

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему: «Імітаційна модель діяльності ремонтного підрозділу засобів зв'язку»

за спеціальністю 122 «Комп'ютерні науки»,
освітньо-професійна програма «Інформаційні технології проектування»

Виконавець роботи: студентка групи ІТ-81 Нестеренко Марина Володимирівна

Кваліфікаційна робота бакалавра
захищена на засіданні ЕК
з оцінкою

_____ «__» _____ 2022 р.

Науковий керівник

(підпис)

к.т.н., доц., Чибіряк Я.І.

(науковий ступінь, вчене звання, прізвище та ініціали)

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____

(підпис)

Суми-2022

Сумський державний університет
 Факультет електроніки та інформаційних технологій
 Кафедра інформаційних технологій
 Спеціальність 122 «Комп'ютерні науки»
 Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ

Зав. кафедри ІТ

_____ В. В. Шендрик
 «__» _____ 2022 р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ

Нестеренко Марина Володимирівна

1 Тема роботи Імітаційна модель діяльності ремонтного підрозділу засобів зв'язку

керівник роботи Чибіряк Яна Іванівна, к.т.н., доцент,

затверджені наказом по університету від «27» квітня 2022 р. № 0301-VI

2 Строк подання студентом роботи «10» червня 2022 р.

3 Вихідні дані до роботи _____
технічне завдання на розробку імітаційної моделі ремонтного підрозділу

4 Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно вирішити) _____

аналіз предметної області, моделювання та проектування імітаційної моделі, програмна реалізація імітаційної моделі

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____

актуальність роботи, постановка задачі, порівняння програмних середовищ

моделювання, вимоги до моделі, проектування імітаційної моделі, створення та підключення бази даних, розробка 3D моделей та введення їх у модель програмна реалізація імітаційної моделі, дослідження моделі ремонтного підрозділу, висновки

6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 02.10.2022

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз предметної області	02.03.2022-13.03.2022	
2	Вибір методів та засобів реалізації моделі ремонтного підрозділу	14.03.2022-22.03.2022	
3	Планування робіт	23.03.2022-18.04.2022	
4	Моделювання та проектування імітаційної моделі	19.04.2022-30.04.2022	
5	Програмна реалізація імітаційної моделі	01.05.2022-23.05.2022	
6	Дослідження моделі ремонтного підрозділу	24.05.2022-31.05.2022	
7	Написання документації	01.06.2022-06.06.2022	

Студент

(підпис)

Нестеренко М.В.

Керівник роботи

(підпис)

к.т.н., доц. Чибіряк Я.І.

РЕФЕРАТ

Тема кваліфікаційної роботи бакалавра «Імітаційна модель діяльності ремонтного підрозділу засобів зв'язку».

Пояснювальна записка складається зі вступу, 3 розділів, висновків, списку використаних джерел із 6 найменувань, додатків. Загальний обсяг роботи – 64 сторінки, у тому числі 41 сторінки основного тексту, 2 сторінки списку використаних джерел, 18 сторінок додатків.

Кваліфікаційну роботу бакалавра присвячено розробці імітаційної моделі ремонтного підрозділу засобів зв'язку. В роботі проведено аналіз предметної області, планування робіт, написано технічне завдання. У роботі виконано моделювання, проектування та програмна реалізація імітаційної моделі ремонтного підрозділу з відображенням статистичних даних. Результатом проведеної роботи є імітаційна модель ремонтного підрозділу засобів зв'язку. Практичне значення роботи полягає у наданні практичних реалізацій щодо покращення роботи ремонтного підрозділу.

У першому розділі було обґрунтовано вибір методів дослідження, проведено аналіз програмних продуктів для реалізації моделі, деталізовано мету та задачі проекту.

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

У третьому розділі описано процес створення бази даних, налаштування візуалізації об'єктів мережі та побудови імітаційної моделі ремонтного підрозділу засобів зв'язку. У моделі реалізовано логіку роботи системи та виведення статистичних даних.

Ключові слова: імітаційна модель, ремонтний підрозділ засобів зв'язку, 3D модель.

ЗМІСТ

ВСТУП.....	6
1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	8
1.1. Обґрунтування методів дослідження	8
1.2. Аналіз продуктів-аналогів	9
1.3. Постановка задачі	13
2. МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ.....	15
2.1. Проектування імітаційної моделі	15
2.2. Проектування моделі бази даних.....	19
3. ПРОГРАМНА РЕАЛІЗАЦІЯ ІМІТАЦІЙНОЇ МОДЕЛІ	22
3.1. Структуризація даних	22
3.2. Розробка тривимірних моделей	26
3.3. Програмна реалізація імітаційної моделі.....	27
3.4. Дослідження роботи імітаційної моделі мережі передачі даних	34
ВИСНОВКИ.....	42
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	44
ДОДАТОК А	46
ДОДАТОК Б.....	54

ВСТУП

Вже важко представити людину без засобів зв'язку. Якщо раніше вони були представлені тільки телефонами, то зараз перелік розширився до смарт-годинників, модемів. Не можна не відмітити значимість безпроводних засобів зв'язку.

Якщо напочатку 19 сторіччя засоби зв'язку були представлені виключно дротовими телефонами, то сьогодні більшість засобів зв'язку відмовилася від проводів, перейшовши на бездротовий зв'язок, отримали акумулятори та стали широкодоступними.

Швидкість зв'язку має ключову роль в сьогоденному суспільстві: швидкість отримання повідомлень може попередити катастрофу, наприклад вчасна евакуація жителів сейсмічно-небезпечного регіону може врятувати тисячі життів, оперативний відгук на дзвінок у швидку допомогу або пожежникам. Все це рятує тисячі життів щодня. І це неможливо було б з застарілими методами зв'язку.

Нажаль у кожного засобу зв'язку іноді трапляються поломки: неправильне використання, прикрі випадковості та звичайне старіння програмного забезпечення. В деяких випадках має місце можливість ремонту та повторного використання пристрою. Саме цим займаються сервіс-центри. Але кількість спеціалістів обмежена нестачею кадрів, їх часу або коштів на найм нових спеціалістів. Саме тоді допомагає правильна організація ресурсів.

Моделювання дозволяє без значних фінансових витрат та ризиків для підприємства експериментувати з знаходженням оптимального розподілу кадрів та об'ємів роботи між ними для отримання максимальної результативності.

Мета: імітаційне моделювання роботи ремонтного підрозділу засобів зв'язку з метою підвищення ефективності надання ремонтних послуг.

Основні задачі для досягнення поставленої мети:

- провести аналіз предметної сфери;
- обґрунтувати вибір методів дослідження;
- розглянути та обґрунтувати вибір програмних засобів реалізації імітаційної моделі;

- побудувати імітаційну модель системи;
- створити 3D-моделі об'єктів;
- дослідити ефективність роботи системи.

1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1. Обґрунтування методів дослідження

Одним з перших методів моделювання є фізичний експеримент. Даний метод базується на створенні моделі досліджуваної системи та проведенні експериментів на даній моделі. До переваг можна віднести велику точність відносно реального об'єкту. Натомість фізичне моделювання потребує великі витрати часу та ресурсів на побудову моделі, оскільки проведення експериментів з реальним об'єктом є дуже ризикованим [1,2].

Іншим методом дослідження системи є аналітичний метод, який полягає у математичному описі системи. Він відрізняється простотою моделі, але не дозволяє аналізувати велику кількість факторів. Зазвичай така модель є статичною. Значні спрощення та відсутність можливості описати паралельні процеси призводять до неможливості точного аналізу складних систем [2].

На відміну від аналітичного, імітаційне моделювання дозволяє створити динамічну модель. Така модель враховує велику кількість факторів, описує елементи системи та взаємодію між ними. Система може перебувати у різних станах, залежно від параметрів. Враховуються не тільки статичні дані, а і випадкові величини, такі як збої системи, що неможливо передбачити при фізичному чи аналітичному моделюванні. Імітаційне моделювання дозволяє зобразити систему, не використовуючи складні математичні функції, тому не дає такої точності як аналітичний метод. Натомість воно потребує набагато менше ресурсів та часу, дозволяє експериментувати з великою кількістю параметрів та швидко отримувати результат. Графічне відображення системи дозволяє краще розуміти процеси які відбуваються у системі. Це допомагає знайти спільну мову між спеціалістами з моделювання систем та спеціалістами, що працюють у галузі, яка моделюється [9-11].

Узагальнена характеристика методів моделювання наведена у таблиці 1.1.

Таблиця 1.1 – Узагальнена характеристика методів моделювання

Метод	Витрати часу	Витрати коштів	Динамічність моделі	Можливість графічного відображення	Врахування випадкових величин
Фізичний експеримент	Великі	Великі	+	-	-
Аналітичний метод	Великі	Середні	-	-	-
Імітаційне моделювання	Малі	Малі	+	+	+

Для розробки даного проекту було обрано метод імітаційного моделювання, оскільки він потребує найменших витрат часу та коштів, при цьому враховуючи випадкові величини та дає можливість створити динамічну модель для проведення експериментів з графічним відображеннями [3-5].

1.2. Аналіз продуктів-аналогів

Для вибору програмного продукту для побудови імітаційної моделі ремонтного підрозділу було розглянуто існуючі програми та створено порівняльну таблицю (табл 1.2).

Розглянемо можливості кількох програмних продуктів детальніше [6]. Anylogic підтримує 3D візуалізацію моделі та відображення статистики, але створення звітів доступне виключно у веб-версії. Також присутня інтеграція з базами даних Access та SQL Server. Є можливість розширення функціоналу за допомогою мови програмування Java. Приклад візуалізації моделі виконаної в Anylogic зображено на рисунку 1.2.1.



Рисунок 1.2.1 – Приклад візуалізації моделі в Anylogic

Arena підтримує 3D візуалізацію моделі, відображення статистики та творення звітів у CSV файлі [8]. Є можливість інтеграції з базами даних FoxPro, Excel, Oracle, Informix, Access та розширення функціоналу за допомогою мови програмування Visual Basic. Приклад візуалізації моделі виконаної в Arena зображено на рисунку 1.2.2.

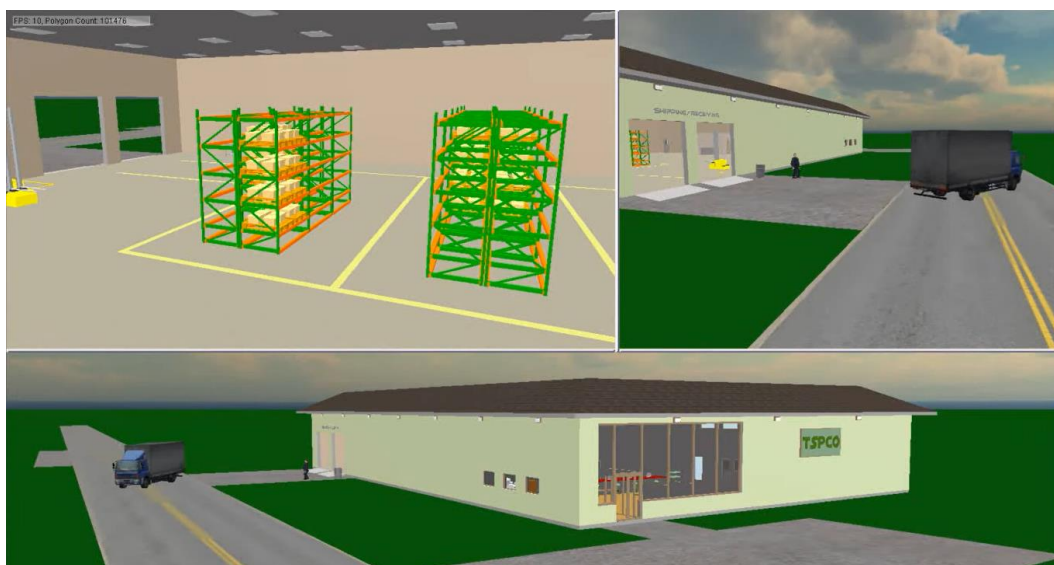


Рисунок 1.2.2 – Приклад візуалізації моделі в Arena

Simulink підтримує 3D візуалізацію моделі, відображення статистики та творення звітів у PDF, Word та HTML [7]. Є можливість інтеграції з базами даних SQLite, Cassandra, MongoDB, Neo4j та розширення функціоналу за допомогою мови програмування С. Приклад візуалізації моделі та графічного інтерфейсу виконаних в Simulink зображено на рисунку 1.2.3.

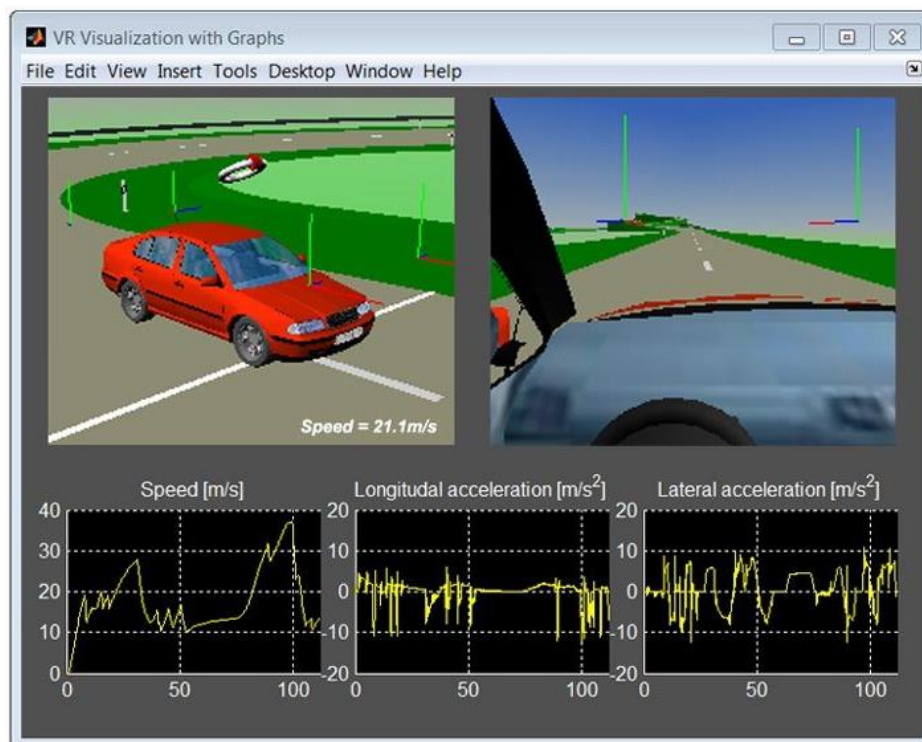


Рисунок 1.2.3 – Приклад візуалізації моделі в Simulink

Таблиця 1.2 - Порівняльна таблиця продуктів-аналогів

	Anylogic	FlexSim	AweSim	Arena	Enterprise Dynamics	Simul8	Matlab (Simulink)	ExtendSim	Vensim
3D візуалізація	+	+	+	+	+	+	+	+	-
Відображення статистичних даних	+	+	+	+	+	+	+	+	+
Інтеграція з базами даних	+(Access, SQL Server)	+(SQL Server, MySQL, Oracle)	+(Dbase, Access, FoxPro, Excel)	+(FoxPro, Excel, Oracle, Informix, Access)	+(Industry Specific Database)	+(Excel, Access, Oracle, SQL Server)	+(SQLite, Cassandra, MongoDB, Neo4j)	+(ADO)	+(Oracle, SQL Server)
Розширення функціоналу за допомогою мов програмування	+(Java)	+(C++, FlexScript)	+(C, Visual Basic)	+(Visual Basic)	+(4DScript)	+(Visual Logic)	+(C)	+(ModL)	+(Visual Basic, C, C++)
Можливість створення інтерфейсу користувача	-	+	-	-	+	+	+	-	-
Створення звітів	-(тільки на веб-платформі)	+	+(Excel, Word)	+(CSV)	-	-	+(PDF, Word, HTML)	-	-

Для моделювання ремонтного підрозділу було обрано FlexSim, оскільки він дозволяє відображати стан систему та статистичні дані у реальному часі, має можливість підключення до баз даних, таких як SQL Server, MySQL та Oracle. Також даний програмний продукт дає можливість розширення функціоналу за допомогою мови програмування C++, створення інтерфейсу користувача та звітів [9-11]. Приклад візуалізації моделі виконаної в FlexSim зображено на рисунку 1.2.4.

