

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Сумський державний університет**  
**Центр заочної, дистанційної та вечірньої форм навчання**  
**Кафедра інформаційних технологій**

«До захисту допущено»

В.о. завідувача кафедри

\_\_\_\_\_ Світлана ВАЩЕНКО

\_\_\_\_\_ 2023 р.

**КВАЛІФІКАЦІЙНА РОБОТА**  
**на здобуття освітнього ступеня магістр**

зі спеціальності 122 «Комп'ютерні науки»,

освітньо-професійної програми «Інформаційні технології проектування»

на тему: «Імітаційне моделювання роботи редакційно-видавничого відділу»

Здобувача (ки) групи ІТ.мз-21с \_\_\_\_\_ Нестеренко Марини Володимирівни  
(шифр групи) (прізвище, ім'я, по батькові)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

\_\_\_\_\_ Марини Нестеренко  
(підпис) (Ім'я та ПРІЗВИЩЕ здобувача)

Керівник \_\_\_\_\_ к.т.н., доцент, Яна Чибіряк \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, Ім'я та ПРІЗВИЩЕ) (підпис)

**Суми – 2023**

Сумський державний університет  
Центр заочної, дистанційної та вечірньої форм навчання  
Кафедра інформаційних технологій  
Спеціальність 122 «Комп'ютерні науки»  
Освітньо-професійна програма «Інформаційні технології проектування»

**ЗАТВЕРДЖУЮ**

В.о. завідувача кафедри ІТ

Світлана ВАЩЕНКО

«\_\_\_\_\_» \_\_\_\_\_ 2023 р.

## **ЗАВДАННЯ**

**на кваліфікаційну роботу магістра студентів**

Нестренко Марині Володимирівні

(прізвище, ім'я, по батькові)

**1 Тема кваліфікаційної роботи** «Імітаційне моделювання роботи редакційно-видавничого відділу»

затверджена наказом по університету від «08» листопада 2023 р. № 1249-VI

**2 Термін здачі студентом кваліфікаційної роботи** « 15 » \_\_\_\_\_ грудня \_\_\_\_\_ 2023 р.

**3 Вхідні дані до кваліфікаційної роботи** \_\_\_\_\_  
технічне завдання на розробку імітаційної моделі редакційно-видавничого відділу

**4 Зміст розрахунково-пояснювальної записки (перелік питань, що їй належить розробити)** \_\_\_\_\_  
аналіз предметної області, моделювання та проектування імітаційної моделі, програмна реалізація імітаційної моделі, дослідження роботи моделі

**5 Перелік графічного матеріалу (з точним зазначенням обов'язкових слайдів презентації)** \_\_\_\_\_  
актуальність роботи, постановка задачі, вимоги до моделі, порівняння програмних середовищ моделювання, обґрунтування методів дослідження, проектування імітаційної моделі, створення та підключення бази даних, розробка 3D моделей та введення їх у модель, програмна реалізація імітаційної моделі, дослідження моделі редакційно-видавничого відділу, висновки

**6. Консультанти випускної роботи із зазначенням розділів, що їй стосуються:**

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

Дата видачі завдання \_\_\_\_\_ 25.09.2023 \_\_\_\_\_.

Керівник \_\_\_\_\_  
(підпис)

Завдання прийняв до виконання \_\_\_\_\_  
(підпис)

### КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Аналіз предметної області	25.09.2023-02.10.2023	
2	Вибір методів та засобів реалізації моделі ремонтного підрозділу	03.10.2023-10.10.2023	
3	Планування робіт	11.10.2023-16.10.2023	
4	Моделювання та проектування імітаційної моделі	17.10.2023-23.10.2023	
5	Програмна реалізація імітаційної моделі	24.10.2023-18.11.2023	
6	Дослідження моделі ремонтного підрозділу	19.11.2023-26.11.2023	
7	Написання документації	27.11.2023-01.12.2023	

Магістрант \_\_\_\_\_

Марина НЕСТЕРЕНКО

Керівник роботи \_\_\_\_\_

к.т.н., доц. Яна ЧИБІРЯК

## АНОТАЦІЯ

Тема кваліфікаційної роботи магістра «Імітаційне моделювання роботи редакційно-видавничого відділу».

Пояснювальна записка складається зі вступу, 5 розділів, висновків, списку використаних джерел із 36 найменувань, додатків. Загальний обсяг роботи – 75 сторінок, у тому числі 44 сторінки основного тексту, 5 сторінок списку використаних джерел, 21 сторінка додатків.

Актуальність роботи полягає в наданні рекомендацій щодо підвищення ефективності процесів редакційно-видавничого відділу.

Об'єкт: діяльність редакційно-видавничого відділу

Предмет: модель потоку обробки друкованих видань у редакційно-видавничому відділі.

Мета роботи: розробка імітаційної моделі редакційно-видавничого відділу для дослідження ефективності його роботи.

У першому розділі було проведено аналіз предметної області та доведено що редакційно-видавничий відділ відноситься до систем масового обслуговування.

У другому розділі було деталізовано мету та задачі проекту, проведено аналіз програмних засобів реалізації моделі та обґрунтовано вибір методів дослідження.

У третьому розділі було створено концептуальної моделі процесу розробки та використання імітаційної моделі у нотації IDEF0, діаграма варіантів використання, концептуальну схему імітаційної моделі OFD та ER діаграми для створення бази даних.

У четвертому розділі описано процес створення бази даних за допомогою PhpMyAdmin та MySQL, налаштування візуалізації об'єктів відділу за допомогою Blender та побудови імітаційної моделі редакційно-видавничого відділу.

У п'ятому розділі було досліджено показники ефективності, налаштовано модуль Experimententer, проведено експеримент та проаналізовано його результати, надано рекомендації для підвищення ефективності.

Ключові слова: імітаційна модель, редакційно-видавничий відділ, 3D модель.

## ЗМІСТ

ЗМІСТ .....	5
ВСТУП.....	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ .....	8
1.1 Огляд існуючих проблем .....	8
2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ.....	9
2.1 Мета та задачі дослідження.....	9
2.2 Вибір програмних засобів реалізації.....	12
2.3 Обґрунтування методів дослідження .....	17
3 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ.....	18
3.1. Проектування імітаційної моделі .....	18
3.2 Проектування моделі бази даних.....	24
4 ПРОГРАМНА РЕАЛІЗАЦІЯ ІМІТАЦІЙНОЇ МОДЕЛІ .....	27
4.1 Структуризація даних .....	27
4.2 Розробка тривимірних моделей .....	33
4.3 Програмна реалізація імітаційної моделі.....	34
5 ДОСЛІДЖЕННЯ РОБОТИ СИСТЕМИ .....	41
ВИСНОВКИ.....	48
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	50
ДОДАТОК А. Планування робіт .....	55
А.1 Ідентифікація мети ІТ-проекту .....	55
А.2 Планування змісту структури робіт ІТ-проекту.....	56
А.3 Побудова календарного графіку виконання ІТ – проекту .....	59
ДОДАТОК Б. Лістинг програмного коду .....	64

## ВСТУП

В сучасному світі поліграфічні видання відіграють надзвичайно важливу роль в передачі інформації та сприяють розвитку культури і освіти. Вони використовуються в різних сферах, включаючи мас-медіа, рекламу, освіту, наукові дослідження та багато інших галузей.

Проте виробництво поліграфічних матеріалів пов'язане з численними труднощами, оскільки їх якість вимагає постійного контролю. Помилки під час друку, недоліки кольоропередачі, а також інші проблеми можуть призвести до низької якості поліграфічного видання, що зменшує його привабливість та читабельність. До того ж кожна помилка несе за собою необхідність її виправити, тобто витратити час та ресурси на перевипуск бракованої партії.

Поліграфічне виробництво і так часто вимагає великих фінансових витрат, пов'язаних з придбанням обладнання та сировинних матеріалів. Велика кількість відходів може призвести до надмірних втрат ресурсів. Окрім того замовлення поліграфічних матеріалів часто пов'язані з обмеженими строками. Затримки в виробництві можуть створювати серйозні проблеми для замовників та репутації виробництва.

Моделювання дозволяє проводити експерименти щодо оптимального розподілу праці та ресурсів у підприємствах без значних витрат та ризиків. Завдяки цьому можна виявити та усунути можливі проблеми виробництва перед фактичним запуском процесу, оптимізувати виробничі процеси та підвищити продуктивність. Імітаційне моделювання поліграфічного видання допомагає зменшити витрати, ризики та збільшити якість продукції.

Мета: розробка імітаційної моделі редакційно-видавничого відділу для дослідження ефективності його роботи.

Основні задачі для досягнення поставленої мети:

- Дослідити існуючі проблеми у системах масового обслуговування та виконати аналіз методів дослідження;
- Обрати програмні засоби реалізації імітаційної моделі;

- Розробити концептуальну модель процесів редакційно-видавничого відділу
- Розробити базу даних та реалізувати імітаційну модель системи
- Дослідити роботу редакційно-видавничого, провести експерименти з метою підвищення ефективності роботи

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

## 1.1 Огляд існуючих проблем

Системи масового обслуговування широко використовуються для моделювання та оптимізації різних процесів [12-18], таких як обслуговування клієнтів, виробництво, транспортування та охорона здоров'я, які включають у себе перебування в чергах. Однак створення та впровадження систем масового обслуговування може становити певний виклик, призводячи до виникнення різних проблем. Серед таких може бути трудомісткий збір даних про систему, вибір неправильної моделі, помилки при аналізі моделі, підтримка актуальності моделі систем що швидко розвиваються. Такі проблеми впливають на важливі показники ефективності системи, до яких відносять:

- середнє число зайнятих каналів обслуговування;
- середній час очікування заявки;
- середня довжина черги;
- середнє число заявок у сфері обслуговування.

Редакційно-видавничій відділ відноситься до систем масового обслуговування, оскільки до даної системи надходять заявки на обробку за допомогою наявних у системі каналів обслуговування. При цьому заявки спочатку потрапляють у чергу, з якої по одній надходять до каналу обслуговування та після обробки виходять з системи. Також система є розімкненою системою масового обслуговування, оскільки об'єкти що обслуговуються в СМО надходять ззовні та після обробки залишають систему. Оскільки в системі міститься декілька каналів призначених для одночасного обслуговування різних заявок, система є багатоканальною. Тож проблемні задачі і питання у системах масового обслуговування можна вважати актуальними і для редакційно-видавничого відділу.



## 2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ

### 2.1 Мета та задачі дослідження

Метою даного проекту є розробка імітаційної моделі редакційно-видавничого відділу для дослідження ефективності його роботи.

Для досягнення мети проекту необхідно виконати наступні задачі:

- Дослідити існуючі проблеми у системах масового обслуговування та виконати аналіз методів дослідження;

Необхідно провести аналіз літератури та існуючих досліджень, щоб виявити актуальні проблеми у системах масового обслуговування.

Для вибору методу дослідження необхідно порівняти відомі методи, наприклад аналітичний метод, фізичний експеримент та імітаційне моделювання. Визначити їх сильні та слабкі сторони та обрати той що буде підходити для дослідження даної системи

- Обрати програмні засоби реалізації імітаційної моделі;

Для створення імітаційної моделі порівняємо програмні продукти та їх характеристики, такі як інтеграція з базами даних, можливість розширення функціоналу за допомогою мов програмування, можливість створення звітів, відображення статистичних даних та 3D візуалізація моделі

- Розробити концептуальну модель робочих процесів редакційно-видавничого відділу

Для створення концептуальної моделі використаємо Object Flow Diagram (OFD). OFD є підходящим засобом для візуалізації потоків об'єктів в системі, оскільки він дозволяє точно визначати об'єкти та їх взаємодії, а також забезпечує легкість сприйняття через графічне відображення цих потоків. На діаграмі мають бути відображені усі етапи які проходить видання, такі як реєстрація та визначення параметрів видання головним редактором, вичитування видання редактором відповідної спеціалізації та категорії, повернення авторам

на доробку, повторна вчитка редактором, вчитка коректором, визначення способу друку видання, верстка видання та підготовка оригінал-макету видання до друку

- Розробити базу даних та реалізувати імітаційну модель системи

Для побудови бази даних будемо використовувати реляційну СУБД, оскільки вона є зручнішою для розуміння завдяки структурованості даних у вигляді таблиць. Також завдяки SQL можна будувати запити різної складності, що було б важко реалізувати з нереляційною СУБД. Для збереження даних будемо використовувати 6 таблиць: таблиці «Тип» та «Підтип» будуть містити назву та ймовірність призначення даного параметру, таблиці «Верстання» та «Підготовка оригінал-макету» будуть містити значення для розрахунку часу обробки різних типів видань, а в таблиці «Параметр» будуть записані мінімальні, максимальні та можливі значення різних параметрів для певних типів видань. Також для таблиць будуть визначені первинні та зовнішні ключі та зв'язки між таблицями.

Для реалізації імітаційної моделі необхідно розмістити та поєднати всі елементи, вказані на Object Flow Diagram (OFD), в обраному програмному забезпеченні. Після чого на потрібних етапах присвоювати виданню та елементам системи відповідні значення з бази даних.

- Дослідити роботу редакційно-видавничого відділу, провести експерименти з метою підвищення ефективності роботи

Необхідно визначити ключові показники системи, такі як завантаженість працівників та обсяг черг, для аналізу продуктивності. Аналіз показників ефективності допоможе визначити можливість оптимізації ресурсів та вдосконалення робочих процесів для досягнення оптимальних результатів.

Виходячи з результатів імітації та аналізу показників ефективності, врахувати кращі практики, що виявилися під час імітаційних

експериментів, висунути пропозиції щодо вдосконалення робочих процесів та розробити на їх основі конкретні рекомендації для підвищення ефективності.

## 2.2 Вибір програмних засобів реалізації

Для вибору програмного продукту для створення імітаційної моделі редакційно-видавничого відділу було проведено аналіз наявних програм і підготовлено порівняльну таблицю (таблиця 1.1).

Розглянемо можливості кількох програмних продуктів детальніше. Simio підтримує 3D візуалізацію моделі та відображення статистики, але можливість створення звітів відсутня [1]. Також присутня інтеграція з базами даних OBDB, OLEDB, Oracle, SQL Server та Access. Є можливість розширення функціоналу за допомогою мови програмування C#. Приклад візуалізації моделі виконаної в Simio зображено на рисунку 2.1.

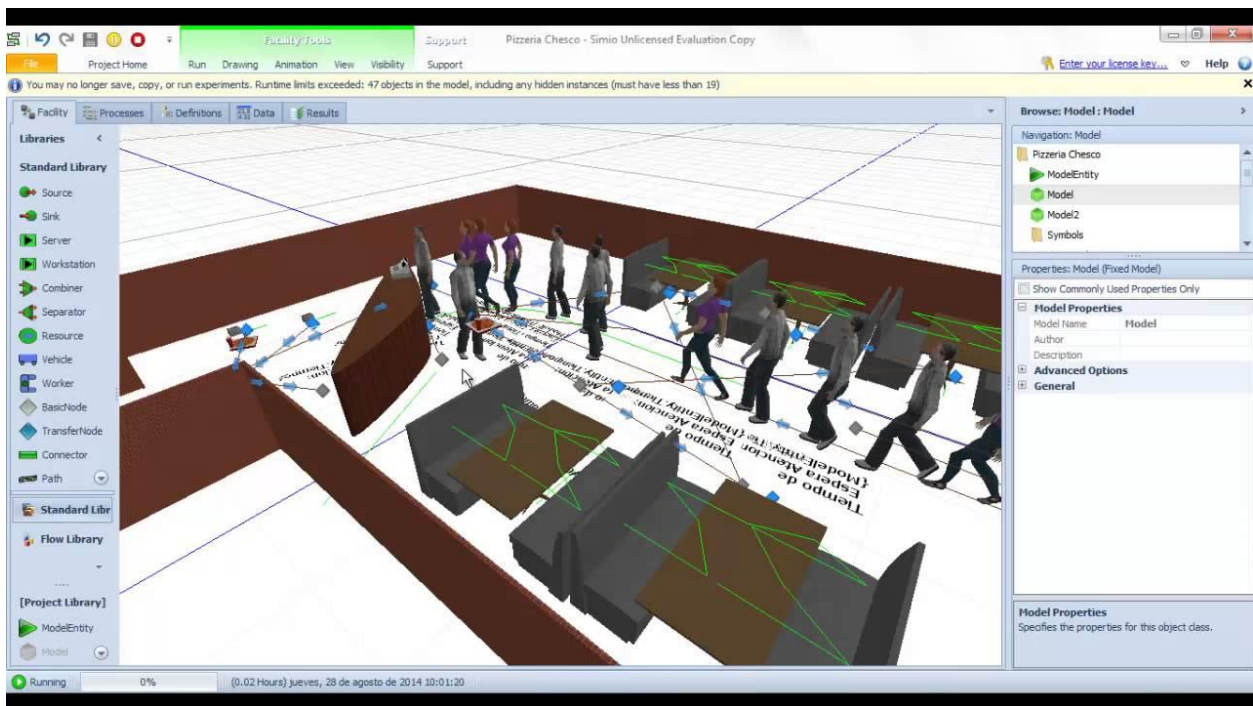


Рисунок 2.1 – Приклад візуалізації моделі в Simio

ProModel підтримує 3D візуалізацію моделі, відображення статистики та творення звітів у Excel [2]. Є можливість інтеграції з базами даних SQL та розширення функціоналу за допомогою мов програмування C# та Visual Basic [3]. Приклад візуалізації моделі виконаної в ProModel зображено на рисунку 2.2.

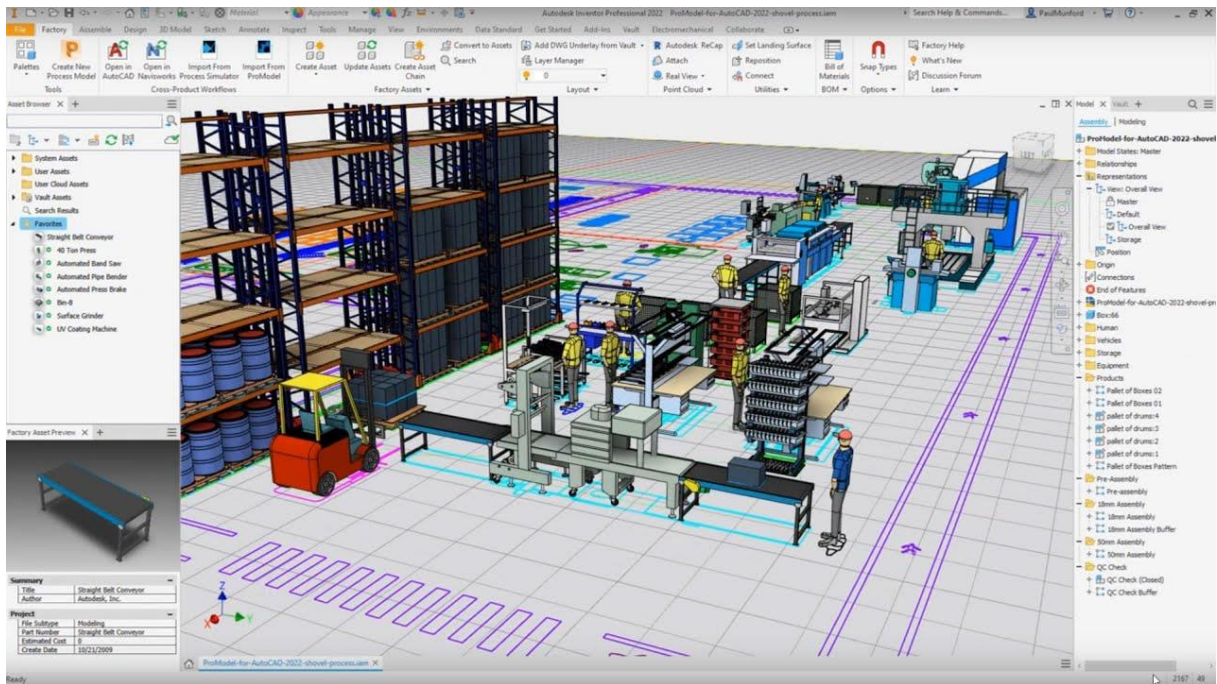


Рисунок 2.2 – Приклад візуалізації моделі в ProModel

Witness (Lanner Group) підтримує 3D візуалізацію моделі, відображення статистики та створення звітів [4-5]. Є можливість інтеграції з базами даних MySQL, OLEDB, Access та розширення функціоналу за допомогою мов програмування C# та Visual Basic. Приклад візуалізації моделі та графічного інтерфейсу виконаних в Witness (Lanner Group) зображено на рисунку 2.3.

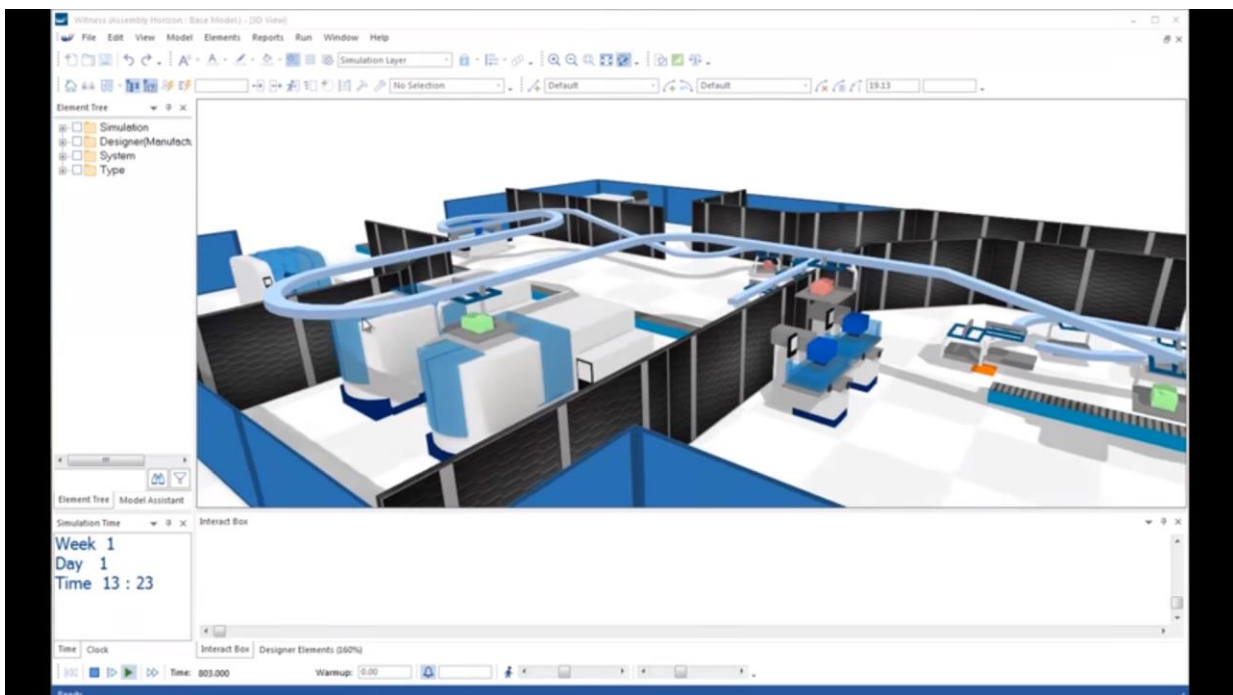


Рисунок 2.3 – Приклад візуалізації моделі в Witness

Technomatrix Plant Simulation підтримує 3D візуалізацію моделі, відображення статистики та творення звітів у HTML та TXT. Є можливість інтеграції з базами даних OBDB, Oracle, SQLite, Access та розширення функціоналу за допомогою мов програмування C та SimTalk [6]. Приклад візуалізації моделі та графічного інтерфейсу виконаних в Technomatrix Plant Simulation зображено на рисунку 2.4.

