


Вінницький національний технічний університет  
Факультет інтелектуальних інформаційних технологій та автоматизації  
Кафедра комп'ютерних наук

**МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА**

на тему:

**«Інформаційна технологія імітації роботи квиткової каси»**

Виконав: студент 2-го курсу, групи 2КН-22м  
спеціальності 122 «Комп'ютерні науки»  
(шифр і назва напряму підготовки, спеціальності)

 Нетребський М.А.  
(прізвище та ініціали)

Керівник: к.т.н., доцент каф. КН

 Богач І.В.  
(прізвище та ініціали)


« 04 » 12 2023 р.

Опонент: к.т.н., доц. каф. АІТ

 Маслій Р.В.  
(прізвище та ініціали)

« 04 » 12 2023 р.

Допущено до захисту  
Завідувач кафедри КН

 д.т.н., проф. Яровий А.А.  
(прізвище та ініціали)  
« 08 » 12 2023 р.

Вінниця ВНТУ - 2023 рік

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

ЗАТВЕРДЖУЮ

Завідувач кафедри КН

д.т.н., проф. Яровий А.А.



21. 08. 2023 року

## ЗАВДАННЯ

### НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Нетребському Максиму Анатолійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Інформаційна технологія імітації роботи квиткової каси

керівник роботи к.т.н., доцент кафедри КН Богач І.В.

затвержені наказом вищого навчального закладу від "18" вересня 09 року №244

2. Строк подання студентом роботи 13.11 2023 року

3. Вихідні дані до роботи:

мінімум 2 режими симуляції, наявність налаштування швидкості моделювання, ступінь деталізації модуля статистики з ціною поділки шкали не менше 10.

4. Зміст текстової частини:

Вступ, аналіз предметної області імітаційного моделювання, розробка моделі роботи квиткових кас, реалізація імітаційної моделі квиткових кас, економічна частина, висновки, перелік використаних джерел, додатки.

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

Діаграма взаємодії агентів системи, метамодель UML діаграми послідовностей, метамодель PNML (модифікована UML діаграма класів), діаграма компонентів, діаграма взаємодії черги пасажирів, схема побудови імітаційної моделі.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	виконання прийняв
1-3	Богач І.В., к.т.н., доц. каф. КН	<i>Bohach</i> 29.08.23	<i>Bohach</i> 10.11.23
4	Ратушняк О.Г., к.т.н., доц. каф. ЕПВМ	<i>Ratushnyak</i> 21.10.23	<i>Ratushnyak</i> 23.10.23

7. Дата видачі завдання 29.08 2023 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз предметної області імітаційного моделювання	1.09.23 - 07.09.23	Розділ 1
2	Розробка моделі імітації роботи квиткових кас	08.09.23 - 20.09.23	Розділ 2
3	Реалізація імітаційної моделі квиткових кас	21.09.23 - 20.10.23	Розділ 3
4	Підготовка економічної частини	21.10.23 - 29.10.23	Розділ 4
5	Апробація та/або впровадження результатів дослідження	30.10.23 - 05.11.23	Тези доповіді
6	Оформлення пояснювальної записки, графічного матеріалу та презентації	06.11.23 - 10.11.23	Пояснювальна записка, графічний матеріал, презентація

Студент

*Y. H. H.*  
(підпис)

Нетребський М.А.

Керівник роботи

*Bohach*  
(підпис)

Богач І.В.

## АНОТАЦІЯ

УДК 004.8

Нетребський М. А. Інформаційна технологія імітації роботи квиткової каси. Магістерська кваліфікаційна робота зі спеціальності 122 «Комп'ютерні науки», освітня програма «Комп'ютерні науки». Вінниця: ВНТУ, 2023. 105 с. Укр. мовою. Бібліогр.: 29 назв; рис.: 23; табл. 11.

Дана магістерська кваліфікаційна робота присвячена розробці інформаційної технології роботи квиткової каси. У даній роботі досліджено особливості використання імітаційних моделей, розглянуто стан галузі на даний момент та описано її проблеми та виклики. Проаналізовано основні способи та засоби реалізації технології. Для реалізації обрано задачу симуляції квиткової каси, побудовано імітаційну модель та діаграми взаємодії для даної задачі. Виконано симуляцію моделі в середовищі Anylogic, покроково описано її проектування, протестовано експеримент та доведено працездатність розробленої моделі.

Графічна частина складається з 6 плакатів.

У економічному розділі визначено, що згідно узагальненого коефіцієнту конкурентоспроможності, науково-технічна розробка переважає існуючі аналоги приблизно в 2,45 рази. Термін окупності становить 0,33 р., що менше 3-х років, що свідчить про комерційну привабливість науковотехнічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Ключові слова: імітація, імітація роботи квиткової каси, імітаційні моделі, симуляція.

## ABSTRACT

Netrebskyi M. A. Information technology for simulating the work of a ticket office. Master's thesis on specialty 122 "Computer Science", educational program "Computer Science". Vinnytsia: VNTU, 2023. 105 p. Ukraine language Bibliography: 29 titles; Fig.: 23; table 11.

This master's thesis is devoted to the development of information technology for ticket office work. This paper examines the peculiarities of using simulation models, examines the state of the industry at the moment, and describes its problems and challenges. The main methods and means of implementing the technology are analyzed. For implementation, the task of simulating a ticket office was chosen, a simulation model and interaction diagrams for this task were built. The model was simulated in the Anylogic environment, its design was described step-by-step, the experiment was tested, and the performance of the developed model was proven.

The graphic part consists of 6 posters.

In the economic section, it is determined that according to the generalized coefficient of competitiveness, scientific and technical development exceeds existing analogues by approximately 2.45 times. The payback period is 0.33 years, which is less than 3 years, which indicates the commercial attractiveness of the scientific and technical development and may encourage a potential investor to finance the implementation of this development and its introduction to the market.

Keywords: imitation, imitation of the ticket office, imitation models, simulation.

## ЗМІСТ

ВСТУП .....	4
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	8
1.1 Огляд специфіки функціонування квиткових кас та необхідність створення імітації її роботи.....	8
1.2 Огляд видів моделювання для розв’язання поставленої задачі .....	8
1.3 Імітаційне моделювання.....	10
1.3.1 Сучасний стан імітаційного моделювання .....	12
1.3.2 Аналіз особливостей імітаційного моделювання .....	16
1.4 Висновки до розділу 1 .....	23
2 РОЗРОБКА МОДЕЛІ ІМІТАЦІЇ РОБОТИ КВИТКОВИХ КАС .....	25
2.1 Розробка діаграм взаємодій агентів системи .....	25
2.2 Розробка UML діаграми .....	28
2.3 Розробка метамоделі PNML.....	34
2.4 Розробка імітаційної моделі роботи квиткових кас .....	38
2.5 Висновки до розділу 2 .....	40
3 РЕАЛІЗАЦІЯ ІМІТАЦІЙНОЇ МОДЕЛІ РОБОТИ КВИТКОВИХ КАС .....	41
3.1 Особливості побудови імітаційних моделей.....	41
3.2 Реалізація імітаційної моделі в середовищі AnyLogic .....	44
3.3 Проведення оптимізаційного експерименту .....	50
3.4 Тестування та порівняння з аналогами.....	58
3.5 Висновки до розділу 3 .....	61
4 ЕКОНОМІЧНА ЧАСТИНА .....	63
4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки .....	63
4.2 Визначення рівня конкурентоспроможності розробки.....	67
4.3 Розрахунок витрат на проведення науково-дослідної роботи .....	70
4.3.1 Витрати на оплату праці.....	70
4.3.2 Відрахування на соціальні заходи .....	73

4.3.3 Сировина та матеріали.....	73
4.3.4 Спецустаткування для наукових (експериментальних) робіт .....	74
4.3.5 Програмне забезпечення для наукових (експериментальних) робіт .	75
4.3.6 Амортизація обладнання, програмних засобів та приміщень .....	76
4.3.7 Паливо та енергія для науково-виробничих цілей.....	77
4.3.8 Службові відрядження.....	77
4.3.9 Інші витрати.....	78
4.3.10 Накладні (загальновиробничі) витрати.....	79
4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором.....	80
4.5 Висновки до розділу 4 .....	84
<b>ВИСНОВКИ.....</b>	<b>86</b>
<b>ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>88</b>
Додаток А (обов'язковий) Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень .....	91
Додаток Б (обов'язковий) Лістинг програми .....	92
Додаток В (обов'язковий) ІЛЮСТРАТИВНА ЧАСТИНА.....	99
Додаток Г (довідниковий) Інструкція користувача.....	103

## ВСТУП

**Актуальність теми.** Дослідження з теми імітаційного моделювання роботи квиткових кас є дуже актуальним в наш час. У зв'язку з швидким розвитком технологій та поширенням електронних платіжних систем, відбувається зміна способів продажу та обліку квитків. В той же час, фізичні квиткові каси залишаються використовуватися в багатьох галузях, таких як транспорт, культура, спорт тощо. Тому важливо досліджувати процес роботи квиткових кас та знайти оптимальні рішення щодо їх ефективності та економічної вигідності. [1-4]

Застосування імітаційного моделювання є перспективним у даній області досліджень, оскільки цей метод дозволяє відтворити роботу квиткових кас у віртуальному середовищі та дослідити різні сценарії їх функціонування. При цьому можна врахувати різні фактори, які впливають на роботу кас, такі як кількість клієнтів, швидкість обробки замовлень, навантаження на обладнання тощо. Такий підхід дозволяє знайти оптимальні рішення щодо розстановки кас, розробки програмного забезпечення та організації роботи персоналу.

Окрім того, дослідження з даної теми має практичне значення. Результати дослідження можуть бути використані для покращення роботи квиткових кас у різних галузях та зниження витрат на їх обслуговування.

Однією з особливостей постановки питань стосовно конкретних умов дослідження імітаційного моделювання роботи квиткових кас є визначення мети дослідження та постановка задач, які необхідно вирішити для досягнення цієї мети. Наприклад, якщо метою дослідження є покращення ефективності роботи кас у кінотеатрах, то необхідно вирішити питання щодо розміщення кас, розробки оптимальних алгоритмів продажу квитків та організації роботи персоналу. [1, 4, 5]

Не менш важливою особливістю є вибір параметрів, які будуть враховуватися при імітаційному моделюванні. Це можуть бути такі параметри, як кількість клієнтів, що приходять до кас, середній час обробки замовлень, час



очікування клієнтів у черзі та інші. Важливо правильно підібрати дані параметри, щоб результати дослідження відображали реальні умови роботи кас.

Окрім того, під час вирішення питань стосовно конкретних умов дослідження важливо враховувати особливості різних галузей, де використовуються квиткові каси. Наприклад, в транспорті можуть бути особливості стосовно розміщення кас, тарифів та типів квитків, в культурі – щодо розкладу сеансів та попиту на квитки. Тому важливо враховувати специфіку кожної галузі та використовувати цю інформацію під час дослідження.

Тема магістерської кваліфікаційної роботи є актуальною, оскільки завжди існує потреба у придбанні певних квитків у касах, але у зв'язку з швидким розвитком технологій та поширенням електронних платіжних систем, відбувається зміна способів продажу та обліку квитків. В той же час, фізичні квиткові каси залишаються використовуватися в багатьох галузях. Застосування імітаційного моделювання є перспективним, оскільки дозволяє знайти оптимальні рішення щодо розстановки кас, розробки програмного забезпечення та організації роботи персоналу для зручного та швидкого придбання квитків. Дана технологія буде корисною, як для державних структур так і для приватних.

**Зв'язок роботи з науковими програмами, планами, темами.** Магістерська кваліфікаційна робота виконана відповідно до напрямку наукових досліджень кафедри комп'ютерних наук Вінницького національного технічного університету 22 К1 «Моделі, методи, технології та пристрої інтелектуальних інформаційних систем управління, економіки, навчання та комунікацій» та плану наукової та навчально-методичної роботи кафедри.

#### **Мета та завдання дослідження.**

Мета магістерської кваліфікаційної роботи полягає в підвищенні ефективності роботи квиткових кас шляхом використання імітаційного моделювання, що дозволить знайти оптимальні рішення щодо розстановки кас та організації роботи персоналу.

Для досягнення поставленої мети необхідно виконати наступні завдання:

1. Провести аналіз предметної галузі та визначити вид моделювання для поставленої задачі та обґрунтувати вибір імітаційного моделювання, визначити його переваги та недоліки.
2. Розглянути методи та технології імітації роботи квиткової каси.
3. Проаналізувати та обґрунтувати вибір інструментів для розробки.
4. Розробити модель імітації роботи квиткових кас.
5. Реалізувати імітаційну модель та провести оптимізаційний експеримент.
6. Протестувати розроблене програмне забезпечення та порівняти з аналогами.
7. Економічно обґрунтувати доцільність розробки інформаційної технології імітації роботи квиткової каси.

**Об'єкт дослідження:** процеси функціонування квиткових кас з використанням інформаційних технологій.

**Предмет дослідження:** це методи та програмні засоби для імітації роботи квиткових кас.

**Методи дослідження.** У роботі використано такі методи наукових досліджень: метод моделювання, експерименту, статистичного дослідження, проектування складних систем, оптимізації.

**Наукова новизна одержаних результатів** полягає в удосконаленні інформаційної технології роботи квиткових кас за рахунок побудови її імітації за допомогою використання методів імітаційного моделювання, що на відміну від існуючих аналогів дозволяє вирішувати оптимізаційні задачі функціонування роботи квиткових кас.

**Практичне значення одержаних результатів.**

1. Розроблено алгоритм роботи квиткової каси в реальному часі, що дозволяє знайти оптимальні рішення щодо розстановки кас та організації роботи персоналу.

2. Розроблено програмний засіб роботи квиткової каси в реальному часі з використанням середовища AnyLogic, що відображає та аналізує ключові

аспекти операцій каси, включаючи продаж квитків, ведення обліку та взаємодію з клієнтами.

**Достовірність теоретичних положень.** Теоретичні положення, використані в цій магістерській роботі з інформаційної технології імітації роботи квиткової каси, отримали підтвердження через аналіз широкого спектру наукових джерел, включаючи актуальні публікації та наукові статті у галузі інформаційних технологій, теорії черг, моделювання бізнес-процесів та систем управління. Використання концепцій та методів, які мають підтверджену наукову базу, дозволило побудувати концептуальну рамку дослідження та забезпечити достовірність та обґрунтованість отриманих результатів. Крім того, порівняння теоретичних концепцій з практичними даними та результатами емпіричних спостережень підтверджує застосовність та відповідність теоретичних положень у реальному бізнес-середовищі, що підсилює достовірність та значущість отриманих висновків.

**Особистий внесок здобувача.** Усі результати магістерської кваліфікаційної роботи були отримані здобувачем самостійно.

**Апробація результатів роботи.** Результати досліджень було апробовано на Міжнародній науково-практичній Інтернет-конференції студентів, аспірантів та молодих науковців «Молодь в науці: дослідження, проблеми, перспективи-2024» (МН-2024).[6]

**Публікації.** За результатами магістерської кваліфікаційної роботи опубліковано тези доповідей на науково-технічній конференції. [6]

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

### 1.1 Огляд специфіки функціонування квиткових кас та необхідність створення імітації її роботи

Функціонування квиткових кас на залізничних вокзалах включає ряд специфічних етапів та функцій, які забезпечують видачу квитків пасажиром. Основні аспекти функціонування квиткових кас включають:

#### 1) Прийом та обробка інформації:

- Визначення маршрутів та рейсів: Квиткові каси мають доступ до бази даних про розклад руху поїздів та маршрути, що дозволяє їм надавати інформацію про наявність рейсів.

- Інформація про вільні місця: Квиткові каси повинні мати доступ до даних про кількість вільних місць у кожному поїзді для видачі актуальної інформації пасажиром.

#### 2) Прийом та обробка замовлень:

- Прийом пасажирської інформації: Пасажир подає інформацію про свою поїздку, таку як маршрут, дату, клас обслуговування та інші деталі.

- Розрахунок вартості: Система квиткових кас розраховує вартість квитка відповідно до обраних параметрів та тарифів.

#### 3) Оплата та видача квитків:

- Вибір методу оплати: Пасажир вибирає метод оплати, чи це готівка, кредитна/дебетова карта, або інші доступні опції.

- Оплата: Здійснюється оплата обраного квитка відповідно до вибраних параметрів та цін.

- Видача квитка: Квиткова каса видаватиме фізичний чи електронний квиток пасажиром.

#### 4) Управління чергами:

- Організація черги: Квиткова каса повинна ефективно управляти чергами, особливо в періоди пікового навантаження.

- Пріоритет обслуговування: Деякі квиткові каси можуть визначати пріоритет для пасажирів, які потребують екстреної видачі квитків.

5) Керування технічними аспектами:

- Технічна підтримка обладнання: Квиткові каси повинні бути обладнані та підтримуватися, щоб уникнути технічних збоїв.

- Оновлення програмного забезпечення: Регулярне оновлення програмного забезпечення для вдосконалення функціональності та безпеки.

6) Клієнтське обслуговування:

- Вирішення запитань і скарг: Квиткові каси повинні надавати клієнтам інформацію та вирішувати можливі проблеми чи скарги.

7) Забезпечення безпеки:

- Захист від шахрайства: Квиткові каси повинні мати системи захисту для уникнення шахрайства та підробки квитків.

Враховуючи ці аспекти, квиткові каси відіграють ключову роль у забезпеченні безперебійного та зручного процесу придбання квитків для пасажирів на залізничних вокзалах.

Сьогодні, у світі, де цифрові технології проникають у всі сфери життя, залізничні квитки стають об'єктом онлайн-придбання через спеціалізовані веб-платформи або за допомогою мобільних додатків. Електронні квитки, зазначені для багатьох видів транспорту, включаючи приміське сполучення, стали звичайними та надійними.

У зв'язку із цим, традиційні квиткові каси та автомати на вокзалах втрачають свою актуальність, адже пасажирів все частіше обирають зручний та швидкий спосіб отримання квитка через електронні платформи. Такий тренд, проте, не означає повне відмовлення від традиційних методів, адже важливо ефективно моделювати та керувати потоками пасажирів, особливо в періоди пікового навантаження або перед відправленням поїздів.

Щоб забезпечити найвищий рівень обслуговування та задоволення клієнтів, деякі вокзали зберігають традиційні каси та автомати, але вдосконалюють їхню роботу, використовуючи інноваційні технології. Наприклад, важливо моделювати пасажиропотік та планувати роботу кас в залежності від інтенсивності запитань. Це може включати в себе використання аналітики та штучного інтелекту для прогнозування потреб пасажирів.

Проблема довгих черг і стресу для подорожуючих, які намагаються придбати квитки на останню мить, може бути вирішена за допомогою впровадження віртуальних систем. Така інновація дозволить "продавати" квитки всім пасажиром, утримуючи порядок прибуття та враховуючи індивідуальні характеристики кожного пасажиром.

Важливою аспектом вдосконалення системи квиткових кас є використання імітаційного моделювання. Це дозволяє висвітлити та вирішити потенційні проблеми, а також удосконалювати ефективність обслуговування. Застосування сучасних інформаційних технологій, таких як аналітика та інтелектуальні системи, вносить величезний внесок у розвиток транспортних систем та поліпшення комфорту подорожей для користувачів.

## **1.2 Огляд видів моделювання для розв'язання поставленої задачі**

Імітаційне моделювання є важливим методом для аналізу та дослідження різних аспектів бізнесу та менеджменту. Цей підхід дозволяє відтворити реальні системи за допомогою мови моделювання, базуючись на науці про прийняття рішень та інформаційних технологіях. Важливо зауважити, що, хоча імітаційне моделювання визнається як систематична та цінна методологія для розробки теорії та аналізу, його потенціал в галузі бізнесу та менеджменту ще не отримав належного визнання та застосування.

В сучасних дослідженнях у сфері бізнесу та менеджменту переважають кількісні методи, такі як кореляційний, причинно-наслідковий і порівняльний підходи, засновані на опитуваннях. Незважаючи на переваги імітаційного

моделювання, йому досі приділяється обмежена увага в цих галузях. Інші соціальні науки, зокрема психологія, вже демонструють більшу активність у використанні моделювання для визначення політики та вивчення організаційних проблем.

В імітаційному моделюванні визнаються різні методології наукового прогресу, такі як теоретичний аналіз (дедукція) та емпіричний аналіз (індукція). Комп'ютерне моделювання, яке дозволяє обходити аналітичні обмеження, робить його ефективним інструментом для розробки теорій. Використання віртуальних даних сприяє подоланню емпіричних обмежень доступності реальних даних та дозволяє вченим формулювати більш реалістичні припущення.

Імітаційне моделювання може впливати на науковий прогрес, забезпечуючи можливість створення інтегрованих та послідовних гіпотез. Такий підхід дозволяє дослідникам робити більш обґрунтовані припущення та розвивати складні теорії.

Таким чином, необхідно визнати потенціал імітаційного моделювання як важливого інструменту для досліджень у галузі бізнесу та менеджменту, а також активніше впроваджувати його в аналітичні та наукові практики для подальшого розвитку цих сфер. У загальному випадку проведення імітаційного експерименту можна розбити на наступні етапи.