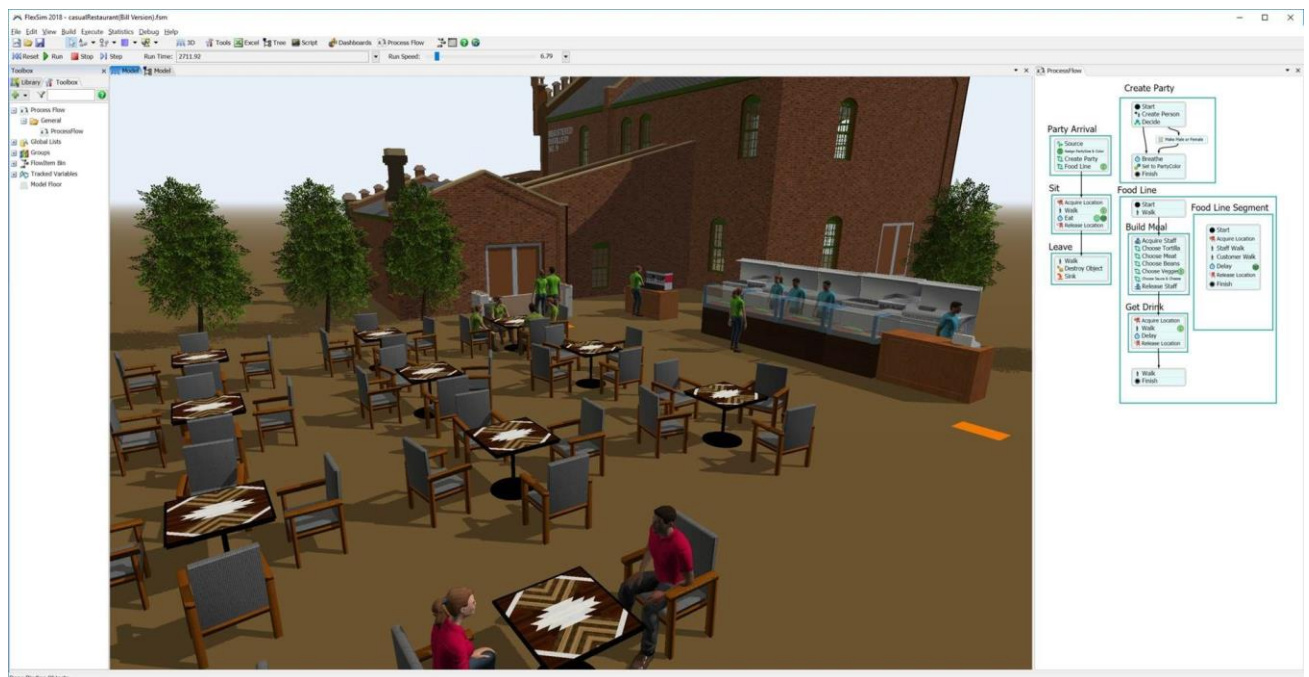


Рисунок 1.2.4 – Приклад візуалізації моделі в FlexSim

1.3. Постановка задачі

Метою даного проекту є імітаційне моделювання роботи ремонтного підрозділу засобів зв'язку з метою підвищення ефективності надання ремонтних послуг.

Основні вимоги до створюваної моделі є наступними:

- модель повинна відображати структуру існуючої системи;
- процеси, які відтворюються в моделі, повинні відповідати процесам, які протікають у реальній системі;
- модель повинна візуально відображати об'єкти реальної системи.

Для досягнення мети проекту необхідно виконати наступні задачі:

1. необхідно проаналізувати процеси, які відбуваються у ремонтному підрозділі засобів зв'язку;
2. необхідно проаналізувати існуючі підходи до дослідження систем, виявити їх переваги та недоліки та вибрати метод дослідження для системи ремонтного підрозділу;
3. необхідно проаналізувати функціонал існуючих програмних засобів для імітаційного моделювання та обрати програмний засіб для побудови імітаційної моделі системи ремонтного підрозділу;
4. у середовищі FlexSim виконати побудову імітаційної моделі системи ремонтного підрозділу засобів зв'язку;
5. з використанням програмного продукту Blender створити 3D-моделі об'єктів системи та імпортувати їх до імітаційної моделі FlexSim;
6. дослідити залежності кількості виконаних заявок та ймовірності виконання заявок всіх типів від інтервалу надходження їх у ремонт та ймовірностей надходження кожного з типів заявок.

Вимоги до проекту описані у технічному завданні на розробку проекту (додаток А).

Для реалізації даної моделі було обрано програмне забезпечення для створення імітаційної моделі FlexSim, для роботи з базами даних MySQL, для створення 3D моделей Blender.

2. МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ

2.1. Проектування імітаційної моделі

2.1.1. Створення концептуальної моделі процесу розробки імітаційної моделі

За допомогою IDEF0 опишемо функціональний аспект імітаційної моделі. Функціональне моделювання імітаційної моделі ремонтного підрозділу в нотації IDEF0 представлено на рисунку 2.1.1.

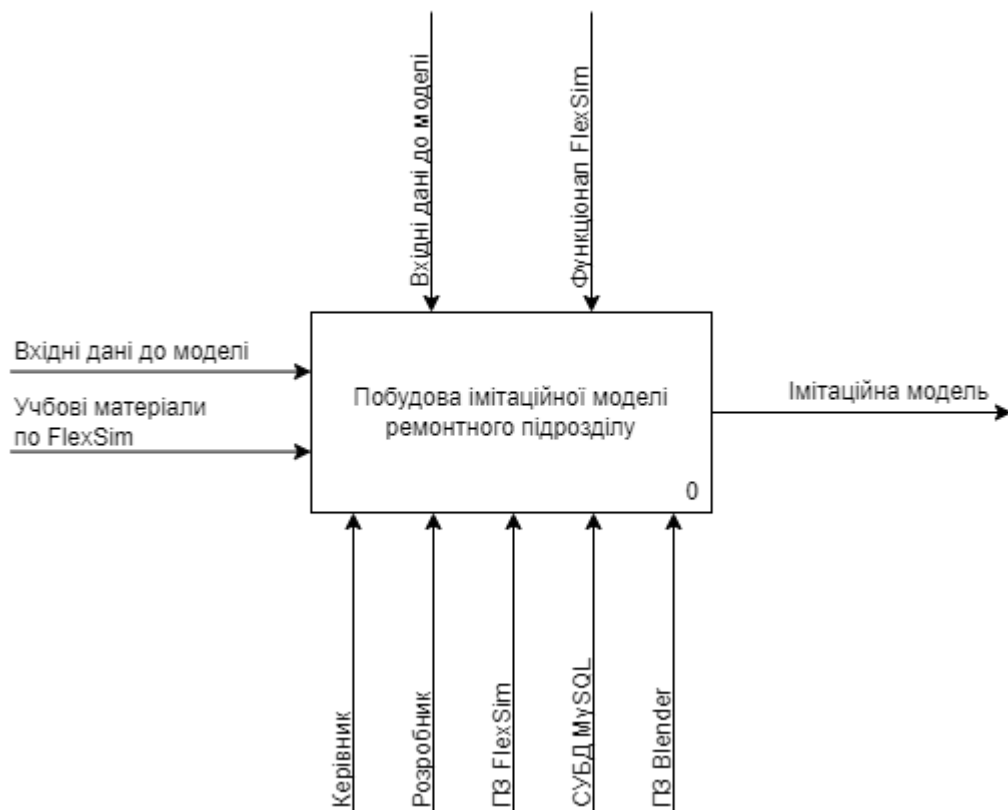


Рисунок 2.1.1 – Контекстна діаграма процесу розробки імітаційної моделі

Елементи діаграми процесу розробки імітаційної моделі:

- Процес: побудова імітаційної моделі ремонтного підрозділу засобів зв'язку;
- Вхід: вхідні дані до моделі, учбові матеріали до Flexsim;
- Управління: вхідні дані до моделі, функціонал ПЗ FlexSim, у якому реалізується імітаційна модель;

- Механізми: керівник, розробник, ПЗ FlexSim, СУБД MySQL для розробки та управління базую даних (БД), ПЗ Blender для створення 3D моделей;
- Вихід: імітаційна модель, готова для використання.

Деталізуємо основний опис концептуальної моделі розробки імітаційної моделі. Декомпозиція першого рівня в нотації IDEF представлена на рисунку 2.1.2.

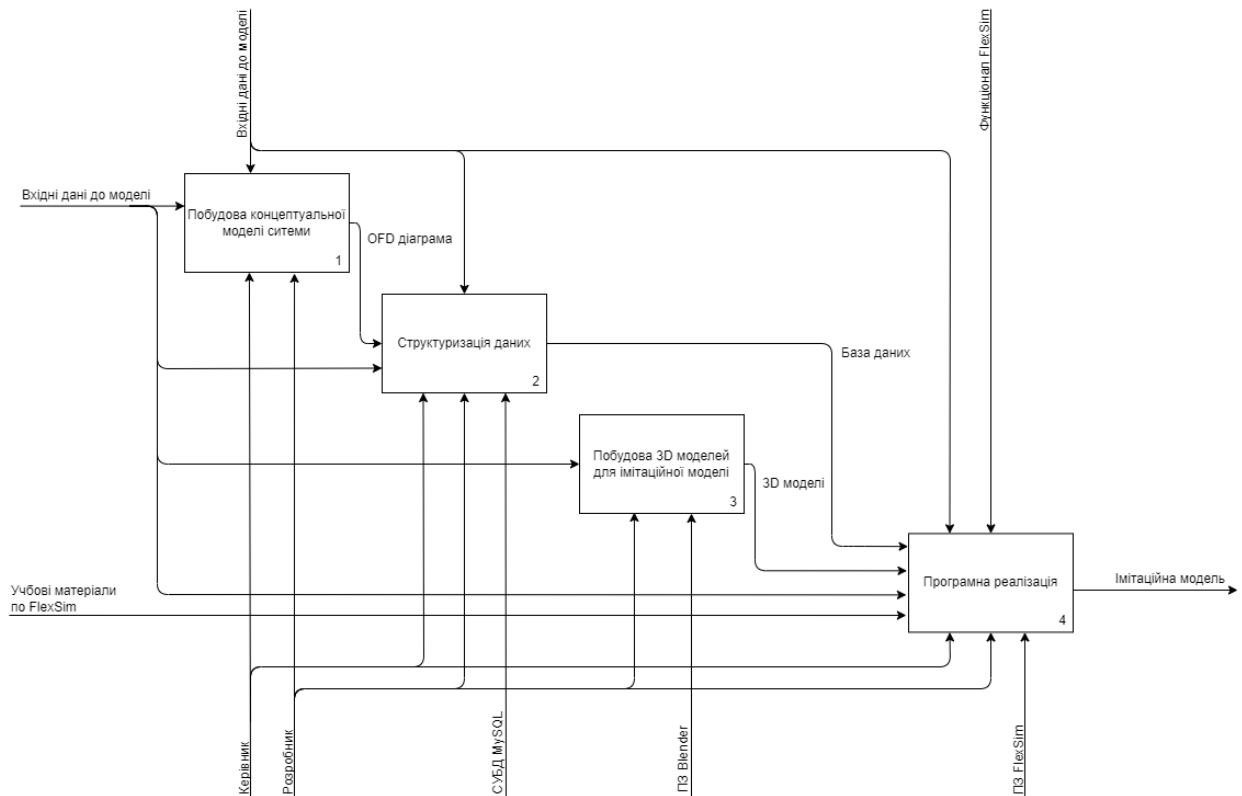


Рисунок 2.1.2 – Декомпозиція першого рівня процесу розробки імітаційної моделі

Процеси, виділені у діаграмі декомпозиції розробки імітаційної моделі:

- побудова концептуальної моделі;
- структуризація даних;
- побудова 3D моделей для імітаційної моделі;
- програмна реалізація.

2.1.2. Створення концептуальної моделі процесу використання імітаційної моделі

Функціональна модель процесу використання імітаційної моделі представлена на рисунку 2.1.3.



Рисунок 2.1.3 – Контекстна діаграма процесу використання імітаційної моделі

Елементи діаграми процесу використання імітаційної моделі:

- Процес: дослідження імітаційної моделі ремонтного підрозділу;
- Вхід: імітаційна модель, готова до використання;
- Управління: функціонал ПЗ FlexSim, постановка задачі;
- Механізми: користувач, ПЗ FlexSim, СУБД MySQL;
- Вихід: практичні рекомендації щодо покращення роботи РП.

Декомпозиція першого рівня в нотації IDEF представлена на рисунку 2.1.4.

