling_products_margin = 2 #Похибка процесу реалізації продукції

def get_values():
    pass

def modulate(cost_per_unit, number_of_purchased_resources, credit, other_expenses,
             zp_product, zp_realization,
             made_products, resources_per_unit, unit_price, sold_per_iteration, sales_competitor, bank,
             composition_of_resources, composition, money, time, procurement_of_resources,
             procurement_of_resources_margin, funds_for_other_expenses, funds_for_other_expenses_margin,
             production_process, production_process_margin, process_of_selling_products,
             process_of_selling_products_margin):
    made = 0 #Виготовлено

    scope_of_implementation = 0 #Обсяг реалізації

    procurement_of_resources_value = procurement_of_resources #Закупівля ресурсів (значення користувача)

    funds_for_other_expenses_value = funds_for_other_expenses #Кошти на інші витрати (значення користувача)

    production_process_value = production_process #Процес виробництва продукції (значення користувача)

```

```

    process_of_selling_products_value = process_of_selling_products #Процес реалізації продукції
(значення користувача)

    procurement_of_resources_iterations = 0

    funds_for_other_expenses_iterations = 0

    production_process_iterations = 0

    process_of_selling_products_iterations = 0

    composition_of_resources_dict = {} # Потрібен для графіка

    composition_dict = {}# Потрібен для графіка

    for day in range(1, time):

        if procurement_of_resources == 0:

            procurement_of_resources_iterations += 1

            # Закупівля ресурсів = Закупівля ресурсів (значення користувача) + (Похибка закупівлі
ресурсів)

            procurement_of_resources = procurement_of_resources_value +
random.randint(procurement_of_resources_margin*-1, procurement_of_resources_margin)

            composition_of_resources += number_of_purchased_resources

        if funds_for_other_expenses == 0:

            funds_for_other_expenses_iterations += 1

            # Кошти на інші витрати = Кошти на інші витрати (значення користувача) + (Похибка коштів
на інші витрати)

            funds_for_other_expenses = funds_for_other_expenses_value +
random.randint(funds_for_other_expenses_margin*-1, funds_for_other_expenses_margin)

        if production_process == 0:

            production_process_iterations += 1

            # Процес виробництва продукції = Процес виробництва продукції (значення користувача) +
(Похибка процесу виробництва продукції)

```

```
production_process = production_process_value + random.randint(production_process_margin*-1,
production_process_margin)
```

```
for i in range(made_products):
```

```
    if composition_of_resources - resources_per_unit >= 0:
```

```
        composition_of_resources -= resources_per_unit
```

```
        composition += 1
```

```
        made += 1
```

```
composition_of_resources_dict[day] = composition_of_resources
```

```
composition_dict[day] = composition
```

```
if process_of_selling_products == 0:
```

```
    process_of_selling_products_iterations += 1
```

```
    # Процес реалізації продукції = Процес реалізації продукції (значення користувача) +  
(Похибка процесу реалізації продукції)
```

```
    process_of_selling_products = process_of_selling_products_value +  
random.randint(process_of_selling_products_margin * -1, process_of_selling_products_margin)
```

```
    if composition - sold_per_iteration >= 0:
```

```
        scope_of_implementation += sold_per_iteration
```

```
        composition -= sold_per_iteration
```

```
procurement_of_resources -= 1
```

```
funds_for_other_expenses -= 1
```

```
production_process -= 1
```

```
process_of_selling_products -= 1
```

```
# Витрати за період моделювання = Ціна за одиницю ресурсу * Кількість закупівельних ресурсів *  
Закупівля ресурсів ітерацій
```

```
costs = cost_per_unit*number_of_purchased_resources*procurement_of_resources_iterations #Витрати  
++
```

```
print(f'Витрати за період моделювання = {costs}')
```

```

# Інші витрати = Інші витрати * Кошти на інші витрати ітерацій

other_costs = other_expenses*funds_for_other_expenses_iterations #Інші ++

print(f'Інші витрати = {other_costs}')

# Витрат на виробництво (ЗП) = ЗП виробництво * кількість місяців у році

costs_product = zp_product * 12

print(f'Витрат на виробництво = {costs_product}')

# Витрат на реалізацію (ЗП) = ЗП реалізація * кількість місяців у році

costs_realization = zp_realization * 12

print(f'Витрат на реалізацію = {costs_realization}')

print(f'Виготовлено = {made}')