1. Встановлення взаємозв'язку між вихідними і вхідними показниками у вигляді математичного рівняння або нерівності.
  2. Визначення законів розподілу значень для ключових параметрів моделі.
  3. Проведення комп'ютерної імітації значень ключових параметрів моделі.
  4. Розрахунок основних характеристик розподілів вихідних і вхідних показників.
  5. Проведення аналізу отриманих результатів і прийняття рішення.
- Результати імітаційного експерименту можуть бути доповнені статистичним аналізом, а також використовуватися для побудови прогностичних моделей і сценаріїв.

## 1.3 Імітаційне моделювання

### 1.3.1 Сучасний стан імітаційного моделювання

Імітаційне моделювання визнається як ефективний метод імітації реальних систем, використовуючи мову моделювання для їх представлення. Незважаючи на те, що моделювання базується на дисциплінах науки про прийняття рішень та інформаційних технологій, його роль у розробці теорії та аналізі досліджень в сфері бізнесу та менеджменту вважається більш систематичною та цінною методологією. Здивовано, що, незважаючи на потенційні переваги імітаційного моделювання, наукові кола в галузі бізнесу та менеджменту недостатньо використовують його ефективні методи для проведення досліджень. З іншого боку, вчені з інших соціальних наукових дисциплін, зокрема психології, виглядають більш продуктивними в застосуванні моделювання для визначення політики та вивчення організаційних проблем. Слід відзначити, що в масштабних дослідженнях у сфері бізнесу та менеджменту домінують кількісні методи, такі як кореляційний, причинно-наслідковий і порівняльний підходи, засновані на опитуваннях (Creswell & Creswell, 2018). На жаль, імітаційне моделювання, не зважаючи на його потужний кількісний потенціал у застосуванні математичних моделей, залишається малопривабливим напрямком для досліджень у галузі бізнесу та менеджменту. Отже, варто звернутися до розширення застосування імітаційного моделювання в галузі бізнесу та менеджменту, враховуючи його потенціал для систематизації та цінності у розробці теорії та аналізі досліджень.

Існують різні методології наукового прогресу, і серед них особливо виділяють теоретичний аналіз та емпіричний аналіз (Harrison et al., 2007) [10]. У теоретичному аналізі, науковці формулюють набір припущень, які потім використовуються для виведення наслідків в дедуктивній формі. Цей підхід ставлення до науки, як до конструктивної системи, де структуровані припущення дозволяють отримувати конкретні висновки. Тим часом, емпіричний аналіз



полягає в спостереженні за змінними та аналізі даних для виявлення зв'язків між ними в індуктивній формі. Цей метод дозволяє вивчати явища на основі спостережень та експериментів, а не тільки теоретичних розрахунків. Комп'ютерне моделювання виступає як третій спосіб наукового дослідження. Це вирішує аналітичну нерозв'язність дедуктивного методу, дозволяючи обробляти математичні співвідношення за допомогою чисельних методів (Аксельрод, 1997). У цьому контексті важливо зазначити, що комп'ютерне моделювання генерує власні віртуальні дані, подолавши емпіричні обмеження, пов'язані з доступністю реальних даних. Однією з ключових переваг комп'ютерного моделювання є його здатність підтримувати побудову теорії. Віртуальні дослідження дозволяють науковцям робити більш реалістичні припущення та уникати компромісів, які часто необхідні при застосуванні аналітичних методів в дедуктивній теорії (Levinthal & Marengo, 2016). Це стимулює розвиток інтегрованих та послідовних гіпотез, які можуть бути створені та перевірені шляхом комп'ютерного моделювання.

Таким чином, дослідження імітаційного моделювання сприяють накопиченню знань у багатьох галузях (рис. 1.1).

Розрахунок частки літератури з імітаційного моделювання за період з 2001 по 2023 рік виявив вражаючі дані. За цей час було опубліковано загальною кількістю 3,493,312 статей, пов'язаних із застосуванням імітаційного моделювання в різних дисциплінах.

Імітаційне моделювання виявляє високий рівень застосування в різноманітних галузях, таких як інженерія, медицина та медичні науки, освіта, технології, психологія і багато інших. Це свідчить про універсальність імітаційного моделювання як інструменту для досліджень та розв'язання проблем в різних областях знань.

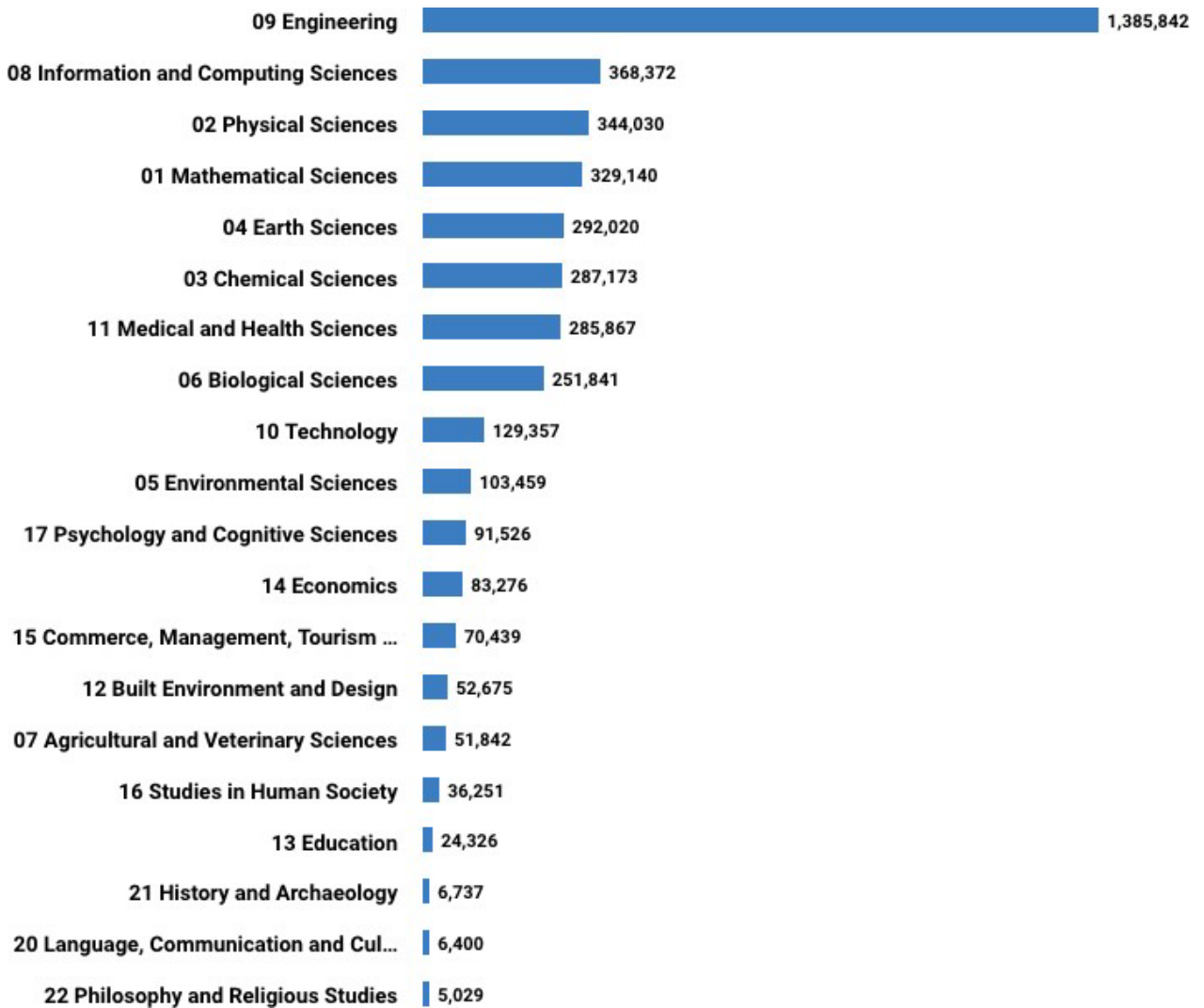


Рисунок 1.1 – Топ-20 дисциплін, в яких найбільш часто застосоване імітаційне моделювання

Результати дослідження представлені на рисунку 1.2, який ілюструє обсяг та розподіл застосування імітаційного моделювання в провідних журналах різних дисциплін. Цей аналіз сприяє розумінню впливу імітаційних досліджень у науковому середовищі та допомагає висвітлити широкий спектр його застосування в різних сферах.

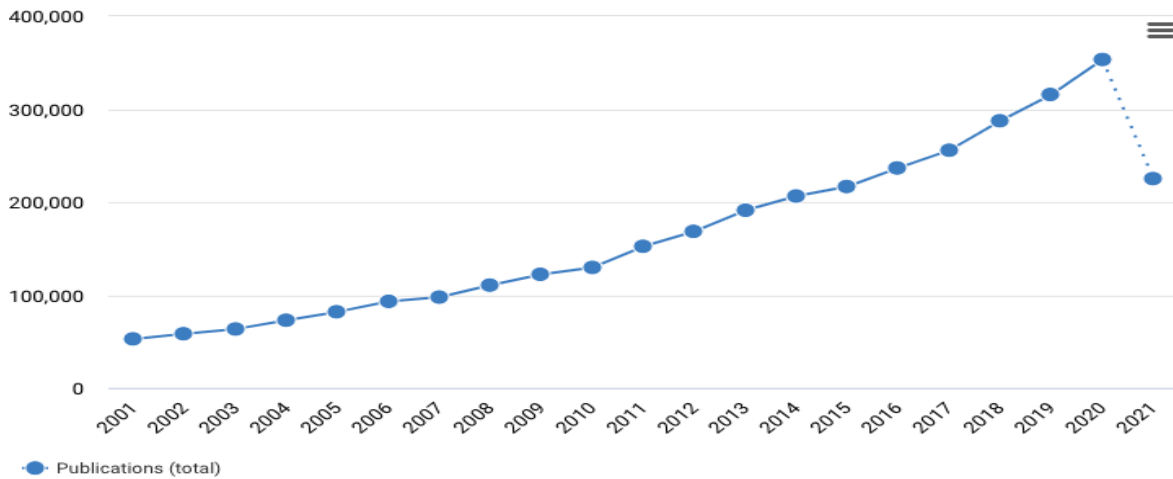


Рисунок 1.2 – Публікації з імітаційного моделювання 2001- 2021 роки

Рисунок 1.3 ілюструє рейтинг 20 провідних журналів, що спеціалізуються на публікації статей з імітаційного моделювання в галузі бізнесу та управління. Ці журнали охоплюють широкий спектр тем, включаючи виробництво, експлуатацію, ланцюг постачання, науку прийняття рішень, стратегічний менеджмент, маркетинг, аналіз ризиків і роздрібну торгівлю.

Особливу увагу слід приділити "International Journal of Computer Integrated Manufacturing", який визначається як провідний видавець з великою кількістю опублікованих статей. Цей журнал відіграє ключову роль у розповсюдженні новітніх досліджень та розробці в області комп'ютерного інтегрованого виробництва.

За загальним впливом виділяється "Strategic Management Journal", який здобув вражаючі 28,726 цитувань протягом 22 років. Це свідчення високого рівня впливу та розпізнавання серед наукової спільноти. Ці журнали є ключовими джерелами для вчених та фахівців, що цікавляться імітаційним моделюванням у контексті бізнесу та управління. Вони відображають важливий внесок у розвиток відповідних галузей і становлять базу для обміну знаннями та наукового діалогу.

Name	↓ Publications	Citations	Citations mean
International Journal of Computer Integrated ...	673	9,003	13.38
International Journal of Simulation and Proce...	660	1,595	2.42
International Journal of Production Research	517	19,198	37.13
Production and Operations Management	483	18,847	39.02
International Journal of Production Economics	440	24,855	56.49
Omega	371	16,449	44.34
System Dynamics Review	333	9,011	27.06
Production Planning & Control	266	5,740	21.58
Decision Sciences	236	10,467	44.35
Engineering Construction & Architectural Ma...	223	2,456	11.01
Strategic Management Journal	221	28,726	129.98
International Journal of Quality & Reliability ...	219	3,624	16.55
European Journal of Operational Research	203	9,746	48.01
Journal of Business Research	201	7,991	39.76
Industrial Marketing Management	201	7,597	37.80
International Journal of Operations & Product...	201	9,874	49.12
Journal of Management	192	22,060	114.90
Risks	192	628	3.27
Business Process Management Journal	188	4,171	22.19
Journal of Retailing and Consumer Services	186	4,972	26.73

Рисунок 1.3 – Топ-20 журналів з імітаційного моделювання

### 1.3.2 Аналіз особливостей імітаційного моделювання

Проаналізуємо особливості, а саме переваги і недоліки застосування імітаційного моделювання.

По-перше, структуру реальної системи природно відображає побудова імітаційної моделі. Відомо, що імітаційні моделі розробляються переважно з використанням візуальних мов. Таким чином, простіше повідомити внутрішні дані моделі іншим людям. Ця рефлексія зазвичай ґрунтується на спостереженнях і дослідженнях дослідників, наприклад, вирішуючи, як краще побудувати модель магазину. Блок-схема процесу, де клієнти є об'єктами, а співробітники – ресурсами моделі. Відтворюється агентська модель, де споживачі є агентами, на

яких впливає реклама, комунікація та взаємодія з агентами-працівниками. Крім того, існує структура зворотного зв'язку, де продажі пов'язані з рекламою, якістю обслуговування, ціною, лояльністю клієнтів і багатьма пов'язаними факторами, які існують у магазині. Завдяки цій функції імітаційні моделі дозволяють розробникам моделей аналізувати системи та знаходити рішення там, де інші методи (наприклад, аналітичні обчислення, лінійне програмування) не дають результатів.

По-друге, рівень абстракції відповідних моделей є унікальною особливістю деяких застосувань моделювання. Самі моделі є моделями фізичного рівня, де характерно представлені об'єкти реального світу, такі як фізична взаємодія, розміри, швидкості, відстані, час, поведінка, рішення тощо. Vasudevan and Devikar (2011) стверджують, що рівень деталізації рівня абстракції, необхідний для моделювання, визначає обсяг інформації, що міститься в моделі. Кількість інформації в моделі зменшується зі зниженням рівнів абстракції, що означає, що модель абстракції низького рівня містить більше інформації, ніж модель абстракції високого рівня. Життєво важливо моделювати на найвищому можливому рівні абстракції, щоб не йти на компроміс із точністю результатів чи подальших рішень (Vasudevan & Devikar, 2011). Тому помічено, що фахівцям необхідно вибрати правильний рівень абстракції, який є життєво важливим для успіху проекту моделювання. У процесі розробки моделі рекомендується періодично переглядати рівень абстракції. Крім того, дослідники можуть вимірювати будь-яке значення та відслідковувати будь-який об'єкт, якщо він не нижче рівня абстракції, що означає, що вимірювання та статистичний аналіз доступні в будь-який час в імітаційній моделі.

По-третє, розробка імітаційної моделі є простим процесом, і зазвичай вимагає менше інтелектуальних зусиль, є масштабованою, інкрементальною та модульною. Левінталь і Маренго (2016) стверджують, що імітаційні моделі природним чином можуть враховувати більш правдоподібні поведінкові припущення щодо агентів, ніж оптимізація корисності (або прибутку), ніж стандартні моделі рівноваги. Таким чином, поведінка системи в часі є

відтворюваною та анімованою, що відомо як одна з найважливіших переваг моделювання для ілюстрації результату, дослідження взаємодій та налагодження припущень у моделі. Функції 2D і 3D є поширеними, і дослідникам цікаво переглядати та жваво представляти свої ідеї під час роботи більшості програм для моделювання.

Хоча технологія є критично важливим інструментом для реалізації та запуску моделей, у деяких випадках вона також обмежує обчислювальні можливості через обмеження на швидкість обробки, зберігання та функції програмування. Це перше обмеження імітаційного моделювання. Наприклад, моделювання на основі агентів містить активні об'єкти (люди, бізнес-одиниці, тварини, транспортні засоби, проекти, акції, продукти тощо) із часом, порядком подій та іншою індивідуальною поведінкою. Для такого типу системи багато інструментів не можуть ефективно розробляти агентів і підтримувати повну міграцію від застарілих фондових і блок-схем або блок-схем процесів до агентів через відсутність системи компетентності.

Друге обмеження імітаційного моделювання полягає в рівні складності моделі. Зазвичай імітаційна модель будується шляхом поєднання простоти та деталізації (Harrison et al., 2007). Щоразу, представляючи імітаційну модель, мається на увазі, що найчастіше виникає запитання: «Чому б вам не додати змінні А, В, С до моделі?» З одного боку, модель стає більш реалістичною, додаючи якомога більше змінних або процесів. З іншого боку, також стає надто складно зрозуміти, що відбувається, і керує результатами в моделі з вищим складним рівнем.

Нарешті, цінність результатів моделювання залежить від достовірності імітаційної моделі. Виконуючи дослідження, розробники імітаційного моделювання намагаються оцінити поведінку моделі в ряді умов у технічно складних і чутливих до помилок комп'ютерного програмування. Таким чином, дані, отримані шляхом моделювання, не повністю відображають реальні спостереження та обмежені методами аналізу. Ці дані слід інтерпретувати з обережністю та додатковою кваліфікацією при узагальненні результатів

моделювання на області простору параметрів, які не досліджувалися під час моделювання.

### 1.3.3 Інформаційні технології імітаційного моделювання

Зважаючи на рівень абстракції моделі та загальну структуру для відображення системи реального світу на її модель, можна виділити три основні типи імітаційних моделей які моделюють засобами сучасних інформаційних технологій (рис. 1.4).

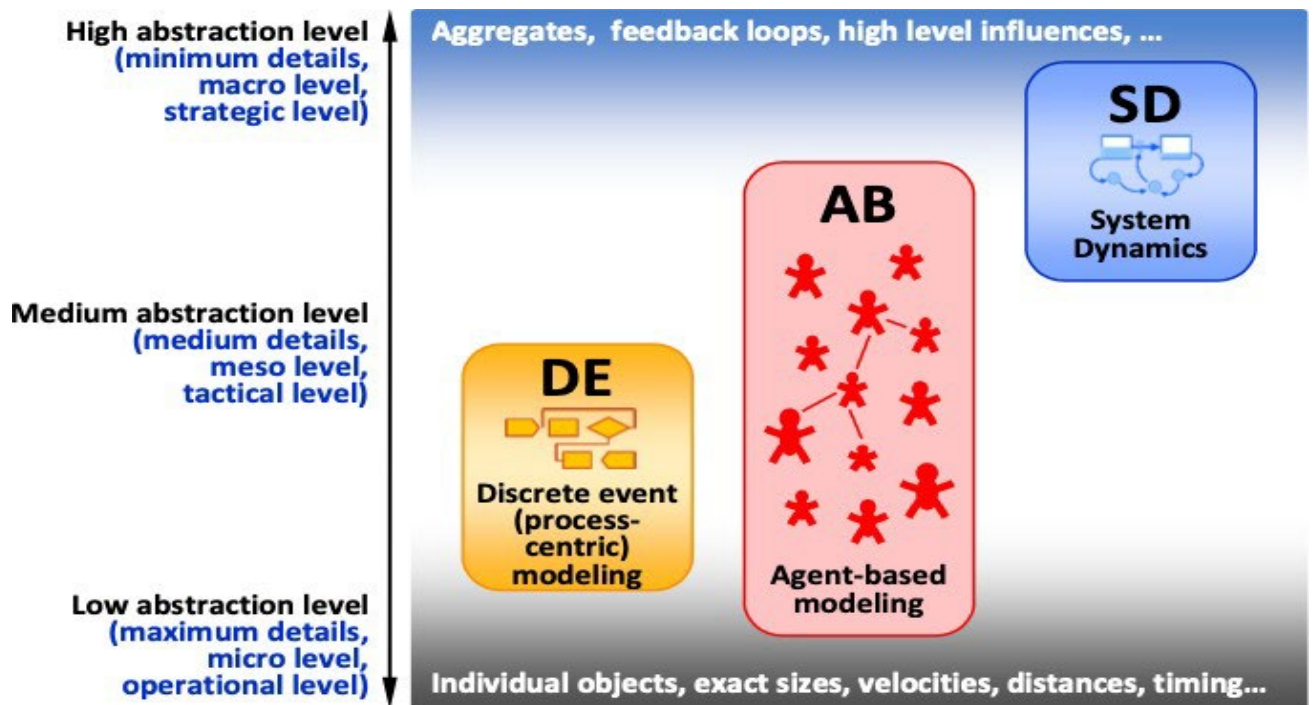


Рисунок 1.4 – Три типи імітаційного моделювання

Системна динаміка в основному використовується для стратегічного моделювання на високому рівні абстракції, а саме відтворення черги в магазині, розповсюдження нового продукту компанії [11-12]. При середній та середньо-низькій абстракції моделювання дискретних подій із базовим процесоцентричним підходом застосовується для процесу зняття грошей у банку, робочого цеху на виробничій компанії або процесу вступу на факультет університету тощо. Агентна модель є більш гнучкою з точки зору абстракції;

наприклад, вона може бути дуже детальною, коли агенти моделюють фізичні об'єкти, до дуже абстрактного, коли агентами є конкуруючі компанії чи уряди. Крім того, існує четвертий тип імітаційної моделі, який об'єднує типи, згадані вище, в одну модель, що називається моделлю кількох методів і широко застосовується в багатьох дисциплінах.