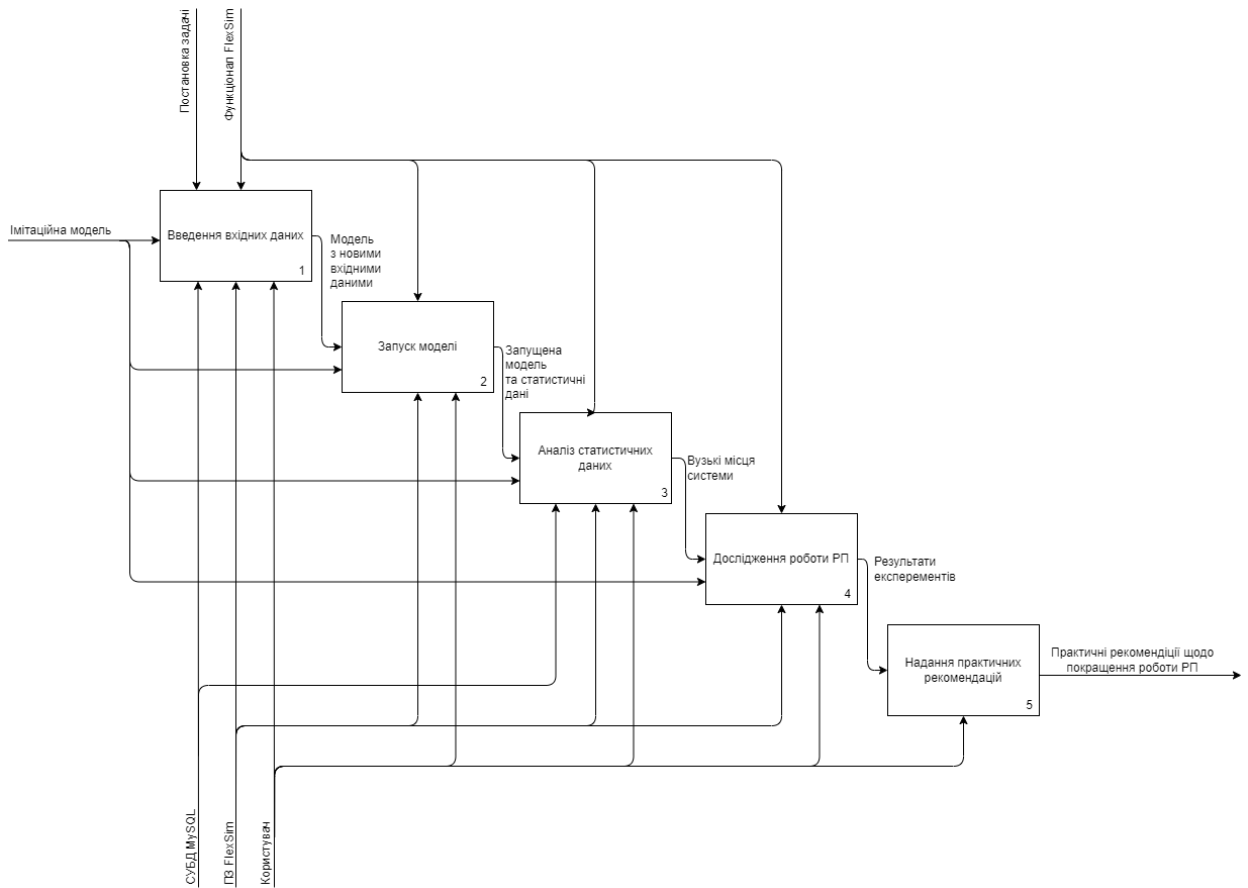


Рисунок 2.1.4 – Декомпозиція першого рівня процесу використання імітаційної моделі

Етапи, виділені у діаграмі декомпозиції процесу використання імітаційної моделі:

- введення вхідних даних;
- запуск моделі;
- аналіз статистичних даних;
- дослідження роботи РП;
- надання практичних рекомендацій.

2.1.3. Створення концептуальної схеми імітаційної моделі

OFD діаграма зображена на рисунку 2.1.5. В систему входять заявки, яким призначаються мітки з ймовірностями вказаними таблично, після чого вони

поступають у чергу заявок Order Queue. З неї заявки розподіляються диспетчерами по типам засобів зв'язку для очікування виконання заявки. Певний відсоток заявок не приймається до ремонту. Далі заявки потрапляють до майстра, який виконує її впродовж часу, визначеного на основі типу засобу зв'язку та виду поломки. Після виконання заявка виходить з системи.

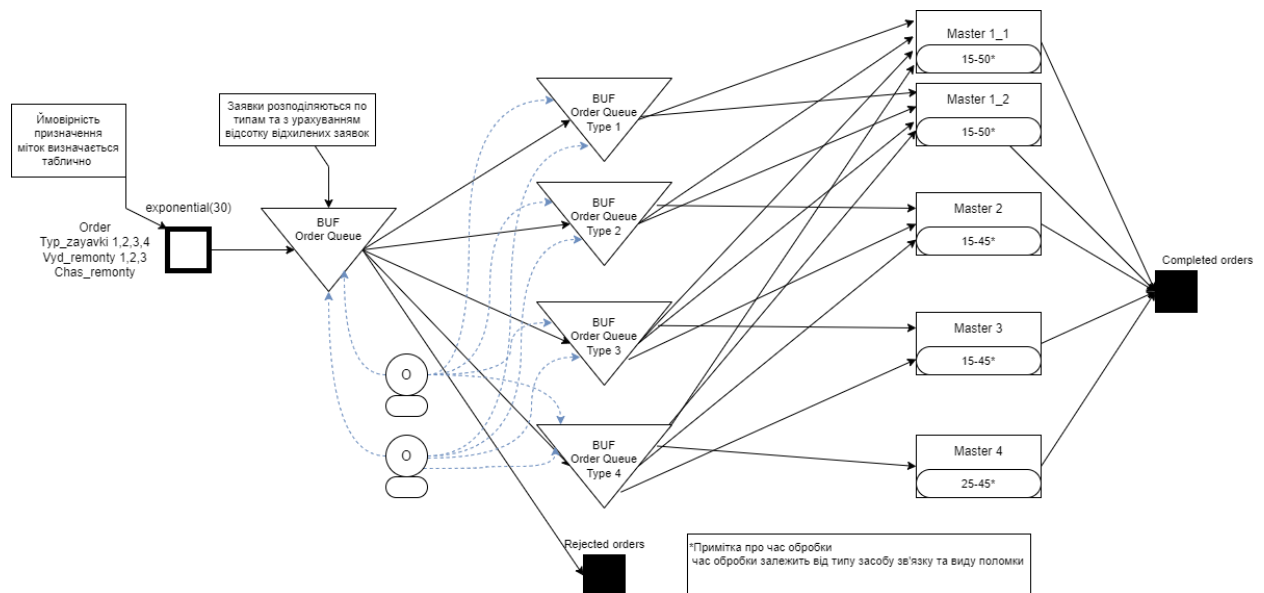


Рисунок 2.1.5 – OFD діаграма

2.2. Проектування моделі бази даних

Для збереження вхідних даних створимо базу даних з наступними таблицями:

Device – тип засобу зв'язку та ймовірність його надходження до системи;

Breakdown – вид поломки та ймовірність його надходження до системи;

Time – час виконання заявки.

Побудуємо ER діаграму для відображення сутностей та зв'язків між ними (рис. 2.2.1).

Для відображення первинних та зовнішніх ключів і зв'язків між ними побудуємо логічну модель (рис. 2.2.2).

Фізична модель дає найповнішу уяву про структуру бази даних, оскільки містить не тільки таблиці з атрибутами, а і тип даних кожного з стовпців (рис. 2.2.3).

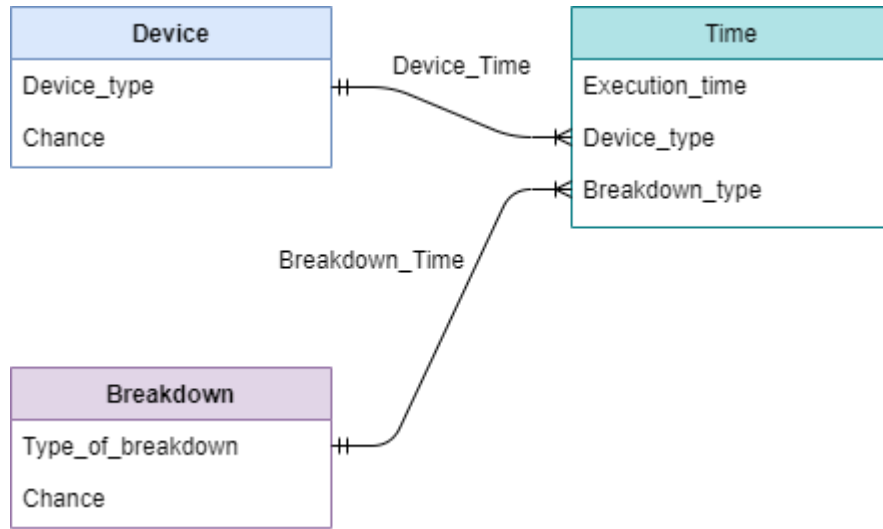


Рисунок 2.2.1 – Концептуальна модель бази даних

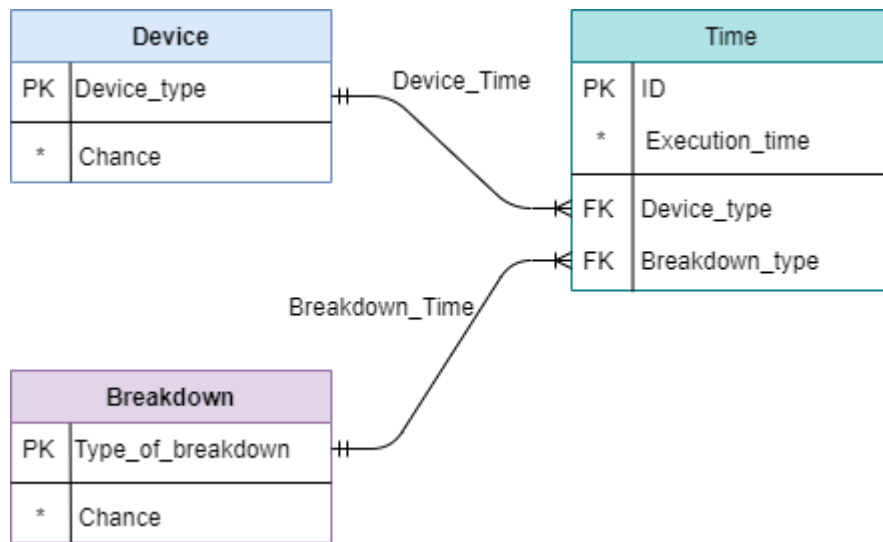


Рисунок 2.2.2 – Логічна модель бази даних

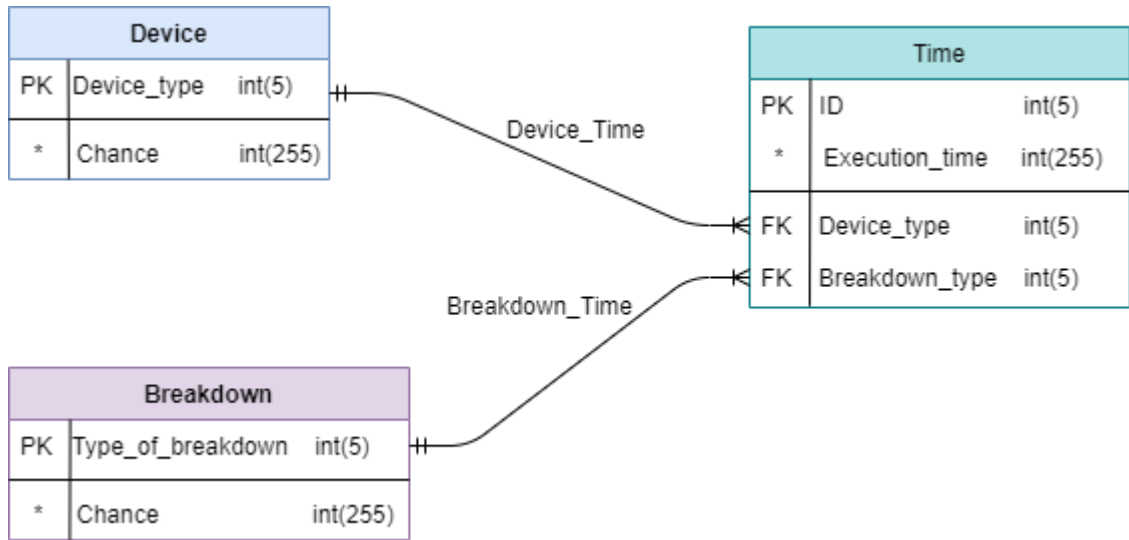


Рисунок 2.2.3 – Фізична модель бази даних

3. ПРОГРАМНА РЕАЛІЗАЦІЯ ІМІТАЦІЙНОЇ МОДЕЛІ

3.1. Структуризація даних

Вхідні дані до імітаційної моделі будуть зберігатися у базі даних, створеної за допомогою веб-додатку PhpMyAdmin та СУБД MySQL. Створюємо таблиці відповідно до фізичної моделі бази даних та вносимо дані (рис. 3.1.1-3.1.3). Позначення стовпців вказані в таблиці 3.1.1.

Таблиця 3.1 – Опис полів таблиць бази даних

Поле таблиці	Опис
Device_type	Тип засобу зв'язку
Type_of_breakdown	Вид поломки
Chance	Ймовірність
Execution_time	Час виконання ремонту

```
CREATE TABLE Device(
    Device_type integer NOT NULL,
    Chance integer NOT NULL,
    PRIMARY KEY (Device_type));
INSERT INTO `Device` (`Device_type`, `Chance`) VALUES (1, 20), (2, 30), (3, 25),
(4, 25);
```

```
CREATE TABLE Breakdown(
    Type_of_breakdown integer NOT NULL,
    Chance integer NOT NULL,
    PRIMARY KEY (Type_of_breakdown));
INSERT INTO `Breakdown` (`Type_of_breakdown`, `Chance`) VALUES (1, 50), (2,
25), (3, 25);
```

```

CREATE TABLE Time(
    Time_ID integer NOT NULL,
    Device_type integer NOT NULL,
    Type_of_breakdown integer NOT NULL,
    Execution_time integer NOT NULL,
    PRIMARY KEY (Time_ID),
    FOREIGN KEY (Device_type) REFERENCES Device (Device_type),
    FOREIGN KEY (Type_of_breakdown) REFERENCES Breakdown
(Type_of_breakdown));
INSERT INTO `Time` (`Time_ID`, `Device_type`, `Type_of_breakdown`,
`Execution_time`) VALUES (1, 1, 1, 30), (2, 1, 2, 40), (3, 1, 3, 50), (4, 2, 1, 20), (5, 2,
2, 30), (6, 2, 3, 40), (7, 3, 1, 15), (8, 3, 2, 25), (9, 3, 3, 35), (10, 4, 1, 25), (11, 4, 2, 35),
(12, 4, 3, 45);

```

				Device_type	Chance
<input type="checkbox"/>				1	20
<input type="checkbox"/>				2	30
<input type="checkbox"/>				3	25
<input type="checkbox"/>				4	25

Рисунок 3.1.1 – Заповнена таблиця Device

				Type_of_breakdown	Chance
<input type="checkbox"/>				1	50
<input type="checkbox"/>				2	25
<input type="checkbox"/>				3	25

Рисунок 3.1.2 – Заповнена таблиця Breakdown

				Time_ID	Device_type	Type_of_breakdown	Execution_time
<input type="checkbox"/>				1	1	1	30
<input type="checkbox"/>				2	1	2	40
<input type="checkbox"/>				3	1	3	50
<input type="checkbox"/>				4	2	1	20
<input type="checkbox"/>				5	2	2	30
<input type="checkbox"/>				6	2	3	40
<input type="checkbox"/>				7	3	1	15
<input type="checkbox"/>				8	3	2	25
<input type="checkbox"/>				9	3	3	35
<input type="checkbox"/>				10	4	1	25
<input type="checkbox"/>				11	4	2	35
<input type="checkbox"/>				12	4	3	45

Рисунок 3.1.3 – Заповнена таблиця Time

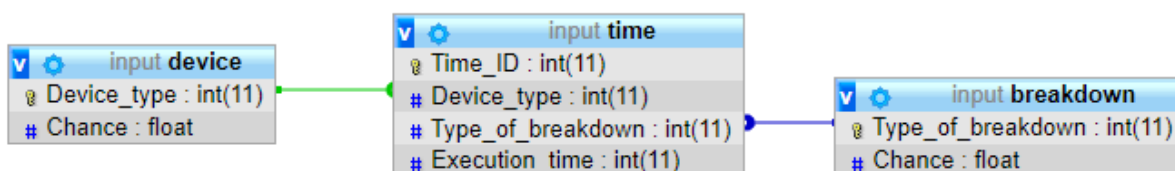


Рисунок 3.1.4 – Структура БД

Підключаємо базу даних до імітаційної моделі за допомогою Database Connector (рис. 3.1.5).