# Продано за день = Обсяг реалізації / Час основний

avg_per_day = round(scope_of_implementation / time, 2)

print(f'Продано за день = {made}')

# Собівартість одиниці = (Витрати за період моделювання + Інші витрати) / Виготовлено

unit_cost = math.trunc((costs+other_costs+costs_product+costs_realization)/made) #Собівартість
одиниці ++

print(f'Собівартість одиниці = {unit_cost}')

# Собівартість загальна = Собівартість одиниці * Виготовлено

cost = unit_cost * made #Собівартість ++

print(f'Собівартість загальна = {cost}')

print(f'Обсяг реалізації = {scope_of_implementation}')

# Виторг = Обсяг реалізації * Собівартість одиниці

proceeds = scope_of_implementation * unit_price #Виторг ++

print(f'Виторг = {proceeds}')

# Чистий Прибуток = Виторг - (Витрати за період моделювання - Інші витрати)

net_profit = proceeds - (costs+other_costs+costs_product+costs_realization) #Чистий Прибуток ++

print(f'Чистий прибуток = {net_profit}')

```

```

# Рентабельність = (Обсяг реалізації * (Ціна одиниці - Собівартість одиниці)) / Обсяг реалізації

# profitability = round((scope_of_implementation*(unit_price-unit_cost))/scope_of_implementation, 2)
#Рентабельність ++

profitability = round(net_profit/(costs+other_costs+costs_product+costs_realization) * 100 ,2) #
Рентабельність ++

print(f'Рентабельність = {profitability}\n')

print(f'Склад = {composition}\n')

print(f'Гроші на рахунку = {money+net_profit}\n')

return {'Витрати за період':costs, 'Інші витрати': other_costs, 'Витрат на виробництво':costs_product,
'Витрат на реалізацію':costs_realization, 'Виготовлено':made, 'Продано за день':avg_per_day, 'Собівартість
одиниці':unit_cost,

        'Собівартість загальна':cost, 'Обсяг реалізації':scope_of_implementation, 'Виторг':proceeds,
'Чистий прибуток':net_profit,

        'Рентабельність':profitability, 'Склад':composition, 'Гроші на рахунку':money+net_profit, 'Склад
ресурсів графік':composition_of_resources_dict,

        'Склад продукції графік':composition_dict}

GUI.py import tkinter

import tkinter.messagebox

import customtkinter as tk

import main

from tkinter import filedialog

import matplotlib.pyplot as plt

from PIL import Image

from matplotlib.figure import Figure

from matplotlib.backends.backend_tkagg import (FigureCanvasTkAgg)

import darkdetect

import os

import numpy as np

```

```

import datetime

tk.set_appearance_mode("System") # Modes: "System" (standard), "Dark", "Light"

tk.set_default_color_theme("blue") # Themes: "blue" (standard), "green", "dark-blue"

data = { }

class App(tk.CTk):

    def __init__(self):

        super().__init__()

        # configure window

        self.title("Моделювання бізнес-процесів підприємства")

        self.geometry(f"{1100}x{580}")

        # configure grid layout (4x4)

        self.grid_columnconfigure((0, 3), weight=0)

        self.grid_columnconfigure((1, 2), weight=1)

        self.grid_rowconfigure((0, 1), weight=1)

        # create scrollable frame

        self.sidebar_frame = tk.CTkScrollableFrame(self, width=350, corner_radius=5, label_text="Параметри")

        self.sidebar_frame.grid(row=0, column=0, rowspan=4, sticky="ns")

        self.cost_per_unit = tk.CTkLabel(self.sidebar_frame, text="Ціна за одиницю ресурсу:")

```