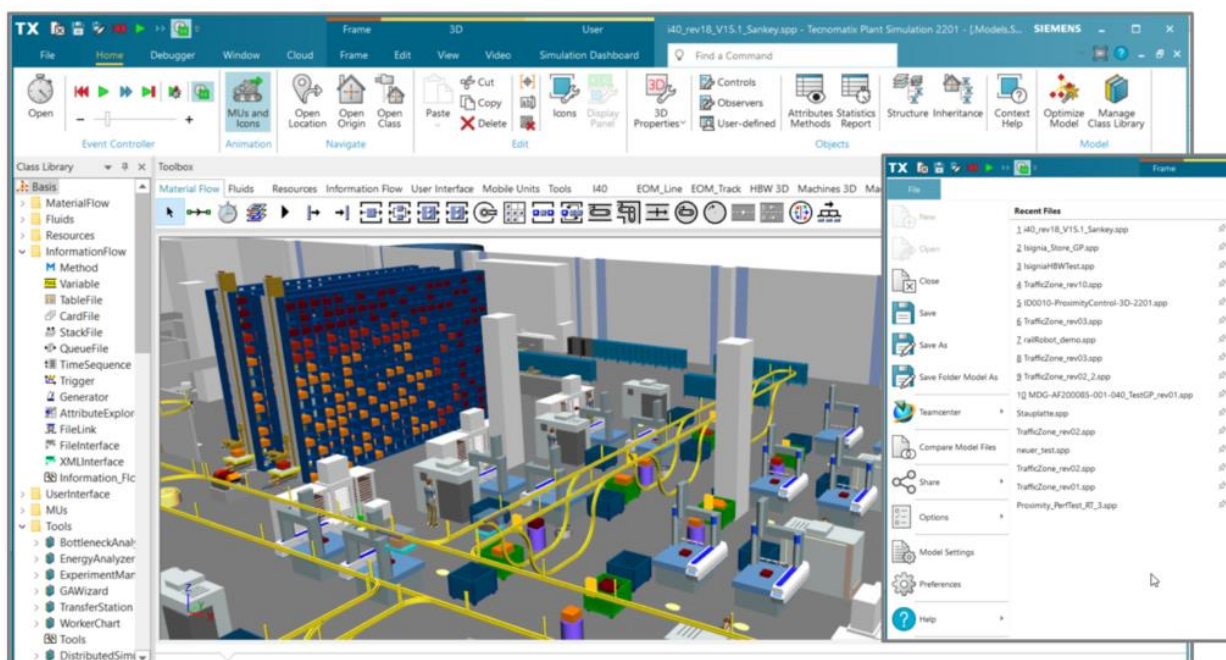


Рисунок 2.4 – Приклад візуалізації моделі в Technomatrix Plant Simulation



Таблиця 2.1 –Порівняльна таблиця продуктів-аналогів

	Simio	ProModel	Withness	Technomatix Plant Simulation	Simcad Pro	FlexSim
3D візуалізація	+	+	+	+	+	+
Відображення статистичних даних	+	+	+	+	+	+
Інтеграція з базами даних	+(OBDB, OLEDB, Oracle, SQL Server, Access)	+(SQL)	+(MySQL, OLEDB, Access)	+(OBDB, Oracle, SQLite, Access)	+(Access, SQLite)	+(SQL Server, MySQL, Oracle)
Розширення функціоналу за допомогою мов програмування	+(C#)	+(C++, Visual Basic)	+(C#, Visual Basic)	+(C, SimTalk)	+(C++, C#, Visual Basic)	+(C++, FlexScript )
Можливість створення інтерфейсу користувача	-	-	-	-	-	+
Створення звітів	-	+(Excel)	+	+(HTML, TXT)	+	+

FlexSim був вибраний для моделювання редакційно-видавничого відділу через його здатність відобразити стан системи та статистичні дані в режимі реального часу [19-21]. Програма також надає можливість підключення до різних баз даних, таких як SQL Server, MySQL та Oracle. Крім того, дане програмне забезпечення дозволяє розширювати свій функціонал за допомогою мови програмування C++, а також забезпечує можливість створення інтерфейсу користувача та генерації звіті. Приклад візуалізації моделі, створеної в FlexSim, представлено на рисунку 2.5.

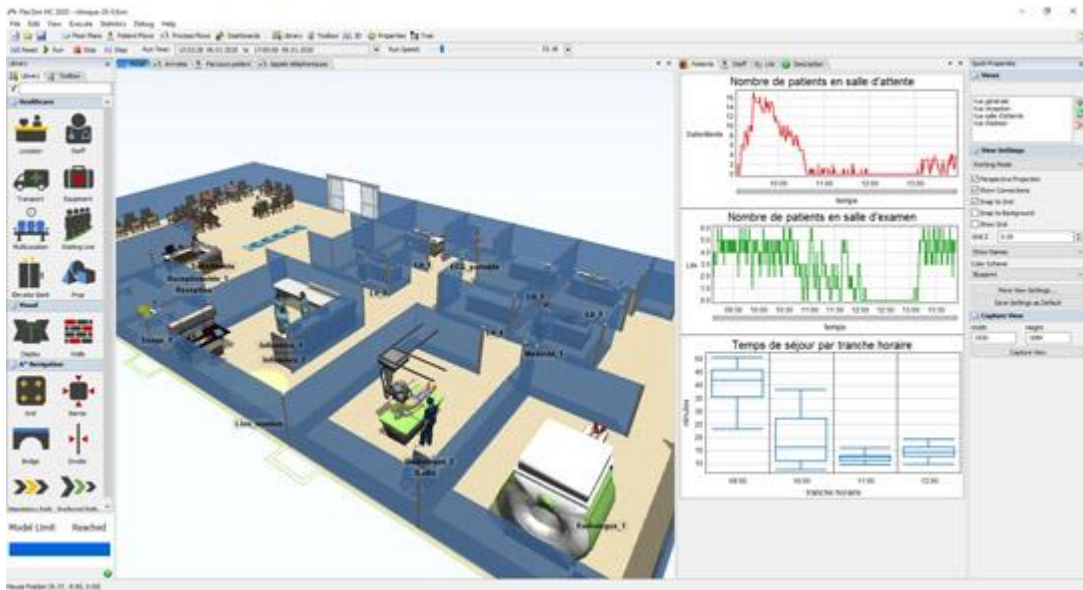


Рисунок 2.5 – Приклад візуалізації моделі в FlexSim



### 2.3 Обґрунтування методів дослідження

Одним з методів дослідження системи є аналітичний метод моделювання — підхід, що передбачає математичний опис об'єкта дослідження. До переваг аналітичного методу включають простоту моделі та її ефективність у знаходженні оптимальних рішень [7, 33-34]. Однак цей метод вимагає значних спрощень, які можуть призвести до неточності результатів [25, 29-31].

Фізичний експеримент вважається одним із перших методів моделювання, який застосовується і дотепер. Фізичні моделі відтворюють систему зі збереженням її фізичних властивостей, надають високу точність та повну інформацію про систему [35-36]. Але є і значні недоліки: високі витрати часу та коштів на створення прототипів та проведення експериментів, особливо з великими та складними об'єктами [8].

Метод імітаційного моделювання базується на побудові детальної моделі системи [24]. Імітаційне моделювання може застосовуватися і для складних систем [9-11], оскільки забезпечує достатній рівень деталізації та дозволяє показати систему у динаміці [26-27]. Це є головною відмінністю імітаційного моделювання від аналітичного. Окрім того велика кількість процесів не може бути описана простими рівняннями, тому робота з випадковими величинами є вагомим перевагою при виборі імітаційного методу моделювання [28]. Також порівняно з фізичною моделлю, на побудову імітаційної моделі витрачається набагато менше часу та коштів. Окрім того імітаційна модель дозволяє швидше змінювати різні параметри системи [22-23], що дає більшу гнучкість у порівнянні з іншими методами моделювання.

Завдяки порівнянню переваг та недоліків різних методів моделювання, для розробки даного проекту було обрано імітаційне моделювання.

## 3 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ

### 3.1. Проектування імітаційної моделі

#### 3.1.1. Побудова концептуальної моделі процесу розробки імітаційної моделі

Застосовуючи методологію IDEF0, виконаємо опис функціонального аспекту імітаційної моделі редакційно-видавничого відділу. Графічне представлення функціонального моделювання імітаційної моделі в нотації IDEF0 подано на рисунку 3.1.

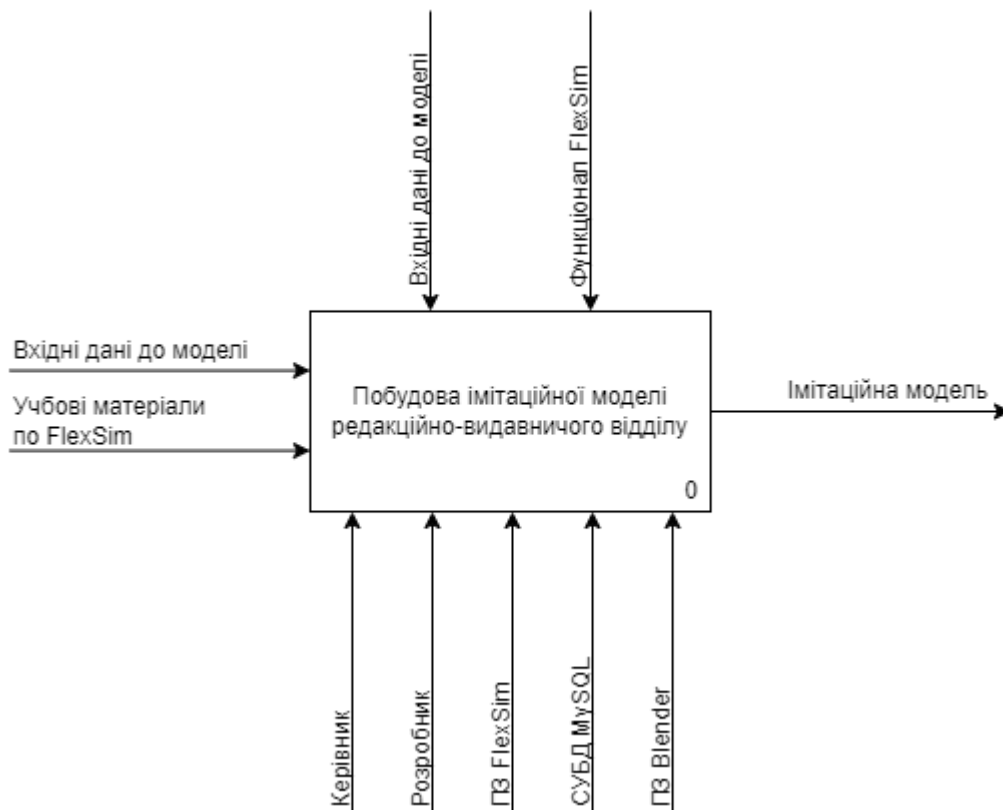


Рисунок 3.1 – Контекстна діаграма процесу розробки імітаційної моделі редакційно-видавничого відділу

Розробимо декомпозицію першого рівня процесу розробки імітаційної моделі редакційно-видавничого відділу (рис. 3.2). Це дозволить визначити послідовність етапів та взаємодію усіх аспектів процесу розробки імітаційної моделі.

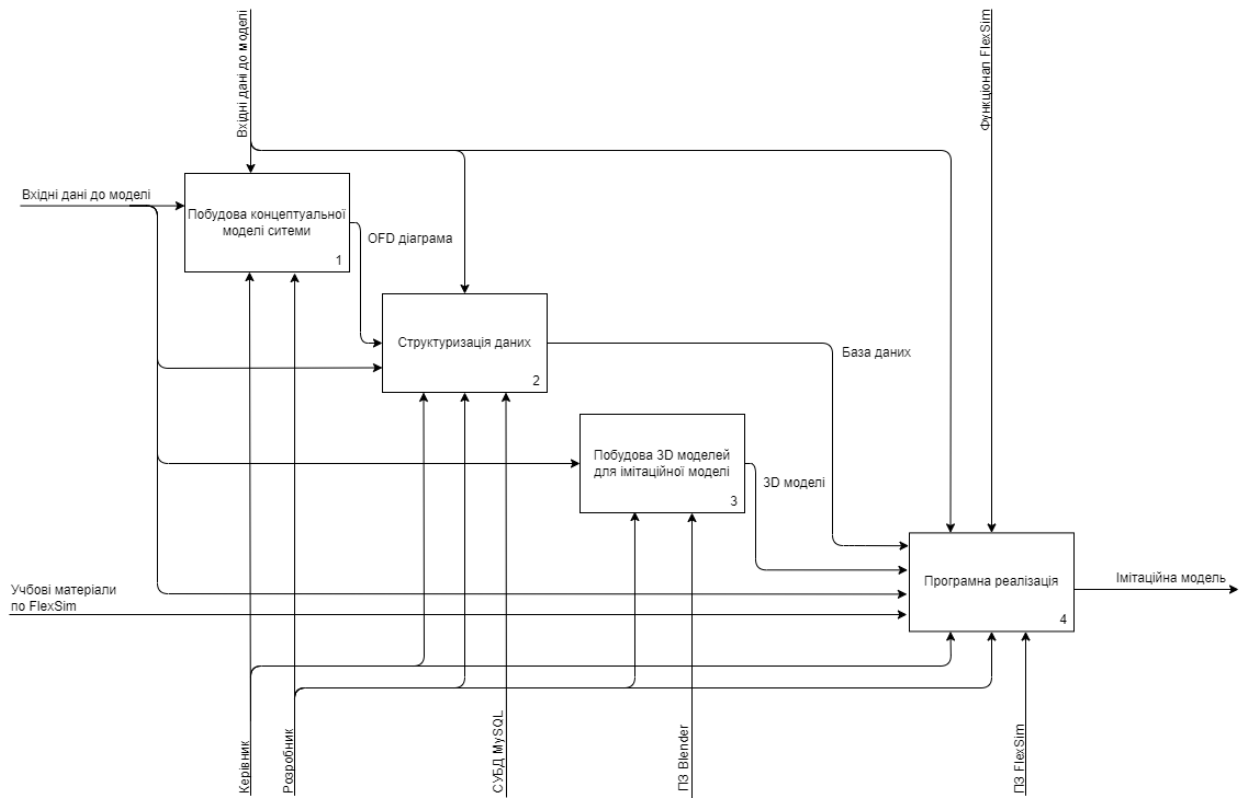


Рисунок 3.2 – Діаграма декомпозиції першого рівня процесу розробки імітаційної моделі

### 3.1.2. Побудова концептуальної моделі процесу використання імітаційної моделі

Побудуємо функціональну модель процесу використання імітаційної моделі (рис. 3.3) та її декомпозицію(рис. 3.4).

Етапи, виділені у діаграмі декомпозиції процесу використання імітаційної моделі:

- запуск імітаційної моделі;
- аналіз показників ефективності
- побудова варіантів прийняття рішення (ПР);
- перевірка варіантів ПР на імітаційній моделі;
- надання практичних рекомендацій.



Рисунок 3.3 – Контекстна діаграма процесу використання імітаційної моделі редакційно-видавничого відділу

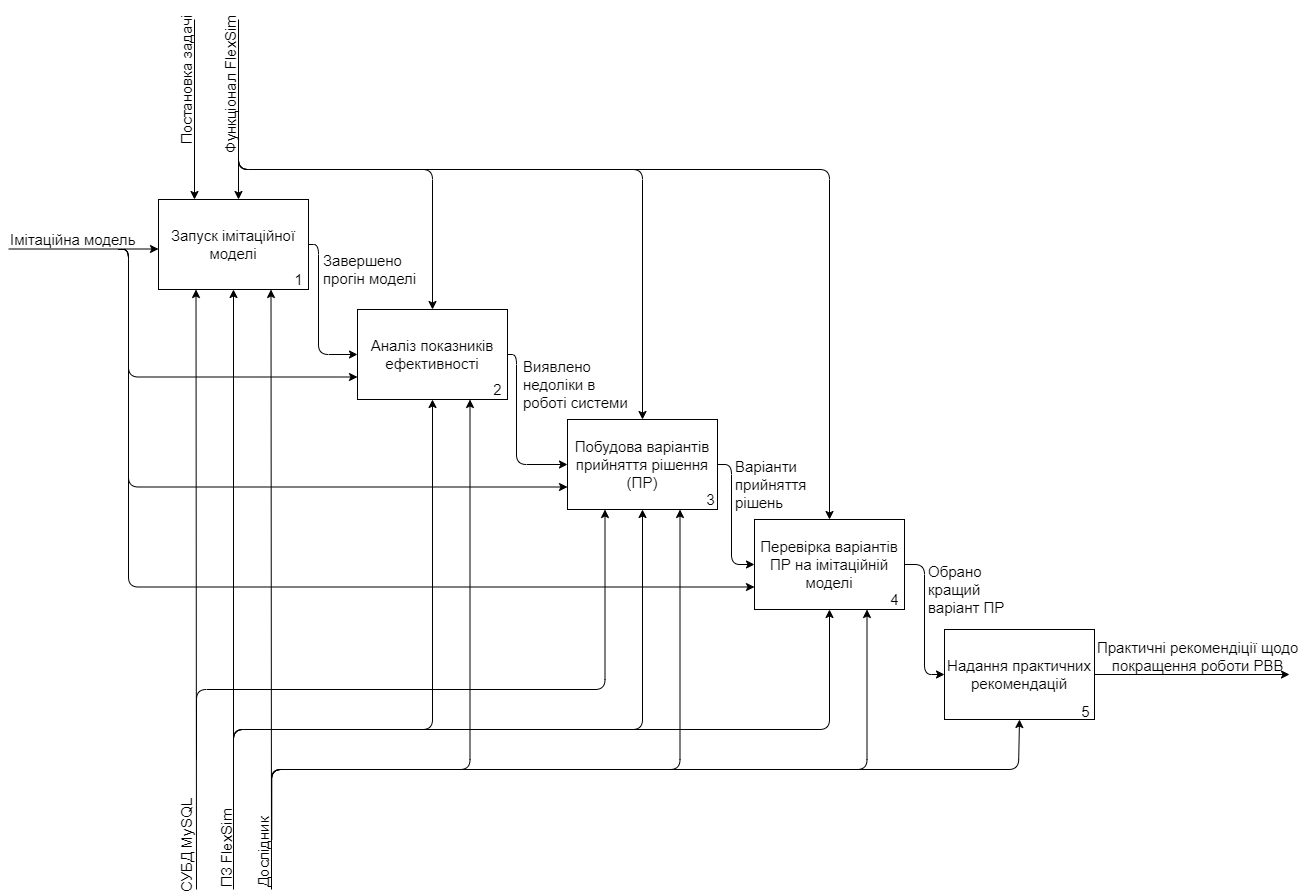


Рисунок 3.4 – Діаграма жеконпозиції першого рівня процесу використання імітаційної моделі

Діаграма варіантів використання імітаційної моделі редакційно-видавничого відділу зображена на рисунку 3.5

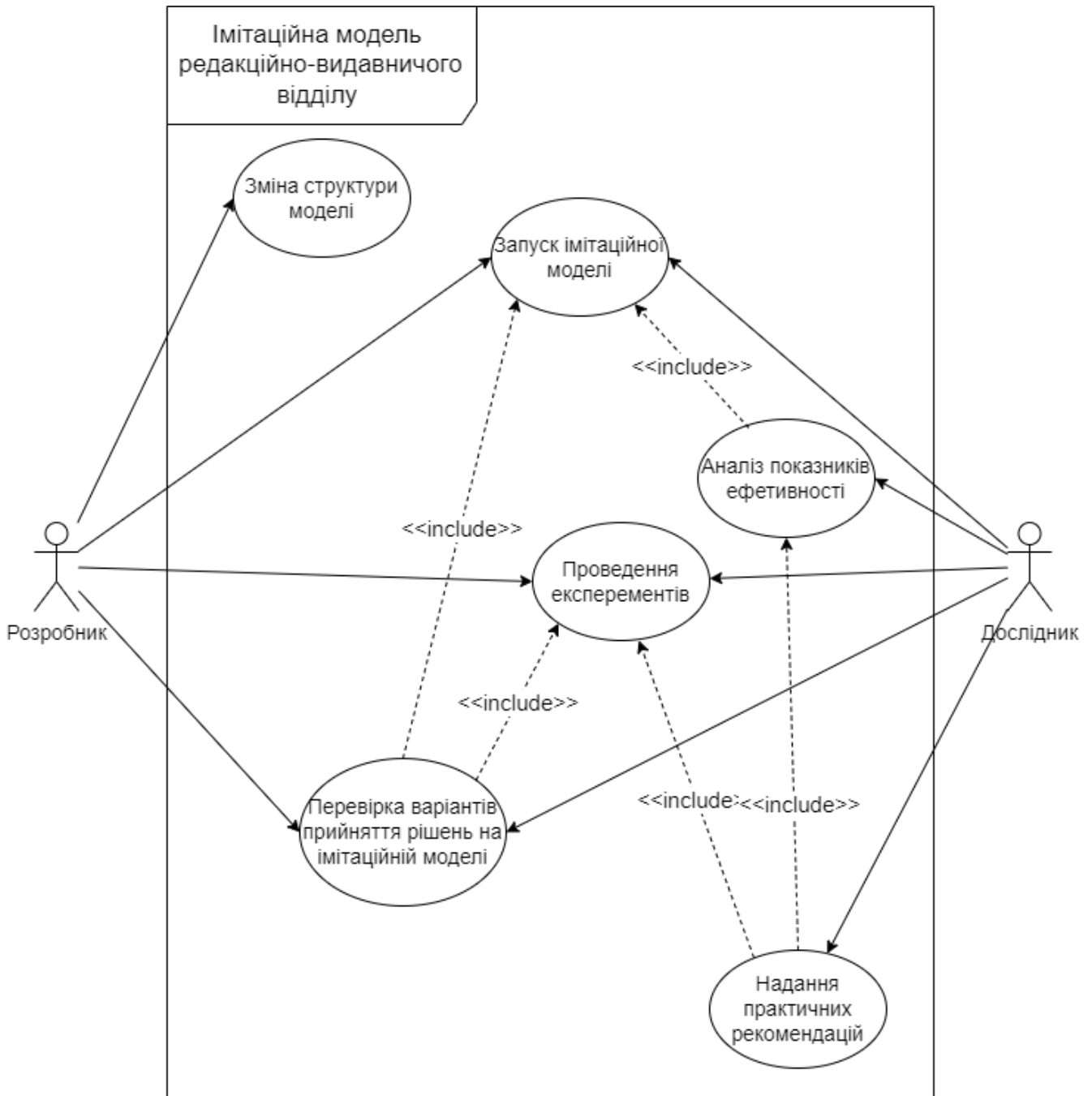


Рисунок 3.5 – Діаграма використання імітаційної моделі редакційно-видавничого відділу

### 3.1.3. Створення концептуальної схеми імітаційної моделі

До відділення надходить автор з рукописом, який потрапляє в чергу до директора. Директор виконує реєстрацію, визначення параметрів майбутнього видання та передає його на обробку одному із шести редакторів, враховуючи їх спеціалізацію та стаж роботи. Спрямування видання має співпадати із спеціалізацією редактора, також наукові видання можуть вичитуватися редакторами I категорії, а ненаукові – редакторами II категорії. При цьому якщо у черзі до редактора II категорії знаходиться більше 20 видань, а науковий редактор вільний від роботи, то певний відсоток видань з черги редактора II категорії переходить у чергу редактора I категорії відповідної спеціалізації. Редактори виправляють помилки пов'язані зі змістом тексту за час, який розраховується на основі формату, розміру шрифту та міжрядкового інтервалу видання.

Після цього рукопис повертається автору на доробку. Від автора видання потрапляє до того ж редактора, який з ним працював і розглядається редактором поза чергою. Час повторного розгляду рукопису також розраховується на основі формату, розміру шрифту та міжрядкового інтервалу видання.