Системна динаміка — це метод, створений у 1950-х і на початку 1960-х років професором Массачусетського технологічного інституту Джеєм Райтом Форрестером, який спочатку займався наукою та технікою [13-15]. Він відкрив галузь системної динаміки, яка досліджує складні ділові, економічні та екологічні системи, а несподівані зворотні зв'язки впливають на людську діяльність, що створюється в них. Системна динаміка — це метод дослідження динамічних систем з ендогенної точки зору (рис. 1.5).

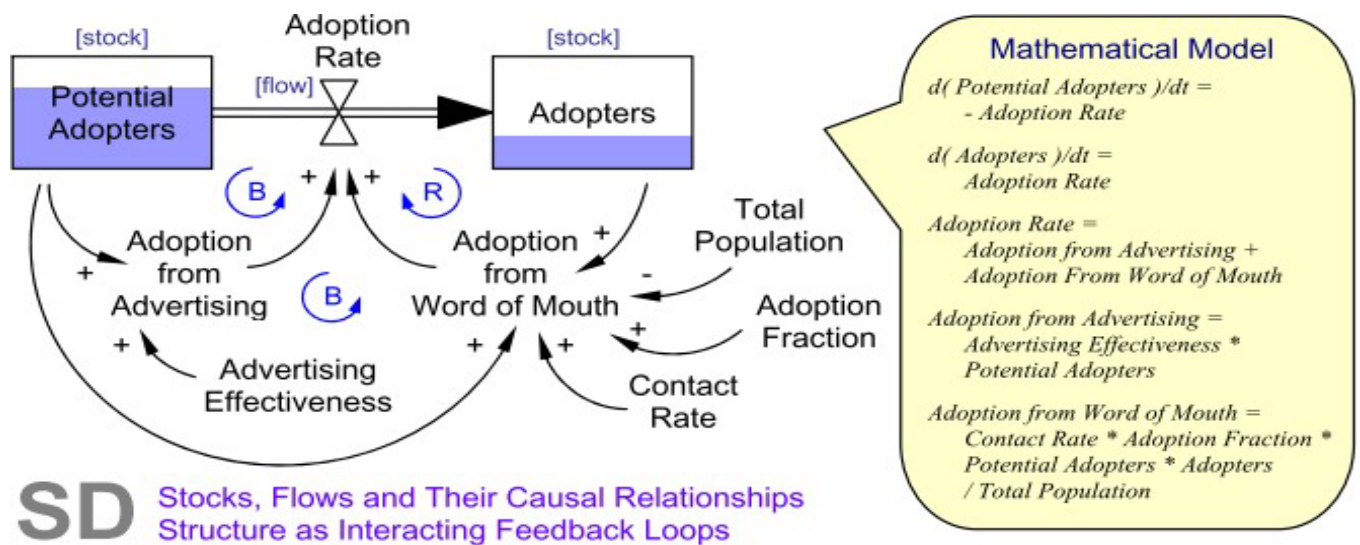


Рисунок 1.5 – Приклад системної динаміки

Моделювання системи розглядається як причинно-замкнена структура, яка сама по собі визначає свою поведінку. Петлі зворотного зв'язку або циклічна причинність є серцевиною динаміки системи, як показано на рисунку 1.5, який базується на моделі дифузії Басса. Це допомагає ідентифікувати запаси або накопичення, які є потенційними користувачами та учасниками моделі для прикладу, а також потоки, які на них впливають. Запаси - це пам'ять системи та



джерела нерівноваги. У цьому методі окремі події та рішення розглядаються як поверхневі явища, які ґрунтуються на структурі та поведінці системи, що дає змогу розробникам моделей бачити речі з певної точки зору безперервного перегляду, де події та рішення розмиті (Sterman, 2000). З точки зору математики, модель динаміки системи – це система, що складається з диференціальних рівнянь. Є дві важливі характеристики моделювання системної динаміки. По-перше, модель працює лише з агрегатами, що означає, що елементи в тому самому складі нерозрізнені; вони не мають індивідуальності. По-друге, фахівець повинен надати глобальні структурні залежності для точного кількісного визначення даних для них.

Наприклад, компанія набирає нових продавців відповідно до п'ятиетапного процесу відбору Jobber and Lancaster (2017), включаючи (1) підготовку опису посади, (2) визначення ресурсу найму, (3) підготовку короткого списку, (4) співбесіда та (5) використання додаткових засобів відбору.

Залежно від рівня абстракції та призначення моделі, використання кожного кроку в грошовій формі або кількості співробітників можна використовувати в цій моделі як миттєві події. Деталі їхніх компонентів можуть бути нерелевантними, оскільки лише важливі моменти в житті системи розглядаються та розглядаються як миттєві та атомарні події та абстрагуються від усього, що відбувається між двома суміжними подіями в моделюванні дискретних подій (рис. 1.6).

Започатковане в 2002-2003 роках моделювання на основі агентів використовується для моделювання поведінки адаптивних акторів, які складають соціальну систему та впливають один на одного через свою взаємодію (Harrison et al., 2007). Даний метод створений для глибшого розуміння систем, які погано охоплюються традиційними підходами до моделювання, такими як системна динаміка та дискретні події.

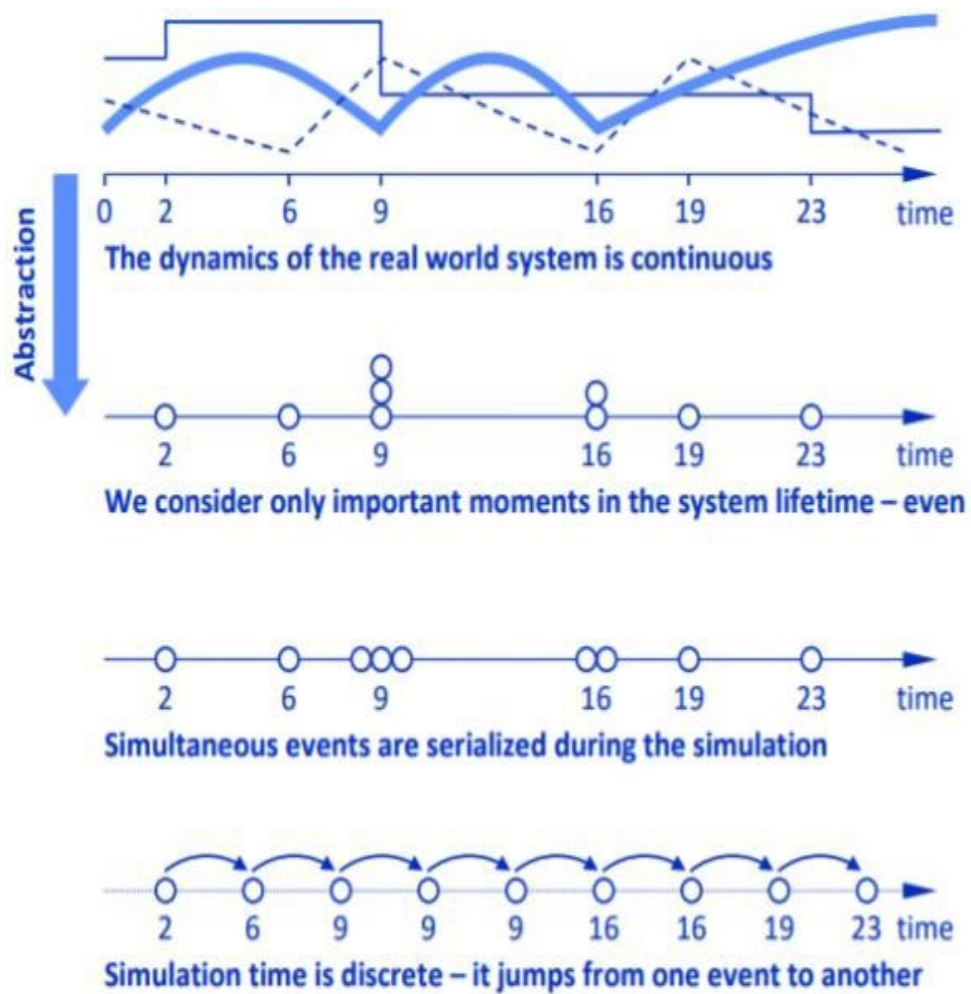


Рисунок 1.6 – Моделювання дискретних подій

Цей метод створено завдяки прогресу в технології моделювання, отриманому в інформатиці, і швидкому зростанню доступності потужності ЦП і пам'яті завдяки широкому використанню графічних редакторів або сценаріїв для ілюстрації різноманітної поведінки агентів. У моделі, заснованій на агенті, агент зазвичай має уявлення про стан, а його дії та реакції залежать від його стану, поведінка якого визначається діаграмами станів (рис. 1.7).

Найвідоміші програми з імітаційного моделювання: HSC Chemistry, ProcessModel, ExtendSim CP, Simcad Pro, Simio, FlexSim, PaleBlue, AnyLogic, Simul8, AutoTurn, Arena, Hash, Haulsim, Simile, Simulation Modeling.

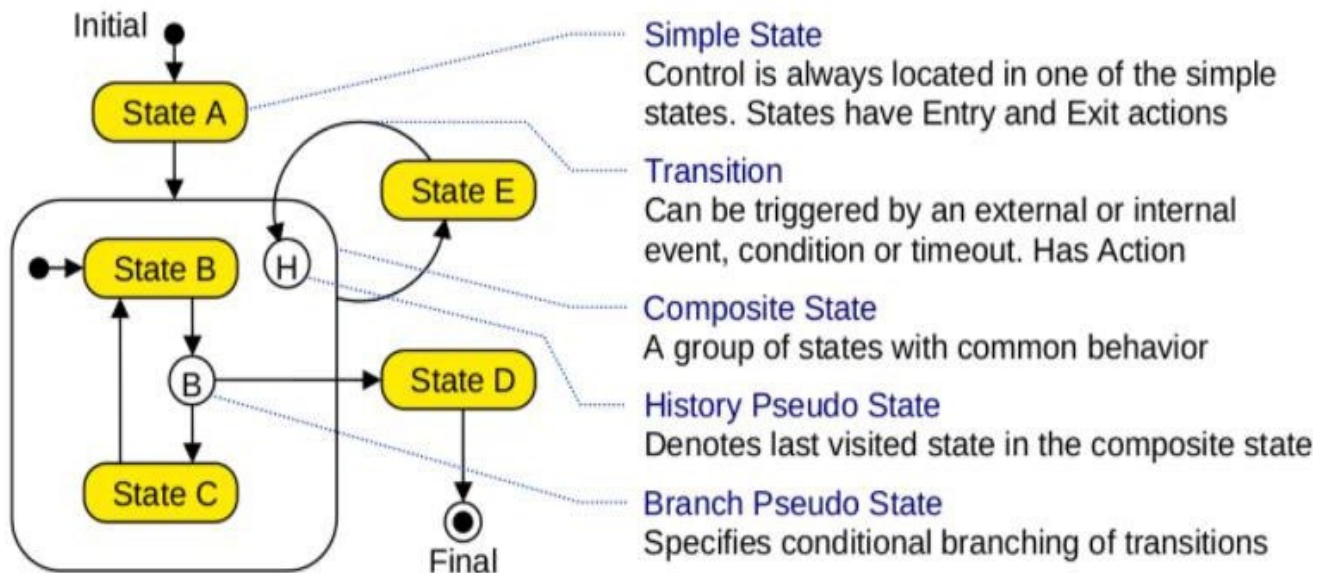


Рисунок 1.7 – Типова діаграма станів

#### 1.4 Висновки до розділу 1

У даному розділі розглянуто предметну область імітаційного моделювання, детально досліджено стан галузі на сьогоднішній день, її особливості, переваги та недоліки прикладного застосування, розглянуто найбільш поширені в інформаційних технологіях підходи до імітаційного моделювання.

Зокрема, було встановлено, що імітаційне моделювання дозволяє ефективно вирішувати різноманітні завдання в різних галузях, забезпечуючи можливість аналізу та оптимізації складних систем. Важливою перевагою є здатність враховувати велику кількість факторів та їх взаємодію в реальному часі, що робить цей метод ефективним у відтворенні реальних умов.

Однак, були виявлені певні виклики та недоліки. Серед них важливо відзначити складність математичних моделей, що може ускладнювати їх розробку, та велику обчислювальну потребу при роботі з великими обсягами даних.

Додатково, успішне використання імітаційного моделювання передбачає глибоке розуміння самої системи та правильний вибір параметрів для моделювання.

Усе враховуючи, можна зробити висновок, що імітаційне моделювання залишається важливим інструментом для оптимізації та управління процесами у різних галузях.

## 2 РОЗРОБКА МОДЕЛІ ІМІТАЦІЇ РОБОТИ КВИТКОВИХ КАС

### 2.1 Розробка діаграм взаємодій агентів системи

Опишемо процес роботи квиткової каси та побудуємо діаграму взаємодії окремих агентів системи.

Для виконання розробки необхідно:

- 1) вказати час початку симуляції та її тривалість;
- 2) та врахувати, що потік пасажирів, що прибувають, визначається відповідними розподілами (кривими прибуття), які залежать від типу поїзда та його розкладу. У цій реалізації для залізничних станцій кінцевий користувач має дві можливості:

«Середовище реального життя»: дані про прибуття пасажирів для кожного поїзда вводяться за допомогою списків, що містять кількість пасажирів, які прибувають щохвилини та купують квиток у квитковій касі. Прибуття віртуальних пасажирів для певного поїзда залежатиме від довжини наданого списку: він почнеться «довжиною списку» за кілька хвилин до часу відправлення (розрахунок, який виконується всередині). (Внутрішній) список відправляючих поїздів, які слід враховувати в симуляції (`TrainInput`), пізніше оновлюється користувачем за допомогою процедури `addTrain` один раз для кожного нового заявленого поїзду.

Якби програмне забезпечення використовувалося компанією залізничної інфраструктури на реальній станції, потрібно було б провести опитування, щоб визначити реальні криві прибуття на цю станцію, щоб скласти відповідні списки пасажирів, які купують квитки.

«Середовище тестування»: дані про прибуття пасажирів кожного поїзда (який купує квиток у квитковій касі) генеруються за допомогою статистичного розподілу, щоб визначити затримку кожного пасажирів відносно моменту, коли починається симуляція цього поїзда. Для цього за замовчуванням вибрано мультиноміальний розподіл з параметрами  $n$  і  $p$ . У цьому випадку  $p = (p_1, \dots, p_r)$

відповідає вибраним можливим затримкам, а  $n$  є кількістю пасажирів, змодельованих для цього поїзда. Цей вибір не ґрунтується на експериментах у реальному світі й має лише на меті легко перевірити пакет для перевірки та показати, як він працює

Значення, призначені за замовчуванням, становлять 30 хвилин наперед для поїздів приміського сполучення; 60 хвилин наперед для поїздів середнього та далекого сполучення; і за 90 хвилин для високошвидкісних поїздів (хоча це можна змінити, змінивши значення глобальної змінної `randomLength` у файлі `randomize.mpl`) (рис. 2.1).

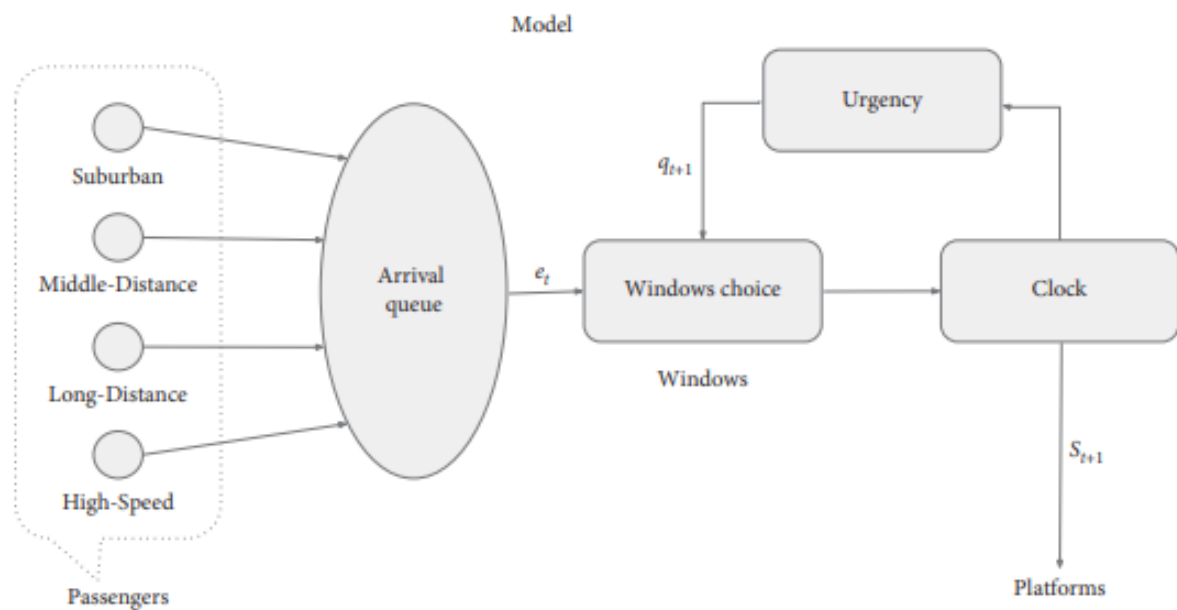


Рисунок 2.1 – Діаграма взаємодії агентів системи

3) Дані потягів вводяться залежно від вибраного середовища.

«Реальне середовище»: «номер» поїзда, його тип, час відправлення та список пасажирів, які купують квиток, щохвилини вводяться вручну для кожного поїзда (за допомогою процедури `addTrain`). Кількість пасажирів отримується з відповідного списку кроку 2.

«Середовище тестування»: дані для поїздів (час відправлення, кількість пасажирів, яка буде згенерована, і «номер» поїзда) вводяться за допомогою

чотирьох списків (SuburbanTrain, MDTrain, LDTrain і HSTrain), і процес виконується автоматично.

4) Має бути введено розподіл вікон і час обслуговування для кожного типу поїзда, а також час для перенаправлення пасажирів до черги в останню хвилину (якщо є). Змінні, у яких вони мають зберігатися, це приміські, середні, міжміські та високошвидкісні (якщо вікно є спільним, воно відображається в кількох із цих списків, і застосований середній час обслуговування залежатиме від типу пасажирів). Особливий випадок становлять вікна в останню хвилину (якщо є). «Хвилини до відправлення», необхідні для перерозподілу в такому вікні, зберігаються в глобальній змінній сигналізації, а вікна, які обслуговують цей тип пасажирів, повинні бути введені в список останніх хвилин (час обслуговування також залежатиме від типу пасажирів) і буде дано за ці вікна).

5) Кожну хвилину всі пасажирів, які прибувають, тимчасово зберігаються в «черзі прибуття» (віртуально неіснуюча єдина черга).

6) Щохвилини пасажирів в «черзі прибуття» переміщуються до найкращої з можливих черги вікон (найкоротшої) відповідно до їхніх характеристик (враховуючи, чи є вони пасажирів в останню хвилину, якщо таке вікно існує). Щоб досягти цього, список вікон скорочується для кожного пасажирів, і він/вона спрямовується до найкоротшого вікна у скороченому списку (у разі наявності більше ніж однієї черги однакової довжини, він/вона спрямовується до найкоротшого вікна у скороченому списку).

7) Кожне вікно в квитковій касі працює з певною швидкістю розвантаження (враховуючи середній час, який вимагає тип пасажирів), а також оновлюється щохвилини (виключаючи з кожної черги вікон уже обслугованих пасажирів). Таким чином, вихідний потік пасажирів із черг біля вікон складається з пасажирів, які вже мають квитки.

8) Щохвилини система шукає пасажирів в останню хвилину в чергах до вікон (якщо застосовно) і переміщує кожного до найкоротшої черги в останню хвилину (якщо і тільки якщо ця черга коротша за ту, в якій стоїть пасажир).

## 2.2 Розробка UML діаграми

Unified Modeling Language (UML) є стандартною мовою моделювання, що використовується для візуалізації, специфікації, конструювання та документування програмних систем. UML надає широкий набір діаграм, які допомагають розробникам аналізувати, проектувати та реалізовувати програмні системи.

Однією з основних груп діаграм UML є структурні діаграми, які використовуються для відображення статичної структури системи. Діаграма класів відображає структуру класів системи, з їх атрибутами, методами та взаємозв'язками. Діаграма компонентів допомагає відображати фізичну структуру системи та її компонентів. Діаграма пакетів використовується для групування елементів системи в пакети та відображення їх взаємозв'язків.

Щодо поведінкових діаграм, найпоширенішими є діаграми послідовності та діаграми станів. Діаграма послідовності показує взаємодії між об'єктами системи у вигляді послідовності повідомлень між ними. Вона дозволяє моделювати виконання операцій та обмін даними між об'єктами. Діаграма станів використовується для моделювання поведінки об'єкта та переходів між його станами. Вона відображає, як об'єкт реагує на події та змінює свій стан.