```
self.number_of_purchased_resources = tk.CTkLabel(self.sidebar_frame, text="Кількість закупаємих  
ресурсів:")

self.credit = tk.CTkLabel(self.sidebar_frame, text="Кредит:")

self.other_expenses = tk.CTkLabel(self.sidebar_frame, text="Інші витрати:")

self.zp_production = tk.CTkLabel(self.sidebar_frame, text="ЗП виробництво:")

self.zp_realization = tk.CTkLabel(self.sidebar_frame, text="ЗП реалізація:")

self.made_products = tk.CTkLabel(self.sidebar_frame, text="Кількість одиниць зробленої продукції:")

self.resources_per_unit = tk.CTkLabel(self.sidebar_frame, text="Кількість ресурсів на одиницю продукції:")

self.unit_price = tk.CTkLabel(self.sidebar_frame, text="Ціна одиниці:")

self.sold_per_iteration = tk.CTkLabel(self.sidebar_frame, text="Кількість проданих за 1 раз:")

self.sales_competitor = tk.CTkLabel(self.sidebar_frame, text="Кількість продажів конкурента:")

self.bank = tk.CTkLabel(self.sidebar_frame, text="Банк:")

self.composition_of_resources = tk.CTkLabel(self.sidebar_frame, text="Склад ресурсів:")

self.composition = tk.CTkLabel(self.sidebar_frame, text="Склад готової продукції:")

self.money = tk.CTkLabel(self.sidebar_frame, text="Гроші на рахунку:")

self.time = tk.CTkLabel(self.sidebar_frame, text="Час основний:")

self.procurement_of_resources = tk.CTkLabel(self.sidebar_frame, text="Закупівля ресурсів:")

self.procurement_of_resources_margin = tk.CTkLabel(self.sidebar_frame, text="Похибка закупівлі  
ресурсів:")

self.funds_for_other_expenses = tk.CTkLabel(self.sidebar_frame, text="Кошти на інші витрати:")

self.funds_for_other_expenses_margin = tk.CTkLabel(self.sidebar_frame, text="Похибка коштів на інші  
витрати:")

self.production_process = tk.CTkLabel(self.sidebar_frame, text="Процес виробництва продукції:")

self.production_process_margin = tk.CTkLabel(self.sidebar_frame, text="Похибка процесу виробництва  
продукції:")
```

```

self.process_of_selling_products = tk.CTkLabel(self.sidebar_frame, text="Процес реалізації продукції:")

self.process_of_selling_products_margin = tk.CTkLabel(self.sidebar_frame, text="Похибка процесу реалізації продукції:")

# Create entry fields for inputting values

width_of_Entry = 50

self.cost_per_unit_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.number_of_purchased_resources_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.credit_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.other_expenses_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.zp_production_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.zp_realization_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.made_products_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.resources_per_unit_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.unit_price_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.sold_per_iteration_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.sales_competitor_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.bank_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.composition_of_resources_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.composition_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.money_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.time_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.procurement_of_resources_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.procurement_of_resources_margin_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

```

```

self.funds_for_other_expenses_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.funds_for_other_expenses_margin_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.production_process_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.production_process_margin_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.process_of_selling_products_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

self.process_of_selling_products_margin_entry = tk.CTkEntry(self.sidebar_frame, width=width_of_Entry)

# Set the default values for the entry fields

def insert_parameters(cost_per_unit_entry,    number_of_purchased_resources_entry,    credit_entry,
other_expenses_entry,

                    zp_production_entry,    zp_realization_entry,    made_products_entry,    resources_per_unit_entry,
unit_price_entry,

                    sold_per_iteration_entry,    sales_competitor_entry,    bank_entry,    composition_of_resources_entry,
composition_entry,    money_entry,    time_entry,

                    procurement_of_resources_entry,                    procurement_of_resources_margin_entry,
funds_for_other_expenses_entry,    funds_for_other_expenses_margin_entry,

                    production_process_entry,                    production_process_margin_entry,
process_of_selling_products_entry,

                    process_of_selling_products_margin_entry):

self.cost_per_unit_entry.delete(0, 'end')

self.cost_per_unit_entry.insert(0, cost_per_unit_entry)

self.number_of_purchased_resources_entry.delete(0, 'end')

self.number_of_purchased_resources_entry.insert(0, number_of_purchased_resources_entry)

self.credit_entry.delete(0, 'end')

self.credit_entry.insert(0, credit_entry)

self.other_expenses_entry.delete(0, 'end')

self.other_expenses_entry.insert(0, other_expenses_entry)

self.zp_production_entry.delete(0, 'end')

```

```
self.zp_production_entry.insert(0, zp_production_entry)

self.zp_realization_entry.delete(0, 'end')

self.zp_realization_entry.insert(0, zp_realization_entry)

self.made_products_entry.delete(0, 'end')

self.made_products_entry.insert(0, made_products_entry)