Далі рукопис передається одному з двох технічних редакторів, рівномірно завантажених роботою. На основі кількості сторінок та тиражу визначається спосіб друку видання. Наступним чином відбувається верстання та підготовка оригінал-макету видання до друку. Час за який це відбувається визначається за допомогою формату та типу видання.

В останню чергу відбувається підпис видання до друку головним редактором.

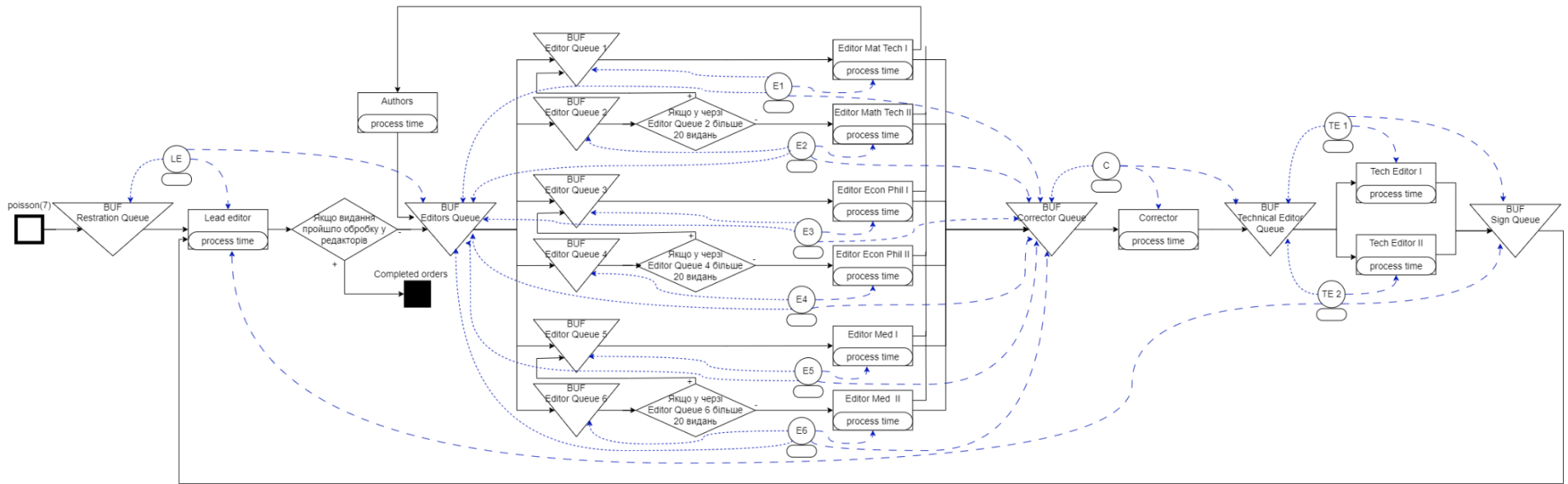


Рисунок 3.6 – OFD діаграма

### 3.2 Проектування моделі бази даних

Для збереження вхідних даних використовуємо базу даних що включатиме наступні таблиці:

- Type – тип видання та ймовірність його надходження до системи;
- Subtype – підтип видання та ймовірність його надходження до системи
- Parameters – параметри видання, такі як спрямування, кількість сторінок, формат видання, міжрядковий інтервал, розмір шрифту, спосіб виготовлення.
- Publication\_layout – параметри формули для визначення часу яке витрачає редактор на верстання видання;
- Preparation\_of\_the\_original\_layout– параметри формули для визначення часу яке витрачає редактор на підготовка оригінал-макету видання до друку.

ER-діаграма, або Entity Relationship Diagram, представляє собою ключовий інструмент для визначення логічної структури баз даних. Ця діаграма допомагає визначити та зобразити зв'язки між сутностями. Концептуальна модель (рис. 3.7) є простою, але потужною візуалізацією, яка моделює дані, що система отримує з вимог до бази даних.

Логічна модель (рис. 3.8) є деталізованою версією концептуальної моделі, включаючи в себе визначення первинних та зовнішніх ключів для кожної сутності. Ця модель надає більше інформації про структуру елементів даних та їх взаємозв'язки, зосереджуючись на важливому аспекті управління даними – ключах, що визначають унікальність записів та забезпечують зв'язки між таблицями.

Фізична модель (рис. 3.9) надає найбільш детальний огляд того, як дані будуть зберігатися та взаємодіяти у реальній базі даних, включаючи таблиці, стовпці, типи даних та ключі.



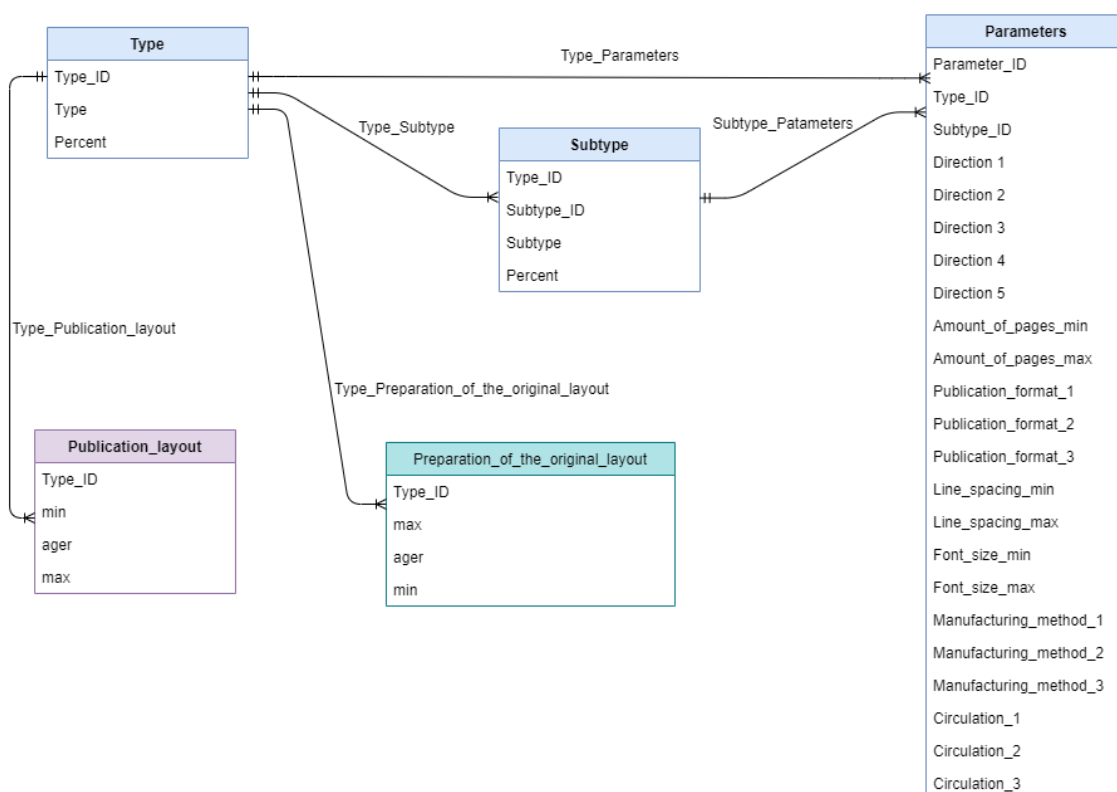


Рисунок 3.7 – Концептуальна модель бази даних

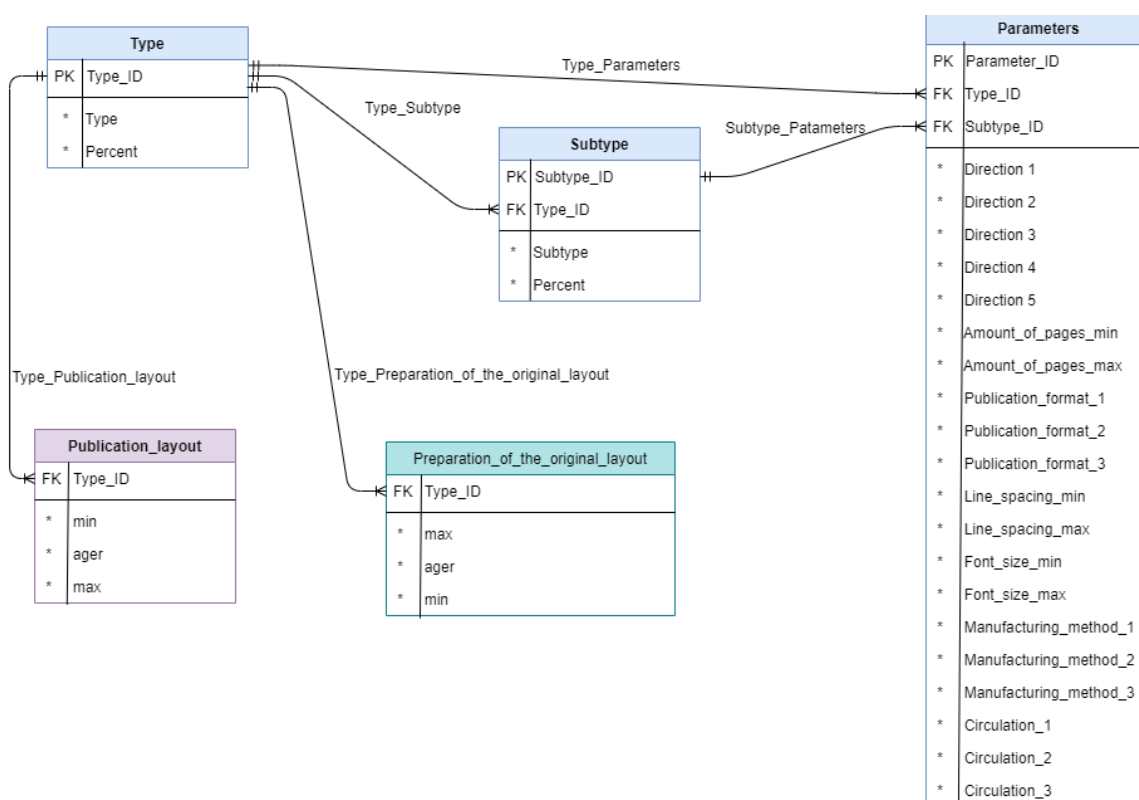


Рисунок 3.8 – Логічна модель бази даних

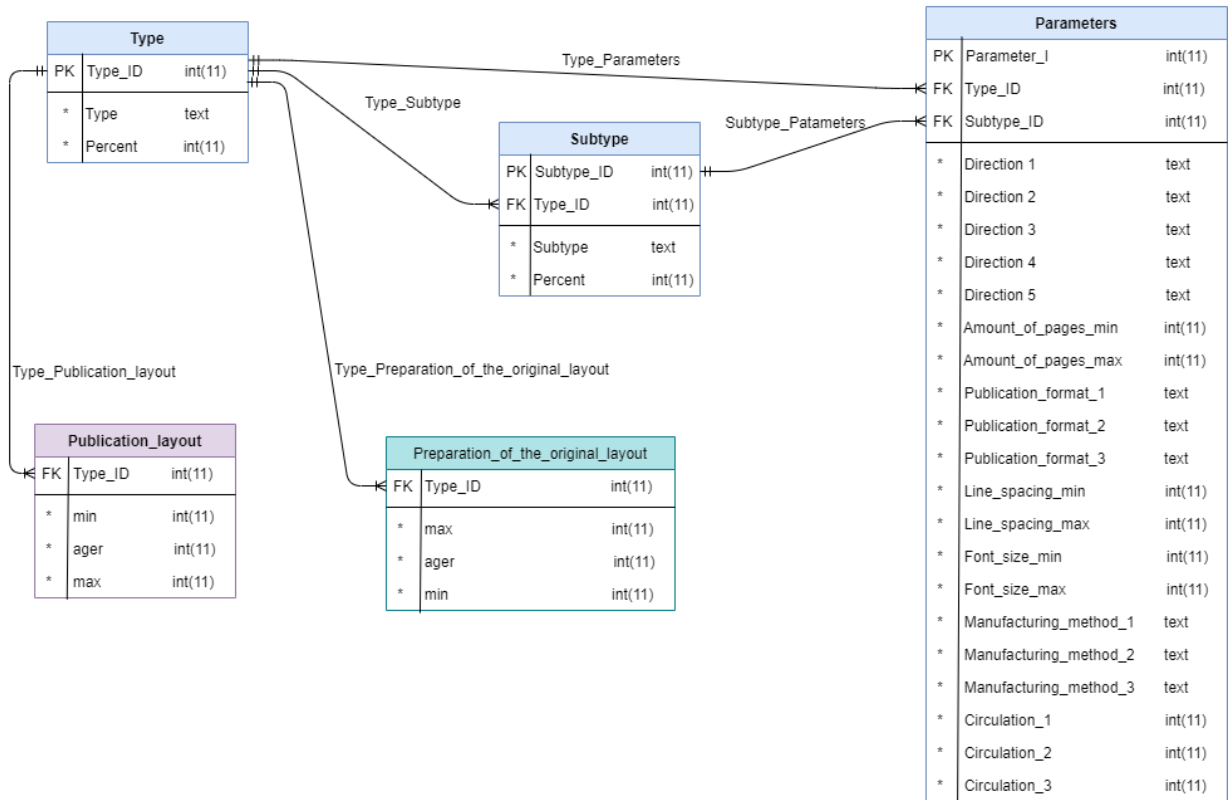


Рисунок 3.9 – Фізична модель бази даних

## 4 ПРОГРАМНА РЕАЛІЗАЦІЯ ІМІТАЦІЙНОЇ МОДЕЛІ

### 4.1 Структуризація даних

Вхідні дані для імітаційної моделі зберігаємо у базі даних, яку створено за допомогою веб-додатку PhpMyAdmin та СУБД MySQL. Таблиці створюємо відповідно до фізичної моделі бази даних, і вносимо дані (див. рис. 4.1-4.6). Позначення стовпців подано в таблиці 4.1.

Таблиця 4.1 –Опис полів таблиць бази даних

Поле таблиці	Опис
Type	Тип видання
Subtype	Підтип видання
Percent	Відсоток надходження певного типу або підтипу видання
direction 1-5	Спрямування видання
Amount_of_pages_min	Мінімальне значення кількості сторінок
Amount_of_pages_max	Максимальне значення кількості сторінок
Publication_format_1-3	Варіанти формату видання
Line_spacing_min	Мінімальне значення міжрядкового інтервалу
Line_spacing_max	Максимальне значення міжрядкового інтервалу
Font_size_min	Мінімальне значення розміру шрифту
Font_size_max	Максимальне значення розміру шрифту
Manufactoring_method_1-3	Варіанти способу виготовлення
Circulation_1-3	Тираж для різних способів виготовлення
min	Значення параметру мінімуму для визначення часу обробки у технічного редактора
ager	Значення параметру середнього значення для визначення часу обробки у технічного редактора
max	Значення параметру максимуму для визначення часу обробки у технічного редактора
method_of_publication	Спосіб друку

Type_ID	Type	Percent
1	official publication	3
2	social and political	12
3	scientific	8
4	popular	11
5	production and practical	9
6	educational	20
7	literary and artistic	12
8	reference	5
9	advertising	20

Рисунок 4.1 – Заповнена таблиця Type

Type_ID	Subtype_ID	Subtype	Percent
1	1	regulatory edition	3
1	2	standards	2
1	3	specifications	50
1	4	instructions	45
3	5	monographs	10
3	6	dissertation abstracts	12
3	7	theses of reports	25
3	8	materials of conferences	25
3	9	collections of scientific papers	13
3	10	preprints	15
5	11	practical advisors	33
5	12	guidelines	33
5	13	methodical manuals	34
6	14	educational programmes	25
6	15	textbooks	12
6	16	manuals	13
6	17	practicums	25
6	18	workbooks	25
8	19	reference books	25
8	20	catalogs	25
8	21	guides	25
8	22	price lists	25
9	23	avenues	40
9	24	posters	20
9	25	booklets	40

Рисунок 4.2 – Заповнена таблиця Subtype

Parameter_ID	Type_ID	Subtype_ID	direction 1	direction 2	direction 3	direction 4	direction 5	Amount_of_pages_min
1	1	1	philological	technical scientific	economical scientific	medical scientific	NULL	20
2	1	2	technical	technical scientific	NULL	NULL	NULL	20
3	1	3	technical	NULL	NULL	NULL	NULL	20
4	1	4	philological	technical	NULL	NULL	NULL	10
5	2	NULL	philological	NULL	NULL	NULL	NULL	10
6	3	5	mathematical scientific	technical scientific	economical scientific	medical scientific	NULL	50
7	3	6	mathematical scientific	technical scientific	economical scientific	medical scientific	NULL	15
8	3	7	mathematical scientific	technical scientific	economical scientific	medical scientific	NULL	25
9	3	8	mathematical scientific	technical scientific	economical scientific	medical scientific	NULL	25
10	3	9	mathematical scientific	technical scientific	economical scientific	medical scientific	NULL	25
11	3	10	mathematical scientific	NULL	NULL	NULL	NULL	8
12	4	NULL	philological	NULL	NULL	NULL	NULL	10

Рисунок 4.3 – Заповнена таблиця Parameters

Type_ID	min	ager	max
1	5	8	10
2	3	5	10
3	7	10	13
4	7	10	13
5	7	10	13
6	7	10	13
7	3	5	10
8	4	8	13
9	10	17	22

Рисунок 4.4 – Заповнена таблиця Publication\_layout

Type_ID	min	ager	max
1	2	4	8
2	1	3	6
3	3	5	7
4	3	5	7
5	3	5	7
6	3	5	7
7	1	3	5
8	2	4	6
9	5	7	9

Рисунок 4.5 – Заповнена таблиця Preparation\_of\_the\_original\_layout

method_of_publication_ID	method_of_publication
1	offset
2	digital
3	stencil

Рисунок 4.6 – Заповнена таблиця Method\_of\_publication

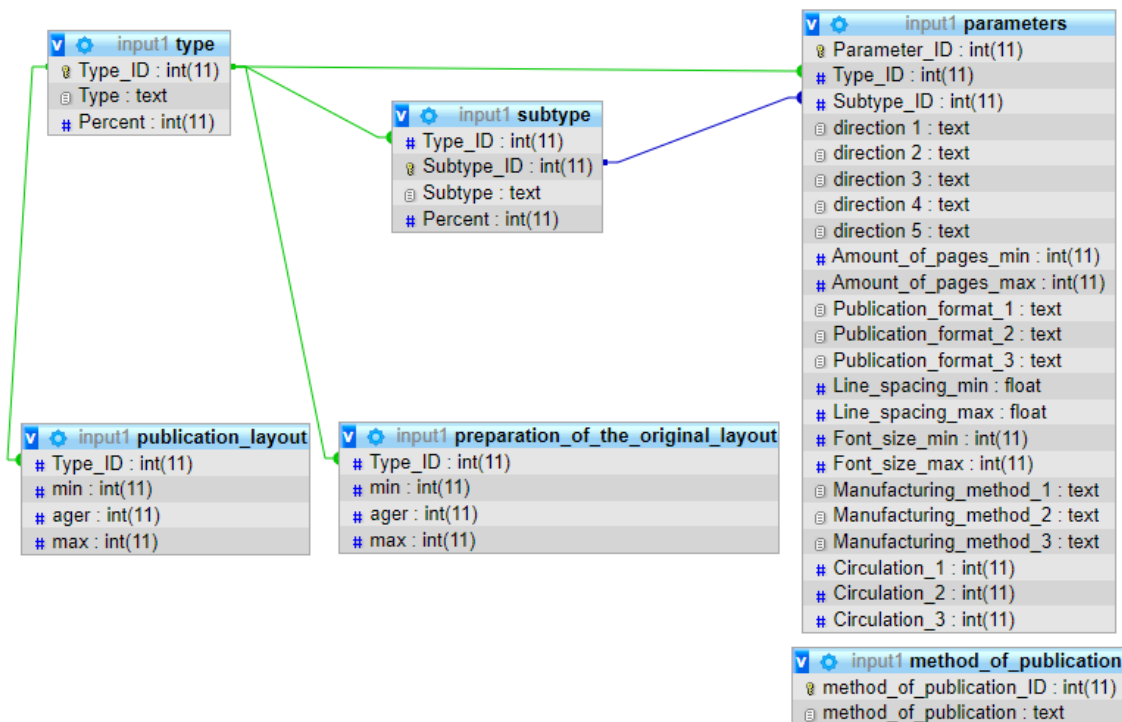


Рисунок 4.7 – Структура бази даних

Підключаємо базу даних до імітаційної моделі за допомогою Database Connector (рис. 4.8) та імпортуємо таблиці (рис. 4.9-4.14).