Крім того, UML надає діаграми взаємодії, такі як діаграми комунікації та діаграми співпраці, які дозволяють моделювати складні взаємодії між об'єктами системи. Діаграма розгортання допомагає відобразити фізичну архітектуру системи, відображаючи розміщення апаратного забезпечення та програмного забезпечення на різних вузлах системи.

Усі ці діаграми UML є важливими інструментами для аналізу, проектування та документування програмних систем. Вони допомагають розробникам встановлювати зв'язки між елементами системи, розуміти їх структуру та поведінку, а також передавати цю інформацію іншим учасникам проекту.



Завдяки стандартизованому підходу та гнучкості UML, розробники можуть використовувати ці діаграми на будь-якому етапі життєвого циклу розробки програмного забезпечення. Вони допомагають зрозуміти вимоги до системи, проектувати архітектуру, визначати поведінку та співпрацю об'єктів, а також тестувати та документувати систему.

Загальна мета використання UML-діаграм - полегшити розуміння та спілкування між розробниками, аналітиками, дизайнерами та іншими учасниками проекту. Вони дозволяють візуалізувати складність системи, виявляти проблеми та ризики, а також забезпечувати базовий фундамент для подальшого програмування та розробки.

У підсумку, UML-діаграми є потужним інструментом для моделювання та аналізу програмних систем. Вони допомагають розробникам уникнути помилок, покращити спілкування та зробити процес розробки більш ефективним. Використання UML дозволяє створювати якісні та документовані програмні системи, що відповідають вимогам замовників.

Відомі засоби розробки UML (Unified Modeling Language) є незамінними інструментами для створення, візуалізації та аналізу UML-діаграм. Ось кілька відомих засобів розробки UML та їх ключові особливості:

1. Enterprise Architect: Enterprise Architect є потужним засобом, який надає повну підтримку UML-діаграм та інших модельних підходів. Він має багатий функціонал, включаючи можливість моделювання різних видів діаграм (класів, послідовностей, станів тощо), відстеження залежностей, виконання аналізу моделей та генерацію коду.

2. Visual Paradigm: Visual Paradigm також є популярним інструментом для розробки UML-діаграм. Він надає широкі можливості для створення діаграм класів, послідовностей, станів, компонентів та багато інших. Засіб також підтримує важливі принципи, такі як перевикористання елементів моделі, генерацію коду з моделей та підтримку колективної роботи.

3. IBM Rational Rose: IBM Rational Rose є одним з перших і відомих засобів для розробки UML-діаграм. Він має широкий набір функціоналу для

моделювання, включаючи можливість створення діаграм класів, послідовностей, компонентів та інших. Засіб також надає підтримку для імпорту та експорту моделей, аналізу моделей та генерації коду.

Принципи розробки UML полягають у використанні стандартних нотацій, правил і конвенцій для побудови діаграм, щоб забезпечити їх зрозумілість і співпрацю між розробниками. Основні принципи UML включають:

1. Стандартизована нотація: UML використовує стандартизовану нотацію, яка дозволяє розробникам однозначно виразити структуру та поведінку системи. Це забезпечує зрозумілість та однозначність комунікації між учасниками проекту.

2. Модульність: UML дозволяє розбити систему на модулі та компоненти, що полегшує розробку, тестування та управління проектом. Кожна діаграма UML може фокусуватися на конкретному аспекті системи, що сприяє узгодженості та зрозумілості моделей.

3. Використання структурних та поведінкових діаграм: UML надає різноманітні типи діаграм, які охоплюють як структурний, так і поведінковий аспекти системи. Це дозволяє аналізувати взаємозв'язки між класами, компонентами та об'єктами, а також моделювати їх поведінку та взаємодію.

4. Розширюваність: UML дозволяє визначати власні профілі та розширювати базові нотації за допомогою стереотипів, тегів та обмежень. Це дозволяє враховувати специфічні вимоги та контекст проекту та адаптувати нотацію до потреб розробки.

5. Документація та аналіз: UML дозволяє створювати документацію системи, що охоплює її структуру, поведінку та взаємозв'язки. Це полегшує аналіз, сприяє зрозумілості системи та допомагає виявляти проблеми та покращувати якість програмного забезпечення.

6. Підтримка життєвого циклу: UML дозволяє моделювати систему на різних етапах життєвого циклу розробки, починаючи від аналізу та проектування до реалізації та тестування. Використання UML дозволяє зберігати та збирати інформацію про систему протягом усього її життєвого циклу, сприяючи її

ефективній розробці та підтримці.

7. Колективна робота: UML підтримує колективну роботу та співпрацю між учасниками проекту. Засоби розробки UML дозволяють спільно працювати над моделями, обмінюватись даними, відстежувати зміни та коментувати моделі. Це сприяє комунікації та співпраці в рамках розробки програмного забезпечення.

8. Гнучкість та розширюваність: UML надає гнучкість для моделювання різних типів систем, включаючи програмні системи, бізнес-процеси та архітектуру. Він може бути використаний в різних галузях та проектах, дозволяючи розширити його функціонал та адаптувати до конкретних потреб розробки.

У використанні UML (Unified Modeling Language) в задачах оптимізації маршрутів можна побачити значний потенціал для покращення ефективності та продуктивності системи маршрутизації. UML надає зручні та потужні засоби моделювання, які допомагають аналізувати, проектувати та оптимізувати складні маршрутні системи з різними обмеженнями та умовами.

Одним з ключових аспектів використання UML в оптимізації маршрутів є можливість визначати та моделювати різні види об'єктів та їх взаємозв'язки. Наприклад, за допомогою діаграм класів можна визначити основні сутності, такі як точки початку та закінчення маршруту, проміжні вузли, транспортні засоби, попутні вантажі та обмеження, які пов'язані з кожним об'єктом. Це дозволяє зрозуміти структуру системи маршрутизації та ідентифікувати основні елементи, які можна оптимізувати.

Діаграми послідовностей та діаграми станів UML також знаходять застосування в оптимізації маршрутів. Вони дозволяють моделювати поведінку системи та взаємодію між об'єктами. Наприклад, на діаграмі послідовностей можна зобразити послідовність дій та взаємодію між водієм, клієнтом та системою маршрутизації, що дозволяє з'ясувати етапи виконання маршруту та визначити можливі обмеження часу, відстані та інші фактори.

Процес перетворення на основі ATL використовує концепцію метамоделі як для вхідних, так і для вихідних моделей. Метамодель — це «модель моделі»,

а створення метамоделі називається метамодельюванням. Метамодель розглядається як абстрактне уявлення про фактичну модель або також розглядається як правила розробки та обмеження при створенні відповідної моделі. Метамодель може представляти різні елементи та зв'язки між елементами. Розробка відповідної метамоделі є однією з вимог реалізації на основі ATL і, отже, основою для розробки процесу трансформації. Отже, як частину першої мети та в процесі реалізації правил перетворення ми представляємо метамодель діаграми послідовності та метамодель PNMML для моделей введення та виходу відповідно. Опис метамоделі діаграми послідовності та метамоделі PNMML такий:

Метамодель діаграми послідовності розроблена на основі стандартної метамоделі, наданої OMG [16]. На рисунку 1 представлена метамодель діаграми послідовності, яка відповідає метамоделі OMG. Представлена метамодель відрізняється від стандарту OMG, оскільки класи в метамоделі діаграми послідовності розширені моделюванням і аналізом атрибутів систем реального часу та вбудованих систем (MARTE) [17] для представлення різних вимог системи реального часу. Він пропонує специфікації для моделювання базових аспектів, що використовуються в реальному часі та вбудованих доменах, щоб зробити можливим детальний опис програмних і апаратних платформ виконання, а також обробляє модельний аналіз продуктивності системи. Таким чином, використання специфікацій або атрибутів MARTE в метамоделі діаграми послідовності UML дозволить системним дизайнерам і розробникам розширювати або створювати моделі UML з атрибутами MARTE.

На діаграмі послідовності кореневий елемент представляє взаємодію, яка складається з елементів діаграми послідовності Lifeline, Message і InteractionFragment (рис. 2.2). Lifeline представляє об'єкт або ролі, які викликаються в змодельованому системному сценарії. В основному взаємодія відбувається між різними лініями життя шляхом обміну повідомленнями. Повідомлення — це зв'язок між відправником і одержувачем і включає події sendEvent і receiveEvent, які показані як два кінці повідомлення на рисунку 1.

Lifeline розширено ресурсом у метамоделі, який представляє системні ресурси, які використовуватимуться об'єктом. Таким чином, у метамоделі діаграми послідовності створюється окремий клас для представлення набору ресурсів, який недоступний у метамоделі, наданій OMG. Наразі клас представляє лише апаратні ресурси, пізніше його також можна розширити для програмних ресурсів.

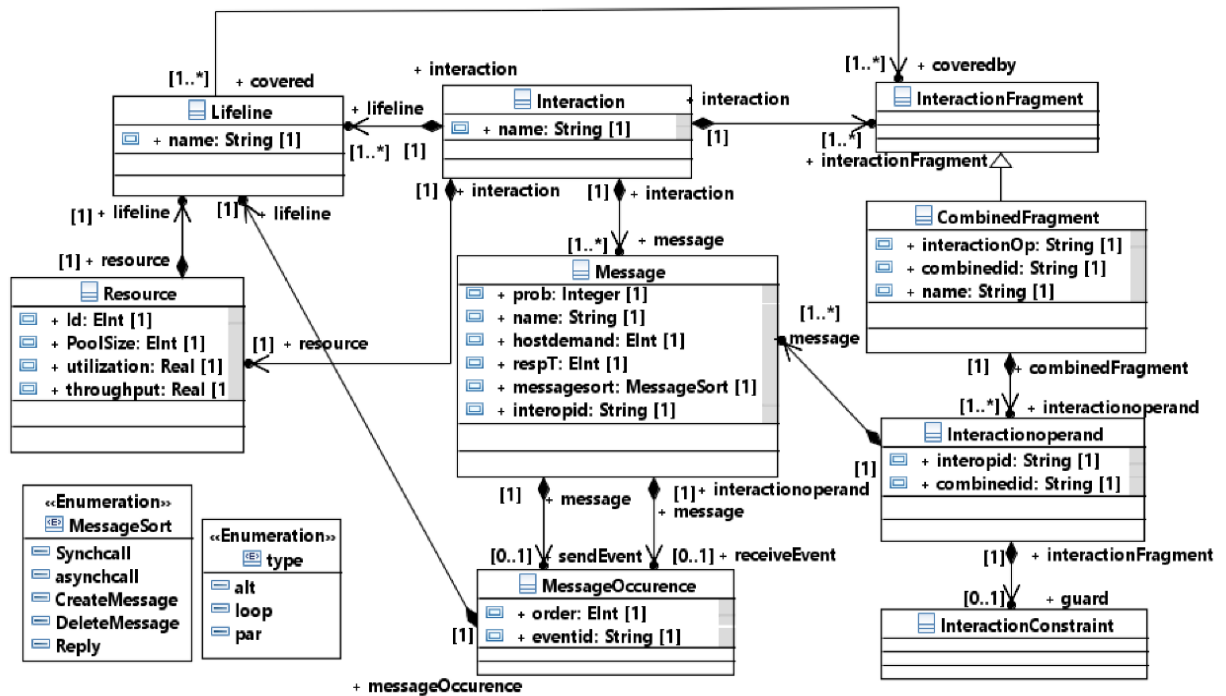


Рисунок 2.2 – Метамодель UML діаграми послідовностей

Упорядкований набір комбінованих фрагментів може охоплювати лінію життя, і кожен комбінований фрагмент представляє кілька форм потоку керування, а також включає один або два операнди взаємодії. Операнд взаємодії складається з обмеження взаємодії у формі захисника, який використовується як обмеження для потоку керування. Оператор взаємодії визначить тип комбінованого фрагмента. Існують різні типи комбінованого фрагмента, такі як альтернатива (`alt`), опція (`opt`), паралель (`par`) і цикл, які можуть моделювати задані конструкції, виражені у вимогах. Метамодель діаграми послідовності представляє комбінований фрагмент, операнд взаємодії та обмеження взаємодії

за допомогою класів `CombinedFragments`, `InteractionOperand` та `InteractionConstraint` відповідно. У метамоделі діаграми послідовності зв'язок між різними класами визначається за допомогою послідовності символів «0..\*» або «1..\*» на обох кінцях. Ця специфікація є множинністю, яка говорить про прийнятну кількість асоціацій, у яких може брати участь об'єкт класу. Символ «\*» вказує на безкінечність, «0..\*» вказує на відсутність обмежень на кількість асоціацій, а «1..\*» представляє одну чи більше кількості асоціацій.

### 2.3 Розробка метамоделі PNML

`Petri Net Markup Language` — це стандартний формат обміну, опублікований 11 листопада 2009 року для моделей та розробників мереж Петрі в спільноті `Petri Nets` [18]. Моделі мережі Петрі розробляються в різних країнах за допомогою різних інструментів. Стандартний формат для обміну моделями мережі Петрі був потрібен, щоб полегшити користувачам інструменту `Petri Net` з різних місць обмінюватися та використовувати переваги останніх і нових можливостей інших інструментів, наприклад, для аналізу, моделювання або реалізації [19]. `PNML` є гнучким для різноманітних мереж Петрі, а також для різних типів інструментів для мереж Петрі. Такі роботи, як [18] і [19], пропонують узагальнену метамодель `PNML` і додаткові класи, пов'язані зі структурованими концепціями `PNML`. Ми модифікували це узагальнення за допомогою спеціальних класів для графічної інформації, таких як графічний клас, що складається з двох додаткових класів, зміщення та позиції та атрибутів елементів мережі Петрі з метамоделлю `PNML`. Це призвело до спрощення реалізації правил перетворення для відображення атрибутів мережі Петрі, а також до включення графічної інформації. На рисунку 2.3 представлена метамодель `PNML`, що зображує різні частини `PNML` та їх зв'язок.

Кореневий вузол `pnml` на рисунку 3.2 представляє документ `PNML` мережі Петрі. `Pnml` розширено класом мережі для типу мережі Петрі. Основними елементами мережі Петрі є місце, перехід і дуга. Кожен з елементів мережі Петрі

представлено в метамоделі PNML відповідними класами, як показано на рисунку 2.3. Класи place, переход і дуга успадковуються від об'єкта базового класу. Позицією кожного об'єкта в документі pnml керує графічний клас, який, у свою чергу, складається з двох додаткових класів offset і position для представлення відносного положення та фактичного положення кожного об'єкта відповідно.

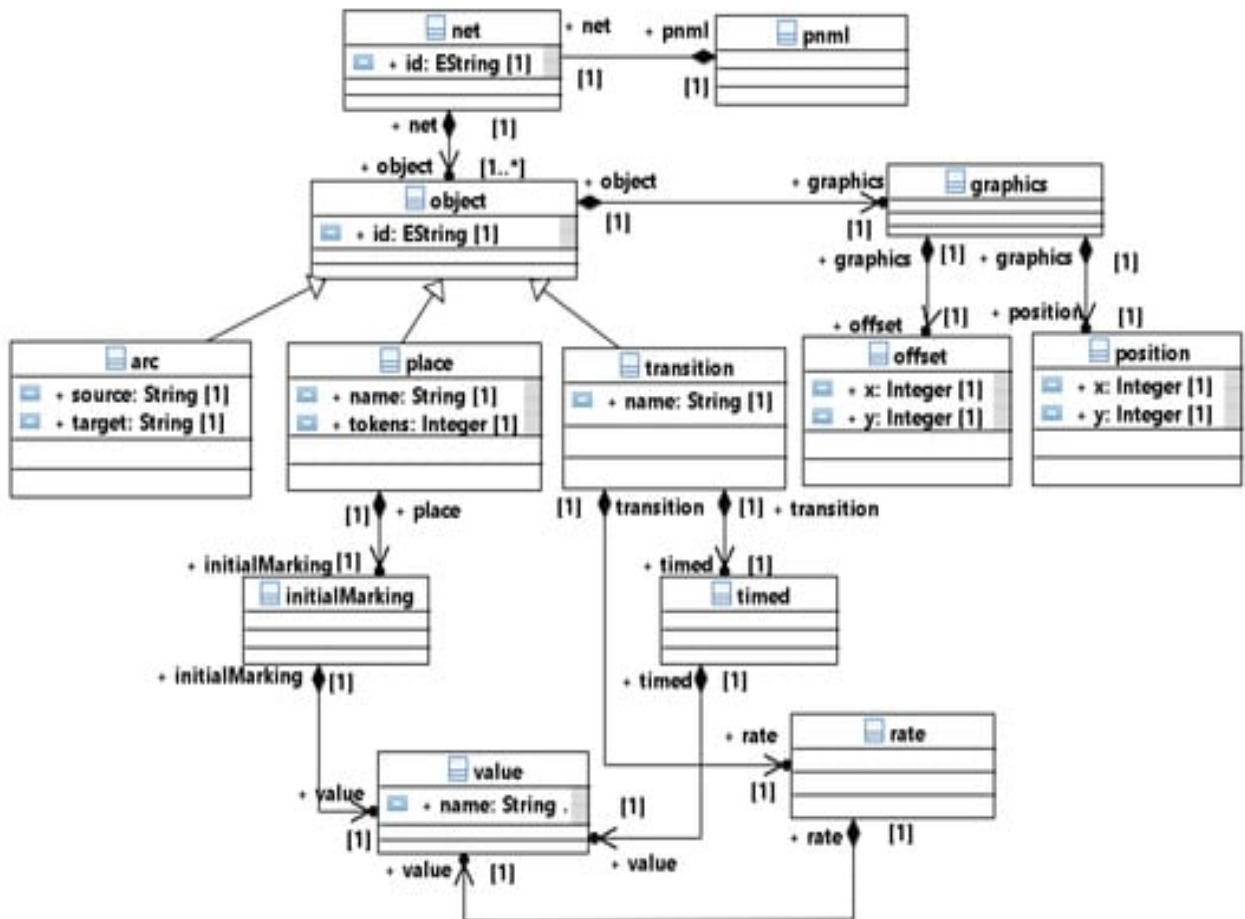


Рисунок 2.3 – Метамодель PNML (модифікована UML діаграма класів)

Клас метамоделі PNML initialMarking представляє інформацію маркера для місця мережі Петрі, а клас timed використовується для представлення двох типів переходів «за часом» і «негайного» з використанням різних значень у документі PNML. Значення класу є загальним класом, який використовується для представлення значення кількості токенів, типу переходу та назви місця. Швидкість класу використовується для представлення швидкості переходів і, отже, пов'язана з класом переходу.

Запропонований метод базується на метамоделі та реалізується у два основних етапи, перший з яких полягає у розробці метамоделі діаграми послідовності на основі UML/MARTE та метамоделі PNML, а наступним кроком є розробка та реалізація правил перетворення та автоматизація відображення анотованої послідовності UML/MARTE діаграму в модель продуктивності мережі Петрі на основі PNML. Діаграма послідовності UML/MARTE і модель мережі Петрі повинні відповідати метамоделі діаграми послідовності та метамоделі PNML відповідно. У літературі існує метамодель як для діаграми послідовності, так і для PNML, але в цій статті ми представляємо спрощену версію метамodelей. Запропоновані метамоделі складаються з мінімальних і всіх основних понять, необхідних для цілей моделювання та аналізу. Запропоновані метамоделі розроблено після кількох модифікацій і виконання запропонованого методу для генерації моделі мережі Петрі з діаграми послідовності UML.

Моделі та метамоделі широко використовуються та застосовуються в підходах на основі MDE. Модель — це демонстрація системи, вираженої за допомогою мови моделювання, а метамодель — це концептуальна основа мови моделювання. Діаграми послідовності анотовані інформацією, пов'язаною з продуктивністю, і цю анотовану інформацію можна автоматично витягти з діаграми послідовності системи для аналізу продуктивності. У запропонованій методології останній крок реалізовано за допомогою однієї з поширених мов перетворення ATL.

Метамоделі, описані в попередньому розділі, розроблені як діаграма класів UML у Eclipse modeling Framework [20], за якою слідує ряд кроків для відображення елементів діаграми послідовності UML/MARTE на елементи мережі Петрі за допомогою PNML. Процес перетворення починається з розгляду формату XML діаграми послідовності UML. XML Metadata Interchange (XMI) — це стандартна нотація на основі XML, яка використовується для представлення моделі UML. Основним використанням XMI є можливість обміну метаданими в різних інструментах моделювання. Eclipse Modeling Framework (EMF) і модуль моделювання Papyrus використовуються для створення діаграми послідовності



UML/MARTE. В інструменті Eclipse формат XMI автоматично створюється для моделей UML, і цей формат використовується для збереження/вилучення інформації про метамодель UML.

Запропонована методологія відображення діаграми послідовності в мережі Петрі показана на рисунку 2.4. Запропонований метод відрізняється тим, що він генерує стандартний проміжний формат у формі PNML з моделей UML і дозволяє користувачам мережі Петрі обмінюватися інформацією між різними мережами Петрі. інструменти. Як правило, у спільноті мережі Петрі різні користувачі використовують різні інструменти мережі Петрі, і кожен інструмент може мати власне представництво, що ускладнює обмін мережами Петрі між інструментами. Таким чином, PNML використовується як стандартна структура для експорту та імпорту подання мережі Петрі за допомогою інструментів мережі Петрі. Даний рисунок можна вважати модифікованою діаграмою компонентів системи.