self.resources_per_unit_entry.delete(0, 'end')

self.resources_per_unit_entry.insert(0, resources_per_unit_entry)

self.unit_price_entry.delete(0, 'end')

self.unit_price_entry.insert(0, unit_price_entry)

self.sold_per_iteration_entry.delete(0, 'end')

self.sold_per_iteration_entry.insert(0, sold_per_iteration_entry)

self.sales_competitor_entry.delete(0, 'end')

self.sales_competitor_entry.insert(0, sales_competitor_entry)

self.bank_entry.delete(0, 'end')

self.bank_entry.insert(0, bank_entry)

self.composition_of_resources_entry.delete(0, 'end')

self.composition_of_resources_entry.insert(0, composition_of_resources_entry)

self.composition_entry.delete(0, 'end')

self.composition_entry.insert(0, composition_entry)

self.money_entry.delete(0, 'end')

self.money_entry.insert(0, money_entry)

self.time_entry.delete(0, 'end')

self.time_entry.insert(0, time_entry)

self.procurement_of_resources_entry.delete(0, 'end')

self.procurement_of_resources_entry.insert(0, procurement_of_resources_entry)

self.procurement_of_resources_margin_entry.delete(0, 'end')

self.procurement_of_resources_margin_entry.insert(0, procurement_of_resources_margin_entry)
```

```

self.funds_for_other_expenses_entry.delete(0, 'end')

self.funds_for_other_expenses_entry.insert(0, funds_for_other_expenses_entry)

self.funds_for_other_expenses_margin_entry.delete(0, 'end')

self.funds_for_other_expenses_margin_entry.insert(0, funds_for_other_expenses_margin_entry)

self.production_process_entry.delete(0, 'end')

self.production_process_entry.insert(0, production_process_entry)

self.production_process_margin_entry.delete(0, 'end')

self.production_process_margin_entry.insert(0, production_process_margin_entry)

self.process_of_selling_products_entry.delete(0, 'end')

self.process_of_selling_products_entry.insert(0, process_of_selling_products_entry)

self.process_of_selling_products_margin_entry.delete(0, 'end')

self.process_of_selling_products_margin_entry.insert(0, process_of_selling_products_margin_entry)

insert_parameters(1, 250, 2000, 10, 50, 20, 15, 2, 6, 10, 50, 1000000, 400, 1, 10000, 365, 90, 5, 27, 3, 7, 2, 5, 2)

# Pack the input widgets onto the frame

padx_for_text = 10

pady_for_text = 5

padx_for_entry = 2

pady_for_entry = 0

self.cost_per_unit.grid(row=0, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.number_of_purchased_resources.grid(row=1, column=0, padx=padx_for_text, pady=pady_for_text,
sticky="W")

self.credit.grid(row=2, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.other_expenses.grid(row=3, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.zp_production.grid(row=4, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

```

```
self.zp_realization.grid(row=5, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.made_products.grid(row=6, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.resources_per_unit.grid(row=7, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.unit_price.grid(row=8, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.sold_per_iteration.grid(row=9, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.sales_competitor.grid(row=10, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.bank.grid(row=11, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.composition_of_resources.grid(row=12, column=0, padx=padx_for_text, pady=pady_for_text,
sticky="W")

self.composition.grid(row=13, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.money.grid(row=14, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.time.grid(row=15, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.procurement_of_resources.grid(row=16, column=0, padx=padx_for_text, pady=pady_for_text,
sticky="W")

self.procurement_of_resources_margin.grid(row=17, column=0, padx=padx_for_text, pady=pady_for_text,
sticky="W")

self.funds_for_other_expenses.grid(row=18, column=0, padx=padx_for_text, pady=pady_for_text,
sticky="W")

self.funds_for_other_expenses_margin.grid(row=19, column=0, padx=padx_for_text, pady=pady_for_text,
sticky="W")

self.production_process.grid(row=20, column=0, padx=padx_for_text, pady=pady_for_text, sticky="W")

self.production_process_margin.grid(row=21, column=0, padx=padx_for_text, pady=pady_for_text,
sticky="W")

self.process_of_selling_products.grid(row=22, column=0, padx=padx_for_text, pady=pady_for_text,
sticky="W")

self.process_of_selling_products_margin.grid(row=23, column=0, padx=padx_for_text, pady=pady_for_text,
sticky="W")

self.cost_per_unit_entry.grid(row=0, column=1, padx=padx_for_entry, pady=pady_for_entry)
```