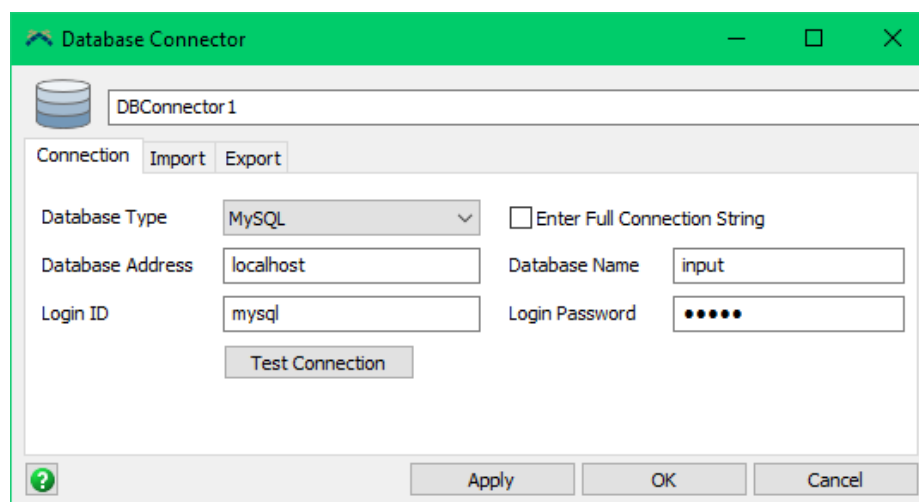


Рисунок 3.1.5 – Підключення до БД

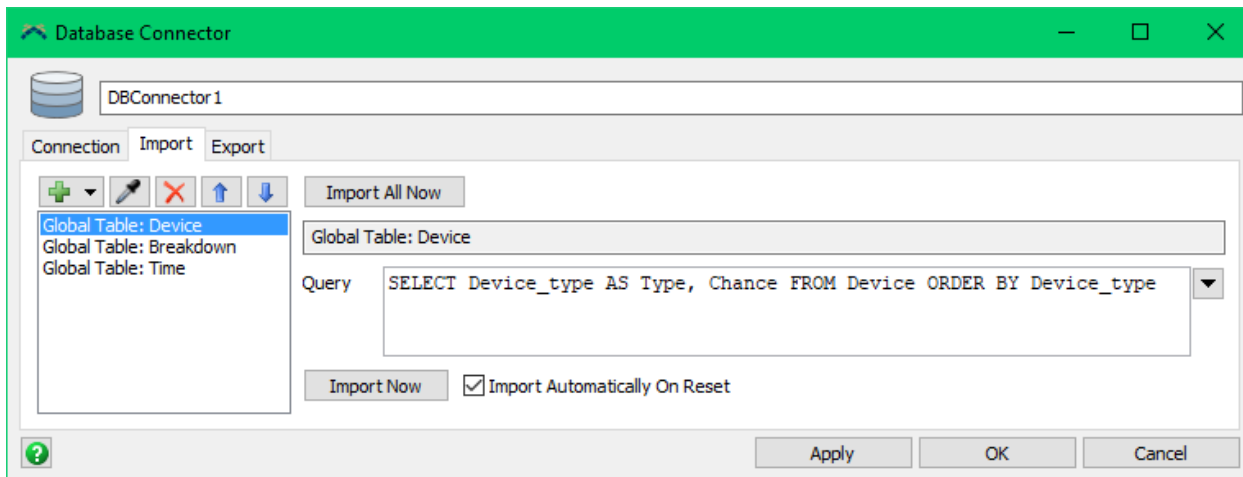


Рисунок 3.1.6 – Імпорт даних у таблицю Device

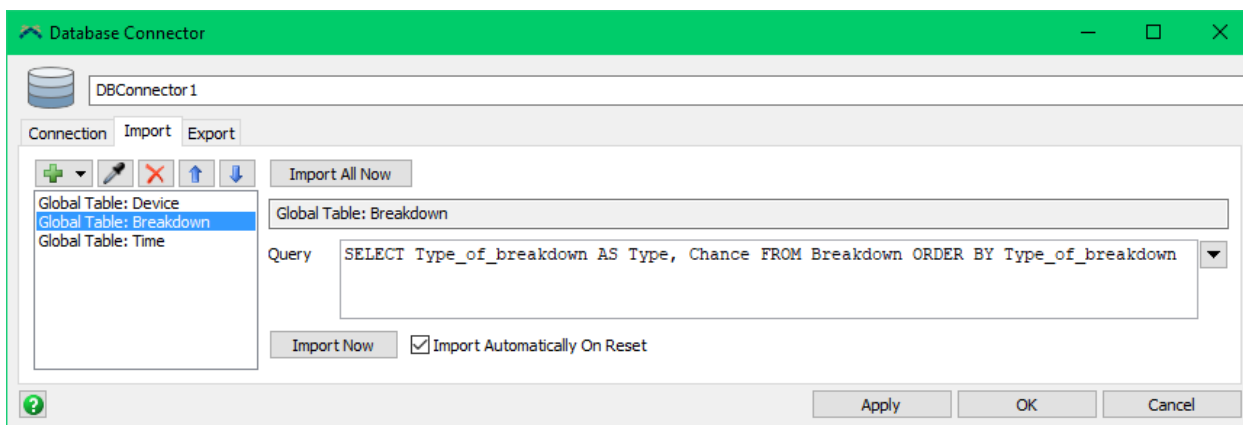


Рисунок 3.1.7 – Імпорт даних у таблицю Breakdown

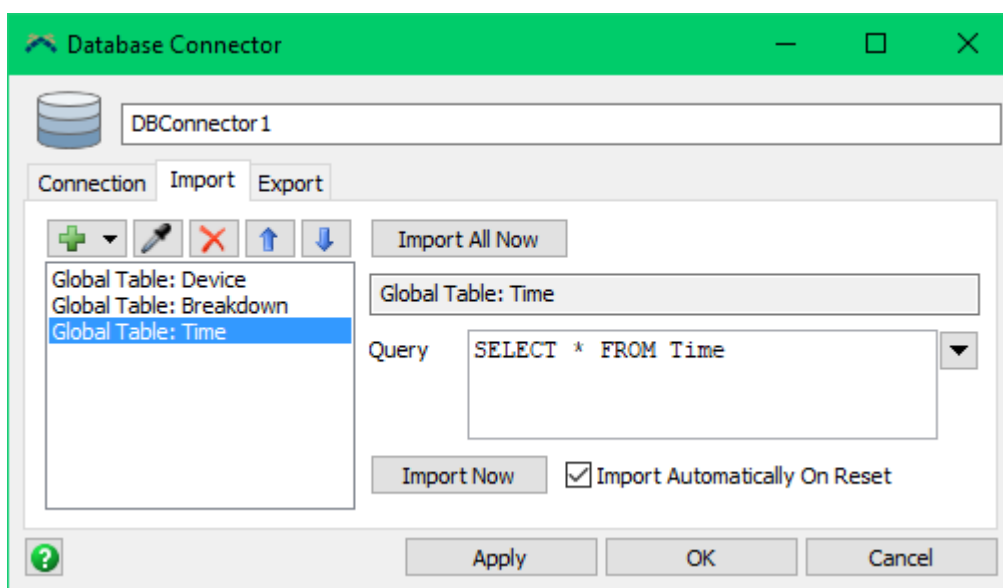


Рисунок 3.1.8 – Імпорт даних у таблицю Time

3.2. Розробка тривимірних моделей

Створюємо 3D моделі для кожного типу засобів зв'язку у програмному середовищі Blender (рис. 3.2.1-3.2.4).



Рисунок 3.2.1 – 3D модель першого типу засобів зв'язку



Рисунок 3.2.2 – 3D модель другого типу засобів зв'язку



Рисунок 3.2.3 – 3D модель третього типу засобів зв'язку

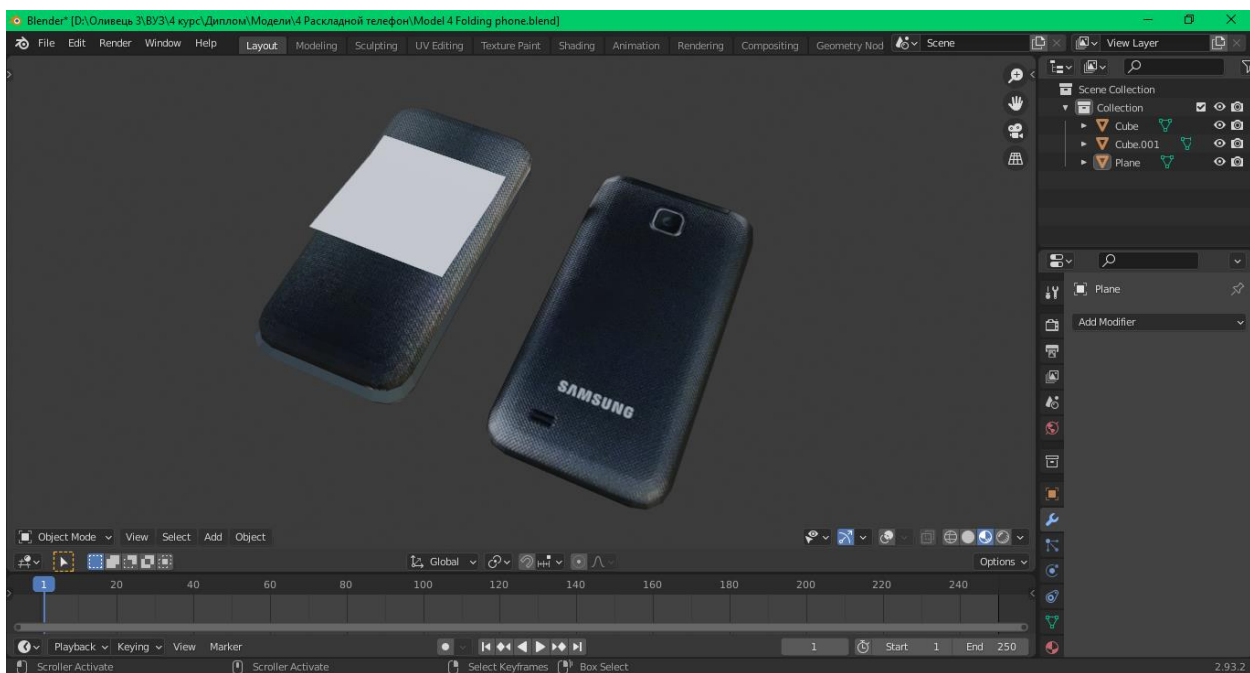


Рисунок 3.2.3 – 3D модель четвертого типу засобів зв'язку

3.3. Програмна реалізація імітаційної моделі

Для створення імітаційної моделі, додаємо об'єкти до робочої області та визначаємо їх параметри. Для джерела заявок створюємо три тригери On Creation,

два з них будуть спиратися на відсоткове співвідношення. Використовуємо таблиці бази даних для призначення міток та їх значень (рис. 3.3.1).

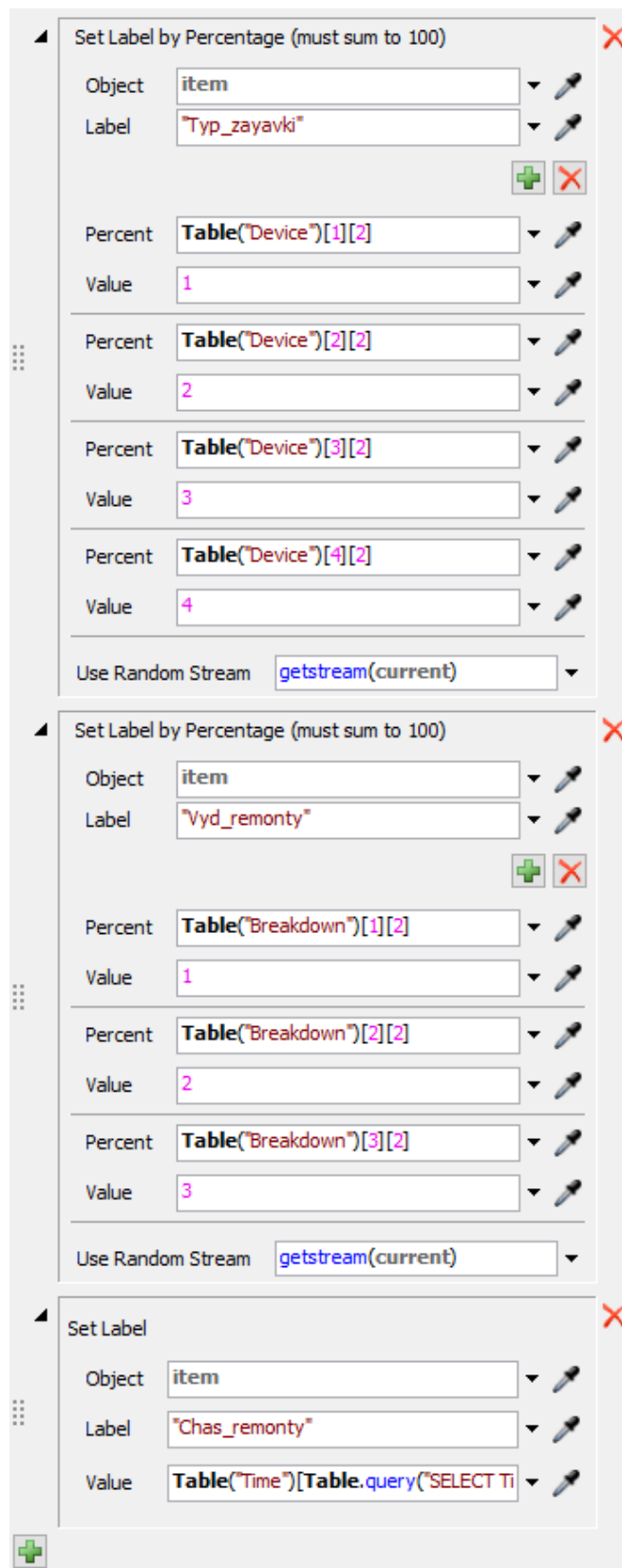


Рисунок 3.3.1 – Призначення мітки типу засобу зв'язку, виду поломки та часу виконання заявки

Для візуального відображення призначених міток редагуємо код тригерів. На основі мітки типу засобу зв'язку призначається модель об'єкта у форматі .fbx (рис 3.3.2). Аналогічно визначається колір стікера на основі значення мітки виду поломки (рис. 3.3.3).