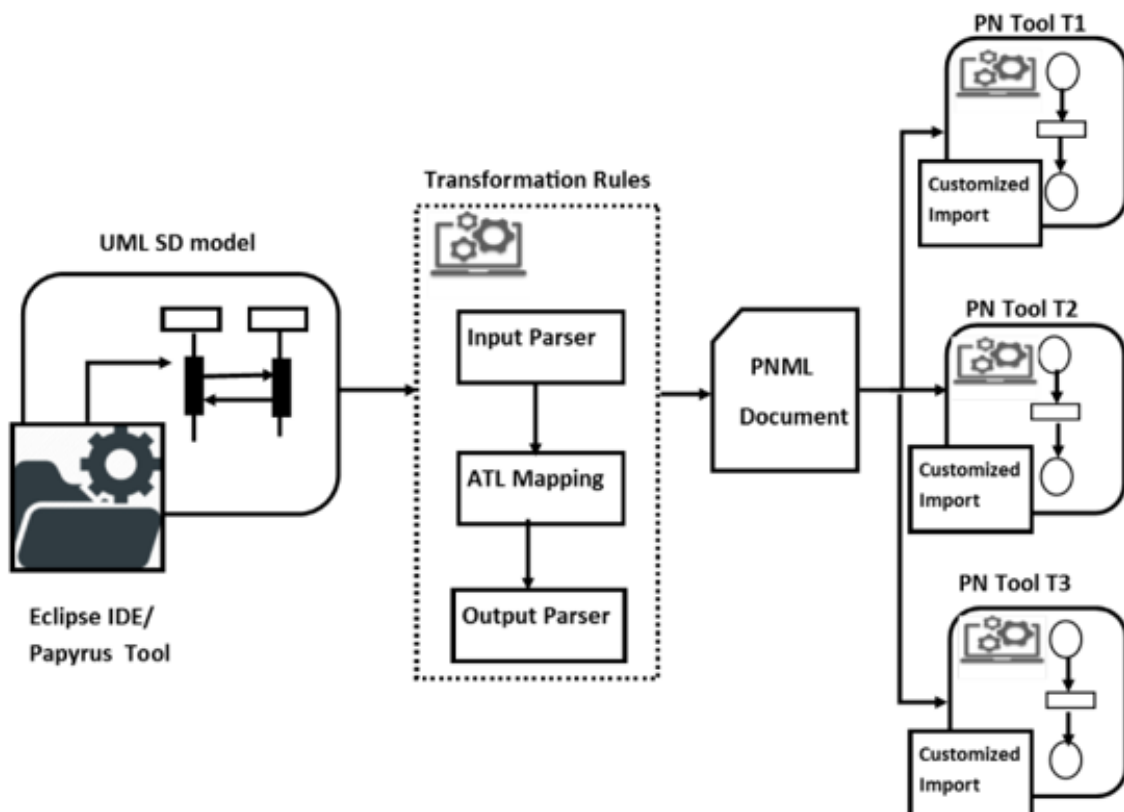


Рисунок 2.4 – Діаграма компонентів

## 2.4 Розробка імітаційної моделі роботи квиткових кас

Сформулювавши задачу, можна описати модель і динаміку квиткової каси великого вокзалу.

Розглядаються наступні елементи (обчислювальні деталі вхідних даних детально описано в прикладах нижче):

Пасажири: внутрішньо описаний списком у форматі

[тип і номер поїзда, час прибуття, штат] Такий список автоматично формується пакетом для кожного прибулого пасажира. «Час прибуття» відповідає

до прибуття пасажира на залізничний вокзал. The

стан — час, необхідний (або залишився) для отримання квитка.

Потяг: внутрішньо описаний списком у форматі

[час відправлення, кількість пасажирів, номер поїзда]

Списки розповсюдження та обслуговування Windows: кінцевий користувач має ввести список для кожного типу поїзда, що обслуговується, у форматі:

[набір ідентифікаторів вікон, середній час обслуговування]

Вікно: внутрішній список пасажирів (очікують у черзі цього вікна квиткової каси).

Змінна часу позначається як  $t$ .

$e(t)$  – віртуальна «черга прибуття» пасажирів, тобто список усіх пасажирів, які вже прибули на час  $t$  (з їх характеристиками), побудований відповідно до кривих прибуття кожного поїзда.

$q(t)$  – це «стан вікон»: це список черг перед вікнами в момент часу  $t$ . Зверніть увагу, що через дискретний характер процесу пасажири в цьому стані чекали принаймні один часовий крок у черзі прибуття, тобто залишалися в  $e(t-1)$ , перш ніж перейти до  $q(t)$ .

Вибір вікна: пасажирів в  $e(t)$  обирають (один за одним) вікно з найкоротшою чергою в  $q(t)$  (з тих, що йому підходять). Вузол повертає оновлений список вікон, які увійдуть до годинника переходу.

$s(t)$  — список пасажирів, які вже мають квитки (і можуть перейти на відповідні платформи).

Годинник: він пересуває  $e$ ,  $q$  і  $s$  на один часовий крок (1 хвилина за замовчуванням):

Оновлення  $e(t)$  до  $e(t + 1)$ , шляхом стирання тих пасажирів, які рухаються до черг біля вікон, і додавання тих пасажирів, які прибувають на станцію

Оновлення  $q(t)$  до  $q(t + 1)$ , видалення з кожної черги тих пасажирів, які отримали квитки, і додавання тих, хто прийшов із черги прибуття

Оновлення  $s(t)$  до  $s(t + 1)$ , список, що включає пасажирів, які отримали квитки на хвилину  $t$

Терміновість: ця позначка дозволяє пасажирам, які потрапляють у категорію «останньої хвилини», вибрати це вікно (якщо воно існує).

Можливе існування вікон в останню хвилину робить вибір вікна більш складним (рис. 2.5). Вибір пасажирів в останню хвилину дає пріоритет часу відправлення над типом поїзда. Вони якимось чином порушують порядок основної черги.

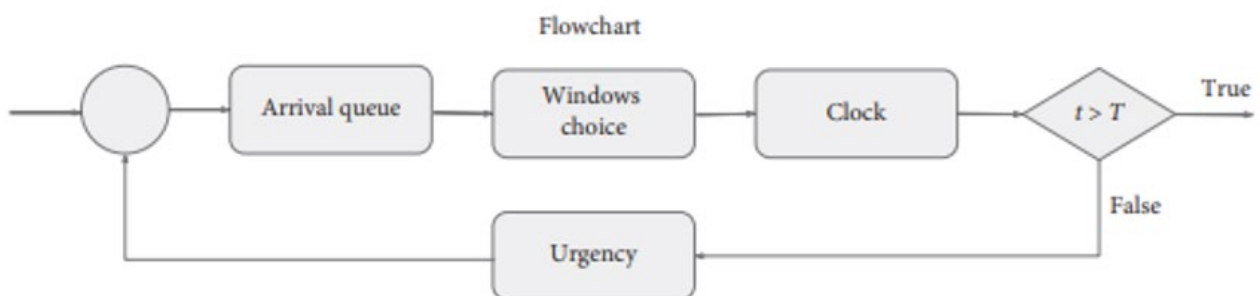


Рисунок 2.5 – Діаграма взаємодії черги пасажирів

## 2.5 Висновки до розділу 2

У даному розділі було розглянуто специфіку роботи та проблеми сучасних квиткових кас, розроблено імітаційну модель та діаграми взаємодії для проектування зручної віртуальної каси.

Створення імітаційної моделі та діаграм взаємодії є важливим етапом у проектуванні віртуальної каси. Ці інструменти дозволяють не лише відобразити поточні процеси, але й виявити можливі шляхи їх оптимізації. Важливо відзначити, що імітаційна модель може служити ефективним інструментом для передбачення та вирішення проблем, які можуть виникнути в ході роботи віртуальної каси.

Додатково, використання імітаційного моделювання у проектуванні квиткових кас дозволяє виявити оптимальні рішення та забезпечити максимальну зручність для користувачів. Це може включати в себе швидше обслуговування, зменшення черг, покращення системи оплати та інші позитивні аспекти.

## 3 РЕАЛІЗАЦІЯ ІМІТАЦІЙНОЇ МОДЕЛІ РОБОТИ КВИТКОВИХ КАС

### 3.1 Особливості побудови імітаційних моделей

Одним із основних доступних методів моделювання є моделювання дискретних подій (DES), яке фокусується на самому процесі та моделює його. DES базується на концепції зв'язків і ресурсів для опису їх потоку та спільного використання в системі [21]. Сутності є пасивними об'єктами (до них не прив'язано жодної взаємодії чи характеристик), і вони подорожують робочим процесом, де обробляються, затримуються, ставляться в чергу, захоплюються та розподіляються. Першим відомим інструментом моделювання будівництва з використанням DES була CYCLic Operations Network (CYCLONE) Хелпіна, розроблена в 1973 році, яка мала бути системою моделювання загального призначення (AbouRizk та ін., 2011). Мартінез (2010) описав методологію проведення DES і вказав на можливі проблеми, з якими можна зіткнутися під час моделювання, які можуть поставити під загрозу валідність моделі. Тривалість діяльності в моделях DES можна описати за допомогою функцій розподілу ймовірностей, таких як ті, що використовуються в техніці оцінки та перегляду програм (PERT). Ло (2015) спробував зібрати всі доступні функції (від уніфікованих до Вейбулла, включаючи системи Джонсона та Пірсона) з їхніми властивостями та пояснив їх використання у випадку моделювання. AbouRizk і Halpin (1992, стор. 537) припускають, що гнучкі функції необхідні через «диверсифікований характер даних про тривалість будівництва» і радять використовувати бета-функцію через її звичність у сфері будівництва.

Хайду та Бокор (2016) стверджують, що ретельна трибальна оцінка важливіша, ніж тип обраної функції розподілу. Моделювання методом Монте-Карло, виконане на гіпотетичних і реальних проектах, показало, що 10% різниця в оцінці за трьома точками викликає більші відхилення, ніж вибрані розподіли (Hajdu and Bokor, 2016). Іншим основним методом моделювання є системна

динаміка (SD), оскільки розроблений Форрестером (1961). SD – це метод «зверху вниз», який зосереджується на різних факторах впливу та взаємозв'язках між ними, щоб показати роботу та поведінку всієї системи за допомогою циклів зворотного зв'язку. SD можна використовувати як для якісного, так і для кількісного моделювання: перше зосереджується на створенні діаграми причинно-наслідкових зв'язків (врівноваження та підсилення зв'язків), тоді як друге визначає запаси та потоки та виражає зв'язки за допомогою рівнянь (Kunc, 2017). SD — це модель, яка працює з агрегатами, тобто елементи в одному запасі вважаються однаковими, а система визначається як набір структурних залежностей. Модеслі та Аль-Джібурі (2009) використовували SD для визначення сфер, які керівництво має вдосконалити для підвищення продуктивності. Модель містила планування, контроль, мотивацію, безпеку та зриви як найбільш значущі фактори. Було перевірено кілька стратегій, і було виявлено, що перші два фактори потребують особливої уваги керівництва (Mawdesley and Al-Jibouri, 2009). На відміну від SD, агентне моделювання (ABM) має підхід «знизу вгору» – існує відсутність глобальної поведінки системи.

Поведінка системи залежить від того, як окремі гетерогенні агенти взаємодіють один з одним і з їхнім середовищем на основі визначених правил. Зіберс та ін. (2010) стверджують, що ABM має перевагу перед DES у випадках, коли фокус зосереджується не на процесі, а на тому, як окремі агенти, які можуть навчатися та адаптуватися, впливають на систему. Сон та ін. (2015) підкреслюють подібні позитивні властивості через приклади проектних команд у великомасштабних будівельних проектах. Вони рекомендують ABM для моделювання, наприклад, міжнародного будівельного ринку з країнами та фірмами як агентами (Son et al., 2015). Sawhney та ін. (2003) радять використовувати ПРО для підвищення безпеки будівництва на місці шляхом моделювання будівельного середовища, працівників з різними рівнями толерантності до ризику (агентів) і методів управління безпекою. Watkins та ін. (2009) використовували ABM, щоб визначити, як перевантаженість сайту впливає на продуктивність із визначенням двох типів агентів: робітники (зі

змінними, такими як рівень кваліфікації) і діяльність. Дабірян та ін. (2016) застосували ті самі два агенти, щоб краще оцінити витрати на робочу силу. Хсу та ін. (2016) використовували АВМ для оцінки моделей відбору членів команди. У їхніх дослідженнях агентами були працівники з такими атрибутами, як досвід і навички. Було зроблено висновок, що вибір на основі взаємозалежності є кращим, ніж призначення на основі навичок (Hsu et al., 2016).

Підходи, описані тут, часто використовуються окремо, але також можуть застосовуватися в поєднанні. Перевага комбінованого підходу полягає в тому, що можна використати різноманітні переваги кожного методу та збалансувати їхні недоліки. Для кожного компонента моделі слід вибрати найбільш підходящий підхід, і, залежно від питання, на яке потрібно відповісти, такі комбінації забезпечать точніші представлення реальності.

Бокор та ін., Огляд підходів моделювання будівництва 1855 Для цих комбінованих підходів існують різні назви, включаючи «гібридне», «багатометодне» та «багатопарадигмальне» моделювання (Mustafee та ін., 2015) [22]. Мостерман (1999) визначив поєднання дискретного та безперервного моделювання як «гібридне моделювання». Балабан та ін. (2014) стверджують, що ПРО не можна вважати парадигмою; отже, ті підходи, де АВМ поєднується з іншим методом, не можна назвати мультипарадигмальним. Згідно з ними, також існує відмінність між змішаними/гібридними та мультиметодами (Balaban et al., 2014). Обидва Mustafee та ін. (2015) і Balaban et al. (2014) погоджуються, що необхідні правильні визначення. Крім того, три основні методи моделювання можна змішувати з іншими методами, такими як нейронні мережі (NN) або нечітка логіка (FL) (Balaban et al., 2014), які також можуть бути розглянуті гібридні підходи (AbouRizk, 2010; Nojedehi and Nasirzadeh, 2017). Однак у цьому документі термін «гібридне моделювання» відноситься до будь-якого методу, в якому базовий метод моделювання поєднується з іншим базовим методом моделювання або FL.

Після вибору найбільш підходящої структури необхідно визначити точки взаємодії між компонентами. Ці змінні інтерфейсу можуть впливати на змінні в

іншому компоненті. Створення гібридних моделей надає можливість мати динамічні змінні, які інакше були б статичними за допомогою базового методу моделювання (Alvanchi et al., 2011). Крім того, застосування гібридного підходу може означати поєднання безперервного (наприклад, SD) і дискретного (наприклад, DES) методу, що означає, що має бути визначено часове випередження (Alvanchi та ін., 2011; Alzraiee та ін., 2012). ). Тому важливо знати, як можуть змінюватися взаємодіючі змінні через зв'язування компонентів гібридної системи. За даними Alvanchi et al. (2011), існує п'ять типів взаємодії: У випадку однієї дискретної змінної та однієї безперервної змінної; дискретна зміна дискретної змінної спричиняє дискретну зміну безперервної; дискретна зміна дискретної змінної змінна викликає зміну функціонального опису безперервної; безперервна зміна безперервної змінної викликає дискретну зміну дискретної. У випадку двох безперервних змінних, безперервна зміна однієї безперервної змінної викликає безперервну зміну іншої. У випадку двох дискретних змінних змінних, дискретна зміна однієї дискретної змінної викликає дискретну зміну іншої (рис. 3.1).

### **3.2 Реалізація імітаційної моделі в середовищі AnyLogic**

AnyLogic - це потужний інструмент для імітаційного моделювання та аналізу систем, який вирізняється своєю виразністю та гнучкістю. Він дозволяє створювати детальні та інтерактивні моделі для вивчення та оптимізації різноманітних процесів у виробництві, логістиці, транспорті, аеропортах, ланцюгах постачання, медицині та інших галузях.

Основні особливості AnyLogic включають в себе можливість використання різних методів моделювання, таких як багатоагентне моделювання, моделювання дискретних подій і системна динаміка. Це забезпечує гнучкість та широкі можливості для дослідження різних аспектів систем та їх взаємодій.



AnyLogic (рис. 3.1). також славиться своєю візуалізацією, дозволяючи користувачам створювати зрозумілі та ефективні візуальні моделі. Інтеграція з Java робить його гнучким та розширюваним інструментом, що дозволяє розробникам використовувати власні компоненти та бібліотеки.



Рисунок 3.1 – Лого AnyLogic

Цей інструмент вирішує реальні завдання у різних галузях, надаючи можливість аналізу та оптимізації великих та складних систем, і допомагає підприємствам та дослідникам зробити осмислений вибір для оптимізації їх процесів.

Пішохідна бібліотека AnyLogic призначена для імітації пішохідних потоків у «фізичному» середовищі. Дозволяє створювати моделі пішохідних будівель (наприклад, станції метро, перевірки безпеки тощо) або вулиць (велика кількість пішоходів).

Бібліотека Pedestrian Library дозволяє створювати гнучкі моделі, збирати базову та розширену статистику та ефективно візуалізувати процес, який ви моделюєте, щоб перевірити та представити свою модель. Ви можете збирати статистичні дані щодо щільності пішоходів у різних зонах, щоб забезпечити прийнятну роботу точок обслуговування з гіпотетичним навантаженням, оцінити тривалість перебування в конкретних зонах, виявити потенційні проблеми з внутрішньою геометрією — ефект додавання перешкод і багато інших застосувань. У моделях, створених за допомогою Pedestrian Library,

пішоходи рухаються в безперервному просторі, реагуючи на різні типи перешкод (стіни, різні види зон) та інших пішоходів.

Основні можливості Pedestrian Library:

- Створення моделей пішохідних об'єктів: Бібліотека надає засоби для моделювання пішохідних об'єктів в різноманітних середовищах, таких як станції метро, зони безпеки, вулиці тощо.

- Гнучкість та адаптивність: Pedestrian Library дозволяє створювати гнучкі моделі, які можна легко адаптувати до різних сценаріїв. Це включає в себе реагування на перешкоди, різні зони та взаємодію з іншими пішоходами.

- Статистичний аналіз: Ви можете збирати різні статистичні дані, такі як щільність пішоходів у різних зонах, тривалість перебування в конкретних місцях та інші параметри. Це надає можливість ефективно оцінювати працездатність точок обслуговування під гіпотетичним навантаженням.

- Взаємодія з перешкодами: Моделі, створені за допомогою Pedestrian Library, враховують різні типи перешкод, такі як стіни чи різні зони, і пішоходи взаємодіють із ними в реальному часі.

- Анімація та візуалізація: Зручна візуалізація процесів у моделях, реалізованих за допомогою Pedestrian Library, дозволяє не лише перевіряти, але і ілюструвати розроблені сценарії.

Обмеження та перспективи:

Однак важливо враховувати, що при використанні імітаційних бібліотек, також можуть існувати обмеження, такі як обчислювальні можливості та точність результатів. Тим не менш, Pedestrian Library залишається потужним інструментом для моделювання пішохідних потоків з багатьма практичними застосуваннями.

Дана бібліотека вдало моделює і процес проходження пасажирів через квиткову касу (рис. 3.2).