```

self.number_of_purchased_resources_entry.grid(row=1,          column=1,          padx=padx_for_entry,
pady=pady_for_entry)

self.credit_entry.grid(row=2, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.other_expenses_entry.grid(row=3, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.zp_production_entry.grid(row=4, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.zp_realization_entry.grid(row=5, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.made_products_entry.grid(row=6, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.resources_per_unit_entry.grid(row=7, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.unit_price_entry.grid(row=8, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.sold_per_iteration_entry.grid(row=9, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.sales_competitor_entry.grid(row=10, column=1, padx=padx_for_text, pady=pady_for_text)

self.bank_entry.grid(row=11, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.composition_of_resources_entry.grid(row=12, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.composition_entry.grid(row=13, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.money_entry.grid(row=14, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.time_entry.grid(row=15, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.procurement_of_resources_entry.grid(row=16, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.procurement_of_resources_margin_entry.grid(row=17,          column=1,          padx=padx_for_entry,
pady=pady_for_entry)

self.funds_for_other_expenses_entry.grid(row=18, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.funds_for_other_expenses_margin_entry.grid(row=19,          column=1,          padx=padx_for_entry,
pady=pady_for_entry)

self.production_process_entry.grid(row=20, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.production_process_margin_entry.grid(row=21, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.process_of_selling_products_entry.grid(row=22, column=1, padx=padx_for_entry, pady=pady_for_entry)

self.process_of_selling_products_margin_entry.grid(row=23,          column=1,          padx=padx_for_entry,
pady=pady_for_entry)

```

```

def get_values():

    global data

    data = {"Ціна за одиницю ресурсу":int(self.cost_per_unit_entry.get()),

            "Кількість закупаємих ресурсів":int(self.number_of_purchased_resources_entry.get()),

            "Кредит":int(self.credit_entry.get()),

            "Інші витрати":int(self.other_expenses_entry.get()),

            "ЗП виробництво":int(self.zp_production_entry.get()),

            "ЗП реалізація":int(self.zp_realization_entry.get()),

            "Кількість одиниць зробленої продукції":int(self.made_products_entry.get()),

            "Кількість ресурсів на одиницю продукції":int(self.resources_per_unit_entry.get()),

            "Ціна одиниці":int(self.unit_price_entry.get()),

            "Кількість проданих за 1 раз":int(self.sold_per_iteration_entry.get()),

            "Кількість продажів конкурента": int(self.sales_competitor_entry.get()),

            "Банк":int(self.bank_entry.get()),

            "Склад ресурсів":int(self.composition_of_resources_entry.get()),

            "Склад готової продукції":int(self.composition_entry.get()),

            "Гроші на рахунку":int(self.money_entry.get()),

            "Час основний":int(self.time_entry.get()),

            "Закупівля ресурсів":int(self.procurement_of_resources_entry.get()),

            "Похибка закупівлі ресурсів":int(self.procurement_of_resources_margin_entry.get()),

            "Кошти на інші витрати":int(self.funds_for_other_expenses_entry.get()),

            "Похибка коштів на інші витрати":int(self.funds_for_other_expenses_margin_entry.get()),

            "Процес виробництва продукції":int(self.production_process_entry.get()),

            "Похибка процесу виробництва продукції":int(self.production_process_margin_entry.get()),

            "Процес реалізації продукції":int(self.process_of_selling_products_entry.get()),

```

```

        "Похибка процесу реалізації продукції":int(self.process_of_selling_products_margin_entry.get())
print(data.values())

values = main.modulate(*data.values())

self.output_frame = tk.CTkTextbox(self)

self.output_frame.grid(row=0, column=3, padx=10, pady=(10,10), sticky="nsew")

self.output_frame.insert("0.0", "Вихідні дані:\n\n"

        f"Витрати за період = {values['Витрати за період']}\n" #Витрати на закупівлю
ресурсів

        f"Інші витрати = {values['Інші витрати']}\n"

        f"Витрат на виробництво = {values['Витрат на виробництво']}\n"

        f"Витрат на реалізацію = {values['Витрат на реалізацію']}\n"