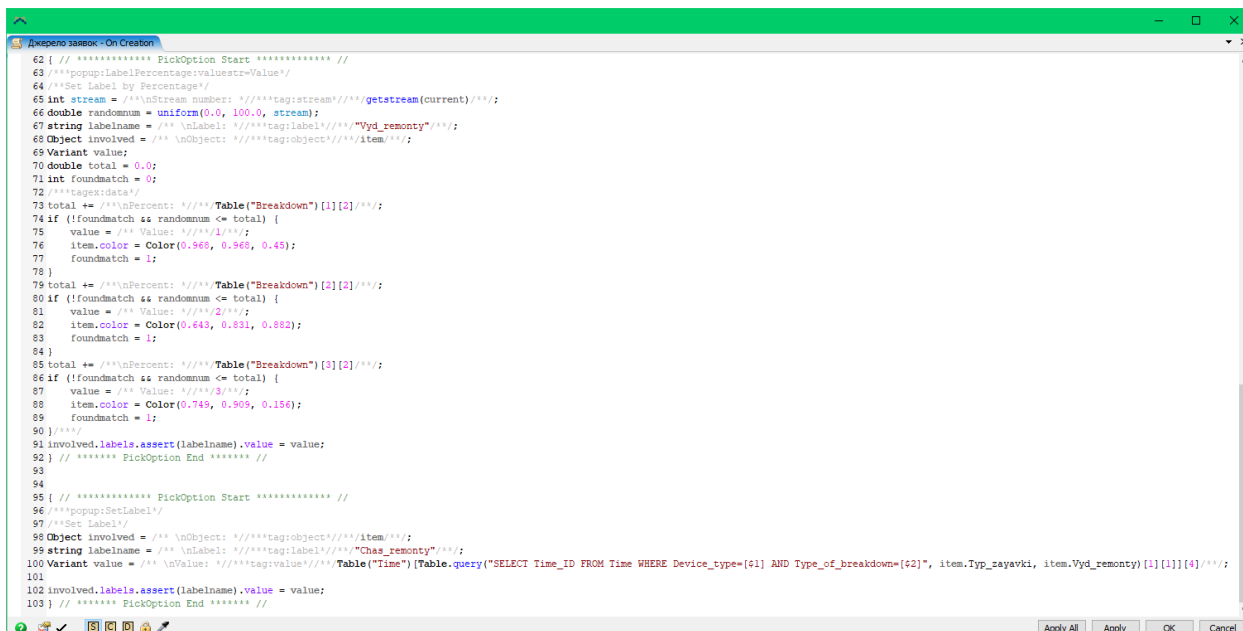


```

1 Object current = ownerobject(c);
2 Object item = param(1);
3 int rownumber = param(2); //row number of the schedule/sequence table
4
5
6 { // ***** PickOption Start ***** //
7 /***popup:LabelPercentage:valuestr=Value*/
8 /**Set Label by Percentage*/
9 int stream = /***\nStream number: /***/tag:stream/***/getstream(current)/***/;
10 double randomnum = uniform(0.0, 100.0, stream);
11 string labelname = /** \nLabel: /***/tag:label/***/"Typ_zayavki"/***/;
12 Object involved = /** \nObject: /***/tag:object/***/item/***/;
13 Variant value;
14 double total = 0.0;
15 int foundmatch = 0;
16 /***tagex:data*/
17 total += /***\nPercent: /***/Table("Device")[1][2]/***/;
18 if (!foundmatch && randomnum <= total) {
19     value = /** Value: /***/1/***/;
20     string shapename = "Model 1 (IPhone).fbx";
21     double theindex = getshapeindex(shapename);
22     sets(shape(item), shapename);
23     setobjectshapeindex(item, theindex);
24     applyshapefactors(item);
25     foundmatch = 1;
26 }
27 total += /***\nPercent: /***/Table("Device")[2][2]/***/;
28 if (!foundmatch && randomnum <= total) {
29     value = /** Value: /***/2/***/;
30     string shapename = "Model 2 (Phone).fbx";
31     double theindex = getshapeindex(shapename);
32     sets(shape(item), shapename);
33     setobjectshapeindex(item, theindex);
34     applyshapefactors(item);
35     foundmatch = 1;
36 }
37 total += /***\nPercent: /***/Table("Device")[3][2]/***/;
38 if (!foundmatch && randomnum <= total) {
39     value = /** Value: /***/3/***/;
40     string shapename = "Model 3 (Smartphone).fbx";
41     double theindex = getshapeindex(shapename);
42     sets(shape(item), shapename);
43     setobjectshapeindex(item, theindex);
44     applyshapefactors(item);
45     foundmatch = 1;
46 }
47 total += /***\nPercent: /***/Table("Device")[4][2]/***/;
48 if (!foundmatch && randomnum <= total) {
49     value = /** Value: /***/4/***/;
50     string shapename = "Model 4 (Folding phone).fbx";
51     double theindex = getshapeindex(shapename);
52     sets(shape(item), shapename);
53     setobjectshapeindex(item, theindex);
54     applyshapefactors(item);
55     foundmatch = 1;
56 }/***/
57 involved.labels.assert(labelname).value = value;
58 // ***** PickOption End ***** //

```

Рисунок 3.3.2 – Призначення мітки типу засобу зв'язку та відповідної моделі об'єкта



```

62 { // ***** PickOption Start ***** //
63 /**popup:LabelPercentage:valuestr=Value*/
64 /**Set Label by Percentage*/
65 int stream = /**\nStream number: /**tag:stream/**/getstream(current)/**/;
66 double randomnum = uniform(0.0, 100.0, stream);
67 string labelname = /** \nLabel: /**tag:label/**/"Vyd_remonty"/**/;
68 Object involved = /** \nObject: /**tag:object/**/item/**/;
69 Variant value;
70 double total = 0.0;
71 int foundmatch = 0;
72 /**tag:data*/
73 total += /**\nPercent: /**/Table("Breakdown")[1][2]/**/;
74 if (!foundmatch && randomnum <= total) {
75 value = /** Value: /**/1/**/;
76 item.color = Color(0.968, 0.968, 0.45);
77 foundmatch = 1;
78 }
79 total += /**\nPercent: /**/Table("Breakdown")[2][2]/**/;
80 if (!foundmatch && randomnum <= total) {
81 value = /** Value: /**/2/**/;
82 item.color = Color(0.643, 0.831, 0.882);
83 foundmatch = 1;
84 }
85 total += /**\nPercent: /**/Table("Breakdown")[3][2]/**/;
86 if (!foundmatch && randomnum <= total) {
87 value = /** Value: /**/3/**/;
88 item.color = Color(0.749, 0.909, 0.156);
89 foundmatch = 1;
90 }/**/
91 involved.labels.assert(labelname).value = value;
92 // ***** PickOption End ***** //
93
94
95 { // ***** PickOption Start ***** //
96 /**popup:SetLabel*/
97 /**Set Label*/
98 Object involved = /** \nObject: /**tag:object/**/item/**/;
99 string labelname = /** \nLabel: /**tag:label/**/"Chas_remonty"/**/;
100 Variant value = /** \nValue: /**tag:value/**/Table("Time")|Table.query("SELECT Time_ID FROM Time WHERE Device_type={@1} AND Type_of_breakdown={@2}", item.Type_sayavki, item.Vyd_remonty)[1][1][4]/**/;
101
102 involved.labels.assert(labelname).value = value;
103 // ***** PickOption End ***** //

```

Рисунок 3.3.3 – Призначення мітки часу виконання заявки, виду поломки та відповідного кольору об'єкта

Для зберігання заявок що надійшли використовуємо стійку на налаштовуємо параметри зберігання. Нижні комірки помічаємо як слоти не призначені для зберігання, для інших виставляємо рівень незаповненості слотів (рис. 3.3.4). Таким чином заявки не будуть зберігатися на підлозі та одна на одній (рис. 3.3.5).

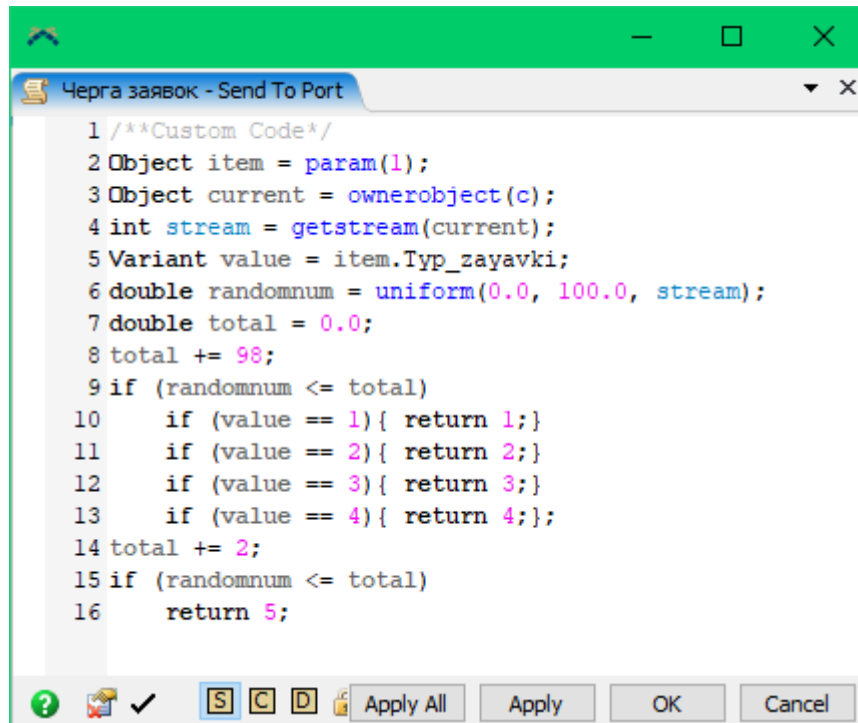


Рисунок 3.3.4 – Правила зберігання заявок у стойці



Рисунок 3.3.5 – Розміщення заявок у стойці

Для розподілу заявок редагуємо код виходу об'єктів з черги заявок (рис. 3.3.6). Таким чином 2 відсотки усіх заявок буде відхилятися, а інші 98 відсотків розподіляться по чергам майстрів відповідно типу засобу зв'язку.



```

1 /**Custom Code*/
2 Object item = param(1);
3 Object current = ownerobject(c);
4 int stream = getstream(current);
5 Variant value = item.Typ_zayavki;
6 double randomnum = uniform(0.0, 100.0, stream);
7 double total = 0.0;
8 total += 98;
9 if (randomnum <= total)
10     if (value == 1){ return 1;}
11     if (value == 2){ return 2;}
12     if (value == 3){ return 3;}
13     if (value == 4){ return 4;};
14 total += 2;
15 if (randomnum <= total)
16     return 5;

```

Рисунок 3.3.6 – Розподіл заявок відповідно типу засобу зв'язку

Розподіл заявок виконується 2 диспетчерами, після чого заявки очікують своєї черги на виконання. Майстри першої групи виконують заявки першого типу. У випадку коли вони відсутні, а майстри другої, третьої та четвертої групи зайняті, вони виконують заявки інших типів. Заявка виконується майстром протягом часу визначеного міткою час виконання заявки. (рис. 3.3.7). Після виконання заявка потрапляє в об'єкт виконанні заявки.



Рисунок 3.3.7 – Процес виконання заявки

Загальний вигляд системи можна побачити на рисунку 3.3.8.

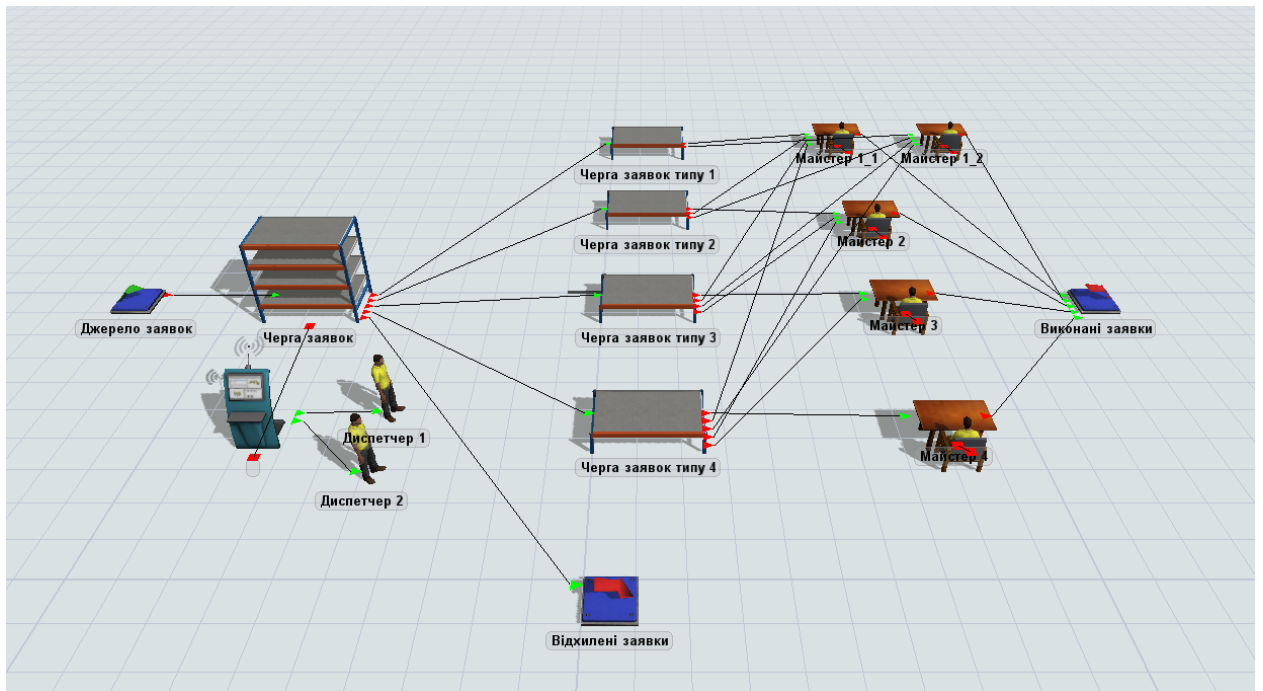


Рисунок 3.3.8 – Загальний вигляд системи

Для відображення статистика створимо інформаційну панель. На рисунку 3.3.9 показано загальну кількість надійшовших та виконаних заявок, розбиття надійшовших та виконаних заявок по типах засобів зв'язку, кількість виконаних заявок за типом засобу зв'язку та видом ремонту, Імовірність виконання заявок за типами та імовірність виконання заявок всіх типів.