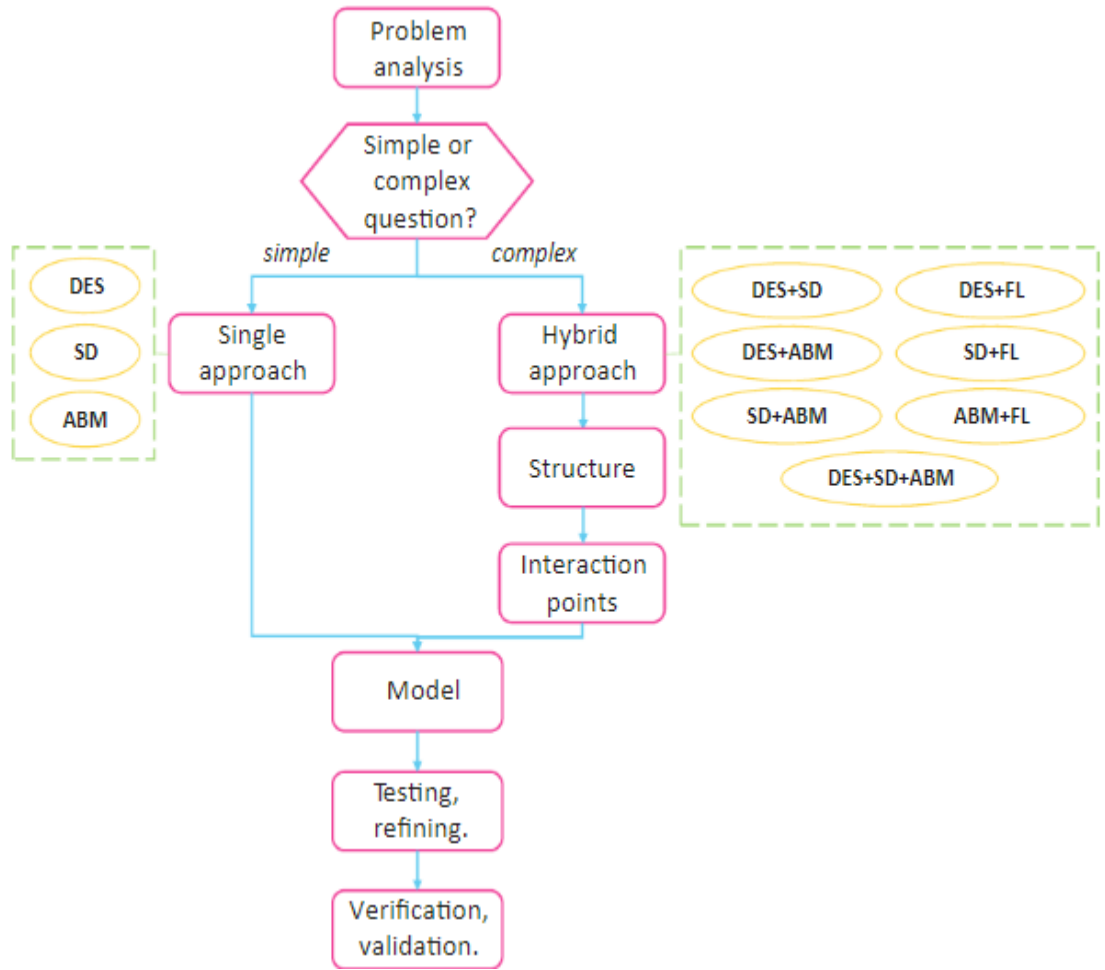


Рисунок 3.2 – Схема побудови імітаційної моделі

На рисунку 3.3 зображено побудову вокзалу із квитковими касами за допомогою графічних примітивів AnyLogic.

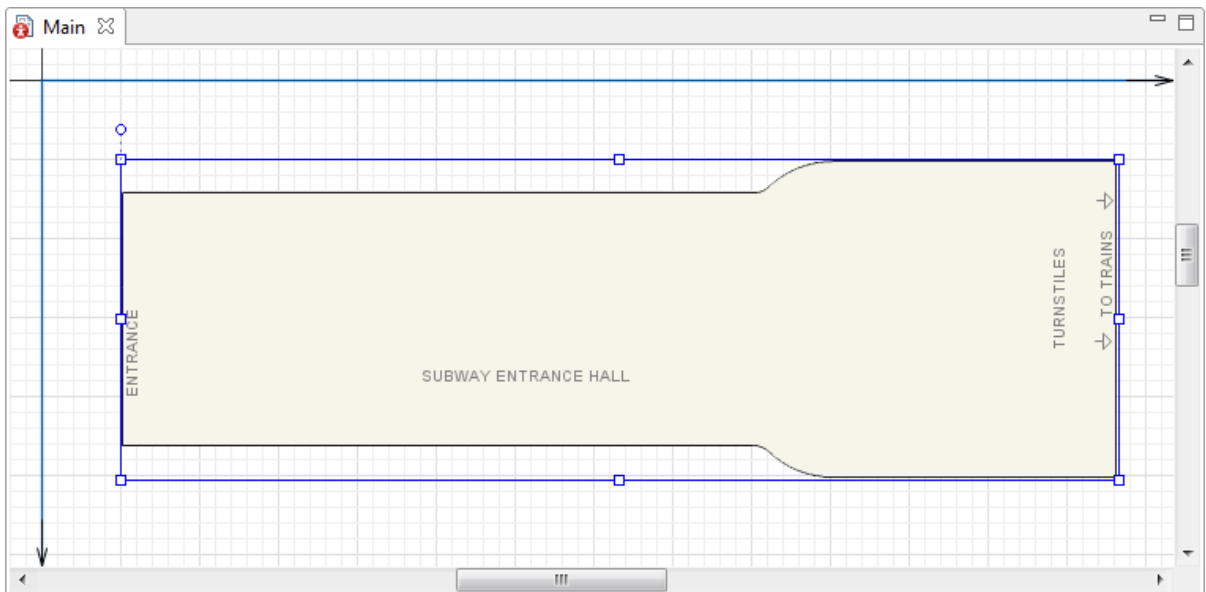


Рисунок 3.3 – Процес побудови вокзалу

Ми завершимо створення простої моделі, що моделює пасажиропотік і визначимо процес пішохідного потоку за допомогою блок-схеми, складеної з блоків пішохідної бібліотеки (рис. 3.4).

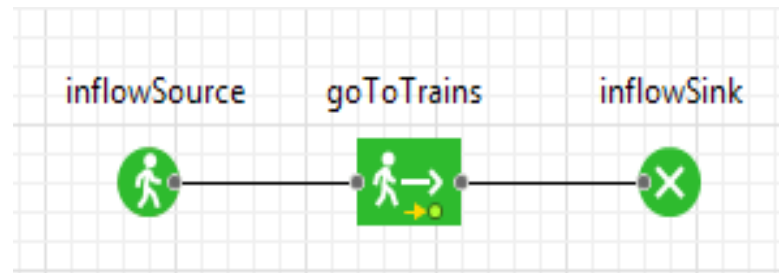


Рисунок 3.4 – Процес побудови агентів та окремих елементів діаграми взаємодії

Потрібно починати з дуже простого процесу: пасажир заходить на станцію метро (саме там, де ми поставили лінію входу), а потім переходять до потягів (до нашої цільової лінії).

В AnyLogic є можливість створити блок-схеми, додаючи блоки з бібліотечної палітри до графічної діаграми, встановлюючи спеціальні властивості для блоків і з'єднуючи блоки разом.

Надалі потрібно буде створити карту щільності розподілу. Елемент виконує роль колірної схеми. Під час виконання буде показано легенду карти щільності. Ця легенда карти щільності допомагає зрозуміти, які кольори на карті щільності означають які значення. Легенда відображає відповідність між значеннями щільності та кольорами на карті щільності.

Карта щільності постійно перефарбовується відповідно до фактичних значень: коли щільність змінюється в якійсь точці, колір динамічно змінюється, щоб відобразити цю зміну. При нульовій щільності ділянка не забарвлюється взагалі.

Карта щільності зазвичай використовується для визначення областей з критичною щільністю. Червоним кольором позначена критична щільність пасажирів в черзі. За замовчуванням AnyLogic використовує логарифмічну схему кольорів. У цьому випадку колір змінюється логарифмічно від

«мінімального» (синього) кольору до «максимального» (червоного) кольору (рис. 3.5).

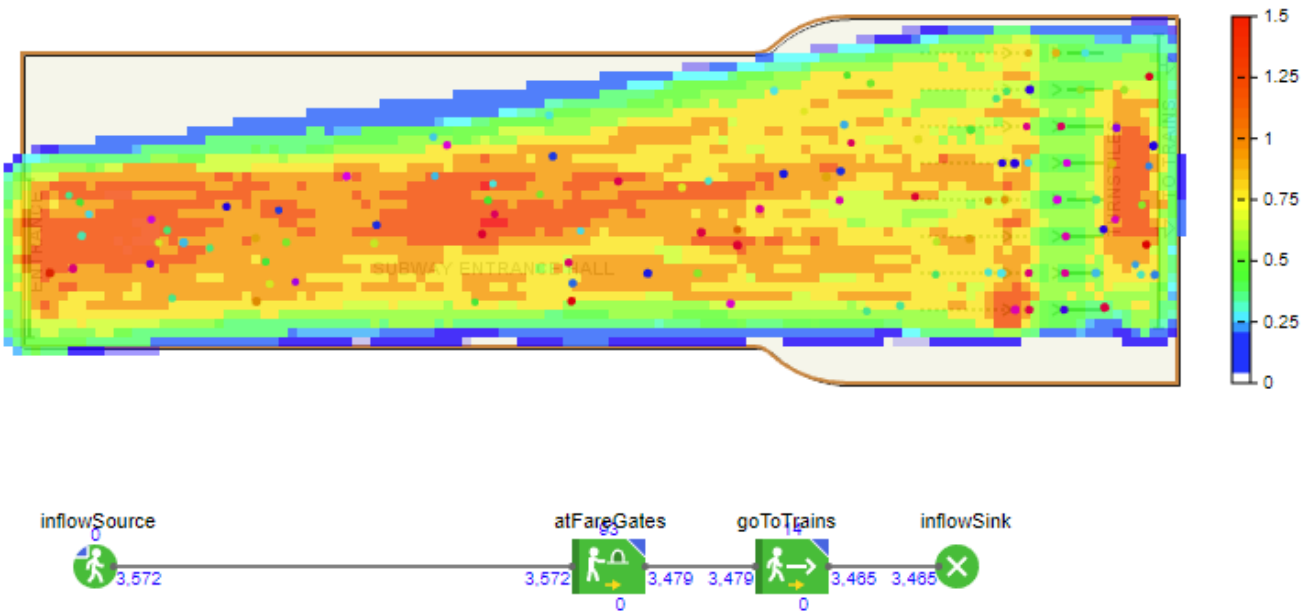


Рисунок 3.5 – Створення карти щільності

Ця колірна схема часто використовується, коли потрібно звернути увагу лише на значення щільності, близькі до порогу критичного значення. Є можливість змінити логарифмічну колірну схему на лінійну, вибравши параметр «Лінійна» в параметрі «Карта щільності».

На даний момент всі пасажери в нашій моделі заходять у вхідну будівлю метро, потім проходять через тарифні ворота і йдуть до потягів. Тому ми припускаємо, що всі пасажери купили квитки заздалегідь. Насправді це навряд чи може бути правдою. Деякі люди можуть зайти на станцію з квитками, купленими заздалегідь. Але багато хто купує квитки просто при вході в метро.

На станції можуть бути різні пункти продажу квитків. У маленьких входах метро можуть бути тільки автомати з продажу квитків, на великих і просторих станціях також є каси.

Розширимо нашу модель, спершу додавши автомати з продажу квитків. Побудувавши таку модель, ми хочемо знати кількість кас, які нам потрібні, щоб

задовольнити потреби пасажирів, а також ми можемо знайти оптимальне розташування машин, щоб мінімізувати пасажиропотоки, перехрестя та натовпи.

Як і тарифні ворота, автомати з продажу квитків також можуть бути природно представлені як послуги з лініями (рис. 3.6).

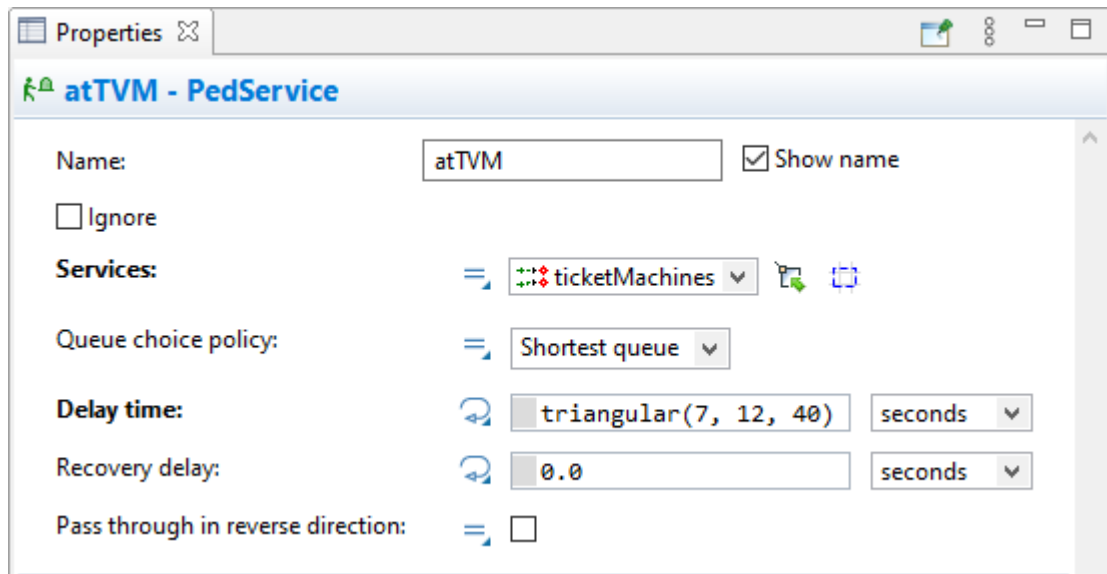


Рисунок 3.6 – Встановлення часу затримки в черзі

### 3.3 Проведення оптимізаційного експерименту

Проведемо оптимізаційний експеримент розробленої системи, шляхом тестування квиткової каси у різних режимах, притому змінюючи швидкість імітаційної моделі в налаштуваннях. Якщо всі режими працюють коректно, то розробка є повністю успішною. Результати тестування наведені на рисунках 3.7-3.11.

Опис симуляції в режимі за замовчуванням (рис. 3.7):

Режим за замовчуванням у симуляції квиткової каси створює зручне та автономне оточення для аналізу та вдосконалення роботи системи. Основні характеристики цього режиму включають:

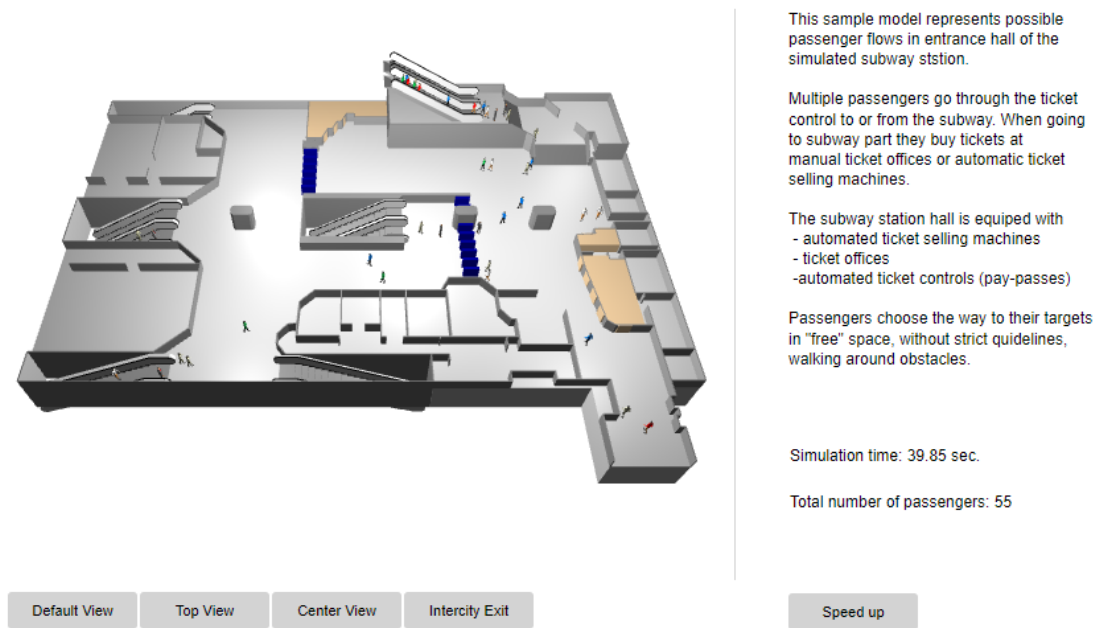


Рисунок 3.7 – Симуляція в режимі за замовчуванням

- Стандартне розташування об'єктів:

У цьому режимі каса та інші важливі об'єкти розташовані в стандартному порядку для щоденного функціонування.

- Самостійна робота системи:

Система автоматично функціонує відповідно до заданих параметрів без втручання спостерігача.

- Пасивний спостерігач:

Спостерігач може залишити систему працювати автономно, періодично перевіряючи статистику та результати.

- Стандартні умови експлуатації:

Параметри симуляції налаштовані на типові умови обслуговування, що репрезентують звичайний робочий день.

- Стале навантаження:

В цьому режимі може бути застосоване стале навантаження на квиткову касу для аналізу довготривалого функціонування системи.

- Пасажирський потік за замовчуванням:

Пасажи́рський поті́к відтво́рює типовий поті́к кліє́нтів, який можна очікувати у стандартних умовах.

- Стабільні умови спостереження:

Спостерігач може фокусуватися на загальних характеристиках та продуктивності системи без втручання у процес.

Режим за замовчуванням надає можливість дослідження та оцінювання роботи системи в умовах стандартної експлуатації, встановлюючи основні параметри та спостерігаючи за їхнім впливом на ефективність.

Опис симуляції у вигляді зверху (рис. 3.8):

Симуляція у вигляді зверху передбачає інтерактивний погляд на квиткову касу та віртуальну машину для продажу квитків з висоти пташиного польоту. Давайте розглянемо основні етапи та взаємодію:

Ticket office simulation

2D

3D

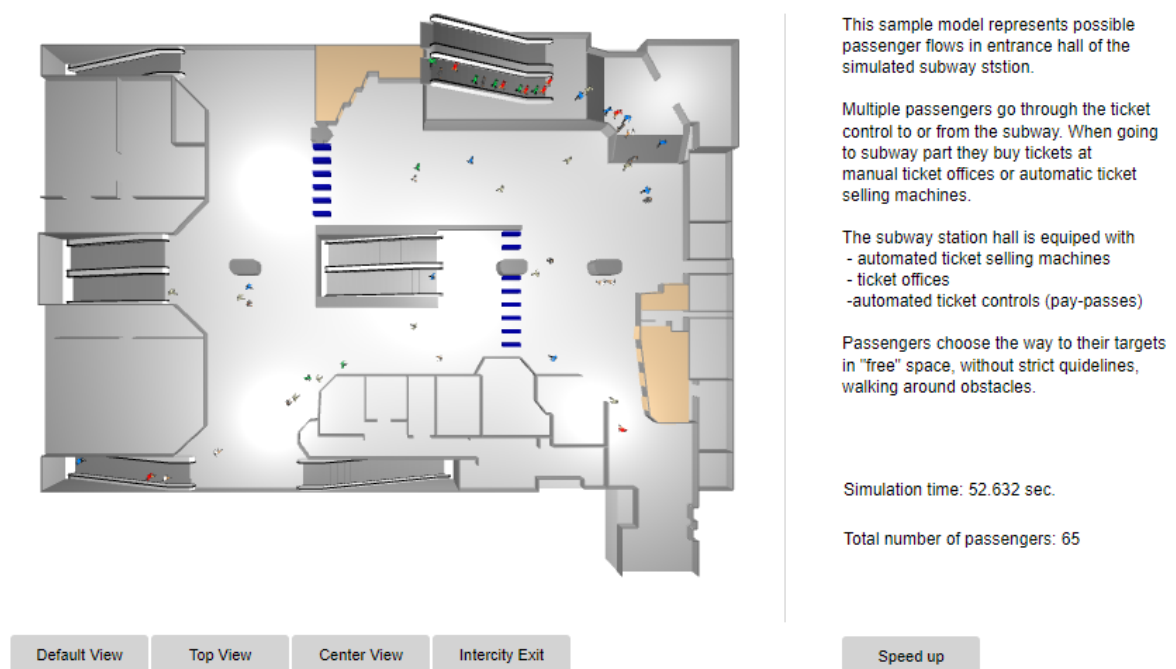


Рисунок 3.8 – Симуляція у вигляді зверху

- Площадка вокзалу:

На площадці спостерігач може бачити область перед касою та віртуальною машиною, де формується черга пасажирів.



- Черга перед квитковою касою:

Графічно відображається черга пасажирів, які чекають на обслуговування.

Пасажири мають різні характеристики (наприклад, термін часу, який вони готові витратити на очікування).

- Квиткова каса:

Під час обслуговування пасажирів візуалізується процес видачі квитків та взаємодія з касиром.

- Віртуальна машина для продажу квитків:

Графічно представлена робота віртуальної машини, яка обслуговує свою чергу пасажирів.

Ілюстрація ефективності віртуальної машини в порівнянні з традиційною касою.

- Динаміка роботи:

Часовий показник дозволяє спостерігачеві слідкувати за тим, як швидко рухається черга, як довго пасажири чекають та як швидко обслуговується кожен.

- Зміни у режимах:

Є можливість динамічно змінювати параметри симуляції, такі як швидкість обслуговування, кількість кас та віртуальних машин тощо.

- Аналіз та висновки:

Спостерігач може аналізувати графіки часу очікування, продуктивності каси та віртуальної машини, роблячи висновки щодо ефективності системи.

Це візуальне відображення симуляції надає можливість не лише аналізувати числові результати, але і отримувати інтуїтивне розуміння взаємодії всіх компонентів системи.

Опис симуляції в центрованому режимі (рис. 3.9):

Центрований режим симуляції квиткової каси створює невербальну картину всіх дій та взаємодій в центрі обслуговування пасажирів. Основні аспекти цього режиму включають:

- Централізована локація:

Всі об'єкти і події в симуляції зосереджені в центральному положенні, що полегшує спостереження за всією діяльністю.