        f"Виготовлено = {values['Виготовлено']}\n"

        f"Собівартість одиниці = {values['Собівартість одиниці']}\n"

        f"Собівартість загальна = {values['Собівартість загальна']}\n"

        f"Обсяг реалізації = {values['Обсяг реалізації']}\n"

        f"Продано за день = {values['Продано за день']}\n"

        f"Виторг = {values['Виторг']}\n"

        f"Чистий прибуток = {values['Чистий прибуток']}\n"

        f"Рентабельність = {values['Рентабельність']}\n"

        f"Склад = {values['Склад']}\n"

        f"Гроші на рахунку = {values['Гроші на рахунку']}")

self.output_frame.configure(state=tk.DISABLED)

plot_top_right(values['Витрати за період'], values['Інші витрати'], values['Виготовлено'],
        values['Собівартість одиниці'], values['Собівартість загальна'],
        values['Обсяг реалізації'], values['Виторг'], values['Чистий прибуток'],
        values['Рентабельність'])

```

```

plot_top_left(values['Виготовлено'])

plot_bot_left(values['Склад ресурсів графік'], values['Склад продукції графік'])

plot_bot_right(values['Витрати за період'], values['Інші витрати'], values['Витрат на виробництво'],
values['Витрат на реалізацію'])

self.output_frame = tk.CTkTextbox(self)

self.output_frame.grid(row=0, column=3, padx=10, pady=(10,10), sticky="nsew")

self.output_frame.insert("0.0", "Вихідні дані:\n\n"

    "Витрати за період = \n"#Витрати на закупівлю ресурсів

    "Інші витрати = \n"

    "Витрат на виробництво = \n"

    "Витрат на реалізацію = \n"

    "Виготовлено = \n"

    "Собівартість одиниці = \n"

    "Собівартість загальна = \n"

    "Обсяг реалізації = \n"

    "Продано за день = \n"

    "Виторг = \n"

    "Чистий прибуток = \n"

    "Рентабельність = \n"

    "Склад = \n"

    "Гроші на рахунку = \n\n")

self.output_frame.configure(state=tk.DISABLED)

self.button = tk.CTkButton(self, text="Змоделювати", command=get_values)

self.button.grid(pady=10)

```

```

if darkdetect.isDark():

    plt.style.use('dark_background')

    plt.rcParams.update({'axes.facecolor': '2b2b2b'})

    thema = '#2b2b2b'

    thema_downloads_frame = '#1d1e1e'

else:

    thema = '#ffffff'

    thema_downloads_frame = 'transparent'

def plot_top_right(costs=0, other_costs=0, made=0, unit_cost=0, cost=0, scope_of_implementation=0,
proceeds=0, net_profit=0, profitability=0):

    # create tabview

    self.tabview = tk.CTkTabview(self)

    self.tabview.grid(row=0, column=2, padx=10, pady=(11, 11), sticky="nsew")

    self.tabview.add("Показники 1")

    self.tabview.add("Показники 2")

    self.tabview.add("Показники 3")

    self.tabview.tab("Показники 1").grid_columnconfigure(0, weight=1) # configure grid of individual tabs

    self.tabview.tab("Показники 2").grid_columnconfigure(0, weight=1)

    self.tabview.tab("Показники 3").grid_columnconfigure(0, weight=1)

    def plot1(costs, other_costs, made, scope_of_implementation):

        data = {'Витрати за період': costs, 'Інші витрати': other_costs, 'Виготовлено': made,

                'Обсяг реалізації': scope_of_implementation}

        courses = list(data.keys())

        values = list(data.values())

        fig = plt.figure(facecolor=f'{thema}')

```

```

# creating the bar plot

plt.bar(courses, values)

# add the axis to the tab using the matplotlib backend for tkinter

canvas = FigureCanvasTkAgg(fig, master=self.tabview.tab("Показники 1"))

canvas.draw()

canvas.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH, expand=2)

def plot2(cost, proceeds, net_profit):

    data = {'Собівартість загальна': cost,

           'Виторг': proceeds, 'Чистий прибуток': net_profit}

    courses = list(data.keys())

    values = list(data.values())

    fig = plt.figure(facecolor=f'{theme}')

# creating the bar plot

plt.bar(courses, values)