<p>Загальна кількість надійшовших заявок</p> <table border="1"> <thead> <tr> <th></th> <th>Кількість</th> </tr> </thead> <tbody> <tr> <td>Джерело заявок</td> <td>143</td> </tr> </tbody> </table>		Кількість	Джерело заявок	143	<p>Загальна кількість виконаних заявок</p> <table border="1"> <thead> <tr> <th></th> <th>Кількість</th> </tr> </thead> <tbody> <tr> <td>Виконані заявки</td> <td>141</td> </tr> </tbody> </table>		Кількість	Виконані заявки	141																															
	Кількість																																							
Джерело заявок	143																																							
	Кількість																																							
Виконані заявки	141																																							
<p>Надійшовші заявки по типам</p> <table border="1"> <thead> <tr> <th>Тип заявки</th> <th>Кількість</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>19</td> </tr> <tr> <td>2</td> <td>50</td> </tr> <tr> <td>3</td> <td>34</td> </tr> <tr> <td>4</td> <td>40</td> </tr> </tbody> </table>	Тип заявки	Кількість	1	19	2	50	3	34	4	40	<p>Виконані заявки по типам</p> <table border="1"> <thead> <tr> <th>Тип заявки</th> <th>Кількість</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>19</td> </tr> <tr> <td>2</td> <td>49</td> </tr> <tr> <td>3</td> <td>34</td> </tr> <tr> <td>4</td> <td>39</td> </tr> </tbody> </table>	Тип заявки	Кількість	1	19	2	49	3	34	4	39																			
Тип заявки	Кількість																																							
1	19																																							
2	50																																							
3	34																																							
4	40																																							
Тип заявки	Кількість																																							
1	19																																							
2	49																																							
3	34																																							
4	39																																							
<p>Кількість виконаних заявок за типом засобу зв'язку та видом ремонту</p> <table border="1"> <thead> <tr> <th>Тип_zayavki</th> <th>Vyd_remonty</th> <th>count</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>7</td> </tr> <tr> <td>1</td> <td>2</td> <td>5</td> </tr> <tr> <td>1</td> <td>3</td> <td>7</td> </tr> <tr> <td>2</td> <td>1</td> <td>25</td> </tr> <tr> <td>2</td> <td>2</td> <td>11</td> </tr> <tr> <td>2</td> <td>3</td> <td>13</td> </tr> <tr> <td>3</td> <td>1</td> <td>20</td> </tr> <tr> <td>3</td> <td>2</td> <td>6</td> </tr> <tr> <td>3</td> <td>3</td> <td>8</td> </tr> <tr> <td>4</td> <td>1</td> <td>21</td> </tr> <tr> <td>4</td> <td>2</td> <td>4</td> </tr> <tr> <td>4</td> <td>3</td> <td>14</td> </tr> </tbody> </table>		Тип_zayavki	Vyd_remonty	count	1	1	7	1	2	5	1	3	7	2	1	25	2	2	11	2	3	13	3	1	20	3	2	6	3	3	8	4	1	21	4	2	4	4	3	14
Тип_zayavki	Vyd_remonty	count																																						
1	1	7																																						
1	2	5																																						
1	3	7																																						
2	1	25																																						
2	2	11																																						
2	3	13																																						
3	1	20																																						
3	2	6																																						
3	3	8																																						
4	1	21																																						
4	2	4																																						
4	3	14																																						
<p>Імовірність виконання заявок по типам</p> <table border="1"> <thead> <tr> <th>Тип_zayavki</th> <th>Protsent_vykonannya</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>100.00</td> </tr> <tr> <td>2</td> <td>98.00</td> </tr> <tr> <td>3</td> <td>100.00</td> </tr> <tr> <td>4</td> <td>97.50</td> </tr> </tbody> </table>		Тип_zayavki	Protsent_vykonannya	1	100.00	2	98.00	3	100.00	4	97.50																													
Тип_zayavki	Protsent_vykonannya																																							
1	100.00																																							
2	98.00																																							
3	100.00																																							
4	97.50																																							
<p>Імовірність виконання заявок всіх типів</p> <table border="1"> <thead> <tr> <th>Protsent_vykonannya</th> </tr> </thead> <tbody> <tr> <td>98.60</td> </tr> </tbody> </table>		Protsent_vykonannya	98.60																																					
Protsent_vykonannya																																								
98.60																																								

Рисунок 3.3.9 – Статистичні дані надійшовших і виконаних заявок та ймовірність їх виконання.

3.4. Дослідження роботи імітаційної моделі мережі передачі даних

Дослідимо залежність кількості виконаних заявок та ймовірності виконання заявок всіх типів від інтервалу надходження їх в ремонт та ймовірностей надходження кожного з типу заявок. Час симуляції визначаємо з 8:00 до 17:00.

Для першого дослідження використаємо вхідні дані, визначені постановкою задачі: інтервал надходження заявок в ремонт $\text{exponential}(30)$, ймовірність надходження заявок першого типу 20%, другого – 30%, третього – 25%, четвертого – 25%. За результатами виконання симуляції, зображеними на рисунку 3.4.1, бачимо що кількість виконаних заявок складає 1072, а ймовірність виконання заявок всіх типів – 97.99%.

Загальна кількість надійшовших заявок		Загальна кількість виконаних заявок	
	Кількість		Кількість
Джерело заявок	1094	Виконані заявки	1072
Надійшовші заявки по типам		Виконані заявки по типам	
Тип заявки	Кількість	Тип заявки	Кількість
1	229	1	220
2	345	2	339
3	245	3	243
4	275	4	270
Кількість виконаних заявок за типом засобу зв'язку та видом ремонту			
Тип_zayavki	Vyd_remonty	count	
1	1	119	
1	2	46	
1	3	55	
2	1	178	
2	2	81	
2	3	80	
3	1	116	
3	2	62	
3	3	65	
4	1	138	
4	2	60	
4	3	72	
Ймовірність виконання заявок по типам			
Тип_zayavki	Protsent_vykonannya		
1	96.07		
2	98.26		
3	99.18		
4	98.18		
Ймовірність виконання заявок всіх типів			
Protsent_vykonannya			97.99

Рисунок 3.4.1 – Перше дослідження

Для другого дослідження змінимо інтервал надходження заявок в ремонт на $\text{exponential}(20)$. За результатами виконання симуляції, зображеними на рисунку 3.4.2, бачимо що кількість виконаних заявок збільшилася до 1631, а ймовірність виконання заявок всіх типів збільшилася до 98.02%.

Загальна кількість надійшовших заявок		Загальна кількість виконаних заявок	
	Кількість		Кількість
Джерело заявок	1664	Виконані заявки	1631
Надійшовші заявки по типам		Виконані заявки по типам	
Тип заявки	Кількість	Тип заявки	Кількість
1	336	1	328
2	508	2	498
3	401	3	393
4	419	4	412
Кількість виконаних заявок за типом засобу зв'язку та видом ремонту			
Тип_zayavki	Vyd_remonty	count	
1	1	182	
1	2	75	
1	3	71	
2	1	256	
2	2	114	
2	3	128	
3	1	186	
3	2	96	
3	3	111	
4	1	213	
4	2	90	
4	3	109	
Ймовірність виконання заявок по типам			
Тип_zayavki	Protsent_vykonannya		
1	97.62		
2	98.03		
3	98.00		
4	98.33		
Ймовірність виконання заявок всіх типів			
Protsent_vykonannya			
			98.02

Рисунок 3.4.2 – Друге дослідження

Для третього дослідження змінимо інтервал надходження заявок в ремонт на $\text{exponential}(20)$, ймовірність надходження заявок другого типу на 25%, а

третього – на 30%. За результатами виконання симуляції, зображеними на рисунку 3.4.3, бачимо що кількість виконаних заявок збільшилася до 1642, а ймовірність виконання заявок всіх типів збільшилася до 98.03%.

Загальна кількість надійшовших заявок		Загальна кількість виконаних заявок	
	Кількість		Кількість
Джерело заявок	1675	Виконані заявки	1642
Надійшовші заявки по типам		Виконані заявки по типам	
Тип заявки	Кількість	Тип заявки	Кількість
1	350	1	341
2	433	2	422
3	473	3	467
4	419	4	412
Кількість виконаних заявок за типом засобу зв'язку та видом ремонту			
Тип_zayavki	Vyd_remonty	count	
1	1	194	
1	2	75	
1	3	72	
2	1	210	
2	2	106	
2	3	106	
3	1	235	
3	2	116	
3	3	116	
4	1	216	
4	2	85	
4	3	111	
Ймовірність виконання заявок по типам			
Тип_zayavki	Protsent_vykonannya		
1	97.43		
2	97.46		
3	98.73		
4	98.33		
Ймовірність виконання заявок всіх типів			
Protsent_vykonannya			
			98.03

Рисунок 3.4.3 – Трете дослідження

Для четвертого дослідження змінимо інтервал надходження заявок в ремонт на $\text{exponential}(20)$, ймовірність надходження заявок першого типу на 30%,

другого – на 25%, третього – на 30%, а четвертого – на 15%. За результатами виконання симуляції, зображеними на рисунку 3.4.4, бачимо що кількість виконаних заявок збільшилася до 1612, а ймовірність виконання заявок всіх типів збільшилася до 98.05%.

Загальна кількість надійшовших заявок		Загальна кількість виконаних заявок	
	Кількість		Кількість
Джерело заявок	1644	Виконані заявки	1612
Надійшовші заявки по типам		Виконані заявки по типам	
Тип заявки	Кількість	Тип заявки	Кількість
1	509	1	498
2	389	2	383
3	501	3	488
4	245	4	243
Кількість виконаних заявок за типом засобу зв'язку та видом ремонту			
Тип_zayavki	Vyd_remonty	count	
1	1	254	
1	2	126	
1	3	118	
2	1	209	
2	2	99	
2	3	75	
3	1	245	
3	2	112	
3	3	131	
4	1	129	
4	2	51	
4	3	63	
Ймовірність виконання заявок по типам			
Тип_zayavki	Protsent_vykonannya		
1	97.84		
2	98.46		
3	97.41		
4	99.18		
Ймовірність виконання заявок всіх типів			
Protsent_vykonannya			
			98.05

Рисунок 3.4.4 – Четверте дослідження

Для четвертого дослідження змінимо інтервал надходження заявок в ремонт на $\text{exponential}(20)$, ймовірність надходження заявок першого типу на 33%,

другого – на 23%, третього – на 27%, а четвертого – на 17%. За результатами виконання симуляції, зображеними на рисунку 3.4.5, бачимо що кількість виконаних заявок збільшилася до 1604, а ймовірність виконання заявок всіх типів збільшилася до 98.22%. Результати дослідження представлені у таблиці 3.4.1.

<p>Загальна кількість надійшовших заявок</p> <table border="1"> <thead> <tr> <th></th> <th>Кількість</th> </tr> </thead> <tbody> <tr> <td>Джерело заявок</td> <td>1633</td> </tr> </tbody> </table>		Кількість	Джерело заявок	1633	<p>Загальна кількість виконаних заявок</p> <table border="1"> <thead> <tr> <th></th> <th>Кількість</th> </tr> </thead> <tbody> <tr> <td>Виконані заявки</td> <td>1604</td> </tr> </tbody> </table>		Кількість	Виконані заявки	1604																															
	Кількість																																							
Джерело заявок	1633																																							
	Кількість																																							
Виконані заявки	1604																																							
<p>Надійшовші заявки по типам</p> <table border="1"> <thead> <tr> <th>Тип заявки</th> <th>Кількість</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>544</td> </tr> <tr> <td>2</td> <td>355</td> </tr> <tr> <td>3</td> <td>454</td> </tr> <tr> <td>4</td> <td>280</td> </tr> </tbody> </table>	Тип заявки	Кількість	1	544	2	355	3	454	4	280	<p>Виконані заявки по типам</p> <table border="1"> <thead> <tr> <th>Тип заявки</th> <th>Кількість</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>533</td> </tr> <tr> <td>2</td> <td>350</td> </tr> <tr> <td>3</td> <td>444</td> </tr> <tr> <td>4</td> <td>277</td> </tr> </tbody> </table>	Тип заявки	Кількість	1	533	2	350	3	444	4	277																			
Тип заявки	Кількість																																							
1	544																																							
2	355																																							
3	454																																							
4	280																																							
Тип заявки	Кількість																																							
1	533																																							
2	350																																							
3	444																																							
4	277																																							
<p>Кількість виконаних заявок за типом засобу зв'язку та видом ремонту</p> <table border="1"> <thead> <tr> <th>Тип_zayavki</th> <th>Vyd_remonty</th> <th>count</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>270</td></tr> <tr><td>1</td><td>2</td><td>134</td></tr> <tr><td>1</td><td>3</td><td>129</td></tr> <tr><td>2</td><td>1</td><td>194</td></tr> <tr><td>2</td><td>2</td><td>88</td></tr> <tr><td>2</td><td>3</td><td>68</td></tr> <tr><td>3</td><td>1</td><td>223</td></tr> <tr><td>3</td><td>2</td><td>99</td></tr> <tr><td>3</td><td>3</td><td>122</td></tr> <tr><td>4</td><td>1</td><td>133</td></tr> <tr><td>4</td><td>2</td><td>63</td></tr> <tr><td>4</td><td>3</td><td>81</td></tr> </tbody> </table>		Тип_zayavki	Vyd_remonty	count	1	1	270	1	2	134	1	3	129	2	1	194	2	2	88	2	3	68	3	1	223	3	2	99	3	3	122	4	1	133	4	2	63	4	3	81
Тип_zayavki	Vyd_remonty	count																																						
1	1	270																																						
1	2	134																																						
1	3	129																																						
2	1	194																																						
2	2	88																																						
2	3	68																																						
3	1	223																																						
3	2	99																																						
3	3	122																																						
4	1	133																																						
4	2	63																																						
4	3	81																																						
<p>Імовірність виконання заявок по типам</p> <table border="1"> <thead> <tr> <th>Тип_zayavki</th> <th>Protsent_vykonannya</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>97.98</td> </tr> <tr> <td>2</td> <td>98.59</td> </tr> <tr> <td>3</td> <td>97.80</td> </tr> <tr> <td>4</td> <td>98.93</td> </tr> </tbody> </table>		Тип_zayavki	Protsent_vykonannya	1	97.98	2	98.59	3	97.80	4	98.93																													
Тип_zayavki	Protsent_vykonannya																																							
1	97.98																																							
2	98.59																																							
3	97.80																																							
4	98.93																																							
<p>Імовірність виконання заявок всіх типів</p> <table border="1"> <thead> <tr> <th>Protsent_vykonannya</th> </tr> </thead> <tbody> <tr> <td>98.22</td> </tr> </tbody> </table>		Protsent_vykonannya	98.22																																					
Protsent_vykonannya																																								
98.22																																								

Рисунок 3.4.5 – П'яте дослідження

	Перше дослідження	Друге дослідження	Третє дослідження	Четверте дослідження	П'яте дослідження
Інтервал надходження заявок	exponential(30)	exponential(20)	exponential(20)	exponential(20)	exponential(20)
Ймовірність надходження заявок 1-го типу	20%	20%	20%	30%	33%
Ймовірність надходження заявок 2-го типу	30%	30%	25%	25%	23%
Ймовірність надходження заявок 3-го типу	25%	25%	30%	30%	27%
Ймовірність надходження заявок 4-го типу	25%	25%	25%	15%	17%
Кількість виконаних заявок	1072	1631	1642	1612	1604
Ймовірність виконання заявок всіх типів	97.99%	98.02%	98.03%	98.05%	98.22%

Таблиця 3.4.1. – результати роботи мережі з різними вхідними даними

Для підвищення ефективності роботи системи, необхідно змінити інтервал надходження заявок на $\text{exponential}(20)$, ймовірність надходження заявок першого типу на 33%, другого – на 23%, третього – на 27%, а четвертого – на 17%. Подібні зміни дозволять збільшити кількість виконаних заявок на 532, а ймовірність виконання всіх типів заявок на 0.23%.

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи були сформовані мета та поставлена задача дослідження, проведено аналіз предметної сфери та виконано аналіз останніх досліджень у напрямку імітаційного моделювання, обрано метод дослідження робочих процесів у ремонтному підрозділі та обрано програмне забезпечення для реалізації імітаційної моделі. Також було розроблено технічне завдання та виконано планування робіт.

Для вибору метода дослідження та програмного забезпечення було складено порівняльні таблиці. В якості метода дослідження було обрано імітаційне моделювання, оскільки він потребує найменших витрат часу та коштів, при цьому враховуючи випадкові величини та дає можливість створити динамічну модель для проведення експериментів з графічним відображеннями. В якості програмного забезпечення для побудови імітаційної моделі було обрано FlexSim, оскільки він дозволяє відображати стан системи та статистичні дані у реальному часі, має можливість підключення до баз даних, розширення функціоналу за допомогою мов програмування, створення інтерфейсу користувача та звітів.

У технічному завданні визначено вимоги до системи, описана структура виробничої системи, склад і зміст робіт із розробки імітаційної моделі.

У плануванні робіт було деталізовано мету методом SMART, проведено планування змісту структури робіт та планування структури організації для впровадження готового проекту, складено календарний план та проведено аналіз ризиків проекту.

Під час проектування імітаційної моделі було створено функціональну модель у нотації IDEF0 та виконано першу декомпозицію, побудовано діаграму потоків (OFD) та розроблено концептуальну, логічну та фізичну моделі бази даних.