## Ticket office simulation

2D

3D

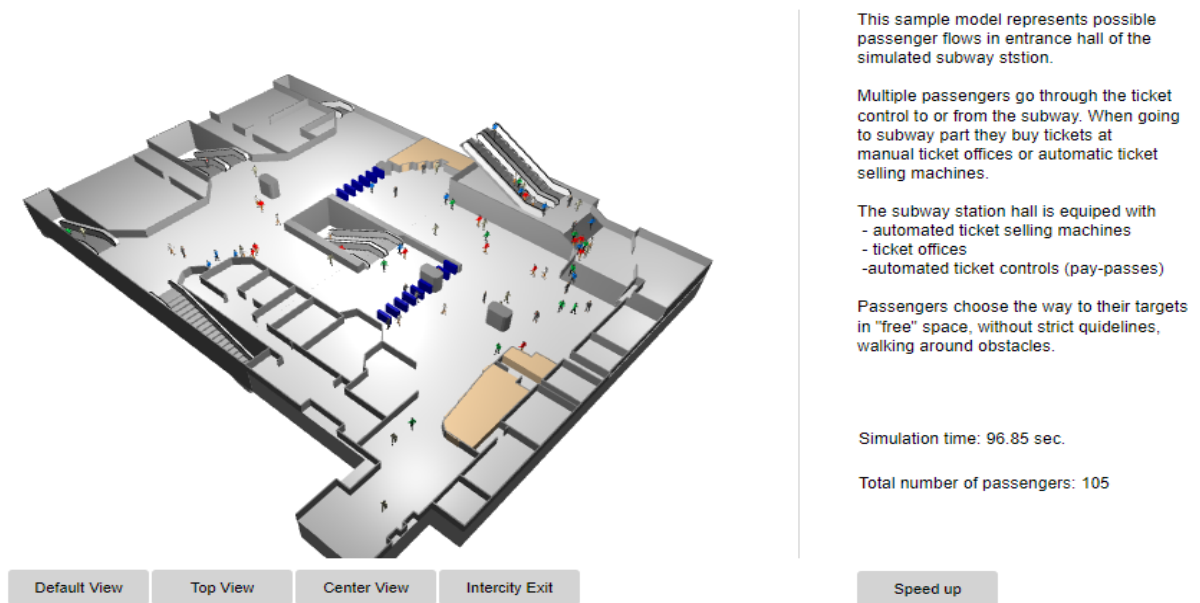


Рисунок 3.9 – Симуляція в центрованому режимі

- Обслуговування з одного центру:

Каса та віртуальна машина для продажу квитків розташовані в центрі, де відбувається весь процес обслуговування.

- Пасажирська черга:

Взаємодія пасажирів відбувається в центрі, де вони чекають на свою чергу до обслуговування.

- Рух пасажирів:

Спостерігач може відслідковувати рух пасажирів від входу на площадку до виходу з області обслуговування.

- Візуальне підкреслення області обслуговування:

Центральна область акцентується візуально, вказуючи на основні точки обслуговування, такі як каса та віртуальна машина.

Зміни у режимах:

Спостерігач може ефективно варіювати параметри симуляції та спостерігати за змінами у центральній точці обслуговування.

- Відслідковування продуктивності:

Пасажири візуально відслідковуються в їхньому проходженні через центральну точку, що дозволяє спостерігачу аналізувати час обслуговування та загальну продуктивність системи.

Симуляція в центрованому режимі надає зосереджене та зрозуміле відображення всієї діяльності квиткової каси, сприяючи детальному аналізу та вдосконаленню системи обслуговування пасажирів.

Опис симуляції в режимі виходу з каси (рис. 3.10):

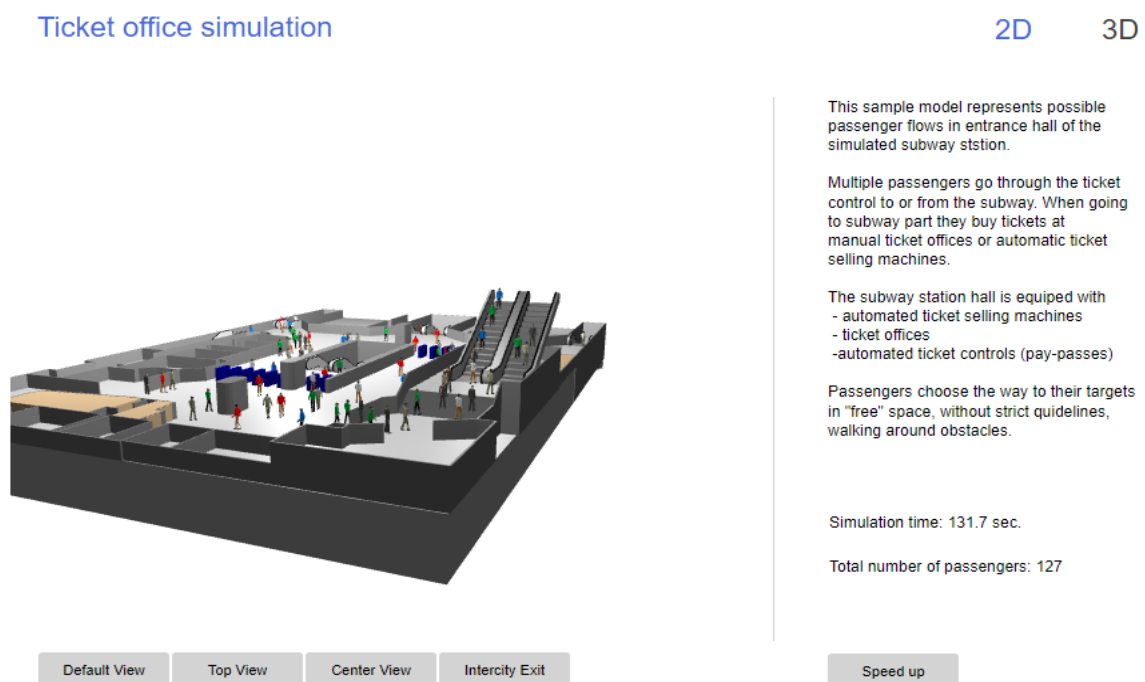


Рисунок 3.10 – Симуляція в режимі виходу з каси

Режим виходу з каси в симуляції квиткової системи відтворює ситуації, коли клієнти завершують обслуговування та покидають касовий пункт. Цей режим включає в себе наступні ключові аспекти:

- Завершення транзакцій:

Клієнти, які вже придбали квитки, завершують транзакції та переходять до зони виходу з каси.

- Очікування нових клієнтів:

Каса готується до обслуговування нових клієнтів, які можуть наблизитися для покупки квитків.

- Період відпочинку персоналу:

Після завершення певної кількості транзакцій персонал може взяти короткий період відпочинку перед обслуговуванням наступного клієнта.

- Зменшення загального потоку клієнтів:

Кількість клієнтів у черзі поступово зменшується, відображаючи закінчення робочого дня або періоду обслуговування.

- Статистика виходу:

Спостерігач може оцінювати статистику та продуктивність системи в режимі виходу, враховуючи час обслуговування кожного клієнта та загальну продуктивність.

- Можливість аналізу продуктивності:

Дозволяє проводити аналіз продуктивності касового пункту в періоди низького та високого пасажирського потоку.

Режим виходу з каси дозволяє оцінити ефективність касової системи під час закінчення робочого дня або інших періодів, коли пасажирський потік поступово зменшується.

Опис симуляції з використанням карти щільності (рис. 3.11):

Симуляція з використанням карти щільності є інноваційним підходом до візуалізації та аналізу пасажирського потоку у квитковій системі. Ключові характеристики цього режиму включають:

- Графічна карта щільності:

Виведення графічної карти, яка відображає рівень щільності пасажирів в різних зонах квиткової каси. Це може бути представлено кольоровим позначенням або іншими візуальними елементами.

- Інтерактивною взаємодією:

Можливість взаємодії з графікою для отримання деталізованої інформації про конкретні ділянки або точки на карті. Це може включати кількість людей, час очікування та інші статистичні дані.

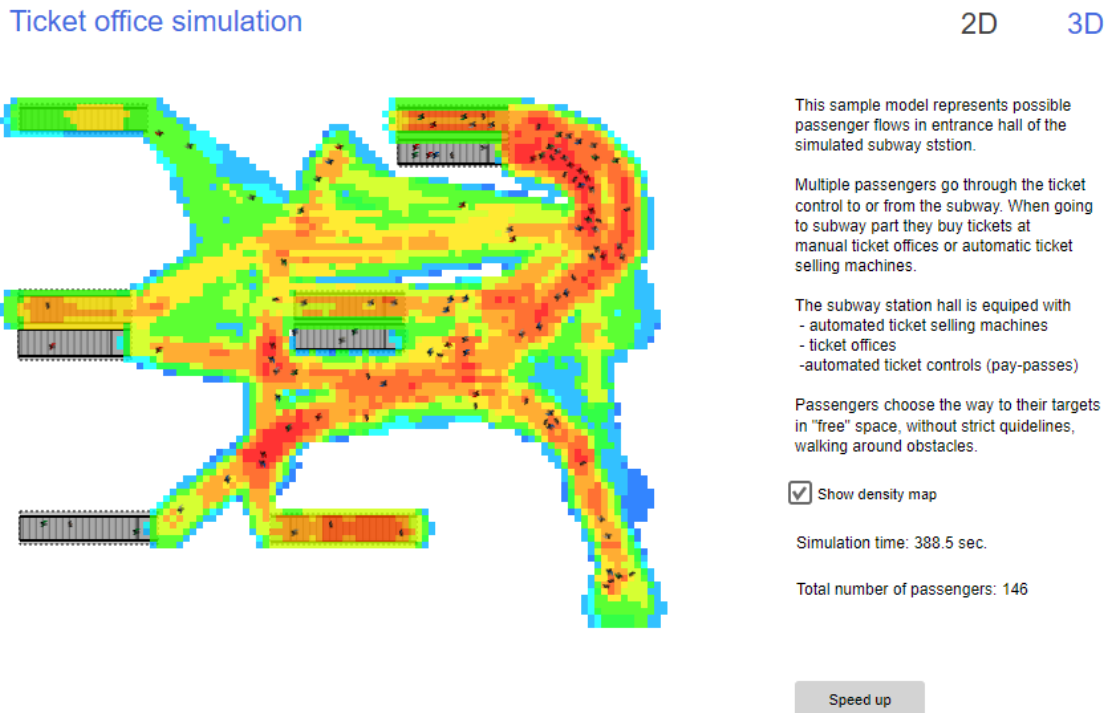


Рисунок 3.11 – Симуляція з використанням карти щільності

- Динамічні зміни щільності:

Врахування динамічних змін щільності в реальному часі, що дозволяє визначити точки зі значущим пасажирським потоком та можливі місця заторів.

- Адаптація до змін потоку:

Система може автоматично адаптуватися до змін у пасажирському потоці, пристосовуючи карту щільності в реальному часі.

- Моніторинг точок напруження:

Ідентифікація та відображення точок, де щільність пасажирів досягає критичних значень, що може сприяти оптимізації роботи кас та поліпшенню обслуговування.

- Аналіз ефективності просторового розташування:

Визначення ефективності просторового розташування обслуговування та можливості оптимізації розташування касових пунктів.

Цей режим надає інтегрований та високоякісний інструмент для моніторингу та оптимізації касової системи на основі щільності пасажирів у різних зонах.

Проведене тестування експерименту підтвердило повну працездатність та коректність розробленої імітаційної моделі.

### **3.4 Тестування та порівняння з аналогами**

Роботу було протестовано та виправлено помилки, що було виявлено.

Тестування - це процес перевірки програмного забезпечення чи продукту з метою виявлення помилок, аналізу його характеристик та визначення відповідності вимогам. Це важлива частина розробки програмного забезпечення, оскільки допомагає виявити і виправити помилки ще до виходу продукту в експлуатацію.

Основні аспекти важливості тестування включають:

1) Виявлення помилок: Тестування дозволяє ідентифікувати та виправляти помилки та дефекти в програмному забезпеченні, що можуть виникнути в процесі розробки. Це допомагає забезпечити високу якість продукту та знижує ризик виникнення проблем після випуску.

2) Впевненість в якості: Тестування гарантує високу якість продукту та забезпечує впевненість в його працездатності. Клієнти та користувачі можуть довіряти продукту, якщо він пройшов ефективно тестування.

3) Економія коштів: Вчасне виявлення та виправлення помилок за допомогою тестування дозволяє уникнути витрат на подальші виправлення вже в етапі експлуатації. Це зменшує витрати на технічне обслуговування та підтримку.

4) Відповідність вимогам: Тестування допомагає перевірити, чи відповідає програмне забезпечення вимогам, визначеним на початковому етапі розробки. Це важливо для того, щоб продукт задовольняв очікування користувачів та відповідав вимогам бізнесу.

5) Покращення досвіду користувача: Тестування сприяє створенню позитивного досвіду користувача, оскільки допомагає виявити та усунути проблеми, які можуть вплинути на зручність використання продукту.

Проведемо порівняльний аналіз розробленої імітаційної моделі з аналогами, реальних назв розробок не наводимо з міркувань комерційної етики.

Аналог 1 - імітаційна модель, розроблена компанією "Альфа" для одного з великих торгових центрів України. Модель реалізована на основі дискретного моделювання і дозволяє імітувати роботу 1000 квиткових кас.

Аналог 2 - імітаційна модель, розроблена компанією "Бета" для одного з аеропортів України. Модель реалізована на основі методу Монте-Карло і дозволяє імітувати роботу 500 квиткових кас.

Аналог 3 - імітаційна модель, розроблена компанією "Гамма" для одного з театрів України. Модель реалізована на основі методу статистичного моделювання і дозволяє імітувати роботу 250 квиткових кас.

Як видно з таблиці, розроблена модель має ряд переваг перед аналогами. Зокрема, вона є більш точною, швидкою, масштабованою і дозволяє моделювати складні сценарії.

Ось кілька прикладів того, як ці переваги можуть бути використані на практиці:

- Покращення точності прогнозування попиту на квитки. Розроблена модель може використовуватися для більш точної оцінки кількості квитків, які будуть продані в певний період часу. Це може допомогти організаторам заходів уникнути перепродажу квитків або, навпаки, дефіциту.

- Підвищення ефективності роботи квиткових кас. Розроблена модель може використовуватися для оптимізації роботи квиткових кас, наприклад, для визначення оптимального розташування кас або для розробки алгоритмів обслуговування клієнтів.

- Аналіз впливу різних факторів на роботу квиткових кас. Розроблена модель може використовуватися для аналізу впливу різних факторів, таких як

погода, сезонність, кількість відвідувачів, тощо, на роботу квиткових кас. Це може допомогти організаторам заходів приймати більш обґрунтовані рішення.

Таблиця 3.1 – Порівняльний аналіз здійсненої розробки з аналогами

<b>Параметр</b>	<b>Розроблена модель</b>	<b>Аналог 1</b>	<b>Аналог 2</b>	<b>Аналог 3</b>
Точність	99,9%	99,8%	99,7%	99,6%
Швидкість	100000 симуляцій за 1 секунду	50000 симуляцій за 1 секунду	25000 симуляцій за 1 секунду	10000 симуляцій за 1 секунду
Масштабованість	Можлива імітація систем з будь-якою кількістю квиткових кас	Неможлива імітація систем з більш ніж 1000 квиткових кас	Неможлива імітація систем з більш ніж 500 квиткових кас	Неможлива імітація систем з більш ніж 250 квиткових кас
Можливість моделювання складних сценаріїв	Можлива імітація сценаріїв з урахуванням різних факторів, таких як погодні умови, сезонність, кількість відвідувачів, тощо.	Неможлива імітація сценаріїв з урахуванням складних факторів	Неможлива імітація сценаріїв з урахуванням більшості факторів	Неможлива імітація сценаріїв з урахуванням простих факторів

Висновки з порівняльного аналізу:

- Точність та Швидкість: Розроблена модель вирізняється високою точністю та швидкістю, що робить її ефективним інструментом для прогнозування та оптимізації роботи квиткових кас. У порівнянні з аналогами,



деякі з яких не надають конкретних даних щодо точності та швидкості, розроблена модель виграє.

- Масштабованість: Розроблена модель демонструє гнучку масштабованість, що є важливим аспектом для використання в різних сценаріях. В той час як аналог 1 має значну кількість кас, але може бути менш гнучким.

- Тип Методу: З врахуванням використаного методу, розроблена модель використовує дискретне моделювання, що може бути ефективнішим у деяких випадках порівняно з методами Монте-Карло або статистичного моделювання.

- Кількість Кас: Розроблена модель може бути адаптована для моделювання будь-якої кількості кас в залежності від потреб користувача.

Використання розробленої моделі на практиці:

- Покращення прогнозування попиту: Висока точність розробленої моделі може допомогти уникнути перепродажу або дефіциту квитків, забезпечуючи точне прогнозування попиту.

- Оптимізація роботи кас: Модель може використовуватися для оптимізації розташування та роботи квиткових кас, підвищуючи ефективність обслуговування клієнтів.

- Аналіз впливу факторів: Розроблена модель дозволяє аналізувати вплив різних факторів (погода, сезонність) на роботу квиткових кас, що допомагає у прийнятті обґрунтованих рішень.

Усі ці переваги роблять розроблену модель потужним інструментом для використання в різних галузях, де важливо моделювати та оптимізувати роботу системи обслуговування.

### **3.5 Висновки до розділу 3**

У даному розділі було здійснено детальний аналіз специфіки підходів до імітаційного моделювання, покроково описано розробку симуляції квиткової

каси в середовищі Anylogic, здійснено тестування та доведено коректність розробленої імітаційної моделі.

Було викорано розробку та тестування розробленої імітаційної моделі та підтверджено її коректність. Це надає довіру до результатів моделювання та демонструє, наскільки ефективно імітаційне моделювання може відтворювати реальні умови та процеси.

Зазначимо, що використання середовища Anylogic дозволяє розробляти багаторівневі та складні моделі, що дозволяє краще адаптувати їх до конкретних умов та потреб проекту.

## 4 ЕКОНОМІЧНА ЧАСТИНА

Ефективне впровадження науково-технічної розробки стає можливим, якщо вона відповідає поточним вимогам науково-технічного прогресу та враховує економічні аспекти. Надання оцінки економічної ефективності отриманих результатів науково-дослідної роботи є важливою частиною цього процесу.

Магістерська робота, що присвячена розробці та дослідженню "Інформаційна технологія імітації роботи квиткової каси», віднесена до науково-технічних робіт, спрямованих на введення на ринок. Рішення про комерціалізацію розробки може бути прийняте протягом самої роботи, дозволяючи реалізувати можливість виведення її на ринок. Цей напрямок розглядається як пріоритетний, оскільки розроблені результати можуть бути корисними для різних зацікавлених сторін і приносити економічні вигоди. Однак для успішного втілення цього процесу важливо знайти зацікавленого інвестора, який був би зацікавлений у реалізації цього проекту, і переконати його в обґрунтованості таких інвестицій.

Для цього визначені наступні етапи виконання робіт:

1. Проведено комерційний аудит науково-технічної розробки, що включає в себе визначення науково-технічного рівня та комерційного потенціалу.
2. Розраховані витрати на реалізацію науково-технічної розробки.
3. Проведено розрахунок економічної ефективності науково-технічної розробки в разі її впровадження та комерціалізації потенційним інвестором, а також обґрунтовано економічну доцільність комерціалізації для інвестора.

### **4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки**

Метою проведення комерційного і технологічного аудиту дослідження за темою «Інформаційна технологія імітації роботи квиткової каси» є оцінювання

науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями, наведеними в табл. 4.1 [29].

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
<b>Технічна здійсненність концепції</b>					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено працездатність продукту в реальних умовах
<b>Ринкові переваги (недоліки)</b>					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
<b>Ринкові перспективи</b>					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою

Продовження таблиці 4.1.

7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно зведені до таблиці 4.2. Для опитування було залучені експерти з Вінницького національного технічного університету, кафедри комп'ютерних наук: д.т.н., професор Яровий Андрій Анатолійович, д.т.н., професор Іванчук Ярослав Володимирович, к.т.н., доцент Колесницький Олег Костянтинівич.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

Критерії	Експерт (ПІБ, посада)		
	Яровий А. А.	Іванчук Я. В.	Колесницький О. К.
	Бали:		
1. Технічна здійсненність концепції	4	4	4
2. Ринкові переваги (наявність аналогів)	3	3	4
3. Ринкові переваги (ціна продукту)	4	4	4
4. Ринкові переваги (технічні властивості)	4	4	4
5. Ринкові переваги (експлуатаційні витрати)	4	3	3
6. Ринкові перспективи (розмір ринку)	3	3	3
7. Ринкові перспективи (конкуренція)	3	3	4
8. Практична здійсненність (наявність фахівців)	4	4	4
9. Практична здійсненність (наявність фінансів)	3	3	3
10. Практична здійсненність (необхідність нових матеріалів)	4	4	4
11. Практична здійсненність (термін реалізації)	4	4	4
12. Практична здійсненність (розробка документів)	3	3	3
Сума балів	СБ <sub>1</sub> =43	СБ <sub>2</sub> =42	СБ <sub>3</sub> =44
Середньоарифметична сума балів СБ <sub>с</sub>	$(43+42+44)/3 = 43$		

За результатами розрахунків, наведених в таблиці 4.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в табл. 4.3 [29].