# add the axis to the tab using the matplotlib backend for tkinter

canvas = FigureCanvasTkAgg(fig, master=self.tabview.tab("Показники 2"))

canvas.draw()

canvas.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH, expand=2)

def plot3(unit_cost, profitability):

    data = {'Собівартість одиниці': unit_cost, 'Рентабельність': profitability}

    courses = list(data.keys())

    values = list(data.values())

```

```

fig = plt.figure(facecolor=f'{thema}')

# creating the bar plot

plt.bar(courses, values)

# add the axis to the tab using the matplotlib backend for tkinter

canvas = FigureCanvasTkAgg(fig, master=self.tabview.tab("Показники 3"))

canvas.draw()

canvas.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH, expand=2)

plot1(costs, other_costs, made, scope_of_implementation)

plot2(cost, proceeds, net_profit)

plot3(unit_cost, profitability)

plot_top_right()

def plot_top_left(made = 0):

    self.tabview = tk.CTkTabview(self)

    self.tabview.grid(row=0, column=1, padx=10, pady=(10, 10), sticky="nsew")

    self.tabview.add("Частка ринку")

    self.tabview.tab("Частка ринку").grid_columnconfigure(0, weight=1)

def plot(made):

    fig = plt.figure(facecolor=f'{thema}')

    y = np.array([made, int(self.sales_competitor_entry.get()*12)])

    #Витрати на закупівлю ресурсів = Витрати за період

    mylabels = ["Компанія", "Конкурент"]

    mycolors = ["#8b4513", "#551756"]

    myexplode = [0.1, 0]

```

```

plt.pie(y, labels = mylabels, explode = myexplode, colors = mycolors, autopct='%1.1f%%', shadow =
True)

canvas = FigureCanvasTkAgg(fig, master=self.tabview.tab("Частка ринку"))

canvas.draw()

canvas.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH, expand=2)

plot(made)

plot_top_left()

def plot_bot_left(line_1 = {}, line_2 = {}):

    self.tabview = tk.CTkTabview(self)

    self.tabview.grid(row=1, column=1, padx=10, pady=(10, 10), sticky="nsew")

    self.tabview.add("Розподіл ресурсів та виготовлення продукції")

    self.tabview.tab("Розподіл ресурсів та виготовлення продукції").grid_columnconfigure(0, weight=1)

def plot(line_1, line_2):

    fig = plt.figure(facecolor=f'{thema}')

    x1 = line_1.keys()

    y1 = line_1.values()

    x2 = line_2.keys()

    y2 = line_2.values()

    plt.plot(x1, y1, label='Ресурси')

    plt.plot(x2, y2, label='Продукція')

    plt.legend()

```

```

        canvas = FigureCanvasTkAgg(fig, master=self.tabview.tab("Розподіл ресурсів та виготовлення
продукції"))

        canvas.draw()

        canvas.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH, expand=2)

plot(line_1, line_2)

plot_bot_left()

def plot_bot_right(costs=1, other_costs=1, costs_product=1, costs_realization=1):

    self.tabview = tk.CTkTabview(self)

    self.tabview.grid(row=1, column=2, padx=10, pady=(10, 10), sticky="nsew")

    self.tabview.add("Розподіл витрат за типами")

    self.tabview.tab("Розподіл витрат за типами").grid_columnconfigure(0, weight=1)

    def plot(costs, other_costs, costs_product, costs_realization):

        fig = plt.figure(facecolor=f'{thema}')

        y = np.array([costs, other_costs, costs_product, costs_realization])

        #Витрати на закупівлю ресурсів = Витрати за період

        mylabels = ["Витрати на закупівлю ресурсів", "Інші витрати", "Витрат на виробництво", "Витрат на
реалізацію"]

        mycolors = ["#336817", "#149478", "#8b4513", "#551756"]

        myexplode = [0, 0, 0, 0]

        plt.pie(y, labels = mylabels, explode = myexplode, colors = mycolors, shadow = True)

        canvas = FigureCanvasTkAgg(fig, master=self.tabview.tab("Розподіл витрат за типами"))

        canvas.draw()

        canvas.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH, expand=2)

```

```

    plot(costs, other_costs, costs_product, costs_realization)

plot_bot_right()

image_path = os.path.join(os.path.dirname(os.path.realpath(__file__)), "Img")