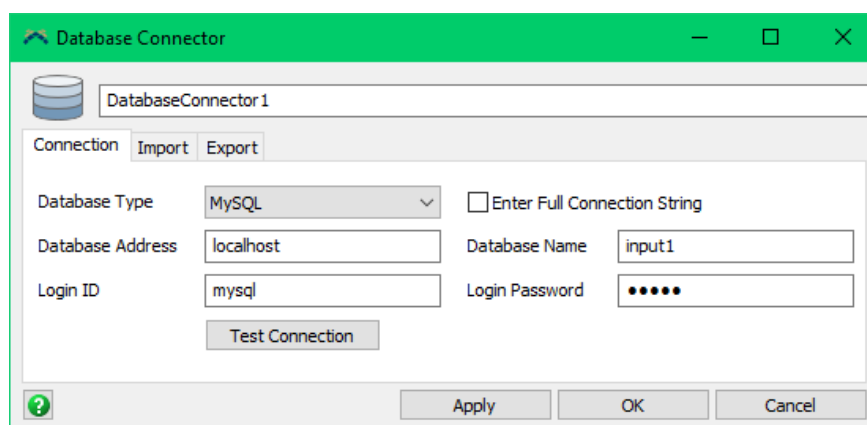


Рисунок 4.8 – Підключення до БД

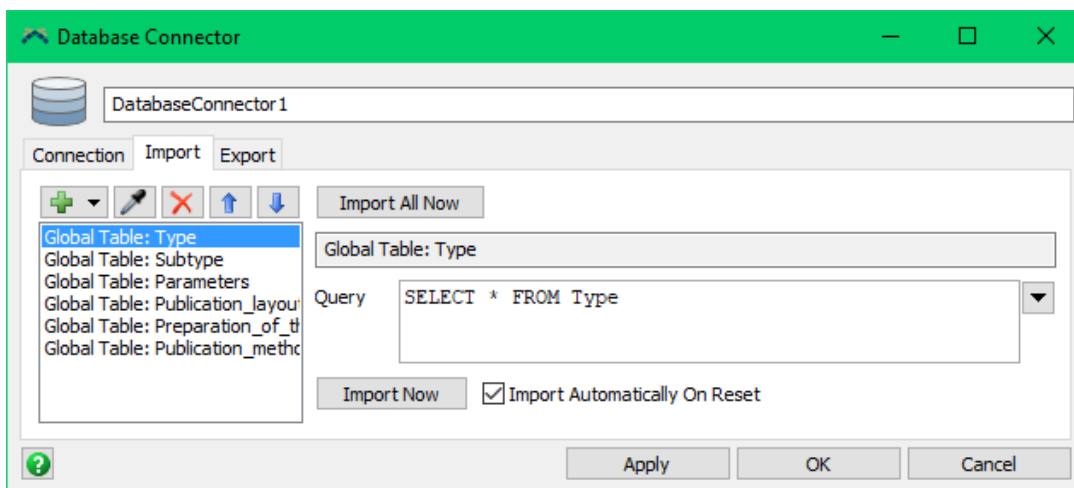


Рисунок 4.9 – Імпорт даних у таблицю Type

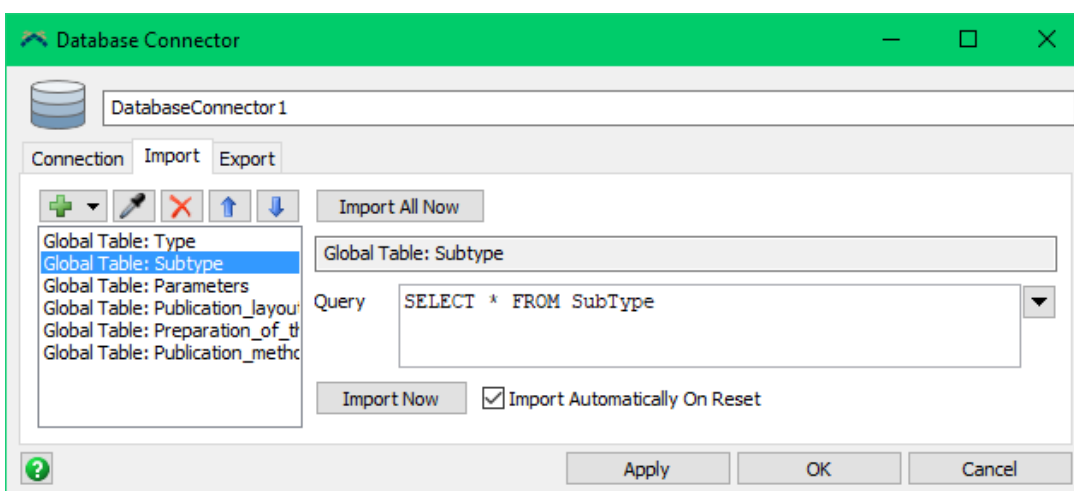


Рисунок 4.10 – Імпорт даних у таблицю Subtype

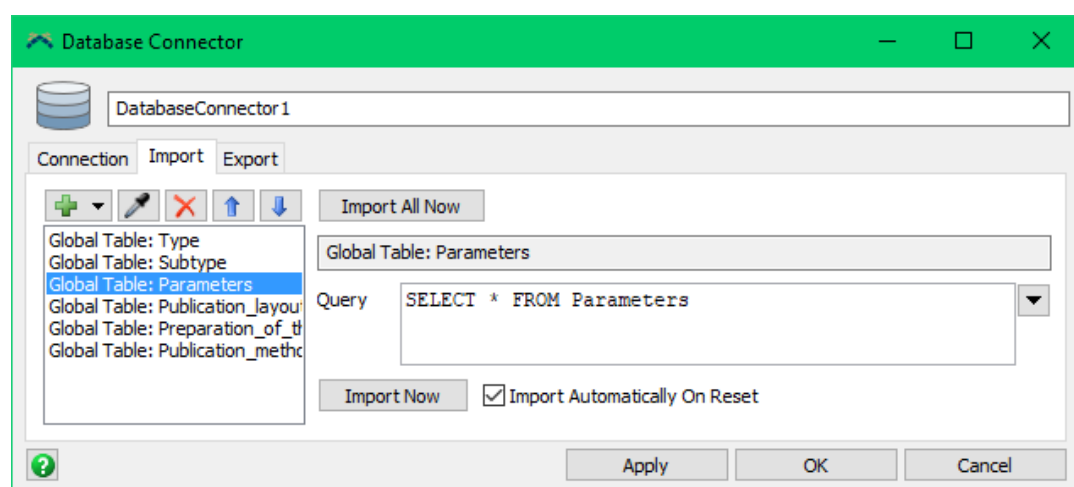


Рисунок 4.11 – Імпорт даних у таблицю Parameters

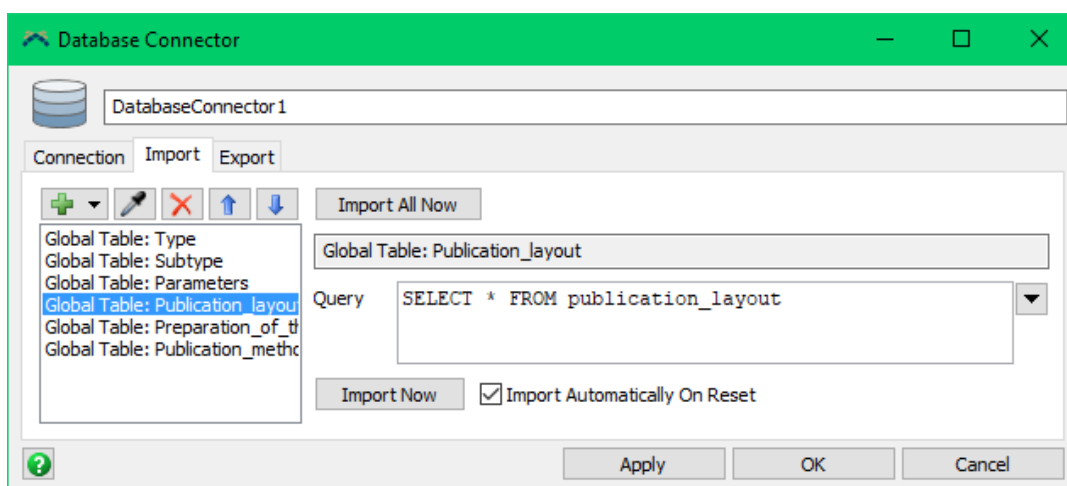


Рисунок 4.12 – Імпорт даних у таблицю Publication\_layout

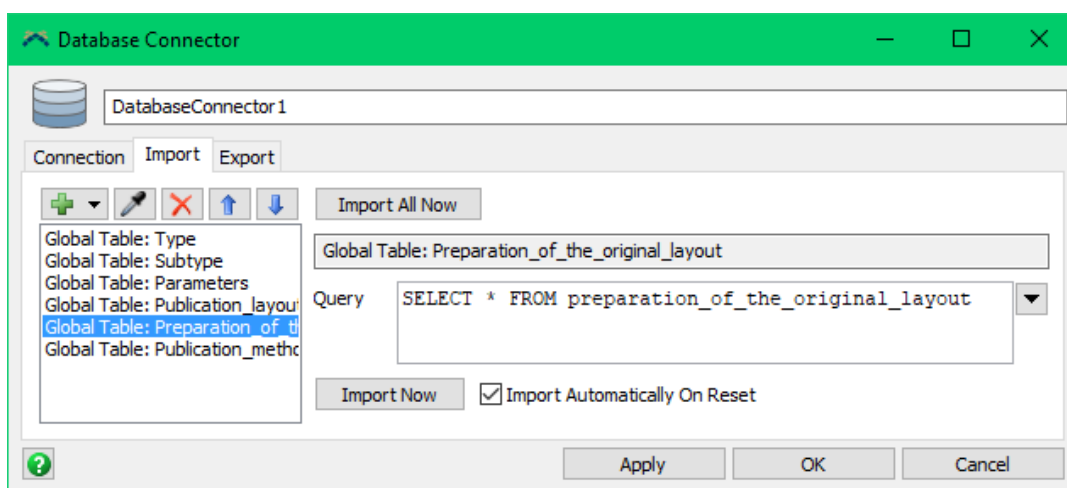


Рисунок 4.13 – Імпорт даних у таблицю Preparation\_of\_the\_original\_layout

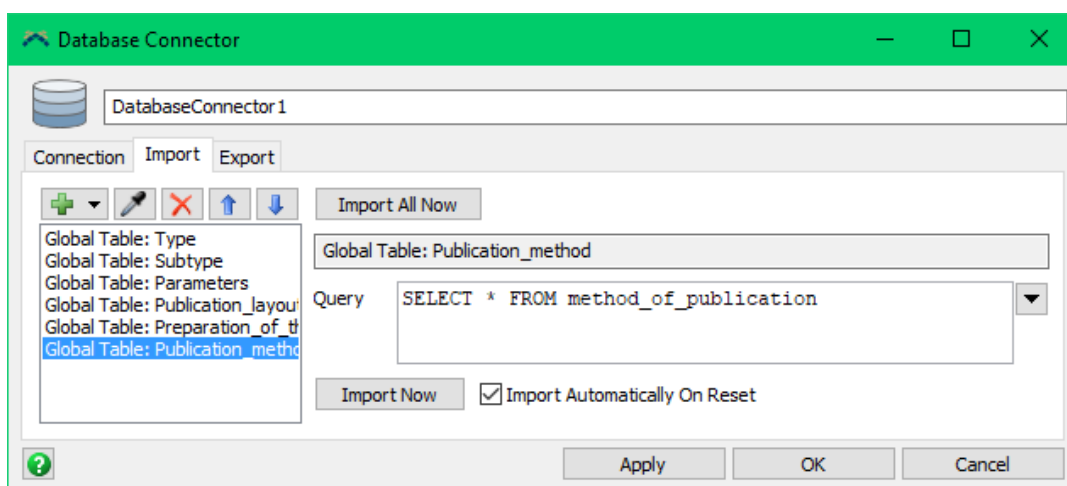


Рисунок 4.14 – Імпорт даних у таблицю Method\_of\_publication



## 4.2 Розробка тривимірних моделей

Створюємо 3D моделі для робочого місця, видання та поштової скриньку куди направляються видання на доробку авторам у програмному продукті Blender (рис. 4.15-4.17).

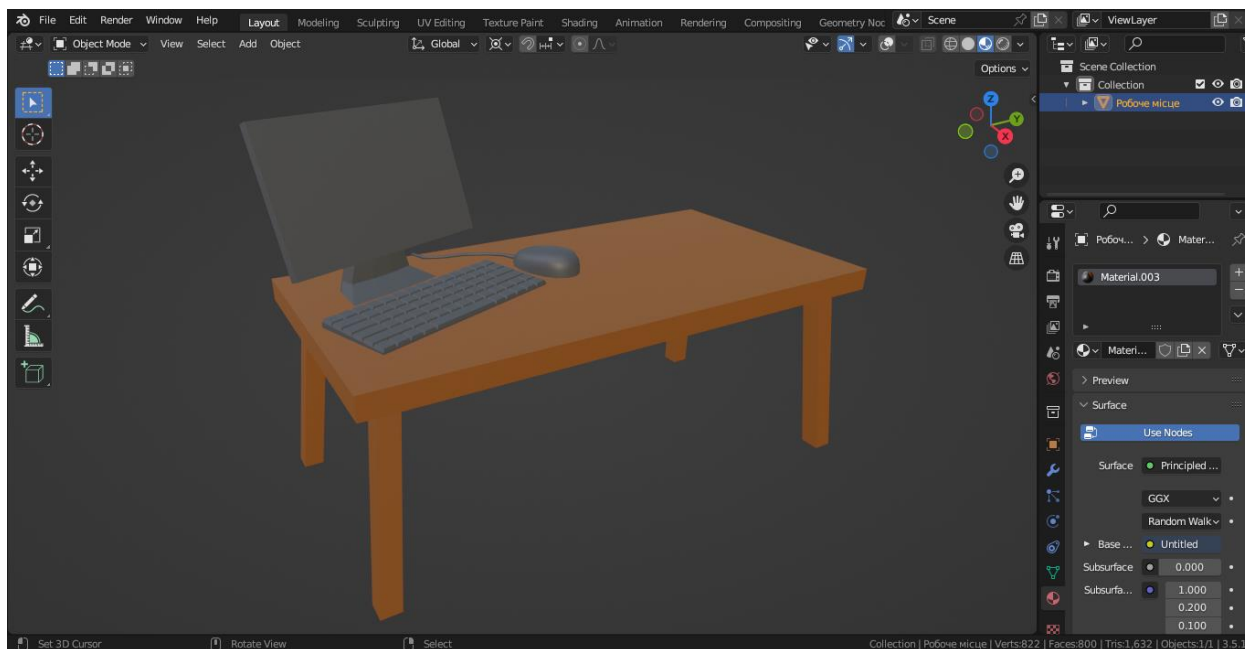


Рисунок 4.15 – 3D модель робочого місця

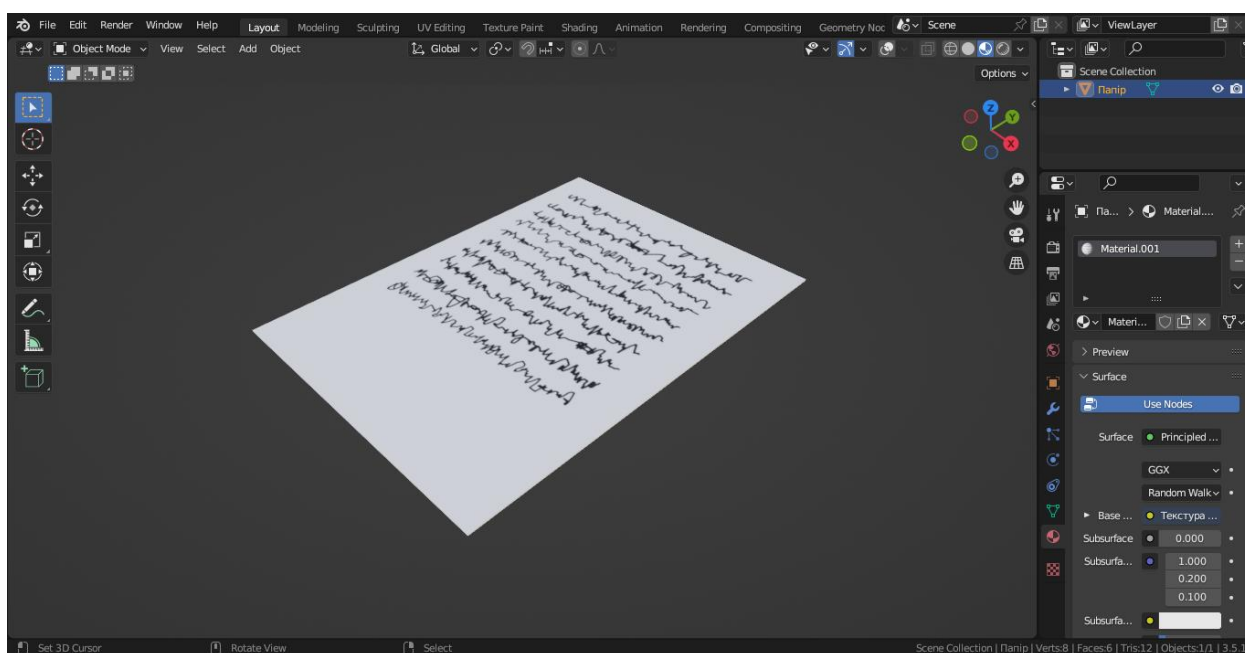


Рисунок 4.16 – 3D модель видання

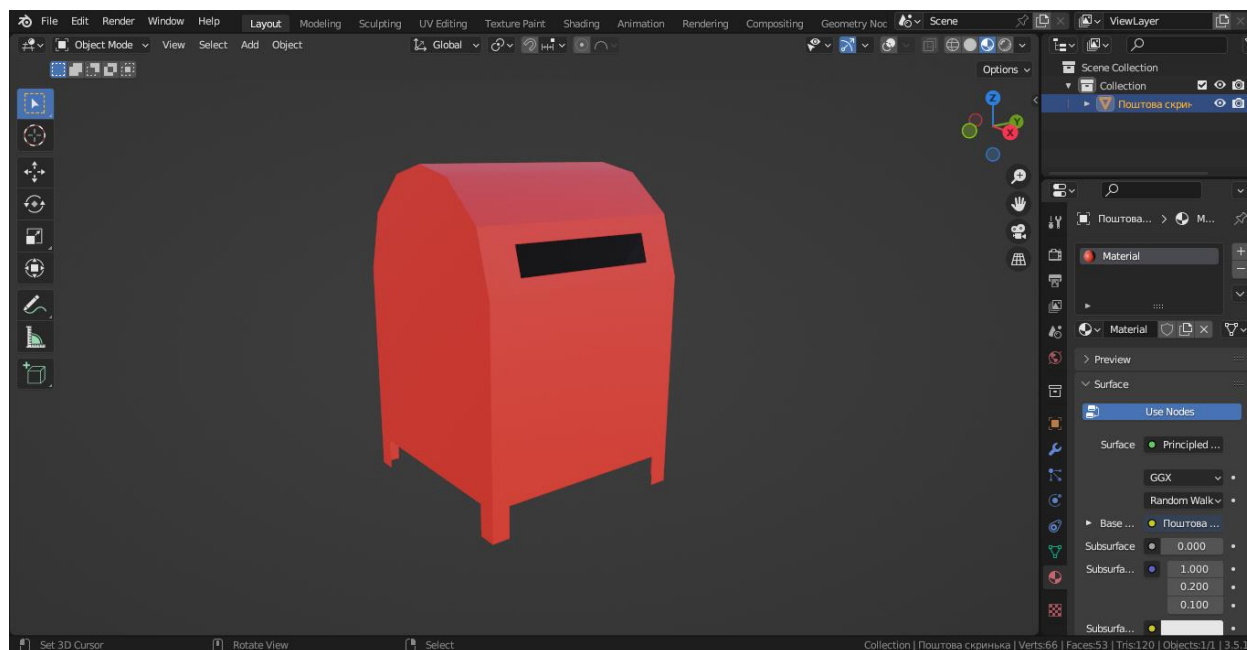


Рисунок 4.17 – 3D модель поштової скриньки

### 4.3 Програмна реалізація імітаційної моделі

Для створення імітаційної моделі, додаємо елементи системи масового обслуговування та встановлюємо між ними зв'язки (рис. 4.18). Після цього налаштовуємо інтервал надходження заявок у систему, час обробки у головного редактора та присвоюємо тип, підтип та параметри видання у головного редактора (рис. 4.19).

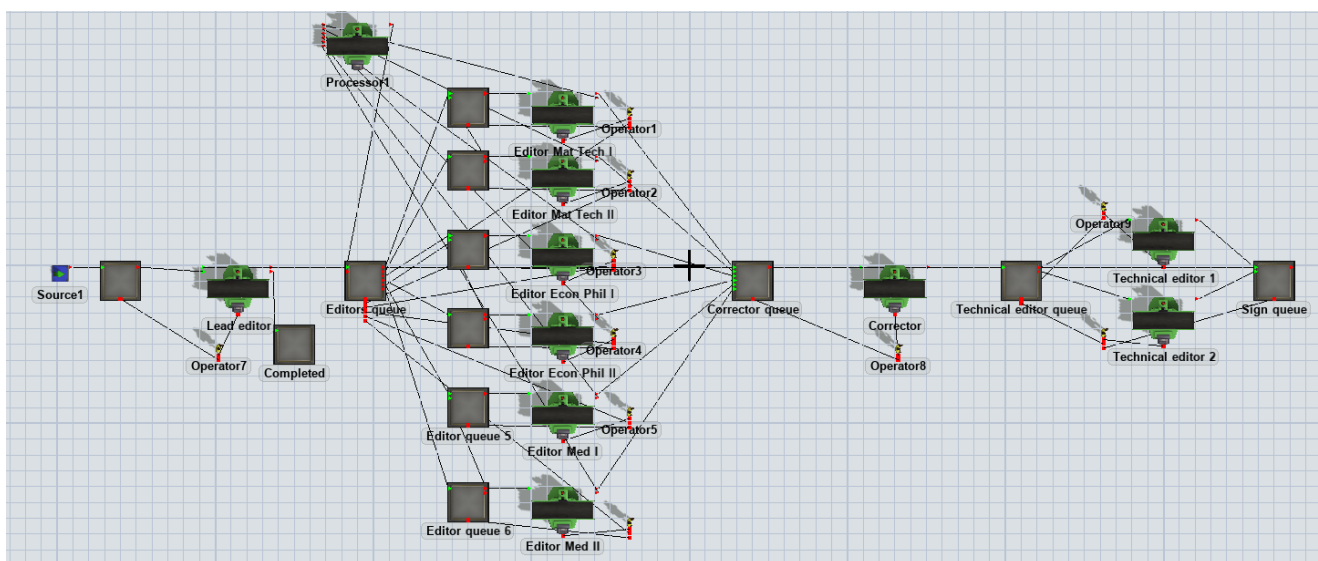


Рисунок 4.18 – Вигляд зв'язків між елементами СМО

```

1 Object current = ownerobject(c);
2 Object item = param(1);
3 if (item.Editor_ID==0){
4   Object current = ownerobject(c);
5   Object item = param(1);
6   int rownumber = param(2);
7   int rownumber1 = param(3);
8   int port = param(4);
9   int stream = getstream(ownerobject(c));
10  double randomnum = uniform(0.0, 100.0, stream);
11  double total = 0.0;
12  int foundmatch = 0;
13  int i=1;
14  for(i=1; i<Table.query("SELECT COUNT(Type_ID) FROM Type")[1][1]+1;i++){
15    total += Table("Type")[i][3];
16    if (!foundmatch && randomnum <= total) {
17      item.labels.assert("Type_ID", Table("Type")[i][1]);
18      item.labels.assert("Type", Table("Type")[i][2]);
19      foundmatch = 1;
20    }
21  }
22
23  randomnum = uniform(0.0, 100.0, stream);
24  total = 0.0;
25  foundmatch = 0;
26  i=1;
27  if(Table.query("SELECT * FROM Subtype WHERE Type_ID={f1}", item.Type_ID).numRows>0){
28    for(i=1; i<Table.query("SELECT COUNT(Type_ID) FROM Subtype WHERE Type_ID={f1}", item.Type_ID)[1][1]+1; i++){
29      total += Table("Subtype")[Table.query("SELECT * FROM Subtype WHERE Type_ID={f1}", item.Type_ID)[i][2]][4];
30      if (!foundmatch && randomnum <= total) {
31        item.labels.assert("Subtype_ID", Table("Subtype")[Table.query("SELECT * FROM Subtype WHERE Type_ID={f1}", item.Type_ID)[i][2]][4]);
32        item.labels.assert("Subtype", Table("Subtype")[Table.query("SELECT * FROM Subtype WHERE Type_ID={f1}", item.Type_ID)[i][2]][4]);
33        foundmatch = 1;
34      }
35    }
36  }else{
37    item.labels.assert("Subtype_ID", 0);
38    item.labels.assert("Subtype", "none");
39  }
40
41  if(Table.query("SELECT * FROM Parameters WHERE Type_ID={f1} AND Subtype_ID={f2}", item.Type_ID, item.Subtype_ID).numRows>0){
42    Array Direction_range = [];
43    Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE Type_ID={f1} AND Subtype_ID={f2}", item.Type_ID, item.Subtype_ID)[1][1]]);
44    if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE Type_ID={f1} AND Subtype_ID={f2}", item.Type_ID, item.Subtype_ID)[1][1]]){
45      Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE Type_ID={f1} AND Subtype_ID={f2}", item.Type_ID, item.Subtype_ID)[1][1]]);
46    }
47    if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE Type_ID={f1} AND Subtype_ID={f2}", item.Type_ID, item.Subtype_ID)[1][1]]){
48      Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE Type_ID={f1} AND Subtype_ID={f2}", item.Type_ID, item.Subtype_ID)[1][1]]);
49    }
50    if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE Type_ID={f1} AND Subtype_ID={f2}", item.Type_ID, item.Subtype_ID)[1][1]]){
51      Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE Type_ID={f1} AND Subtype_ID={f2}", item.Type_ID, item.Subtype_ID)[1][1]]);
52    }
53    item.labels.assert("Direction",Direction_range[duniform(1,Direction_range.length,getstream(current))]);
54
55    item.labels.assert("Amount_of_pages",duniform(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE Type_ID={f1} AND Subtype_ID={f2}", item.Type_ID, item.Subtype_ID)[1][1]]));
56  }
57
58  }
59
60  }
61
62  }
63
64  }
65
66  }
67
68  }
69
70  }
71
72  }
73
74  }
75
76  }
77
78  }
79
80  }
81
82  }
83
84  }
85
86  }
87
88  }
89
90  }
91
92  }
93
94  }
95
96  }
97
98  }
99
100 }