Для збереження та зміни вхідних даних за допомогою веб-додатку PhpMyAdmin та СУБД MySQL було створено та підключено до Flexsim базу даних. Також було створено 3D моделі об'єктів системи у програмному забезпеченні Blender та імпортовано їх до імітаційної моделі.

Було проведено дослідження залежності кількості виконаних заявок та ймовірності виконання заявок всіх типів від інтервалу надходження їх у ремонт та ймовірностей надходження кожного з типів заявок. На основі чого було надано практичні рекомендації щодо покращення ефективності системи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Возможности – инструмент имитационного моделирования Anylogic : вебсайт. URL: <https://www.anylogic.ru/features/> (дата звернення: 05.05.2022)
2. Flexsim 3D Simulation Modeling Software : вебсайт. URL: <https://www.flexsim.com/flexsim/> (Last accessed: 05.05.2022)
3. Arena Simulation Software : вебсайт. URL: <https://www.rockwellautomation.com/en-us/products/software/arena-simulation.html> (Last accessed: 06.05.2022)
4. Online Simulation Software Simul8 : вебсайт. URL: <https://www.simul8.com/software/> (Last accessed: 06.05.2022)
5. Стеценко І.В. Моделювання систем: навч. посіб. Черкаси : Черкаський державний технологічний університет, 2010. 399 с. URL: http://web.kpi.kharkov.ua/auts/wp-content/uploads/sites/67/2017/02/MOCS_Kachanov_posobie.pdf (дата звернення: 10.05.2022)
6. Великодний С. С. Моделювання систем: конспект лекцій. Одеський державний екологічний університет, 2018. – 186 с. URL: http://eprints.library.odeku.edu.ua/id/eprint/708/1/VelykodniySS_Modelirovanie_system_KL_2018.pdf (дата звернення: 11.05.2022)
7. Lavrov, E., Chybiriak, Y., Siryk, O., Logvinenko, V. and Zakharova, A., 2022. Training of Specialists for Adaptive management. Techniques for Teaching Computer Analysis of Automated Production Systems in the FlexSim Environment, CEUR Workshop Proceedings 2022, pp. 106-118.
8. Lavrov, E., et al. A Model for the Organization of Adaptive Dialogue Interaction 'Man-Computer' Taking into Account the Requirements of Reliability and Efficiency, 2021. SCOPUS, www.scopus.com, doi:10.1109/AICT52120.2021.9628939.
9. Lavrov, E., Siryk, O., Kirichenko, I., Barchenko, N. And Chybiriak, Y., 2021. The Methodology of Managed Functional Networks for Organizing Effective and

Adaptive Human-Machine Dialogue in Automated Systems, CEUR Workshop Proceedings 2021, pp. 428-437.

10. Nahornyi, V. V., Lavrov, E. A., & Chybiriak, Y. I. (2019). FORECASTING INDIVIDUAL RESOURCE OF TECHNICAL SYSTEMS. Radio Electronics, Computer Science, Control, (1). <https://doi.org/10.15588/1607-3274-2019-1-5>

11. Chybiriak , Y., Konoplianchenko , I., & Marchenko A. (2020). Technological patterns and mathematical models of the synthesis of a rational sequence of product assembly. COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION, (39), 110-116. <https://doi.org/10.36910/6775-2524-0560-2020-39-19>

ДОДАТОК А**ТЕХНІЧНЕ ЗАВДАННЯ**

**на розробку імітаційної моделі
«Імітаційна модель діяльності ремонтного підрозділу засобів
зв'язку»**

ПОГОДЖЕНО:

Доцент кафедри комп'ютерних наук

_____ Чибіряк Я.І.

Студент групи ІТ-81

_____ Нестеренко М.В.

Суми 2022

1. Призначення й мета створення імітаційної моделі

1.1 Призначення імітаційної моделі

Імітаційна модель призначена для відображення функціонування ремонтного підрозділу та надання рекомендацій для покращення роботи на основі дослідження показників її ефективності.

1.2 Мета створення імітаційної моделі

Імітаційна модель створена для дослідження залежності кількості виконаних заявок та ймовірності виконання заявок всіх типів від інтервалу надходження їх у ремонт та ймовірностей надходження кожного з типів заявок.

1.3 Цільова аудиторія

До цільової аудиторії можна віднести представників ремонтних підрозділів, зацікавлених у підвищенні ефективності роботи відділів при обмеженій кількості працівників.

2 Вимоги до імітаційної моделі

2.1 Вимоги до імітаційної моделі в цілому

2.1.1 Вимоги до структури й функціонування імітаційної моделі

Розробити імітаційну модель, яка відображає роботу ремонтного підрозділу. Під час роботи моделі повинні відображатися статистичні дані: кількість заявок які надійшли всіх та кожного типу засобу зв'язку, кількість виконаних заявок всіх та кожного типу засобу зв'язку, кількість виконаних заявок з всіх та кожного типу ремонту, на основі цих даних розраховується імовірність виконання заявок всіх та кожного типу.

2.1.2 Системні вимоги

Для виконання проекту необхідно мати програмне забезпечення FlexSim 2020 та операційна система Windows 7 або вище.

2.1.3 Вимоги до збереження інформації

Вхідні дані, необхідні для роботи моделі повинні зберігатися у базі даних реалізованій засобами системи управління базами даних MySQL.

2.2 Опис ремонтного підрозділу

2.2.1 Структура ремонтного підрозділу

Структура ремонтного підрозділу зображена на рис. А.1.

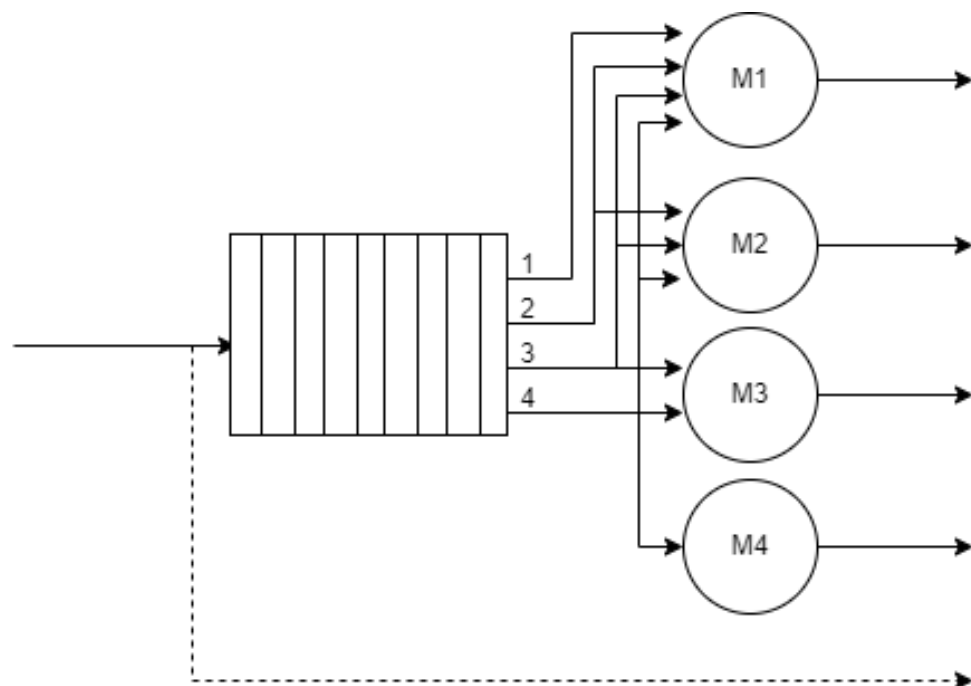


Рисунок А.1 – Схема ремонтного підрозділу

2.2.2 Функціональний опис ремонтного підрозділу

До ремонтного підрозділу засобів зв'язку надходять несправні засоби зв'язку n типів з ймовірностями p_1, p_2, \dots, p_n відповідно. Інтервали часу T_n між двома черговими надходженнями одного типу заявок є випадковими. Кожний засіб зв'язку може вимагати одного з a_1, a_2, \dots, a_k видів ремонту з ймовірностями pa_1, pa_2, \dots, pa_k відповідно.

У фірмі є n_1, n_2, \dots, n_n майстрів для виконання заявок кожного типу відповідно. Майстри n_1 виконують заявки першого типу. Якщо їх немає і фахівці n_2, \dots, n_n груп зайняті, вони виконують заявки цих типів. При цьому заявки першого типу, що надходять, очікують на їх звільнення. Майстри n_2 виконують заявки другого типу. Якщо їх немає і фахівці n_3, n_4, \dots, n_n груп зайняті, вони виконують заявки цих типів. При цьому заявки другого типу, що надходять, очікують їх звільнення. Аналогічні обов'язки і в майстрів інших груп. Тільки майстри n_n виконують заявки одного n типу.

Час виконання заявки n -го типу випадковий, не залежить від майстра, а залежить тільки від виду ремонту: T_{11}, T_{12}, T_{13} – для засобів зв'язку першого типу, T_{21}, T_{22}, T_{23} – для засобів зв'язку другого типу, $\dots, T_{n1}, T_{n2}, \dots, T_{nn}$ – для засобів зв'язку n типу. Значення часу виконання заявки в залежності від типу засобу зв'язку та виду ремонту вказано в табл. А.1.

Таблиця А.1 – Час виконання заявки

Тип засобу зв'язку	Вид ремонту		
	a_1	a_2	a_3
p_1	30	40	50
p_2	20	30	40
p_3	15	25	35
p_4	25	35	45

Прийом та розподіл заявок між групами майстрів здійснюється диспетчерами. Час, що витрачається одним диспетчером однією заявкою, T_1 , випадковий. Диспетчерами не приймаються до ремонту q заявок усіх типів.

2.3 Вимоги до функціонування системи

2.3.1 Потреби користувача

Потреби користувача представлені у таблиці А.2.

Таблиця А.2 – Потреби користувача

ІД	Потреби користувача	Джерело
UN-01	Можливість зміни вхідних даних у моделі	Дослідник
UN-02	Перегляд моделі та процесу імітації	Дослідник
UN-03	Проведення експериментів з моделлю	Дослідник
UN-04	Перегляд статистичних даних моделі	Дослідник
UN-05	Можливість зміни структури моделі (за необхідності)	Дослідник

2.3.2 Функціональні вимоги

На основі потреб користувача були визначені такі функціональні вимоги:

- можливість змінювати вхідні дані;
- керування процесом імітації;
- проведення експериментів;
- відображення статистичних даних;
- можливість змінювати структуру моделі.

2.3.3 Системні вимоги

Даний розділ визначає, розподіляє та вказує на системні вимоги, визначені розробником. Їх перелік наведений в таблиці А.3.

Таблиця А.3 – Системні вимоги

ID	Системні вимоги	Пріоритет	Опис
SR-01	База даних із вхідними даними	М	Зберігає вхідні дані та дає можливість користувачу змінювати вхідні дані
SR-02	Наявність компонента збору статистики	S	Відповідає за збір та виведення статистичних даних
SR-03	Наявність компонента управління процесом імітації	М	Надає можливість запускати, зупиняти модель, масштабувати модельний час по відношенню до реального та ін.
SR-04	Візуалізація у 3D просторі	С	Відповідає за відображення моделі у тривимірному стані як у статичному стані так і в динамічному

Умовні позначення в таблиці А.3:

Must have (M) – вимоги, які повинні бути реалізовані в системі;

Should have (S) – вимоги, які мають бути виконані, але вони можуть почекати своєї черги;

Could have (C) – вимоги, які можуть бути реалізовані, але вони не є центральною ціллю проекту.

2.4 Вимоги до видів забезпечення

2.4.1 Вимоги до інформаційного забезпечення

Реалізація імітаційної моделі та її дослідження відбувається з використанням:

- FlexSim 2020;
- MySQL 8.0;
- Мови програмування C++;
- Blender;

2.4.2 Вимоги до лінгвістичного забезпечення

Імітаційна модель має бути виконана англійською мовою.

2.4.3 Вимоги до програмного забезпечення

Реалізація імітаційної моделі відбувається з використанням:

- FlexSim 2020;
- Windows 10 або вище.

3 Склад і зміст робіт зі створення імітаційної моделі

Докладний опис етапів роботи зі створення імітаційної моделі наведено в таблиці А.4.

Таблиця А.4 – Етапи створення імітаційної моделі

№	Склад і зміст робіт	Строк розробки (у робочих днях)
1	Аналіз предметної області	12 днів
2	Постановка задачі	9 днів
3	Планування робіт	27 днів

4	Проектування імітаційної моделі	6 днів
5	Проектування бази даних	6 днів
6	Створення 3D моделей	6 днів
7	Побудова імітаційної моделі	7 днів
8	Визначення логіки моделі	10 днів
9	Дослідження моделі	8 днів
10	Написання документації	6 днів
	Загальна тривалість робіт	97 днів

4 Вимоги до складу й змісту робіт із введення імітаційної моделі в експлуатацію

Для використання створеної імітаційної моделі користувач повинен мати на своєму комп'ютері встановлену програму FlexSim версії 2020 або вище.

ДОДАТОК Б

Планування робіт

Деталізація мети проекту методом SMART. Продуктом дипломного проекту є імітаційна модель функціонування ремонтного підрозділу засобів зв'язку.

Результати деталізації мети даного проекту представлено в таблиці Б.1.

Таблиця Б.1 – Деталізація мети проекту методом SMART

Specific (конкретна)	Створити імітаційну модель ремонтного підрозділу засобів зв'язку
Measurable (вимірювана)	Імітаційна модель ремонтного підрозділу засобів зв'язку
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 фірми надання ремонтних послуг
Time-framed (обмежена у часі)	Проект має часові обмеження і повинен бути реалізований згідно з календарним планом

Планування змісту структури робіт. Ієрархічна структура робіт (WBS) графічно відображає ієрархію дій необхідних для виконання проекту. Спочатку вказуються основні роботи, після чого виконується декомпозиція до рівня елементарних робіт та вказання строків їх виконання.

WBS діаграма проекту зображена на рис. Б.1.

Планування структури організації, для впровадження готового проекту (OBS). На основі WBS будуємо організаційну структуру проекту (OBS), що відображає розподіл робіт та відносини учасників проекту.

OBS діаграма проекту вказана на рис. Б.2. Список виконавців, що функціонують в проекті знаходиться в табл. Б.2.