self.download_image = tk.CTkImage(Image.open(os.path.join(image_path, "download.png")), size=(50, 50))

self.import_image = tk.CTkImage(Image.open(os.path.join(image_path, "import.png")), size=(50, 50))

self.file_frame = tk.CTkFrame(self, fg_color="transparent")

self.file_frame.grid(row=1, column=3, pady=(30,10), padx=10, sticky="nsew")

self.file_frame.rowconfigure((0), weight = 1)

self.file_frame.columnconfigure(0, weight=1)

self.file_output_frame = tk.CTkFrame(self.file_frame, fg_color=thema_downloads_frame)

self.file_output_frame.grid(row=0, column=0, pady=(0, 10), sticky="nsew")

self.file_output_frame.columnconfigure((0,1,2), weight=1)

label = tk.CTkLabel(self.file_output_frame, fg_color="transparent", text=f"Стан", width=30, height=5)

label.grid(row=0, column=1, pady=5, sticky="nsew")

text2 = tk.CTkLabel(self.file_output_frame, fg_color="transparent", text=f"Ви не зберегли і не імпортували параметри.", wraplength=250)

text2.configure(justify="left")

text2.grid(row=1, column=0, padx=5, pady=5, colspan=3, sticky="w")

def save():

    global text2

    try:

```

```

now = datetime.datetime.now()

date_time = now.strftime("%Y-%m-%d_%H-%M-%S")

if not os.path.exists('./Saves/'):

    os.makedirs('./Saves/')

file_path = os.path.join("./Saves/", f'saved_parameters-{date_time}.txt')

with open(file_path, 'w', encoding='utf-8') as f:

    for key, value in data.items():

        f.write(f'{key}: {value}\n')

text2 = tk.CTkLabel(self.file_output_frame, fg_color="transparent",

                    text=f"Файл було збережено.\n"

                    f"Шлях до файлу: {os.getcwd()}\Saves\saved_parameters-{date_time}.txt",

wraplength=200)

text2.configure(justify="left")

text2.grid(row=1, column=0, padx=5, pady=5, columnspan=3, sticky="w")

except:

text2 = tk.CTkLabel(self.file_output_frame, fg_color="transparent",

                    text=f"Виникла помилка при зберіганні файлу.", wraplength=200)

text2.configure(justify="left")

text2.grid(row=1, column=0, padx=5, pady=5, columnspan=3, sticky="w")

self.button_save = tk.CTkButton(self.file_frame, text="Зберегти параметри", image=self.download_image,

                                fg_color="transparent", compound="top", command=save)

self.button_save.grid(row=1, column=0, pady=(0, 10), sticky="ew")

def import_parameters():

    try:

        filename = filedialog.askopenfilename(initialdir=os.getcwd(), title="Select file",

```

```

filetypes=(("Text files", "*.txt"), ("All files", "*.*"))

with open(filename, 'r', encoding='utf-8') as f:

    lines = f.readlines()

imported_data = {}

for line in lines:

    if ':' in line:

        key, value = line.strip().split(': ')

        imported_data[key] = int(value)

self.cost_per_unit_entry.insert(0, 555)

self.number_of_purchased_resources_entry.insert(0, 555)

self.credit_entry.insert(0, 555)

insert_parameters(*imported_data.values())

main.get_values()

text2 = tk.CTkLabel(self.file_output_frame, fg_color="transparent",

                    text=f"Файл було імпортовано.\n"

                    f"Шлях до файлу: {filename}.txt", wraplength=200)

text2.configure(justify="left")

text2.grid(row=1, column=0, padx=5, pady=5, columnspan=3, sticky="w")

except:

text2 = tk.CTkLabel(self.file_output_frame, fg_color="transparent",

                    text=f"При імпортуванні файлу виникла помилка.\n", wraplength=200)

text2.configure(justify="left")

text2.grid(row=1, column=0, padx=5, pady=5, columnspan=3, sticky="w")

```

```
self.button_import = tk.CTkButton(self.file_frame, text="Імпортувати параметри",  
image=self.import_image,  
  
fg_color="transparent", compound="top", command=import_parameters)  
  
self.button_import.grid(row=2, column=0, pady=(10, 10), sticky="ew")
```

```
# create tabview
```

```
if __name__ == "__main__":
```

```
    app = App()
```

```
    app.mainloop()
```

Результати генерації