```

Рисунок 4.19 – Призначення типу, підтипу та параметрів видання

Після цього розподіляємо видання по редакторам таким чином щоб спеціальність та категорія редактора відповідала спрямуванню видання (рис. 4.20)

```

1 Object item = param(1);
2 Object current = ownerobject(c);
3 /**Custom Code*/
4 if(item.Direction.match(/scientific/).length>0){
5   if(item.Direction.match(/mathematical scientific/).length>0||item.Direction.match(/technical scientific/).length>0){return 1;}
6   if(item.Direction.match(/economical scientific/).length>0||item.Direction.match(/philological scientific/).length>0){return 3;}
7   if(item.Direction.match(/medical scientific/).length>0){return 5;}
8 }else{
9   if(item.Direction.match(/mathematical/).length>0||item.Direction.match(/technical/).length>0){if(opavailable(current,1) && current.outObjects[3].subnodes.length > 20){return 1;}else{return 3;}}
10  if(item.Direction.match(/economical/).length>0||item.Direction.match(/philological/).length>0){if(opavailable(current,3) && current.outObjects[4].subnodes.length > 20){return 3;}else{return 4;}}
11  if(item.Direction.match(/medical/).length>0){if(opavailable(current,5) && current.outObjects[6].subnodes.length > 20){return 5;}else{return 6;}}
12 }

```

Рисунок 4.20 – Розподіл видань по редакторам

Після обробки у редактора заявка повертається на доробку авторам. Щоб видання які пройшли перевірку редактора вперше повернулися авторам на доробку використовується мітка `Return_from_author` (рис. 2.21). Після доробки заявка повертається до редактора у якого вчитувалася в перший раз. Для того щоб заявка яка вже пройшла першу вчитку потрапляла до редактора поза чергою використовується стратегія pull (рис. 4.22)

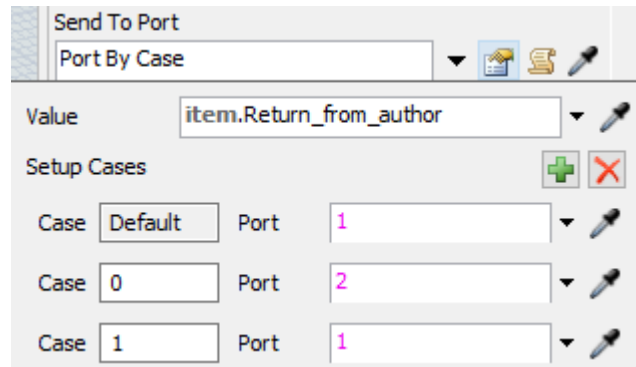


Рисунок 4.21 – Використання мітки `Return_from_author`

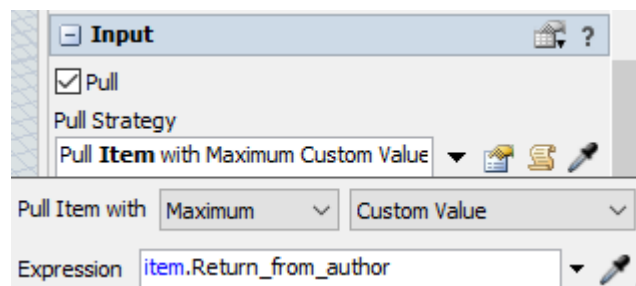


Рисунок 4.22 – Використання стратегії pull

Далі заявка направляється до коректора, якому налаштовуємо час обробки (рис. 4.22). Для відображення часу обробки видання на кожному етапі використовується мітка `time_of_process` (рис. 4.23). Опісля цього видання направляється до технічного редактора, який визначає спосіб друку видання, виконує верстання видання та готує оригінал-макет видання до друку. Налаштовуємо час обробки у технічного редактора (рис. 4.24). Остаточним етапом є підпис у головного редактора, до якого видання повертається після обробки у технічного редактора.

```

1 /**Custom Code*/
2 Object current = ownerobject(c);
3 Object item = param(1);
4 double time_of_process;
5 time_of_process = triangular(
6     171.29*pow(0.5,item.Publication_format.charAt(3).toNum())*(2.3-0.1*item.Font_size)*(1.32-0.13*pow(item.Line_spacing,2)),
7     214.11*pow(0.5,item.Publication_format.charAt(3).toNum())*(2.3-0.1*item.Font_size)*(1.32-0.13*pow(item.Line_spacing,2)),
8     265.93*pow(0.5,item.Publication_format.charAt(3).toNum())*(2.3-0.1*item.Font_size)*(1.32-0.13*pow(item.Line_spacing,2)),
9     getstream(current))*item.Amount_of_pages/2;
10
11 item.labels.assert("time_of_process").value=time_of_process;
12 return time_of_process;

```

Рисунок 4.22– Налаштування часу обробки коректором

Labels	
Editor_ID	5
Return_from_auth	0
time_of_process	1606.03

Рисунок 4.23– Мітка time\_of\_process

```

1 /**Custom Code*/
2 Object current = ownerobject(c);
3 Object item = param(1);
4
5 int time_of_printing_method_determination=duniform(5,7,getstream(current));
6 double time_of_process;
7
8 int minimum=0; int ager=0; int maximum=0;
9 if(Table.query("SELECT * FROM Publication_layout WHERE Type_ID={s1}", item.Type_ID).numRows>0){
10     minimum=Table("Publication_layout")[Table.query("SELECT * FROM Publication_layout WHERE Type_ID={s1}", item.Type_ID)[1][1]][2];
11     ager=Table("Publication_layout")[Table.query("SELECT * FROM Publication_layout WHERE Type_ID={s1}", item.Type_ID)[1][1]][3];
12     maximum=Table("Publication_layout")[Table.query("SELECT * FROM Publication_layout WHERE Type_ID={s1}", item.Type_ID)[1][1]][4];
13 }
14 int time_of_arrangement=16*0.5*item.Publication_format.charAt(3).toNum()*triangular(minimum,ager,maximum)*item.Amount_of_pages;
15
16 if(Table.query("SELECT * FROM Preparation_of_the_original_layout WHERE Type_ID={s1}", item.Type_ID).numRows>0){
17     minimum=Table("Preparation_of_the_original_layout")[Table.query("SELECT * FROM Preparation_of_the_original_layout WHERE Type_ID={s1}", item.Type_ID)[1][1]][2];
18     ager=Table("Preparation_of_the_original_layout")[Table.query("SELECT * FROM Preparation_of_the_original_layout WHERE Type_ID={s1}", item.Type_ID)[1][1]][3];
19     maximum=Table("Preparation_of_the_original_layout")[Table.query("SELECT * FROM Preparation_of_the_original_layout WHERE Type_ID={s1}", item.Type_ID)[1][1]][4];
20 }
21 int time_of_preparing_for_printing=16*0.5*item.Publication_format.charAt(3).toNum()*triangular(minimum,ager,maximum)*item.Amount_of_pages;
22 time_of_process = (time_of_printing_method_determination+time_of_arrangement+time_of_preparing_for_printing)/2;
23 item.labels.assert("time_of_process").value=time_of_process;
24 return time_of_process;

```

Рисунок 4.24 – Визначення часу обробки у технічного редактора

Для навігації операторів по моделі використовується алгоритм А\* до учасників якого додано усі об'єкти системи та операторів (рис. 4.24). Також для відображення приміщення було додано стіни(рис. 4.25). Розміщення усіх елементів моделі зі зв'язками показано на рисунку 4.26. Для більш зручного відображення

використовується режим з перспективою (рис. 4.27). Під час роботи моделі усі черги видань та видання в обробці відображаються на столі редактора (рис. 4.28).

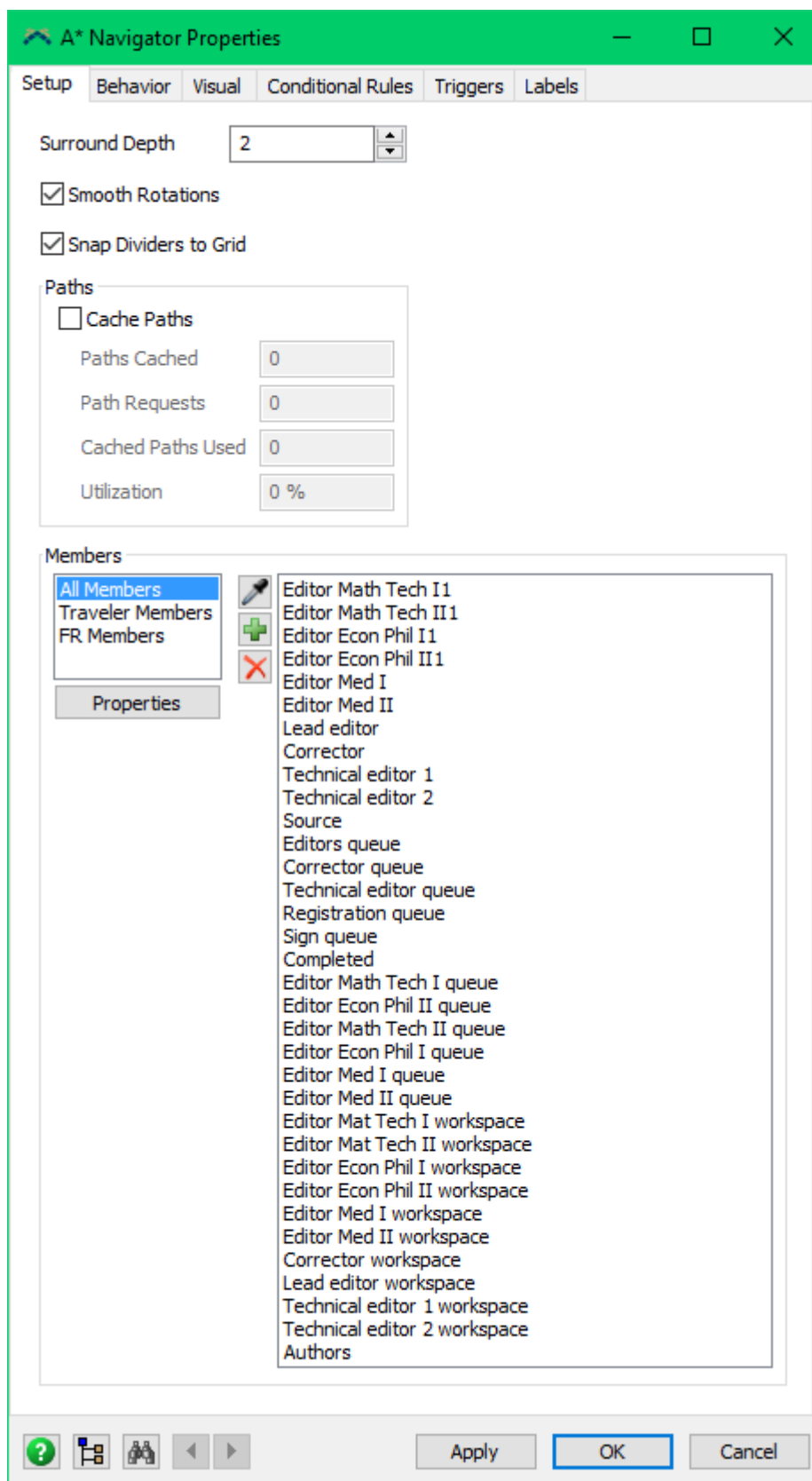


Рисунок 4.24 – 3D модель видання



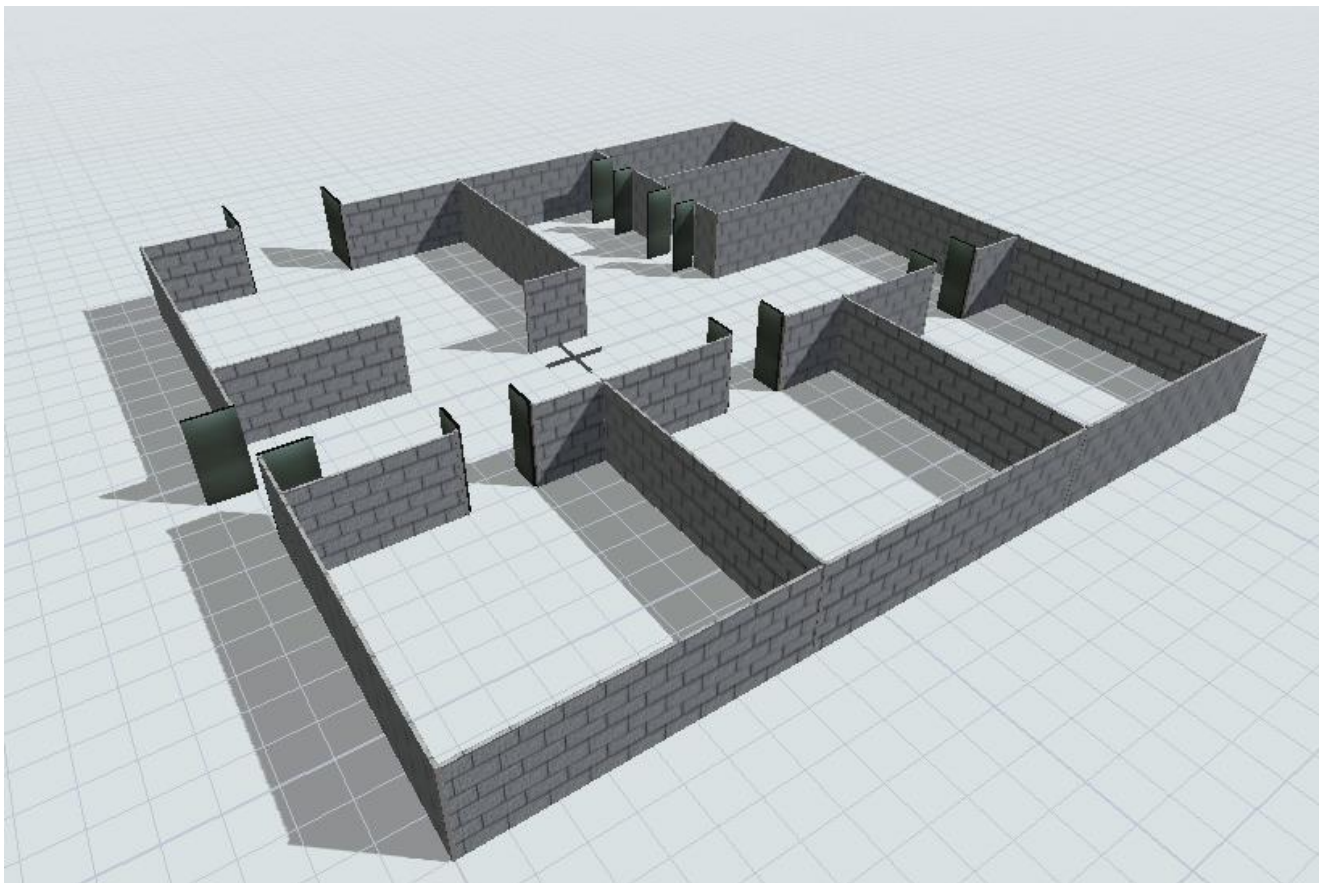


Рисунок 4.25 – Вигляд приміщення

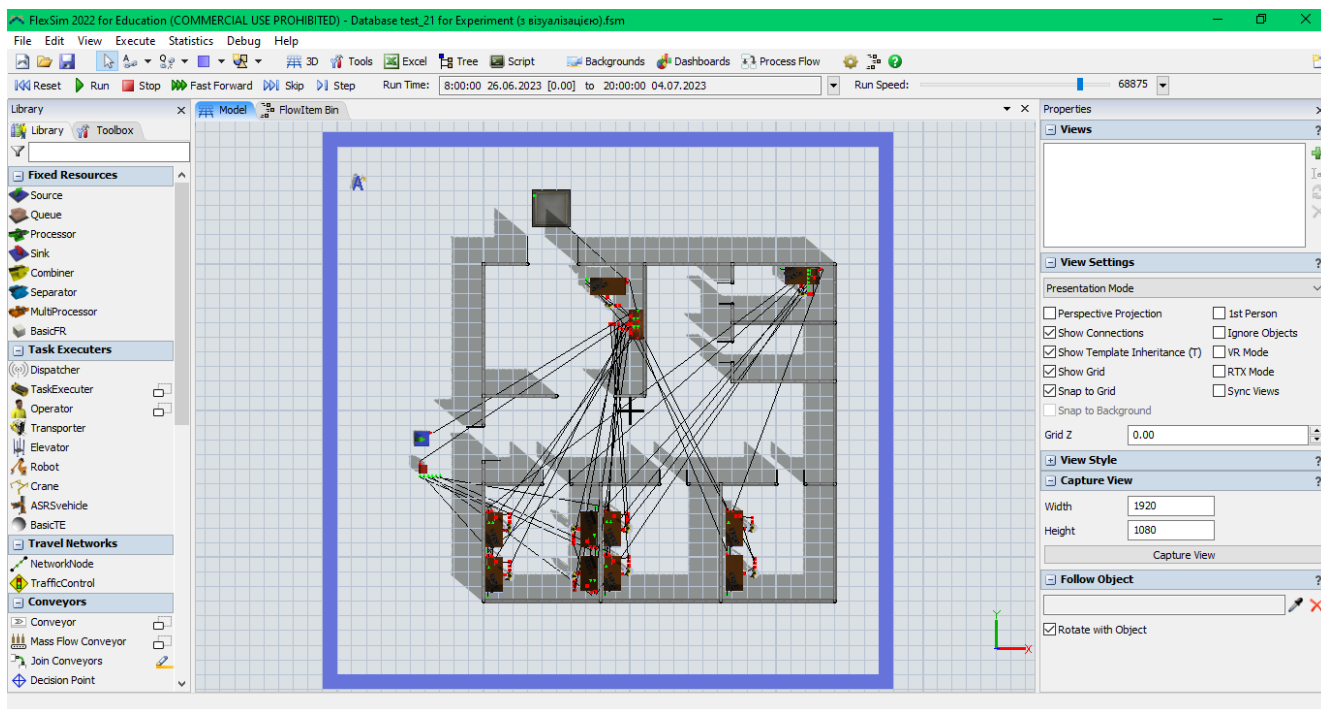


Рисунок 4.26 – Вигляд моделі з відображенням зв'язків

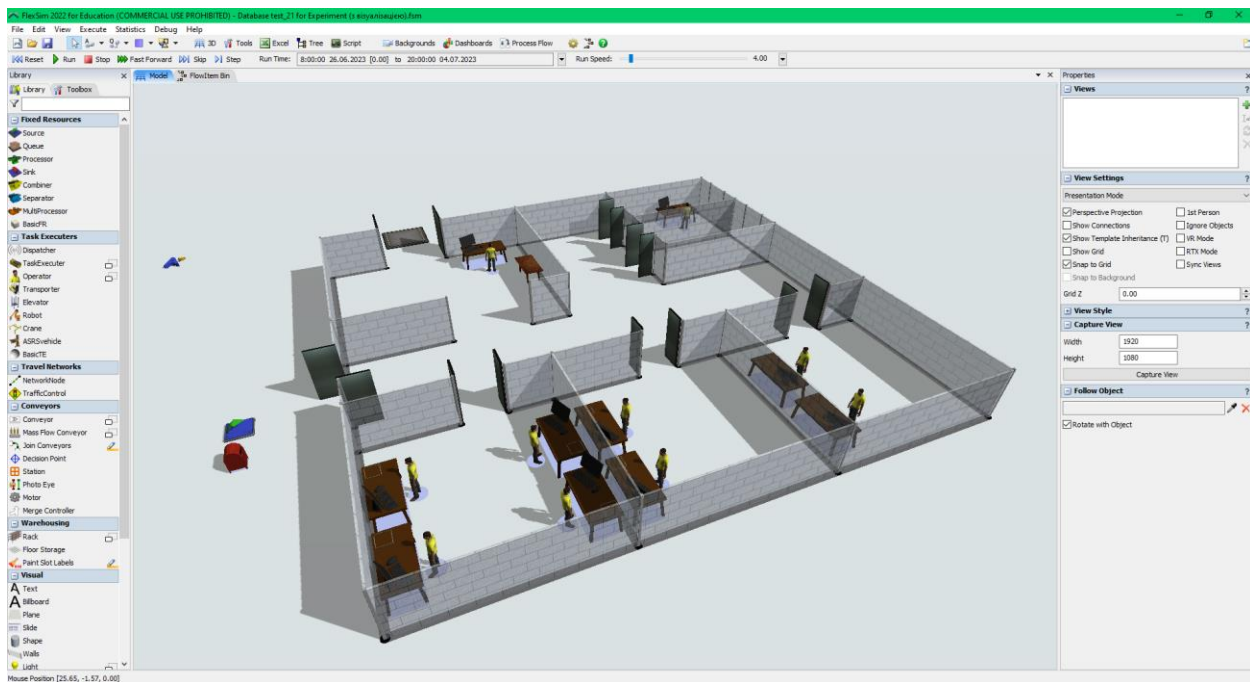


Рисунок 4.26 – Вигляд моделі у перспективі

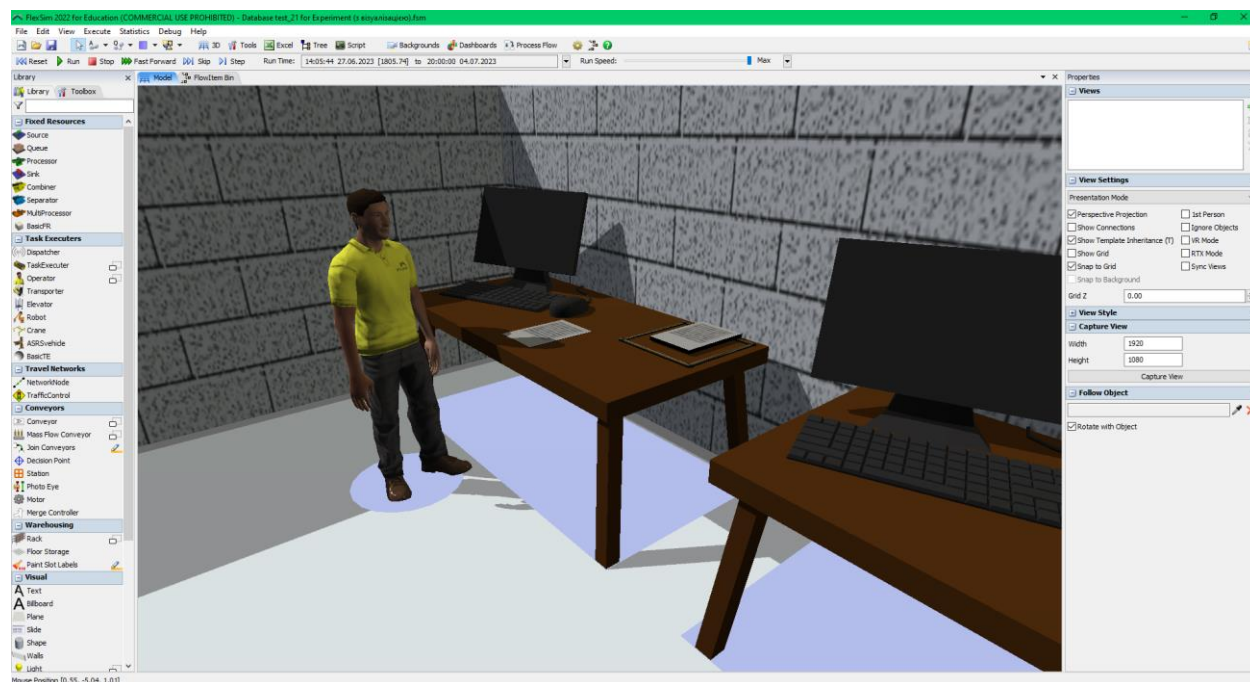


Рисунок 4.28 – Вигляд моделі під час роботи



## 5 ДОСЛІДЖЕННЯ РОБОТИ СИСТЕМИ

У моделі досліджуються такі показники ефективності:

- продуктивність системи;
- завантаженість робочих місць і працівників відділу;
- середні довжини черг;
- час перебування заявок у системі.