Таблиця Б.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

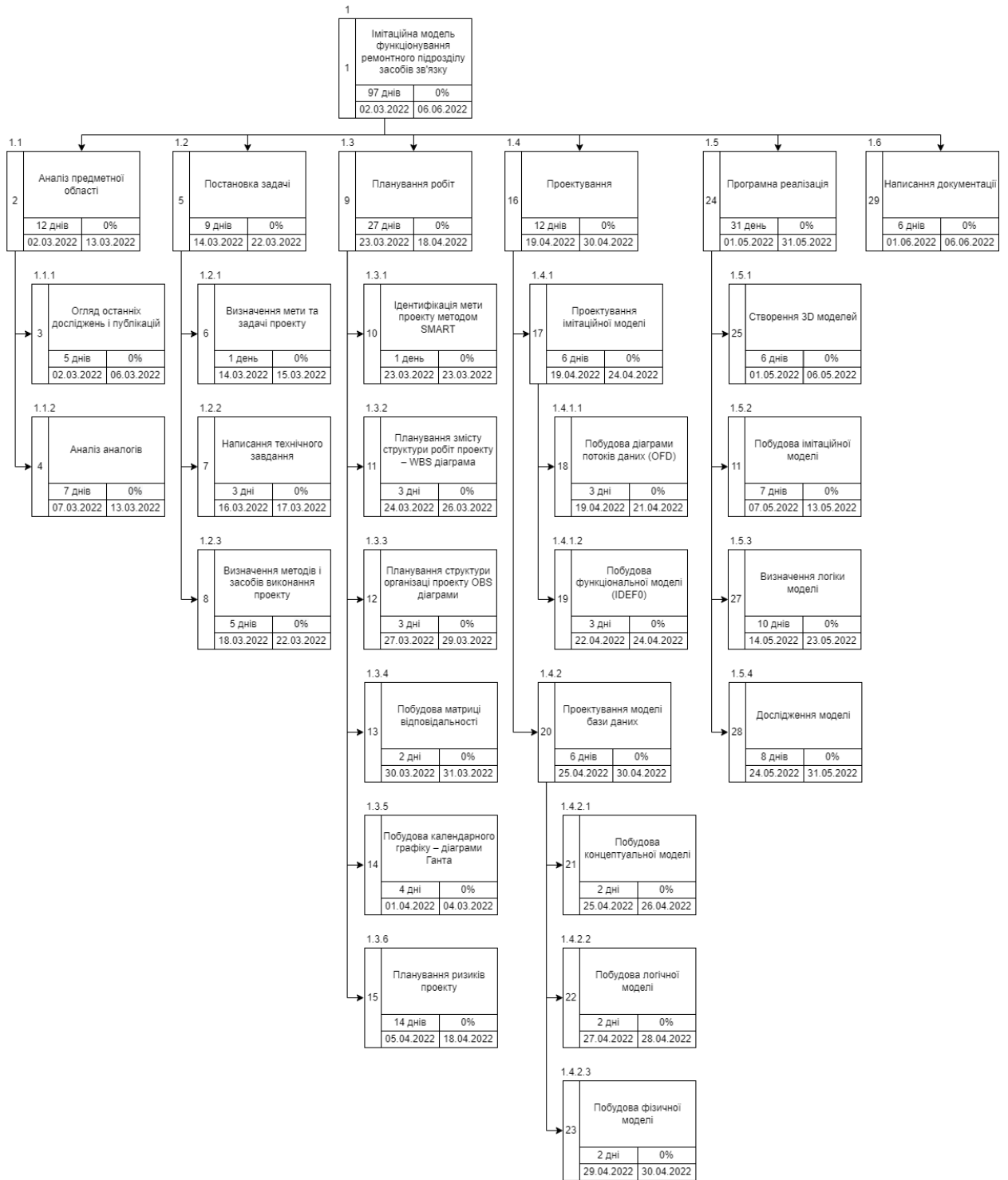


Рисунок Б.1 –WBS структури робіт проекту

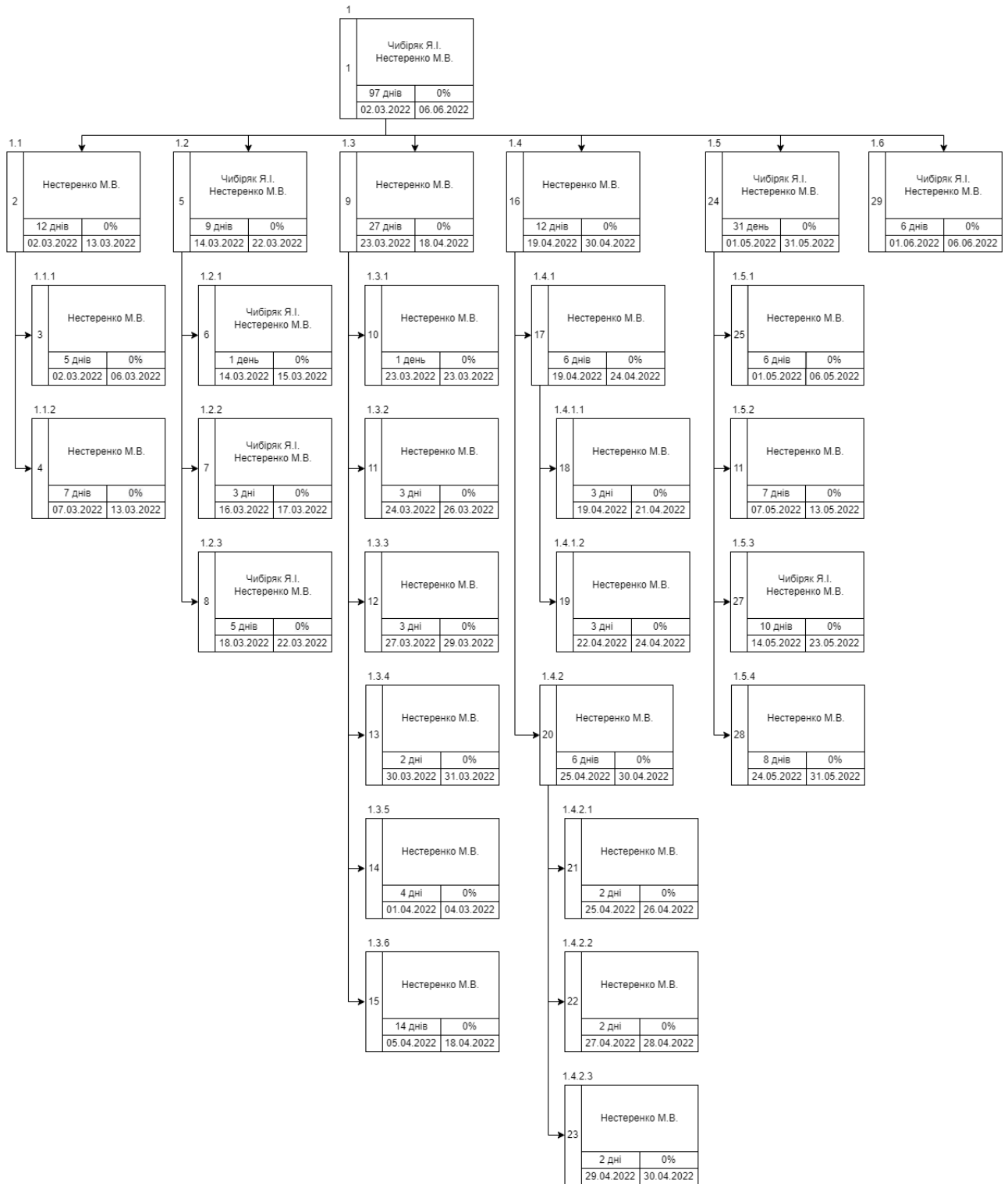


Рисунок Б.2 – Організаційна структура проекту (OBS)

Діаграма Ганта. Для відображення календарного плану використаємо діаграму Ганта, яка заключається у відображенні задач проекту та відповідним проміжкам часу. Відрізки, розташовані відповідно часу виконання задач дають змогу візуалізувати зв'язок між задачами проекту та черговість їх виконання.

Календарний план, побудований за допомогою Microsoft Project 2007 з урахуванням вихідних, святкових днів та обмеженості часових ресурсів, зображено на рис. Б.3.

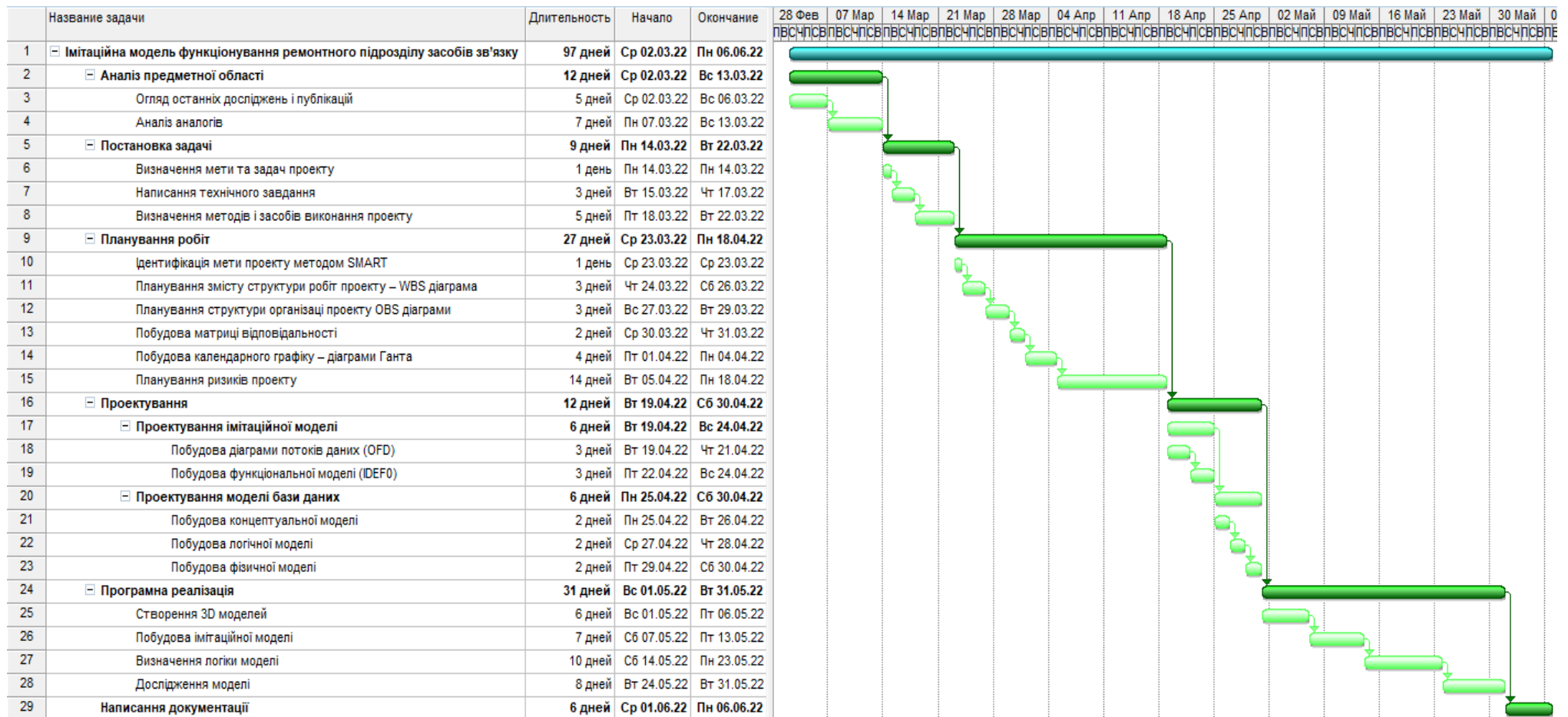


Рисунок Б.3 – Діаграма Ганта

Аналіз ризиків. Виконаємо аналіз ризиків, який включає якісну та кількісну оцінку ризиків. За допомогою якісної оцінки визначимо ризики, причини їх виникнення, рівень ризику та спосіб реагування. Кількісна оцінка заключається у визначенні ймовірності виникнення та ступені впливу ризиків на виконання проекту. Кількісну та якісну оцінку можна застосовувати разом або окремо, залежно від наявного часу. Після оцінки ризиків необхідно розробити план реагування на ризики для виявлення наслідків впливу на проект та зниження негативного впливу.

За допомогою шкали оцінювання показників ризику зображеної в табл. Б.3, визначаємо ймовірності виникнення та вплив ризику на виконання проекту та заносимо у матрицю зображену в табл. Б.4. Опис, ймовірність виникнення, вплив, ранг ризиків та план реагування на нього вказано в табл. Б.6.

Таблиця Б.3 - Шкала оцінювання ймовірності виникнення та впливу ризику на виконання проекту

Оцінка	Ймовірність виникнення	Вплив ризику
1	Дуже мала	Дуже малий
2	Мала	Малий
3	Середня	Середній
4	Велика	Великий
5	Дуже велика	Дуже великий

Таблиця Б.4 – Матриця ймовірності виникнення ризиків та впливу ризику

Ймовірність виникнення	Вплив ризику				
	Дуже малий	Малий	Середній	Великий	Дуже великий
	0,05	0,1	0,2	0,4	0,8
0,9					
0,7				R1	
0,5			R2, R6	R8	R3
0,3				R5, R9	R4, R7
0,1			R10		

- зелений колір – прийнятні ризики;
- жовтий колір – виправданні ризики;
- червоний колір – недопустимі ризики.

На основі отриманого індексу класифікуємо ризики за їх рівнем, вказаним в таблиці Б.5.

Таблиця Б.5 – Шкала оцінювання за рівнем ризику

№	Назва	Межі	Ризики, які входять(номера)
1	Прийнятні	$1 \leq R \leq 2$	10
2	Виправдані	$3 \leq R \leq 4$	2, 5, 6, 9
3	Недопустимі	$6 \leq R \leq 9$	1, 3, 4, 7, 8

Таблиця Б.6 – Оцінка ймовірності виникнення, величини витрат та індексу ризику

ID	Статус ризику	Опис ризику	Ймовірність виникнення	Вплив ризику	Ранг ризику	План А	Тип стратегії реагування	План Б
RS_1	Відкритий	Неоптимальне використання часу	Висока	Високий	8	Створити план виконання робіт та розставити пріоритети. Дотримуватися створеного плану	Пом'якшення	Змінити пріоритети робіт, оптимізувати роботу
RS_2	Відкритий	Недостатня кваліфікація розробників	Середня	Середній	3	Переглянути необхідну літературу або онлайн - ресурси	Пом'якшення	Відвідування курсів для підвищення кваліфікації
RS_3	Відкритий	Помилки проектування	Висока	Високий	9	Розбивати задачі на підзадачі та погоджувати результати робіт з замовником	Пом'якшення	

Продовження таблиці Б.6

ID	Статус ризику	Опис ризику	Ймовірність виникнення	Вплив ризику	Ранг ризику	План А	Тип стратегії реагування	План Б
RS_4	Відкритий	Втрата даних через технічні неполадки	Низька	Високий	7	Зберігати резервні копії в хмарних сховищах або незалежних носіях інформації	Попередження	
RS_5	Відкритий	Відсутність резервних копій	Низька	Високий	4	Налаштувати автоматичне збереження резервних копій	Попередження	Через певні інтервали часу вручну зберігати зміни
RS_6	Відкритий	Зміни ТЗ	Середня	Середній	3	Перед розробкою обговорити функціонал продукту		Приймати виключно зміни, які не впливають на основний функціонал
RS_7	Відкритий	Непрацездатність розробника в зв'язку зі хворобою	Низька	Високий	7	Попередити виникнення хвороби	Попередження	Виділення додаткового часу

Продовження таблиці Б.6

ID	Статус ризику	Опис ризику	Ймовірність виникнення	Вплив ризику	Ранг ризику	План А	Тип стратегії реагування	План Б
RS_8	Відкритий	Нечітке завдання на розробку	Середня	Високий	6	Обговорити вимоги та затвердити завдання на розробку з замовником	Попередження	Чітко описувати невідповідності та складати правки
RS_9	Відкритий	Поява альтернативного продукту	Середня	Високий	4	Провести дослідження альтернативних продуктів, врахувати їх недоліки	Прийняття	
RS_10	Відкритий	Непорозуміння між замовником та розробником	Низька	Низький	2	Створити комфортні умови співпраці	Попередження	Виявити та усунути причину непорозуміння