Пробний прогін моделі, що здійснювався протягом 1 місяця роботи показав наступні значення показників ефективності (рис. 5.1).

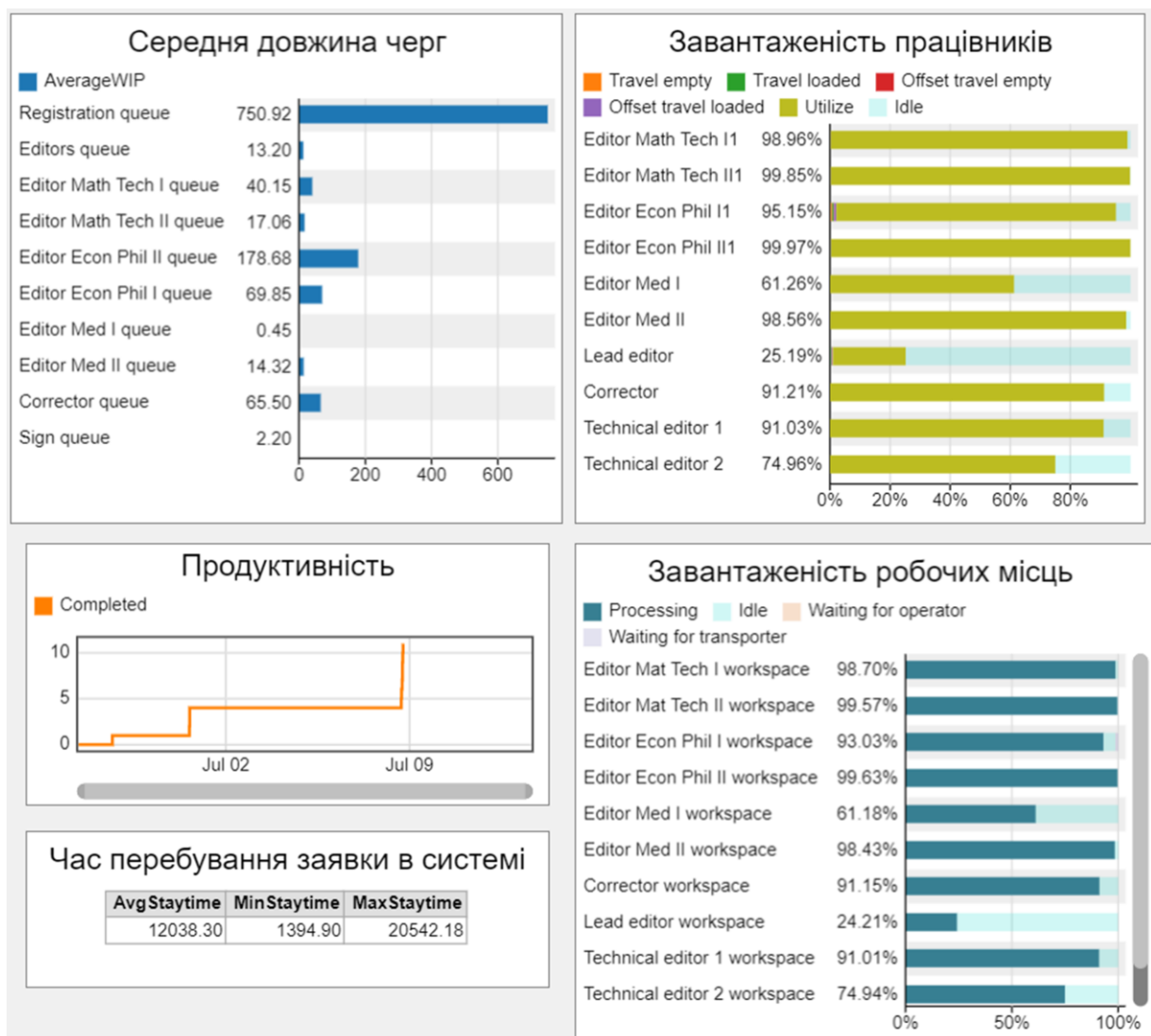


Рисунок 5.1 – Значення показників ефективності системи (пробний прогін протягом 1 місяця)

Аналіз даних показників показав:

- середній час перебування заявки в системі триває 12038.30 хв.  $\approx$  8.5 днів;
- кількість підготовлених видань рівна 10, серед них посібники, підручники та методичні вказівки об'ємом: 100, 21, 17, 139, 101, 37, 247, 60, 25, 10, 54 сторінок;
- зростання середньої довжини черг на таких робочих місцях:
  - головного редактора під час реєстрації видань;
  - редакторів економічно-філологічного спрямування 1-ї та 2-ї категорій;
  - коректора;
- високу завантаженість працівників та робочих місць до 90% і вище для:
  - редакторів математично-технічного спрямування 1-ї та 2-ї категорій;
  - редакторів економічно-філологічного спрямування 1-ї та 2-ї категорій;
  - редактора медичного спрямування 2-ї категорії;
  - коректора;
  - технічного редактора 2-ї категорії.

В імітаційній моделі є можливість варіювати кількістю робочих місць та працівників відділу, і досліджуючи роботу системи, обрати кращий варіант. Але не можна виконувати зміну параметрів моделі і запускати по одному прогону для кожного варіанту. Відгук моделі при такому дослідженні буде недостовірним. Для підвищення достовірності результатів моделювання з моделлю проводять експерименти. FlexSim має вбудований програмний модуль – Experimenter.

Experimenter – це інструмент, який дозволяє запуснути модель з різними вхідними даними і оцінити результат. Щоб використати Експериментатор, потрібно налаштувати множину сценаріїв, кількість реплікацій для кожного сценарію та час прогріву моделі.

Кожен сценарій містить свій набір вхідних даних моделювання.

Кількість реплікацій впливає на точність результатів моделювання і обчислюється за формулою Чебишева або за формулою центральної граничної теореми.

Для забезпечення високої точності ( $\leq 95\%$ ) і довірчої імовірності ( $\leq 95\%$ ) потрібно здійснювати не менше 25 прогонів моделі в межах кожної реплікації.

Пробний прогін дозволив визначити час прогріву моделі, який склав 9 000 хв.

У якості параметрів моделювання обрано кількість робочих місць. У табл. 5.1 приведено перелік досліджуваних факторів та їх діапазони варіювання.

Таблиця 5.1 – Діапазони досліджуваних факторів

Робочі місця спеціалістів	Кількість
Lead editor workspace	[1, 5]
Editor Mat Tech I workspace	[1, 5]
Editor Mat Tech II workspace	[1, 5]
Editor Econ Phil I workspace	[1, 5]
Editor Econ Phil II workspace	[1, 5]
Editor Med I workspace	[1, 5]
Editor Med II workspace	[1, 5]
Technical editor 1 workspace	[1, 5]
Technical editor 2 workspace	[1, 5]
Corrector workspace	[1, 5]

На рис. 5.2 показано налаштування плану проведення експериментів у програмі FlexSim.

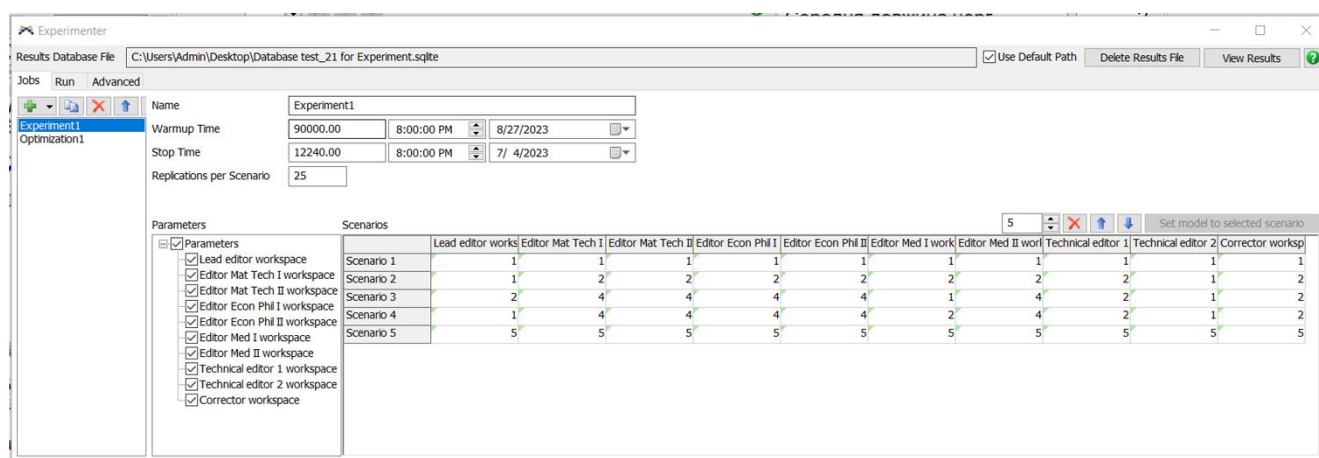


Рисунок 5.2 – Налаштування проведення експерименту

В межах експерименту налаштовано 5 сценаріїв, кожен з яких містить відповідні значення варійованих факторів (кількість робочих місць спеціалістів).

Далі запускаємо експеримент на виконання (рис. 5.3).

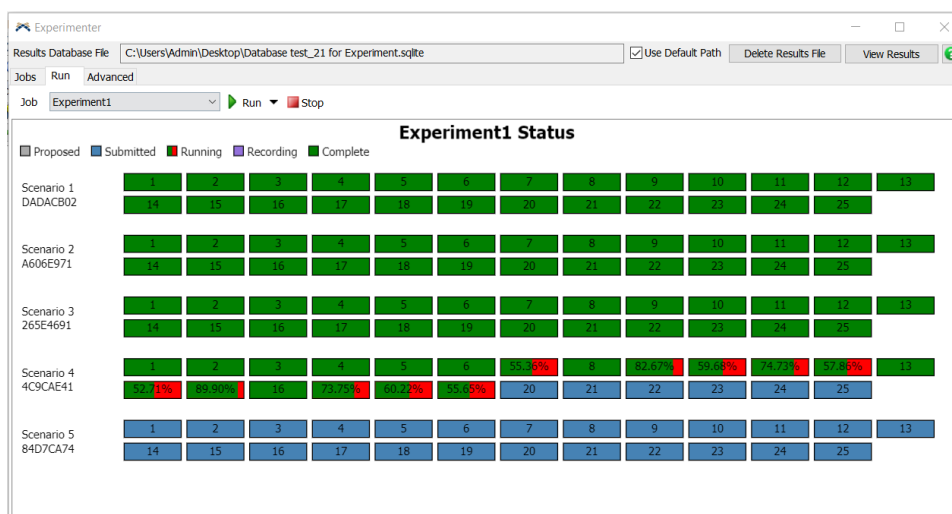


Рисунок 5.3 – Процес виконання експерименту

По закінченню переглядаємо та вивчаємо результати проведеного експерименту (рис. 5.4).

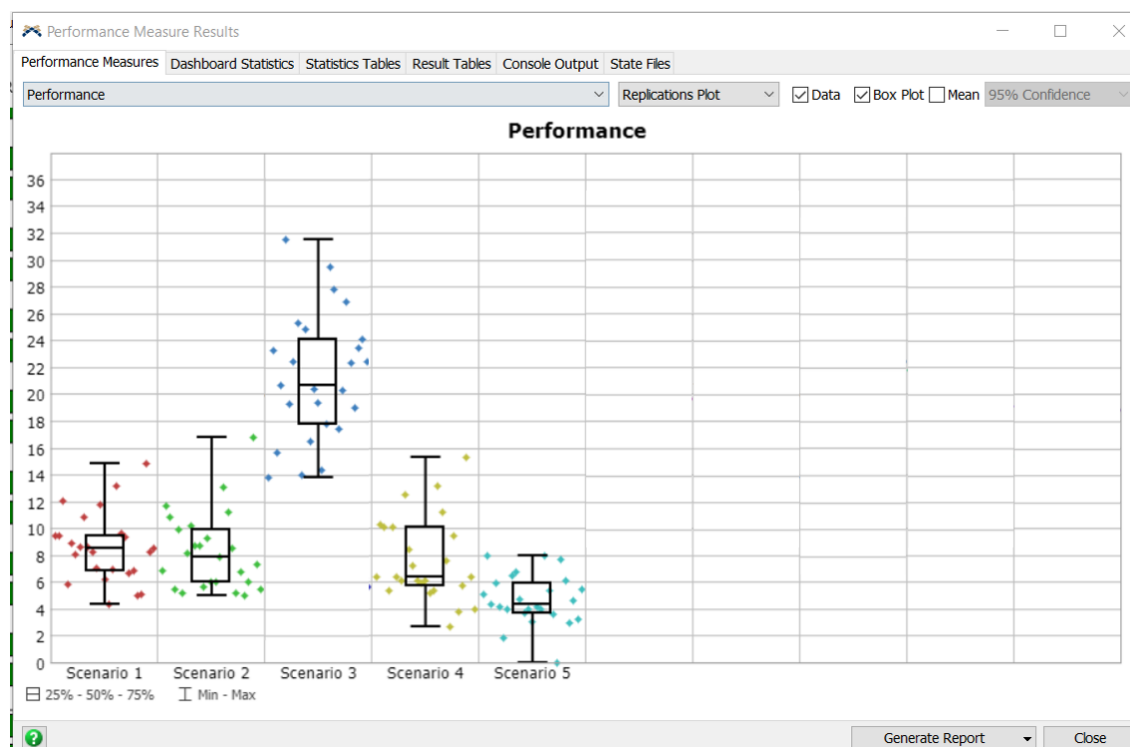


Рисунок 5.4 – Результати проведеного експерименту

За графіком на рис. 5.4 бачимо, що 3-й сценарій експерименту дав найвищі значення досліджуваного показника ефективності – продуктивності системи.

Базовий варіант системи показує сценарій 1. У базовому варіанті ми маємо по 1-му робочому місту на спеціаліста кожної категорії і кожного спрямування. За базовим варіантом ми матимемо не більше 14 видань на місяць, а в середньому близько 9-ти. Тоді як за 3-м сценарієм матимемо не більше 32 видань на місяць, а в середньому близько 21-го видання.

На основі проведеного експерименту можна надати практичні рекомендації по кількості необхідних робочих місць у системі редакційно-видавничого відділу (табл. 5.2) для забезпечення високої продуктивності, спираючись на дані за 3-м сценарієм проведеного експерименту.

Таблиця 5.2 – Рекомендована кількість робочих місць

Робочі місця спеціалістів	Рекомендована кількість спеціалістів
Lead editor workspace	2
Editor Mat Tech I workspace	4
Editor Mat Tech II workspace	4
Editor Econ Phil I workspace	4
Editor Econ Phil II workspace	4
Editor Med I workspace	1
Editor Med II workspace	4
Technical editor 1 workspace	2
Technical editor 2 workspace	1
Corrector workspace	2

Отримані дані по 3-му сценарію було введено до моделі і виконано пробний прогін протягом 1 місяця роботи. Результати показників ефективності відображені на рис. 5.5

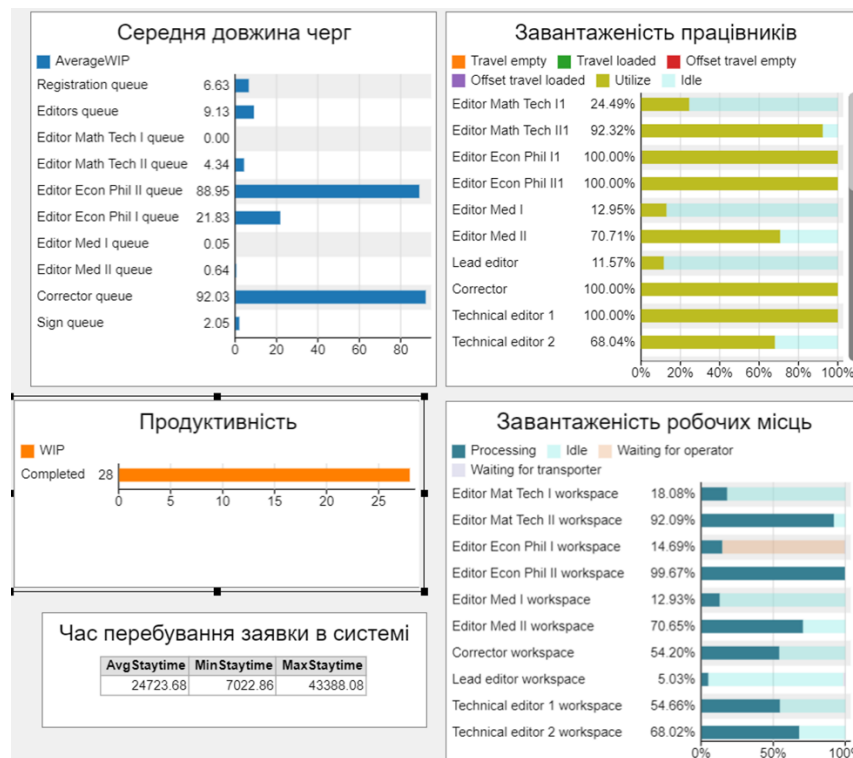


Рисунок 5.5 – Значення показників ефективності системи за вхідними даними 3-го сценарію (пробний прогін протягом 1 місяця)

Порівняння результатів пробних прогонів базового варіанту системи (рис. 5.1) і варіанту системи за 3-м сценарієм (рис. 5.5) показали такі покращення в роботі системи (табл. 5.3).

Таблиця 5.3 – Порівняння показників ефективності для пробних прогонів моделі (1 місяць)

Покращені показники ефективності	Базовий варіант моделі	Варіант моделі за 3-м сценарієм	Результат покращення
<b>Продуктивність (од./міс.)</b>	10	28	Збільшення в 2.8 рази
<b>Середні довжини черг (од.):</b>			Зменшення:
- редакторів	13.20	9.30	в 1.4 р.
- редактора мат.-тех. 1 кат.	40.15	0	на 100%
- редактора мат.-тех. 2 кат.	17.06	4.34	в 4 р.
- редактора екон.-філол. 1 кат.	168.78	88.95	в 2 р.
- редактора екон.-філол. 2 кат.	69.85	21.83	в 3 р.
<b>Завантаженість робочих місць (%):</b>			Зменшення:
- редактор мед. 2 кат.	98.43%	70.65%	в 1.4 р.
- коректор	91.21%	54.20%	в 1.7 р.
- техн. редактор 1 кат.	91.01%	54.66%	в 1.7 р.

Проведений аналіз та порівняння доводять, що результати експерименту за 3-м сценарієм є достовірними, оскільки у результаті пробних прогонів моделі отримано ряд покращень у порівнянні з базовим варіантом системи.

## ВИСНОВКИ

Під час виконання магістерської кваліфікаційної роботи був проведений аналіз предметної області та досліджень, щоб визначити актуальність роботи. Зазначено та деталізовано мету та завдання проекту. У процесі планування робіт було деталізовано мету за методом SMART, побудовано діаграми WBS для ієрархічної структури робіт та OBS для відображення організаційної структури проекту. Створено календарний план у вигляді діаграми Ганта та проведено аналіз ризиків проекту.

Завдяки аналізу методів дослідження було обрано метод імітаційного моделювання, оскільки він забезпечує достатній рівень точності відображення системи, дозволяє працювати з випадковими величинами, показує систему в динаміці та дозволяє швидко змінювати параметри системи. Проведений аналіз програмних засобів реалізації моделі включав порівняльну характеристику продуктів-аналогів. Завдяки цьому було обрано середовище моделювання FlexSim, оскільки дана програма підтримує роботу з базами даних за допомогою SQL та дозволяє розширити функціонал за допомогою використання мови програмування FlexScript.

Під час проектування було побудовано контекстні діаграми процесу розробки та процесу використання у нотації IDEF0 та їх декомпозицію першого рівня. Також розроблено діаграму варіантів використання та діаграму потоку об'єктів (OFD). Для збереження вхідних даних використовується база даних для створення якої було побудовано концептуальну, логічну та фізичну моделі бази даних.

Базу даних було реалізовано за допомогою PhpMyAdmin та СУБД MySQL, 3D моделі були створені у програмному забезпеченні Blender. Засобами ПЗ FlexSim розроблено імітаційну модель редакційно-видавничого відділу. Проведено дослідження показників ефективності системи, налаштування програмного модуля Experimenter, проаналізовано результати експерименту та надано практичні



рекомендації покількості необхідних робочих місць у системі редакційно-видавничого відділу.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Simulation, Production Planning and Scheduling Software | Simio: вебсайт, URL: <https://www.simio.com/> (дата звернення: 29.09.2023)
2. ProModel: Process Simulation Software - BigBear.ai: вебсайт, URL: <https://bigbear.ai/solutions/supply-chain-logistics/promodel-discrete-event-simulator/> (дата звернення: 30.09.2023)
3. (PDF) USER GUIDE PROMODEL: вебсайт, URL: [https://www.researchgate.net/publication/359021844\\_USER\\_GUIDE\\_PROMODEL](https://www.researchgate.net/publication/359021844_USER_GUIDE_PROMODEL) (дата звернення: 01.10.2023)
4. Predictive Digital Twin, Analytics & Simulation Software | Scenario Planning Solutions | Lanner: вебсайт, URL: <https://www.lanner.com/en-gb/> (дата звернення: 02.10.2023)
5. WITNESS SIMULATION SOFTWARE: вебсайт, URL: <https://tarjomefa.com/wp-content/uploads/2017/08/7597-English-TarjomeFa.pdf> (дата звернення: 30.09.2023)
6. Tecnomatix Plant Simulation Help: вебсайт, URL: [https://docs.plm.automation.siemens.com/content/plant\\_sim\\_help/15/plant\\_sim\\_all\\_in\\_one\\_html/en\\_US/tecnomatix\\_plant\\_simulation\\_help/tecnomatix\\_plant\\_simulation\\_help.html](https://docs.plm.automation.siemens.com/content/plant_sim_help/15/plant_sim_all_in_one_html/en_US/tecnomatix_plant_simulation_help/tecnomatix_plant_simulation_help.html) (дата звернення: 01.10.2023)
7. Sarkar, Advait. “Interactive analytical modelling.” (2018): вебсайт, URL: <https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-920.pdf> (дата звернення: 26.09.2023)
8. В. Б. Неруш, В. В. Курдеча.. ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ СИСТЕМ ТА ПРОЦЕСІВ. ): вебсайт, URL: [https://ela.kpi.ua/bitstream/123456789/15598/1/Konspect\\_lekciy\\_Imit\\_modelyr\\_syst\\_process%28CHANGED%29.pdf](https://ela.kpi.ua/bitstream/123456789/15598/1/Konspect_lekciy_Imit_modelyr_syst_process%28CHANGED%29.pdf) (дата звернення: 24.09.2023)
9. Бріцин А. О. ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ, ЯК ВАЖЛИВИЙ ІНСТРУМЕНТ МЕНЕДЖМЕНТУ Всеукраїнська науково-технічна конференція магістрантів і студентів ТДАТУ. Факультет економіки та бізнесу: Всеукраїнська науково-

технічна конференція, збірник тез доповідей (Мелітополь, 18-22 листопада 2019 року);(С. 153)

10. Власюк Ю.О. ОСОБЛИВОСТІ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ ЕКОНОМІЧНИХ СИСТЕМ Збірник наукових праць Таврійського державного агротехнологічного університету (економічні науки); № 4 (28) (С. 303-305)
11. Братушка, С.М. Імітаційне моделювання як інструмент дослідження складних економічних систем [Текст] / С.М. Братушка // Вісник Української академії банківської справи. – 2009. – № 2 (27). – С. 113–118.
12. Lavrov, E., Chybiriak, Y., Siryk, O., Logvinenko, V., Zakharova, A. Training of Specialists for Adaptive management. Techniques for Teaching Computer Analysis of Automated Production Systems in the FlexSim Environment (2022) CEUR Workshop Proceedings, 3104, pp. 106-118.
13. Lavrov, E., Siryk, O., Chybiriak, Y., Danilova, L., Nahornyi, V., Vakal, S. A Model for the Organization of Adaptive Dialogue Interaction 'Man-Computer' Taking into Account the Requirements of Reliability and Efficiency (2021) 2021 IEEE 4th International Conference on Advanced Information and Communication Technologies, AICT 2021 - Proceedings, pp. 31-35.
14. Lavrov, E., Siryk, O., Kirichenko, I., Barchenko, N., Chybiriak, Y. The Methodology of Managed Functional Networks for Organizing Effective and Adaptive Human-Machine Dialogue in Automated Systems (2021) CEUR Workshop Proceedings, 3013, pp. 428-437.
15. Marchenko, A., Antypenko, V., Vashchenko, S., Fedotova, N., Chybiriak, Y., Krasulia, A. A Complex Model of Blended Learning: Using a Project Approach to Organize the Educational Process (2021) Communications in Computer and Information Science, 1486 CCIS, pp. 265-278.
16. Lavrov, E., Chybiriak, Y., Siryk, O., Velykodnyi, D., & Borovyk, V. (2023). An approach to building automated expert assessment systems to support strategic and tactical decision-making. COMPUTER-INTEGRATED TECHNOLOGIES:

EDUCATION, SCIENCE, PRODUCTION, (52), 10-23.  
<https://doi.org/10.36910/6775-2524-0560-2023-52-02>

17. Chybiriak, Y., Zakharova, A., Lavrov, E., Velykodnyi, D., & Borovyk, V. (2023). Models for automated search for rational assembly options for mechanical engineering products. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (52), 32-42. <https://doi.org/10.36910/6775-2524-0560-2023-52-04>
18. Чибіряк Я. І. Метод наскрізного навчання студентів ІТ-спеціальностей імітаційному моделюванню у середовищі FlexSim для пошуку резервів підвищення ефективності автоматизованих систем / Чибіряк Я. І., Баранова І.В., Ніколаєнко К.О. // Комп'ютерно-інтегровані технології: освіта, наука, виробництво. – №42. – Луцьк, 2021. DOI: <https://doi.org/10.36910/6775-2524-0560-2021-42-17>
19. Чибіряк Я.І., Заболотний Д.В., Ніколаєнко К.О. Імітаційне моделювання та дослідження діяльності відділення банку. - Інформатика, математика, механіка : Збірник тез за матеріалами Міжвузівської науково-технічної конференції викладачів, співробітників, аспірантів і студентів. – Суми : СумДУ, 2021. – С. 152-153
20. Чибіряк Я.І., Нестеренко М.В. Моделювання параметрів системи передачі даних за наявності випадкових збоїв. - Інформатика, математика, механіка : Збірник тез за матеріалами Міжвузівської науково-технічної конференції викладачів, співробітників, аспірантів і студентів. – Суми : СумДУ, 2021. – С.147
21. Чибіряк Я.І., Ніколаєнко К.О., Овчаренко К.В. Дослідження впливу послідовності обробки завдань на завантаженість ресурсів інформаційної системи. - Інформатика, математика, механіка : Збірник тез за матеріалами Міжвузівської науково-технічної конференції викладачів, співробітників, аспірантів і студентів. – Суми : СумДУ, 2021. – С.148-149
22. Чибіряк Я. І., Нестеренко М.В. Імітаційна модель функціонування ремонтного підрозділу засобів зв'язку. - Інформатика, математика, автоматика. ІМА :: 2022.

- Міжнародна наукова конференція молодих учених (Суми – Нур-Султан, 18–22 квітня 2022 р.). – Суми : СумДУ, 2022. – С. 129-130
23. Захарова А.М., Чибіряк Я.І., Лавров Е.А. Імітаційне моделювання технологічного процесу складання клапана компресорного. - Інформатика, математика, автоматика. ІМА :: 2022. Міжнародна наукова конференція молодих учених (Суми – Нур-Султан, 18–22 квітня 2022 р.). – Суми : СумДУ, 2022. – С. 126-127
24. Чибіряк Я.І., Устименко К.О., Захарова А.М. Розробка імітаційної моделі локальної обчислювальної мережі. - Інформатика, математика, автоматика. ІМА :: 2023. Міжнародна наукова конференція молодих учених (Суми – Астана, 24–28 квітня 2023 р.). – Суми : СумДУ, 2023. – С. 228
25. Чибіряк Я.І., Захарова А.М., Устименко К.О. Моделювання локальної комп'ютерної мережі. - Інформатика, математика, автоматика. ІМА :: 2023. Міжнародна наукова конференція молодих учених (Суми – Астана, 24–28 квітня 2023 р.). – Суми : СумДУ, 2023. – С. 226-227
26. Simulation of the Production and Transport Problem in the FlexSim Environment вебсайт, URL: <https://sci.ldubgd.edu.ua/jspui/handle/123456789/11598> (дата звернення: 27.09.2023)
27. Modularization Approaches in the Context of Monolithic Simulations вебсайт, URL: <https://publikationen.bibliothek.kit.edu/1000131811/110806863> (дата звернення: 28.09.2023)
28. Role of Random Numbers in Simulations of Economic Processes вебсайт, URL: [https://www.researchgate.net/publication/23535848\\_Role\\_of\\_Random\\_Numbers\\_in\\_Simulations\\_of\\_Economic\\_Processes](https://www.researchgate.net/publication/23535848_Role_of_Random_Numbers_in_Simulations_of_Economic_Processes) (дата звернення: 26.09.2023)
29. ANALYTICAL MODELING, TESTING, AND COMPARISON OF 1-D, 2-D, AND 3-D DEWATERING PROCESS вебсайт, URL: <https://surface.syr.edu/thesis/175/> (дата звернення: 28.09.2023)

30. Comprehensive Analytical Modeling of Laser Powder-Bed/Fed Additive Manufacturing Processes and an Associated Magnetic Focusing Module вебсайт, URL: <https://core.ac.uk/download/pdf/200282759.pdf> (дата звернення: 26.09.2023)
31. Current-Driven Magnetization Dynamics : Analytical Modeling and Numerical Simulation вебсайт, URL: <https://ediss.sub.uni-hamburg.de/handle/ediss/4662> (дата звернення: 226.09.2023)
32. Analytical modeling of material constitutive behaviors and process mechanics in precision machining and additive manufacturing вебсайт, URL: <https://repository.gatech.edu/handle/1853/62307> (дата звернення: 27.09.2023)
33. Analytical modeling of part distortion in additive manufacturing вебсайт, URL: <https://repository.gatech.edu/entities/publication/8158343e-0f05-467b-9e3c-a7b346e51d98> (дата звернення: 25.09.2023)
34. A novel analysis model for optimization performance of Bernoulli serial production system considering rework processes вебсайт, URL: <https://journals.sagepub.com/doi/full/10.1177/0954405419852502> (дата звернення: 27.09.2023)
35. COGNITIVE EFFECTS OF PHYSICAL MODELS IN ENGINEERING IDEA GENERATION вебсайт, URL: <https://core.ac.uk/download/pdf/30268289.pdf> (дата звернення: 25.09.2023)
36. Physical modeling of flow and sediment transport using distorted scale modeling [https://repository.lsu.edu/cgi/viewcontent.cgi?article=4503&context=gradschool\\_theses](https://repository.lsu.edu/cgi/viewcontent.cgi?article=4503&context=gradschool_theses) (дата звернення: 28.09.2023)

## ДОДАТОК А. Планування робіт

### А.1 Ідентифікація мети ІТ-проекту

Деталізація мети проекту методом SMART. Продуктом дипломного проекту є імітаційна модель функціонування редакційно-видавничого відділу.

Результати деталізації методом SMART розміщені у таблиці А.1.

Таблиця А.1 – Деталізація мети проекту методом SMART

Specific (конкретна)	Створити імітаційну модель редакційно-видавничого відділу
Measurable (вимірювана)	Імітаційна модель редакційно-видавничого відділу
Achievable (досяжна)	Для реалізації проекту використовується середовище FlexSim, для роботами з базами даних – MySQL, для створення 3D моделей Blender
Relevant (реалістична)	Для підвищення ефективності роботи редакційно-видавничого відділу
Time-framed (обмежена у часі)	Проект має часові обмеження і повинен бути реалізований згідно з календарним планом

## А.2 Планування змісту структури робіт ІТ-проекту

Планування структури робіт у проекті здійснюється через ієрархічну структуру робіт (WBS), яка графічно відображає послідовність дій, необхідних для виконання завдань. Починаючи з основних робіт, ми розкладаємо їх на менші елементарні завдання та вказуємо терміни їх виконання. WBS діаграма проекту представлена на рисунку А.1. Далі, на основі WBS, створюємо організаційну структуру проекту (OBS), яка відображає розподіл робіт та взаємозв'язки між учасниками. OBS діаграма проекту наведена на рисунку А.2, а список виконавців подано в табличному вигляді у табл. А.2.

Таблиця А.2 – Виконавці проекту

Роль	Ім'я	Проектна роль
Розробник	Нестеренко М.В.	Розробка імітаційної моделі та її логіки, проведення досліджень
Проектувальник	Нестеренко М.В.	Проектування 3D моделей, створення таблиць у СУБД MySQL
Консультант проекту	Чибіряк Я.І.	Формує завдання на розробку проекту
Менеджер проекту	Нестеренко М.В.	Планування робіт, розподіл задач між учасниками, моніторинг виконання робіт



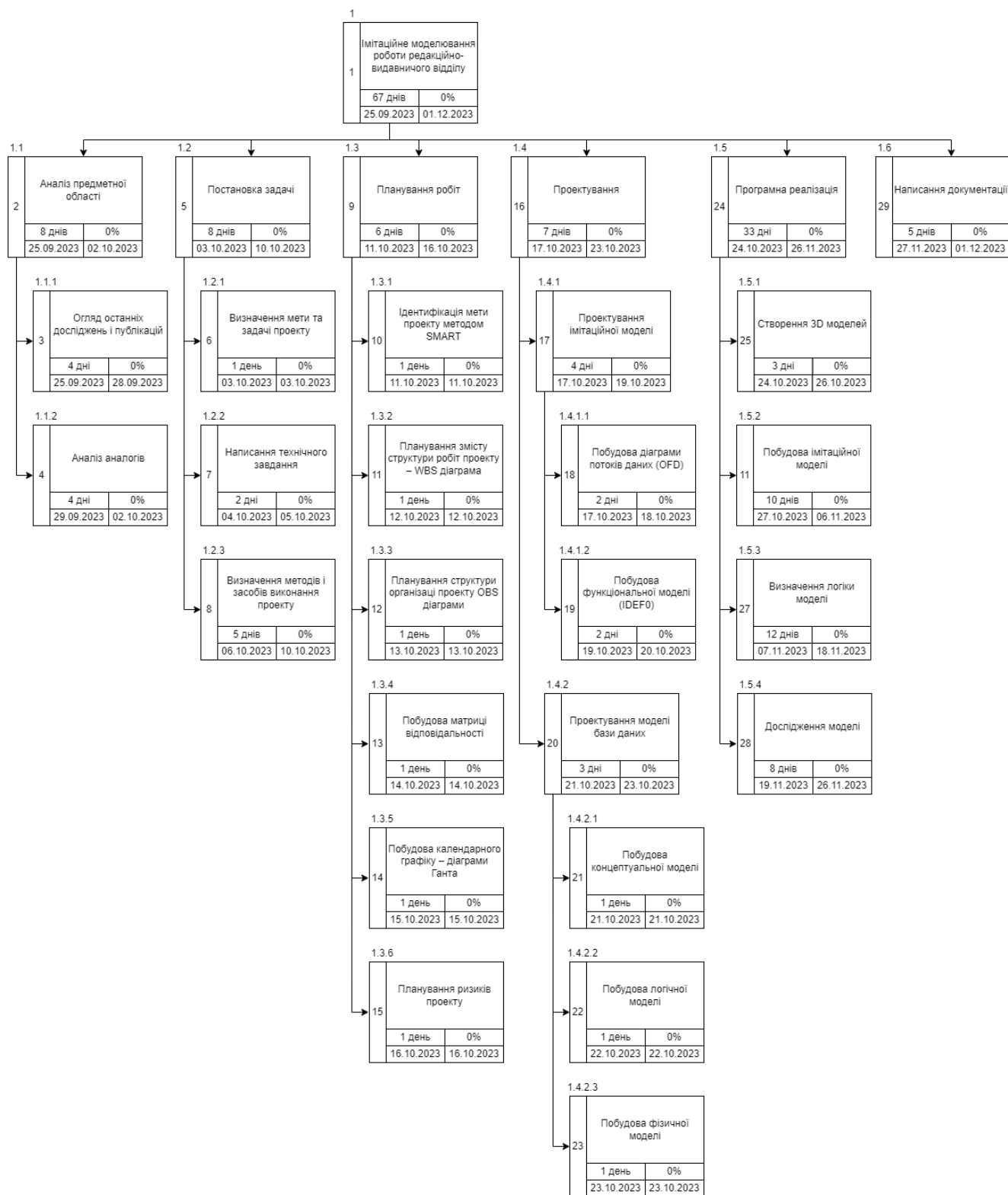


Рисунок А.1 –WBS структури робіт проекту

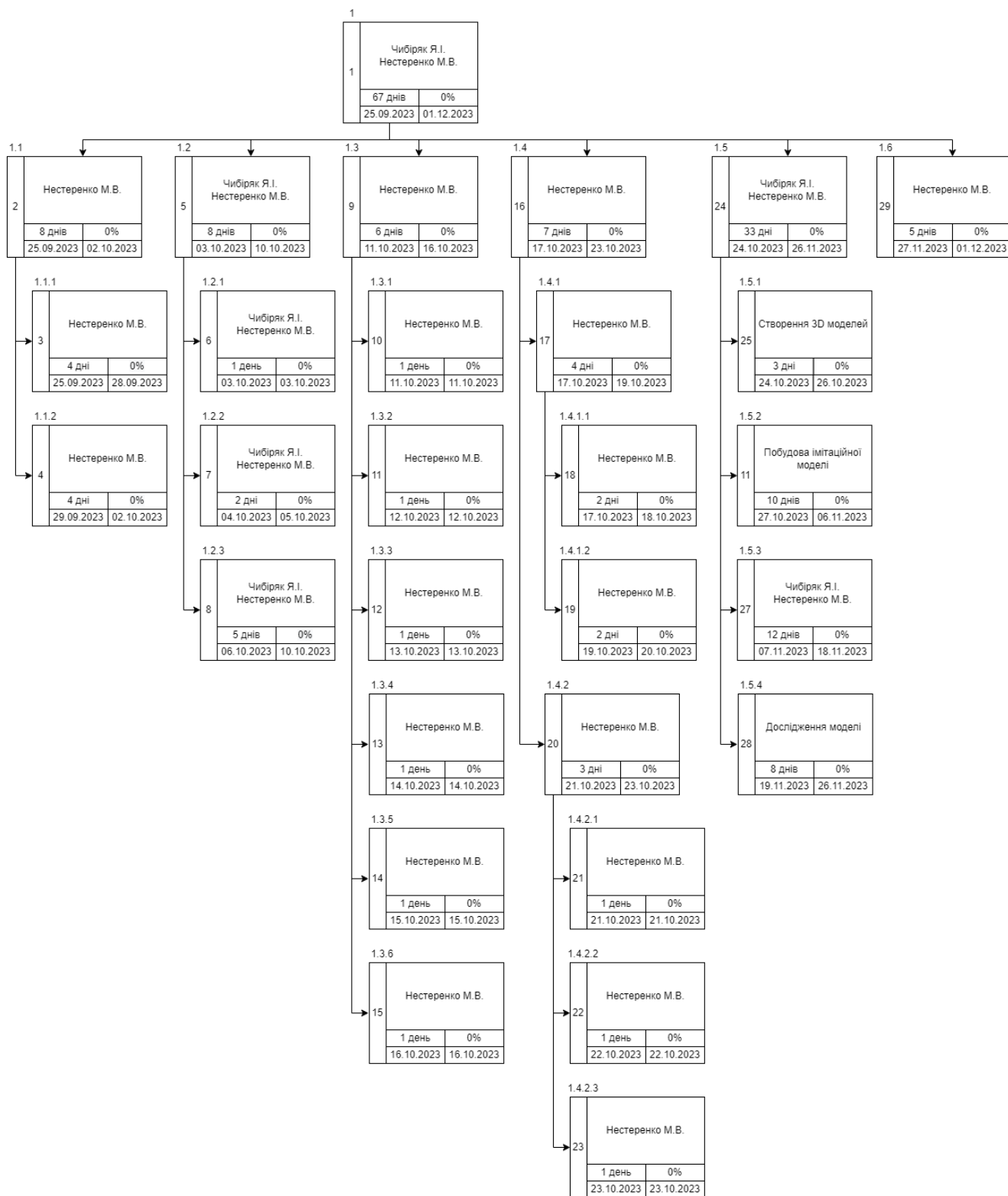


Рисунок А.2 – Організаційна структура проекту (OBS)

### **А.3 Побудова календарного графіку виконання ІТ – проекту**

Графік виконання проекту показано на діаграмі Ганта, створеній за допомогою Microsoft Project. На ній вказано завдання, їх взаємозв'язки та час виконання з урахуванням свят, вихідних та доступних ресурсів.

Діаграма Ганта зображена на рисунку А.3, показує порядок виконання завдань та їх часові рамки.

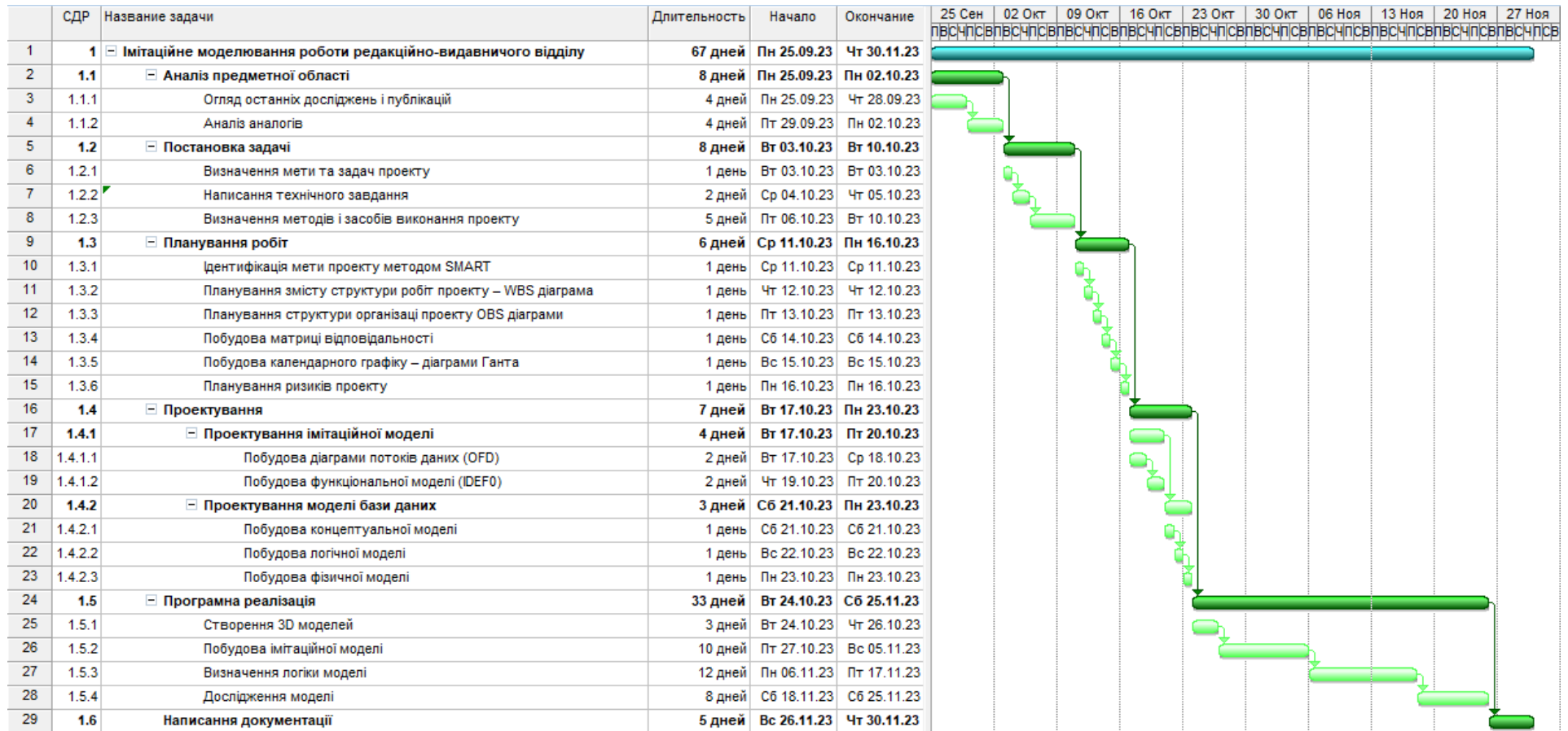


Рисунок А.3 – Діаграма Ганта

#### А.4 Планування ризиків проекту

Управління ризиками. До основних ризиків розробки імітаційної моделі роботи редакційно-видавничого відділу є:

- неоптимальне використання часу
- недостатня кваліфікація розробників
- помилки проектування
- втрата даних через технічні неполадки
- відсутність резервних копій
- зміни ТЗ
- непрацездатність розробника в зв'язку зі хворобою
- нечітке завдання на розробку
- поява альтернативного продукту
- непорозуміння між замовником та розробником

Таблиця Б.2. Ймовірність виникнення і величина ризику

№	Ризики	Виникнення	Втрати
1	Неоптимальне використання часу	4	4
2	Недостатня кваліфікація розробників	3	3
3	Помилки проектування	4	4
4	Втрата даних через технічні неполадки	2	5
5	Відсутність резервних копій	2	4
6	Зміни тз	3	3
7	Непрацездатність розробника в зв'язку зі хворобою	2	4
8	Нечітке завдання на розробку	3	3
9	Поява альтернативного продукту	3	3
10	Непорозуміння між замовником та розробником	2	2



- Недостатня кваліфікація розробників
- Зміни ТЗ
- Нечітке завдання на розробку
- Істотні
  - Неоптимальне використання часу
  - Помилки проектування
- Критичні
  - Відсутні

Класифікація ризиків за рівнем впливу:

- Прийнятні
  - Відсутні
- Виправдані
  - Недостатня кваліфікація розробників
  - Втрата даних через технічні неполадки
  - Відсутність резервних копій
  - Зміни ТЗ
  - Непрацездатність розробника в зв'язку зі хворобою
  - Нечітке завдання на розробку
  - Поява альтернативного продукту
  - Непорозуміння між замовником та розробником
- Неприпустимі
  - Неоптимальне використання часу;
  - Помилки проектування.

## ДОДАТОК Б. Лістинг програмного коду

```
CREATE TABLE type(
  Type_ID integer NOT NULL,
  Type text NOT NULL,
  Percent integer NOT NULL,
  PRIMARY KEY (Type_ID));
```

```
INSERT INTO `type` (`Type_ID`, `Type`, `Percent`) VALUES
(1, 'official publication', 3),
(2, 'social and political', 12),
(3, 'scientific', 8),
(4, 'popular', 11),
(5, 'production and practical', 9),
(6, 'educational', 20),
(7, 'literary and artistic', 12),
(8, 'reference', 5),
(9, 'advertising', 20);
```

```
CREATE TABLE subtype(
  Type_ID integer NOT NULL,
  Subtype_ID integer NOT NULL,
  Subtype text NOT NULL,
  Percent integer NOT NULL,
  PRIMARY KEY (Subtype_ID),
  FOREIGN KEY (Type_ID) REFERENCES type (Type_ID));
```

```
INSERT INTO `subtype` (`Type_ID`, `Subtype_ID`, `Subtype`, `Percent`) VALUES
(1, 1, 'regulatory edition', 3),
(1, 2, 'standards', 2),
(1, 3, 'specifications', 50),
(1, 4, 'instructions', 45),
(3, 5, 'monographs', 10),
(3, 6, 'dissertation abstracts', 12),
(3, 7, 'theses of reports', 25),
(3, 8, 'materials of conferences', 25),
(3, 9, 'collections of scientific papers', 13),
(3, 10, 'preprints', 15),
(5, 11, 'practical advisors', 33),
(5, 12, 'guidelines', 33),
(5, 13, 'methodical manuals', 34),
(6, 14, 'educational programmes', 25),
(6, 15, 'textbooks', 12),
(6, 16, 'manuals', 13),
(6, 17, 'practicums', 25),
(6, 18, 'workbooks', 25),
(8, 19, 'reference books', 25),
(8, 20, 'catalogs', 25),
(8, 21, 'guides', 25),
```



```
(8, '22', 'price lists', '25'),
(9, '23', 'avenues', '40'),
(9, '24', 'posters', '20'),
(9, '25', 'booklets', '40');
```

```
CREATE TABLE Parameters(
  Parameter_ID integer NOT NULL,
  Type_ID integer NOT NULL,
  Subtype_ID integer NULL,
  Direction 1 text NOT NULL,
  Direction 2 text NULL,
  Direction 3 text NULL,
  Direction 4 text NULL,
  Direction 5 text NULL,
  Amount_of_pages_min integer NOT NULL,
  Amount_of_pages_max integer NOT NULL,
  Publication_format_1 text NOT NULL,
  Publication_format_2 text NULL,
  Publication_format_3 text NULL,
  Line_spacing_min float(10) NOT NULL,
  Line_spacing_max float(10) NOT NULL,
  Font_size_min integer NOT NULL,
  Font_size_max integer NOT NULL,
  Manufacturing_method_1 Text NULL,
  Manufacturing_method_2 Text NULL,
  Manufacturing_method_3 Text NULL,
  Circulation_1 integer NULL,
  Circulation_2 integer NULL,
  Circulation_3 integer NULL,
  PRIMARY KEY (Parameter_ID),
  FOREIGN KEY (Type_ID) REFERENCES type (Type_ID),
  FOREIGN KEY (Subtype_ID) REFERENCES subtype (Subtype_ID));
```

```
INSERT INTO `parameters` (`Parameter_ID`, `Type_ID`, `Subtype_ID`, `Amount_of_pages_min`,
`Amount_of_pages_max`, `Publication_format_1`, `Publication_format_2`,
`Publication_format_3`, `Line_spacing_min`, `Line_spacing_max`, `Font_size_min`,
`Font_size_max`, `Manufacturing_method_1`, `Manufacturing_method_2`,
`Manufacturing_method_3`, `Circulation_1`, `Circulation_2`, `Circulation_3`)
VALUES('1','1','1','20','200','A4','A5','1','1.5','8','14','only printed','only electronic','printed and
electronic','50','25','25'),('2','1','2','20','300','A4','A5','1','1.5','8','14','only printed','only
electronic','printed and electronic','50','25','25'),('3','1','3','20','300','A4','A5','1','1.5','8','14','only
printed','only electronic','printed and
electronic','50','25','25'),('4','1','4','10','300','A4','A5','1','1.5','8','14','only printed','only
electronic','printed and
electronic','50','25','25'),('5','2',NULL,'10','200','A4','A5','1','1.5','12','14','only printed','only
electronic','printed and electronic','50','30','30'),('6','3','5','50','350','A4','A5','1','1.5','12','14','only
printed','only electronic','printed and
electronic','200','100','100'),('7','3','6','15','25','A5','','1','1.5','8','12','only printed','only
electronic','printed and electronic','50','25','25'),('8','3','7','25','200','A4','A5','1','1.5','8','14','only
```

```

printed','only electronic','printed and
electronic','200','100','100'),(9,'3','8','25','350','A4','A5','1','1.5','8','14','only printed','only
electronic','printed and
electronic','200','100','100'),(10,'3','9','25','350','A4','A5','1','1.5','8','14','only printed','only
electronic','printed and electronic','100','50','50'),(11,'3','10','8','80','A4','A5','1','1.5','8','14','only
printed','only electronic','printed and
electronic','20','10','10'),(12,'4',NULL,'10','200','A4','A5','1','1.5','12','14','only printed','only
electronic','printed and electronic','50','30','30'),(13,'5','11','8','250','A4','A5','1','1.5','8','14','only
printed','only electronic','printed and
electronic','50','25','25'),(14,'5','12','8','100','A4','A5','1','1.5','8','14','only printed','only
electronic','printed and electronic','50','25','25'),(15,'5','13','50','350','A4','A5','1','1.5','8','14','only
printed','only electronic','printed and
electronic','150','75','75'),(16,'6','14','8','40','A4','A5','1','1.5','8','14','only printed','only
electronic','printed and electronic','3','2','2'),(17,'6','15','50','350','A4','A5','1','1.5','8','14','only
printed','only electronic','printed and
electronic','200','100','100'),(18,'6','16','50','350','A4','A5','1','1.5','8','14','only printed','only
electronic','printed and
electronic','200','100','100'),(19,'6','17','10','350','A4','A5','1','1.5','8','14','only printed','only
electronic','printed and electronic','200','100','100'),(20,'6','18','5','75','A4','A5','1','1.5','8','14','only
printed','only electronic','printed and
electronic','200','100','100'),(21,'7',NULL,'10','400','A3','A4','1','1.5','12','16','only printed','only
electronic','printed and
electronic','1000','500','500'),(22,'8','19','10','350','A4','A5','1','1.5','8','14','only printed','only
electronic','printed and electronic','100','50','50'),(23,'8','20','5','300','A4','A5','1','1.5','5','14','only
printed','only electronic','printed and
electronic','100','50','50'),(24,'8','21','5','50','A4','A5','1','1.5','5','14','only printed','only
electronic','printed and electronic','100','50','50'),(25,'8','22','5','50','A4','A5','1','1.5','5','14','only
printed','only electronic','printed and
electronic','100','50','50'),(26,'9','23','5','50','A4','A5','1','1.5','5','14','only printed','only
electronic','printed and electronic','100','50','50'),(27,'9','24','1','1','A0','A1','A2','1','1.5','17','22','only
printed','only electronic','printed and
electronic','100','50','50'),(28,'9','25','1','1','A4','A5','1','1.5','5','14','only printed','only
electronic','printed and electronic','1000','500','500')
CREATE TABLE `input1`.`method_of_publication` (`method_of_publication_ID` INT NOT NUL
L, `method_of_publication` TEXT NOT NULL, PRIMARY KEY (`method_of_publication_ID`))
ENGINE = InnoDB;
INSERT INTO `method_of_publication` (`method_of_publication_ID`, `method_of_publication`)
VALUES ('1', 'offset'), ('2', 'digital'), ('3', 'stencil');

```

```

CREATE TABLE publication_layout(
  Type_ID integer NOT NULL,
  min INT NOT NULL,
  ager INT NOT NULL,
  max INT NOT NULL,
  FOREIGN KEY (Type_ID) REFERENCES type (Type_ID));
INSERT INTO `publication_layout` (`Type_ID`, `min`, `ager`, `max`) VALUES
('1', '5', '8', '10'),
('2', '3', '5', '10'),
('3', '7', '10', '13'),

```

```
(4, '7', '10', '13'),
(5, '7', '10', '13'),
(6, '7', '10', '13'),
(7, '3', '5', '10'),
(8, '4', '8', '13'),
(9, '10', '17', '22');
```

```
CREATE TABLE preparation_of_the_original_layout(
  Type_ID integer NOT NULL,
  min INT NOT NULL,
  ager INT NOT NULL,
  max INT NOT NULL,
  FOREIGN KEY (Type_ID) REFERENCES type (Type_ID));
INSERT INTO `preparation_of_the_original_layout` (`Type_ID`, `min`, `ager`, `max`) VALUES
(1, 2, 4, 8),
(2, 1, 3, 6),
(3, 3, 5, 7),
(4, 3, 5, 7),
(5, 3, 5, 7),
(6, 3, 5, 7),
(7, 1, 3, 5),
(8, 2, 4, 6),
(9, 5, 7, 9);
```

```
CREATE TABLE `method_of_publication` (
  `method_of_publication_ID` int(11) NOT NULL,
  `method_of_publication` text NOT NULL,
  PRIMARY KEY (method_of_publication_ID));
INSERT INTO `method_of_publication` (`method_of_publication_ID`, `method_of_publication`)
VALUES
(1, 'offset'),
(2, 'digital'),
(3, 'stencil');
```

Lead editor workspace – On Exit

```
Object current = ownerobject(c);
Object item = param(1);
if (item.Editor_ID==0){
  Object current = ownerobject(c);
  Object item = param(1);
  int rownumber = param(2);
  int rownumber1 = param(3);
  int port = param(4);
  int stream = getstream(ownerobject(c));
  double randomnum = uniform(0.0, 100.0, stream);
  double total = 0.0;
  int foundmatch = 0;
  int i=1;
  for(i=1; i<Table.query("SELECT COUNT(Type_ID) FROM Type")[1][1]+1;i++){
```



```

        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID, item.Subtype_ID)[1][1][7]!="")
            Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][7]);
        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID, item.Subtype_ID)[1][1][8]!="")
            Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][8]);

```

```

        item.labels.assert("Direction",Direction_range[duniform(1,Direction_range.length,getstream
(current))]);

```

```

        item.labels.assert("Amount_of_pages",duniform(Table("Parameters")[Table.query("SELEC
T * FROM Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][9],Table("Parameters")[Table.query("SELECT * FROM Parameters
WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][10],getstream(current)))/**/;

```

```

        Array Publication_format_range = [];
        Publication_format_range.push(Table("Parameters")[Table.query("SELECT *
FROM Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][11]);
        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID, item.Subtype_ID)[1][1][12]!="")
            Publication_format_range.push(Table("Parameters")[Table.query("SELECT
* FROM Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][12]);
        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID, item.Subtype_ID)[1][1][13]!="")
            Publication_format_range.push(Table("Parameters")[Table.query("SELECT
* FROM Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][13]);

```

```

        item.labels.assert("Publication_format",Publication_format_range[duniform(1,Publication_f
ormat_range.length,getstream(current))]);

```

```

        Array Line_spacing_range = [];
        Line_spacing_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][14]);
        Line_spacing_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][15]);

```

```

        item.labels.assert("Line_spacing",Line_spacing_range[duniform(1,Line_spacing_range.leng
th,getstream(current))]);

```

```

        item.labels.assert("Font_size",duniform(Table("Parameters")[Table.query("SELECT
* FROM Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][16],Table("Parameters")[Table.query("SELECT * FROM Parameters
WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][17],getstream(current)))/**/;

```

```

        Array Manufacturing_method_range = [];
        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID, item.Subtype_ID)[1][1][18]!= "")

```

```

        Manufacturing_method_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][18]);

```

```

        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID, item.Subtype_ID)[1][1][19]!= "")

```

```

        Manufacturing_method_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][19]);

```

```

        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID, item.Subtype_ID)[1][1][20]!= "")

```

```

        Manufacturing_method_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID,
item.Subtype_ID)[1][1][20]);

```

```

        item.labels.assert("Manufacturing_method",Manufacturing_method_range[duniform(1,Man
ufacturing_method_range.length,getstream(current))]);

```

```

        if(item.Manufacturing_method.match(/only_printed/).length>0)item.labels.assert("Circulati
on",Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE Type_ID=[\$1]
AND Subtype_ID=[\$2]", item.Type_ID, item.Subtype_ID)[1][1][21]);

```

```

        if(item.Manufacturing_method.match(/only_electronic/).length>0)
        item.labels.assert("Circulation",Table("Parameters")[Table.query("SELECT * FROM Parameters
WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID, item.Subtype_ID)[1][1][22]);

```

```

        if(item.Manufacturing_method.match(/printed_and_electronic/).length>0)
        item.labels.assert("Circulation",Table("Parameters")[Table.query("SELECT * FROM Parameters
WHERE Type_ID=[\$1] AND Subtype_ID=[\$2]", item.Type_ID, item.Subtype_ID)[1][1][23]);

```

```

        }else{

```

```

            Array Direction_range = [];

```

```

            Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1]", item.Type_ID)[1][1][4]);

```

```

            if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1]", item.Type_ID)[1][1][5]!= "")

```

```

                Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1]", item.Type_ID)[1][1][5]);

```

```

        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1]", item.Type_ID)[1][1][6]!="")
            Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1]", item.Type_ID)[1][1][6]);
        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1]", item.Type_ID)[1][1][7]!="")
            Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1]", item.Type_ID)[1][1][7]);
        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1]", item.Type_ID)[1][1][8]!="")
            Direction_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1]", item.Type_ID)[1][1][8]);

```

```

        item.labels.assert("Direction",Direction_range[duniform(1,Direction_range.length,getstream
(current))]);

```

```

        item.labels.assert("Amount_of_pages",duniform(Table("Parameters")[Table.query("SELEC
T * FROM Parameters WHERE Type_ID=[\$1]",
item.Type_ID)[1][1][9],Table("Parameters")[Table.query("SELECT * FROM Parameters
WHERE Type_ID=[\$1]", item.Type_ID)[1][1][10],getstream(current)))/**/;

```

```

        Array Publication_format_range = [];
        Publication_format_range.push(Table("Parameters")[Table.query("SELECT *
FROM Parameters WHERE Type_ID=[\$1]", item.Type_ID)[1][1][11]);
        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1]", item.Type_ID)[1][1][12]!="")
            Publication_format_range.push(Table("Parameters")[Table.query("SELECT
* FROM Parameters WHERE Type_ID=[\$1]", item.Type_ID)[1][1][12]);
        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[\$1]", item.Type_ID)[1][1][13]!="")
            Publication_format_range.push(Table("Parameters")[Table.query("SELECT
* FROM Parameters WHERE Type_ID=[\$1]", item.Type_ID)[1][1][13]);

```

```

        item.labels.assert("Publication_format",Publication_format_range[duniform(1,Publication_f
ormat_range.length,getstream(current))]);

```

```

        Array Line_spacing_range = [];
        Line_spacing_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1]", item.Type_ID)[1][1][14]);
        Line_spacing_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[\$1]", item.Type_ID)[1][1][15]);

```

```

        item.labels.assert("Line_spacing",Line_spacing_range[duniform(1,Line_spacing_range.leng
th,getstream(current))]);

```

```

        item.labels.assert("Font_size",duniform(Table("Parameters")[Table.query("SELECT
* FROM Parameters WHERE Type_ID=[\$1]",

```

```
item.Type_ID)[1][1]][16],Table("Parameters")[Table.query("SELECT * FROM Parameters
WHERE Type_ID=[$1]", item.Type_ID)[1][1]][17],getstream(current))/**/;
```

```
    Array Manufacturing_method_range = [];
    if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[$1]", item.Type_ID)[1][1]][18]!="")
```

```
        Manufacturing_method_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[$1]", item.Type_ID)[1][1]][13]);
        if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[$1]", item.Type_ID)[1][1]][19]!="")
```

```
            Manufacturing_method_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[$1]", item.Type_ID)[1][1]][14]);
            if(Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE
Type_ID=[$1]", item.Type_ID)[1][1]][20]!="")
```

```
                Manufacturing_method_range.push(Table("Parameters")[Table.query("SELECT * FROM
Parameters WHERE Type_ID=[$1]", item.Type_ID)[1][1]][15]);
```

```
            item.labels.assert("Manufacturing_method",Manufacturing_method_range[duniform(1,Man
ufacturing_method_range.length,getstream(current))]);
```

```
        if(item.Manufacturing_method.match(/only_printed/).length>0)item.labels.assert("Circulati
on",Table("Parameters")[Table.query("SELECT * FROM Parameters WHERE Type_ID=[$1]
AND Subtype_ID=[$2]", item.Type_ID, item.Subtype_ID)[1][1]][21]);
        if(item.Manufacturing_method.match(/only_electronic/).length>0)
item.labels.assert("Circulation",Table("Parameters")[Table.query("SELECT * FROM Parameters
WHERE Type_ID=[$1] AND Subtype_ID=[$2]", item.Type_ID, item.Subtype_ID)[1][1]][22]);
        if(item.Manufacturing_method.match(/printed_and_electronic/).length>0)
item.labels.assert("Circulation",Table("Parameters")[Table.query("SELECT * FROM Parameters
WHERE Type_ID=[$1] AND Subtype_ID=[$2]", item.Type_ID, item.Subtype_ID)[1][1]][23]);
    }
}
```

Editors queue – Send To Port

Object item = param(1);

Object current = ownerobject(c);

/\*\*Custom Code\*/

```
if(item.Direction.match(/scientific/).length>0){
    if(item.Direction.match(/mathematical scientific/).length>0||item.Direction.match(/technical
scientific/).length>0){return 1;}
    if(item.Direction.match(/economical
scientific/).length>0||item.Direction.match(/philological scientific/).length>0){return 3;}
    if(item.Direction.match(/medical scientific/).length>0){return 5;}
}else{
```



```

    if(item.Direction.match(/mathematical/).length>0||item.Direction.match(/technical/).length>
0){if(opavailable(current,1) && current.outObjects[2].subnodes.length > 20){return 1;}else{return
2;}}
    if(item.Direction.match(/economical/).length>0||item.Direction.match(/philological/).length
>0){if(opavailable(current,3) && current.outObjects[4].subnodes.length > 20){return
3;}else{return 4;}}
    if(item.Direction.match(/medical/).length>0){if(opavailable(current,5) &&
current.outObjects[6].subnodes.length > 20){return 5;}else{return 6;}}
}

```

Editors queue – Transport Resource

```
/**Custom Code*/
```

```
Object current = ownerobject(c);
```

```
Object item = param(1);
```

```
int port = param(2);
```

```
Object destination = param(3);
```

```
double priority = param(4);
```

```
int preempt = param(5);
```

```

switch(port){
    case 1:
        return current.centerObjects[1];
    case 2:
        return current.centerObjects[2];
    case 3:
        return current.centerObjects[3];
    case 4:
        return current.centerObjects[4];
    case 5:
        return current.centerObjects[5];
    case 6:
        return current.centerObjects[6];
    default:
        transportoutcomplete(current, item, port);
        transportincomplete(current.outObjects[port], item, opipno(current, port));
        moveobject(item, current.outObjects[port], port);
}

```

Editor Mat Tech I workspace – Process Time

```
/**Custom Code*/
```

```
Object current = ownerobject(c);
```

```
Object item = param(1);
```

```
double time_of_process;
```

```
if (item.Return_from_author==0){
```

```

    time_of_process= triangular(
        256.93*pow(0.5,item.Publication_format.charAt(3).toNum())*(2.3-
0.1*item.Font_size)*(1.32-0.13*pow(item.Line_spacing,2)),
        321.16*pow(0.5,item.Publication_format.charAt(3).toNum())*(2.3-
0.1*item.Font_size)*(1.32-0.13*pow(item.Line_spacing,2)),

```

```

        385.39*pow(0.5,item.Publication_format.charAt(3).toNum()*(2.3-
0.1*item.Font_size)*(1.32-0.13*pow(item.Line_spacing,2)),
        getstream(current))*item.Amount_of_pages/4;
    }else{
        time_of_process= triangular(
            102.77*pow(0.5,item.Publication_format.charAt(3).toNum()*(2.3-
0.1*item.Font_size)*(1.32-0.13*pow(item.Line_spacing,2)),
            128.46*pow(0.5,item.Publication_format.charAt(3).toNum()*(2.3-
0.1*item.Font_size)*(1.32-0.13*pow(item.Line_spacing,2)),
            154.16*pow(0.5,item.Publication_format.charAt(3).toNum()*(2.3-
0.1*item.Font_size)*(1.32-0.13*pow(item.Line_spacing,2)),
            getstream(current))*item.Amount_of_pages/4;
        }
    item.labels.assert("time_of_process").value=time_of_process;
    return time_of_process;

```

Corrector workspace – Process Time

/\*\*Custom Code\*/

Object current = ownerobject(c);

Object item = param(1);

double time\_of\_process;

time\_of\_process = triangular(

171.29\*pow(0.5,item.Publication\_format.charAt(3).toNum()\*(2.3-  
0.1\*item.Font\_size)\*(1.32-0.13\*pow(item.Line\_spacing,2)),

214.11\*pow(0.5,item.Publication\_format.charAt(3).toNum()\*(2.3-  
0.1\*item.Font\_size)\*(1.32-0.13\*pow(item.Line\_spacing,2)),

265.93\*pow(0.5,item.Publication\_format.charAt(3).toNum()\*(2.3-  
0.1\*item.Font\_size)\*(1.32-0.13\*pow(item.Line\_spacing,2)),  
getstream(current))\*item.Amount\_of\_pages/2;

item.labels.assert("time\_of\_process").value=time\_of\_process;

return time\_of\_process;

Technical editor 1 workspace – Process Time

/\*\*Custom Code\*/

Object current = ownerobject(c);

Object item = param(1);

int time\_of\_printing\_method\_determination=duniform(5,7,getstream(current));

double time\_of\_process;

int minimum=0; int ager=0; int maximum=0;

if(Table.query("SELECT \* FROM Publication\_layout WHERE Type\_ID=[\$1]",

item.Type\_ID).numRows>0){

minimum=Table("Publication\_layout")[Table.query("SELECT \* FROM Publication\_layout  
WHERE Type\_ID=[\$1]", item.Type\_ID)[1][1]][2];

ager=Table("Publication\_layout")[Table.query("SELECT \* FROM Publication\_layout  
WHERE Type\_ID=[\$1]", item.Type\_ID)[1][1]][3];

```

        maximum=Table("Publication_layout")[Table.query("SELECT * FROM Publication_layout
WHERE Type_ID=[$1]", item.Type_ID)[1][1]][4];
    }
    int
    time_of_arrangement=16*0.5^item.Publication_format.charAt(3).toNum()*triangular(minimum,ag
er,maximum)*item.Amount_of_pages;

    if(Table.query("SELECT * FROM Preparation_of_the_original_layout WHERE Type_ID=[$1]",
item.Type_ID).numRows>0){
        minimum=Table("Preparation_of_the_original_layout")[Table.query("SELECT * FROM
Preparation_of_the_original_layout WHERE Type_ID=[$1]", item.Type_ID)[1][1]][2];
        ager=Table("Preparation_of_the_original_layout")[Table.query("SELECT * FROM
Preparation_of_the_original_layout WHERE Type_ID=[$1]", item.Type_ID)[1][1]][3];
        maximum=Table("Preparation_of_the_original_layout")[Table.query("SELECT * FROM
Preparation_of_the_original_layout WHERE Type_ID=[$1]", item.Type_ID)[1][1]][4];
    }
    int
    time_of_preparing_for_printing=16*0.5^item.Publication_format.charAt(3).toNum()*triangular(mi
nimum,ager,maximum)*item.Amount_of_pages;
    time_of_process =
    (time_of_printing_method_determination+time_of_arrangement+time_of_preparing_for_printing)/
    2;
    item.labels.assert("time_of_process").value=time_of_process;
    return time_of_process;

```

Technical editor 1 workspace – On Exit

```
/**Custom Code*/
```

```
Object current = ownerobject(c);
```

```
Object item = param(1);
```

```
int port = param(2);
```

```
if
```

```
(item.Amount_of_pages>=50&&item.Circulation>=100)item.labels.assert("Printing_method",Tabl
e("Publication_method")[1][2]);
```

```
if
```

```
(item.Amount_of_pages<50&&item.Circulation<100)item.labels.assert("Printing_method",Table("
Publication_method")[2][2]);
```

```
if
```

```
((item.Amount_of_pages>=50&&item.Circulation<100)||((item.Amount_of_pages<50&&item.Circ
ulation>=100)))item.labels.assert("Printing_method",Table("Publication_method")[3][2]);
```