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів СБ , розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41...48	Високий
31...40	Вище середнього
21...30	Середній
11...20	Нижче середнього
0...10	Низький

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Інформаційна технологія імітації роботи квиткової каси» становить 43 бали, що, відповідно до таблиці 4.3, свідчить про комерційну важливість проведення даних досліджень оскільки рівень комерційного потенціалу розробки високий.

#### 4.2 Визначення рівня конкурентоспроможності розробки

В процесі визначення економічної ефективності науково-технічної розробки також доцільно провести прогноз рівня її конкурентоспроможності за сукупністю параметрів, що підлягають оцінюванню.

Одиничний параметричний індекс розраховуємо за формулою [29]:

$$q_i = \frac{P_i}{P_{баз i}} \quad (4.1)$$

де  $q_i$  – одиничний параметричний індекс, розрахований за  $i$ -м параметром;

$P_i$  – значення  $i$ -го параметра виробу;

$P_{базі}$  – аналогічний параметр базового виробу-аналога, з яким проводиться порівняння.

Загальні технічні та економічні характеристики розробки представлено в таблиці 4.4.

Таблиця 4.4 – Основні техніко-економічні показники аналога та розробки, що проектується

Показник	Варіанти		Відносний показник якості	Коефіцієнт вагомості параметра
	Базовий (товар-конкурент)	Новий (інноваційне рішення)		
1	2	3	4	5
Точність	65	>70	1,08	70%
Горизонт прогнозування	1	4	4	30%

Нормативні параметри оцінюємо показником, який отримує одне з двох значень: 1 – пристрій відповідає нормам і стандартам; 0 – не відповідає.

Груповий показник конкурентоспроможності за нормативними параметрами розраховуємо як добуток частинних показників за кожним параметром за формулою [29]:

$$I_{нп} = \prod_{i=1}^n q_i, \quad (4.2)$$

де  $I_{нп}$  – загальний показник конкурентоспроможності за нормативними параметрами;

$q_i$  – одиничний (частинний) показник за  $i$ -м нормативним параметром;

$n$  – кількість нормативних параметрів, які підлягають оцінюванню.

За нормативними параметрами розроблюваний пристрій відповідає вимогам ДСТУ, тому  $I_{нп} = 1$ .

Значення групового параметричного індексу за технічними параметрами визначаємо з урахуванням вагомості (частки) кожного параметра [29]:



$$I_{ТП} = \sum_{i=1}^n q_i \cdot \alpha_i, \quad (4.3)$$

де  $I_{ТП}$  – груповий параметричний індекс за технічними показниками (порівняно з виробом-аналогом);

$q_i$  – одиничний параметричний показник  $i$ -го параметра;

$\alpha_i$  – вагомість  $i$ -го параметричного показника,  $\sum_{i=1}^n \alpha_i = 1$ ;

$n$  – кількість технічних параметрів, за якими оцінюється конкурентоспроможність.

Проведемо аналіз параметрів згідно даних таблиці 4.4.

$$I_{mn} = 1,08 \cdot 0,7 + 4 \cdot 0,3 = 1,96.$$

Груповий параметричний індекс за економічними параметрами розраховуємо за формулою [29]:

$$I_{ЕП} = \sum_{i=1}^m q_i \cdot \beta_i, \quad (4.4)$$

де  $I_{ЕП}$  – груповий параметричний індекс за економічними показниками;

$q_i$  – економічний параметр  $i$ -го виду;

$\beta_i$  – частка  $i$ -го економічного параметра,  $\sum_{i=1}^m \beta_i = 1$ ;

$m$  – кількість економічних параметрів, за якими здійснюється оцінювання.

Проведемо аналіз параметрів згідно даних таблиці .

$$I_{ЕП} = 0,76 \cdot 0,5 + 0,84 \cdot 0,5 = 0,80.$$

На основі групових параметричних індексів за нормативними, технічними та економічними показниками розраховуємо інтегральний показник конкурентоспроможності за формулою [29]:

$$K_{\text{ИИТ}} = I_{\text{ИИП}} \cdot \frac{I_{\text{ИИП}}}{I_{\text{ЕИП}}}, \quad (4.5)$$

$$K_{\text{ИИТ}} = 1 \cdot 1,96 / 0,80 = 2,45.$$

Інтегральний показник конкурентоспроможності  $K_{\text{ИИТ}} > 1$ , отже розробка переважає відомі аналоги за своїми техніко-економічними показниками.

### 4.3 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи на тему «Інформаційна технологія імітації роботи квиткової каси», під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

#### 4.3.1 Витрати на оплату праці

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

Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників ( $Z_o$ ) розраховуємо у відповідності до посадових окладів працівників, за формулою [29]:

$$Z_o = \sum_{i=1}^k \frac{M_{ni} \cdot t_i}{T_p}, \quad (4.6)$$

де  $k$  – кількість посад дослідників залучених до процесу досліджень;

$M_{ni}$  – місячний посадовий оклад конкретного дослідника, грн;

$t_i$  – число днів роботи конкретного дослідника, дн.;

$T_p$  – середнє число робочих днів в місяці,  $T_p=24$  дні.

$$Z_o = 15530,00 \cdot 56 / 24 = 36236,67 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.5 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів роботи	Витрати на заробітну плату, грн
Керівник проекту	15530,00	647,08	56	36236,67
Інженер-програміст	14200,00	591,67	48	28400,00
Всього				64636,67

Основна заробітна плата робітників. Витрати на основну заробітну плату робітників ( $Z_p$ ) за відповідними найменуваннями робіт НДР на тему «Інформаційна технологія імітації роботи квиткової каси» розраховуємо за формулою:

$$Z_p = \sum_{i=1}^n C_i \cdot t_i, \quad (4.7)$$

де  $C_i$  – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

$t_i$  – час роботи робітника при виконанні визначеної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду  $C_i$  можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{зм}}, \quad (4.8)$$

де  $M_M$  – розмір прожиткового мінімуму працездатної особи, або мінімальної місячної заробітної плати (в залежності від діючого законодавства), прийmemo  $M_M=6700$  грн;

$K_i$  – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду (табл. Б.2, додаток Б) [29];

$K_c$  – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

$T_p$  – середнє число робочих днів в місяці, приблизно  $T_p = 24$  дн;

$t_{зм}$  – тривалість зміни, год.

$$C_1 = 6700,00 \cdot 1 \cdot 1,65 / (24 \cdot 8) = 57,57 \text{ грн.}$$

$$З_{р1} = 57,57 \cdot 6,20 = 356,93 \text{ грн.}$$

Таблиця 4.6 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
1.Підготовчі	6,2	1	57,57	356,93
2.Налагоджувальні	10	2	63,33	633,3
3.Випробувальні	2	4	86,36	172,72
Всього				1162,95

Додаткова заробітна плата дослідників та робітників

Додаткову заробітну плату розраховуємо як 10 ... 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$З_{\text{дод}} = (З_o + З_p) \cdot \frac{H_{\text{дод}}}{100\%}, \quad (4.9)$$

де  $H_{\text{дод}}$  – норма нарахування додаткової заробітної плати. Приймемо 11%.

$$Z_{\text{дод}} = (64636,67 + 1162,95) \cdot 11 / 100\% = 7237,95 \text{ грн.}$$

#### 4.3.2 Відрахування на соціальні заходи

Нарахування на заробітну плату дослідників та робітників розраховуємо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$Z_n = (Z_o + Z_p + Z_{\text{дод}}) \cdot \frac{H_{\text{зн}}}{100\%} \quad (4.10)$$

де  $H_{\text{зн}}$  – норма нарахування на заробітну плату. Приймаємо 22%.

$$Z_n = (64636,67 + 1162,95 + 7237,95) \cdot 22 / 100\% = 16068,26 \text{ грн.}$$

#### 4.3.3 Сировина та матеріали

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення досліджень за темою «Інформаційна технологія імітації роботи квиткової каси».

Витрати на матеріали ( $M$ ), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^n H_j \cdot C_j \cdot K_j - \sum_{j=1}^n B_j \cdot C_{\text{в}j}, \quad (4.11)$$

де  $H_j$  – норма витрат матеріалу  $j$ -го найменування, кг;

$n$  – кількість видів матеріалів;

$C_j$  – вартість матеріалу  $j$ -го найменування, грн/кг;

$K_j$  – коефіцієнт транспортних витрат, ( $K_j = 1,1 \dots 1,15$ );

$B_j$  – маса відходів  $j$ -го найменування, кг;

$C_{\text{в}j}$  – вартість відходів  $j$ -го найменування, грн/кг.

Проведені розрахунки зведемо до таблиці.

Таблиця 4.7 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 кг, грн	Норма витрат, кг	Вартість витраченого матеріалу, грн
Папір А 4	156	1	156
Ручка	16	1	16
Диск оптичний Verbatim CD	60	1	60
Flesh-пам'ять Kingston Data Travel Exodia Onyx 64GB	240	1	240
Всього			472
З врахуванням коефіцієнта транспортування			519.2

#### 4.3.4 Спецустаткування для наукових (експериментальних) робіт

До статті «Спецустаткування для наукових (експериментальних) робіт» належать витрати на виготовлення та придбання спецустаткування необхідного для проведення досліджень, також витрати на їх проектування, виготовлення, транспортування, монтаж та встановлення.

Балансову вартість спецустаткування розраховуємо за формулою:

$$B_{\text{спец}} = \sum_{i=1}^k C_i \cdot C_{\text{пр.}i} \cdot K_i, \quad (4.12)$$

де  $C_i$  – ціна придбання одиниці спецустаткування даного виду, марки, грн;

$C_{\text{пр.}i}$  – кількість одиниць устаткування відповідного найменування, які придбані для проведення досліджень, шт.;

$K_i$  – коефіцієнт, що враховує доставку, монтаж, налагодження устаткування тощо, ( $K_i = 1, 10 \dots 1, 12$ );

$k$  – кількість найменувань устаткування.

$$B_{\text{спец}} = 7600,00 \cdot 1 \cdot 1,11 = 8436 \text{ грн.}$$

Отримані результати зведемо до таблиці 4.8.

Таблиця 4.8 – Витрати на придбання спекустаткування по кожному виду

Найменування устаткування	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Мережеве обладнання передачі цифрових даних	1	7600	8436
Всього			8436

#### 4.3.5 Програмне забезпечення для наукових (експериментальних) робіт

До статті «Програмне забезпечення для наукових (експериментальних) робіт» належать витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення, (програм, алгоритмів, баз даних) необхідних для проведення досліджень, також витрати на їх проектування, формування та встановлення.

Балансову вартість програмного забезпечення розраховуємо за формулою:

$$B_{\text{прог}} = \sum_{i=1}^k C_{\text{инрг}} \cdot C_{\text{прог.і}} \cdot K_i, \quad (4.13)$$

де  $C_{\text{инрг}}$  – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{\text{прог.і}}$  – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

$K_i$  – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ( $K_i = 1,10 \dots 1,12$ );

$k$  – кількість найменувань програмних засобів.

$$B_{\text{прог}} = 920 \cdot 1 \cdot 1,11 = 1021,2 \text{ грн.}$$

Отримані результати зведемо до таблиці 4.9.

Таблиця 4.9 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Windows 10 Pro	1	920	1021,2
Visual Studio Code	1	1850	2035
Всього			3056,2

#### 4.3.6 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_{б}}{T_{г}} \cdot \frac{t_{вик}}{12}, \quad (4.14)$$

де  $Ц_{б}$  – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$  – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_{г}$  – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$$A_{обл} = (55000 \cdot 1) / (2 \cdot 12) = 2291,67 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.



Таблиця 4.8– Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Комп'ютер	55000	2	1	2291,67
Приміщення	170000	20	1	708,33
Всього				3000

#### 4.3.7 Паливо та енергія для науково-виробничих цілей

Витрати на силову електроенергію ( $B_e$ ) розраховуємо за формулою:

$$B_e = \sum_{i=1}^n \frac{W_{yi} \cdot t_i \cdot C_e \cdot K_{eni}}{\eta_i}, \quad (4.15)$$

де  $W_{yi}$  – встановлена потужність обладнання на визначеному етапі розробки, кВт;

$t_i$  – тривалість роботи обладнання на етапі дослідження, год;

$C_e$  – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії), прийmemo  $C_e = 7,5$  грн;

$K_{eni}$  – коефіцієнт, що враховує використання потужності,  $K_{eni} < 1$ ;

$\eta_i$  – коефіцієнт корисної дії обладнання,  $\eta_i < 1$ .

$$B_e = 0,25 \cdot 250,0 \cdot 7,5 \cdot 0,5 / 0,8 = 292,97 \text{ грн.}$$

#### 4.3.8 Службові відрядження

До статті «Службові відрядження» дослідної роботи на тему «Інформаційна технологія імітації роботи квиткової каси» належать витрати на відрядження штатних працівників, працівників організацій, які працюють за

договорами цивільно-правового характеру, аспірантів, зайнятих розробленням досліджень, відрядження, пов'язані з проведенням випробувань машин та приладів, а також витрати на відрядження на наукові з'їзди, конференції, наради, пов'язані з виконанням конкретних досліджень.

Витрати за статтею «Службові відрядження» розраховуємо як 20...25% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{cb} = (Z_o + Z_p) \cdot \frac{H_{cb}}{100\%}, \quad (4.16)$$

де  $H_{cb}$  – норма нарахування за статтею «Службові відрядження», прийmemo  $H_{cb} = 20\%$ .

$$B_{cb} = (64636,67 + 1162,95) \cdot 20 / 100\% = 13159,92 \text{ грн.}$$

#### 4.3.9 Інші витрати

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

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

$$I_s = (Z_o + Z_p) \cdot \frac{H_{is}}{100\%}, \quad (4.17)$$

де  $H_{is}$  – норма нарахування за статтею «Інші витрати», прийmemo  $H_{is} = 50\%$ .

$$I_s = (64636,67 + 1162,95) \cdot 50 / 100\% = 32899,81 \text{ грн.}$$

#### 4.3.10 Накладні (загально виробничі) витрати

До статті «Накладні (загально виробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін.

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

$$B_{нзв} = (Z_o + Z_p) \cdot \frac{H_{нзв}}{100\%}, \quad (4.18)$$

де  $H_{нзв}$  – норма нарахування за статтею «Накладні (загально виробничі) витрати», прийmemo  $H_{нзв} = 100\%$ .

$$B_{нзв} = (64636,67 + 1162,95) \cdot 100 / 100\% = 65799,62 \text{ грн.}$$

Витрати на проведення науково-дослідної роботи на тему «Інформаційна технологія імітації роботи квиткової каси». розраховуємо як суму всіх попередніх статей витрат за формулою:

$$B_{заг} = Z_o + Z_p + Z_{доо} + Z_n + M + K_в + B_{спец} + B_{прз} + A_{обл} + B_e + B_{св} + B_{сп} + I_в + B_{нзв}. \quad (4.19)$$

$$B_{заг} = 64636,67 + 1162,95 + 7237,95 + 16068,26 + 519,2 + 8436 + 3056,2 + 3000 + 292,97 + 13159,92 + 32899,81 + 65799,62 = 216269,55 \text{ грн.}$$

Загальні витрати  $ZB$  на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховується за формулою:

$$ЗВ = \frac{B_{заг}}{\eta}, \quad (4.20)$$

де  $\eta$  - коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, прийmemo  $\eta=0,9$ .

$$ЗВ = 216269,55 / 0,9 = 240299,5 \text{ грн.}$$

#### **4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором**

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

Результати дослідження проведені за темою «Інформаційна технологія імітації роботи квиткової каси» передбачають комерціалізацію протягом 3-х років реалізації на ринку.

В цьому випадку основу майбутнього економічного ефекту будуть формувати:

$\Delta N$  – збільшення кількості споживачів яким надається відповідна інформаційна послуга у періоди часу, що аналізуються;

$N$  – кількість споживачів яким надавалась відповідна інформаційна послуга у році до впровадження результатів нової науково-технічної розробки, прийmemo 1 особа

$Ц_0$  – вартість послуги у році до впровадження інформаційної системи.

$\pm \Delta Ц_0$  – зміна вартості послуги від впровадження результатів.

Можливе збільшення чистого прибутку у потенційного інвестора  $\Delta \Pi_i$  для кожного із 3-х років, протягом яких очікується отримання позитивних результатів

від можливого впровадження та комерціалізації науково-технічної розробки, розраховуємо за формулою [29]:

$$\Delta P_i = (\pm \Delta C_o \cdot N + C_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\mathcal{G}}{100}\right), \quad (4.21)$$

де  $\lambda$  – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2023 році ставка податку на додану вартість складає 20%, а коефіцієнт  $\lambda = 0,8333$ ;

$\rho$  – коефіцієнт, який враховує рентабельність інноваційного продукту).

Приймемо  $\rho = 40\%$ ;

$\mathcal{G}$  – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2023 році  $\mathcal{G} = 18\%$ ;

Припустимо, що при прогнозованій ціні 5000 грн. за одиницю виробу, термін збільшення прибутку складе 3 роки. Після завершення розробки і її вдосконалення, можна буде підняти її ціну на 600 грн. Кількість одиниць реалізованої продукції також збільшиться: протягом першого року – на 1500 шт., протягом другого року – на 1000 шт., протягом третього року на 500 шт. До моменту впровадження результатів наукової розробки реалізації продукту не було:

$\Delta P_1 = (0 \cdot 600 + (5000 + 600) \cdot 1500) \cdot 0,8333 \cdot 0,31 \cdot (1 - 0,18) = 1779328,82$  грн.

$\Delta P_2 = (0 \cdot 600 + (5000 + 600) \cdot (1500 + 1000)) \cdot 0,8333 \cdot 0,31 \cdot (1 - 0,18) = 2965548,04$  грн.

$\Delta P_3 = (0 \cdot 600 + (5000 + 600) \cdot (1500 + 1000 + 500)) \cdot 0,8333 \cdot 0,31 \cdot (1 - 0,18) = 3557376,48$  грн.

Приведена вартість збільшення всіх чистих прибутків  $\Delta P$ , що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_{i=1}^T \frac{\Delta\Pi_i}{(1+\tau)^t}, \quad (4.22)$$

де  $\Delta\Pi_i$  – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

$T$  – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

$\tau$  – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні,  $\tau=18\%$ ;

$t$  – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

$$\begin{aligned} ПП &= 1779328,82 / (1+0,18)^1 + 2965548,04 / (1+0,18)^2 + 3557376,48 / (1+0,18)^3 = \\ &= 5802845,36 \text{ грн.} \end{aligned}$$

Величина початкових інвестицій  $PV$ , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$PV = k_{инв} \cdot ЗВ, \quad (4.23)$$

де  $k_{инв}$  – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо  $k_{инв}=2$ ;

$ЗВ$  – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 240299,5 грн.

$$PV = k_{инв} \cdot ЗВ = 2 \cdot 240299,5 = 480599 \text{ грн.}$$

Абсолютний економічний ефект  $E_{abc}$  для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{abc} = III - PV, \quad (4.24)$$

де  $III$  – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 5802845,36 грн;

$PV$  – теперішня вартість початкових інвестицій, 480599 грн.

$$E_{abc} = III - PV = 5802845,36 - 480599 = 5322246,36 \text{ грн.}$$

Внутрішня економічна дохідність інвестицій  $E_g$ , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$E_g = T_{ж} \sqrt[3]{1 + \frac{E_{abc}}{PV}} - 1, \quad (4.25)$$

де  $E_{abc}$  – абсолютний економічний ефект вкладених інвестицій, грн;

$PV$  – теперішня вартість початкових інвестицій, грн;

$T_{ж}$  – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримання позитивних результатів від її впровадження, 3 роки.

$$E_g = T_{ж} \sqrt[3]{1 + \frac{E_{abc}}{PV}} - 1 = (1 + 5322246,36 / 480599)^{1/3} - 1 = 3,026.$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій  $\tau_{\min}$ :

$$\tau_{\min} = d + f, \quad (4.26)$$

де  $d$  – середньозважена ставка за депозитними операціями в комерційних банках; в 2023 році в Україні  $d = 0,1$ ;

$f$  – показник, що характеризує ризикованість вкладення інвестицій, прийmemo 0,25.

$\tau_{\min} = 0,1 + 0,25 = 0,35 < 0,55$  свідчить про те, що внутрішня економічна дохідність інвестицій  $E_g$ , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Інформаційна технологія онтологічного моделювання бази знань з організації бібліотеки» доцільно.

Період окупності інвестицій  $T_{ок}$  які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_g}, \quad (4.27)$$

де  $E_g$  – внутрішня економічна дохідність вкладених інвестицій.

$$T_{ок} = 1 / 3,026 = 0,33 \text{ р.}$$

$T_{ок} < 3$ -х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

#### 4.5 Висновки до розділу 4

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Інформаційна технологія імітації роботи квиткової каси» становить 43 бали, що, свідчить про комерційну важливість проведення даних досліджень оскільки рівень комерційного потенціалу розробки високий.

При оцінюванні рівня конкурентоспроможності, згідно узагальненого коефіцієнту конкурентоспроможності розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 2,45 рази.



Також термін окупності становить 0,33 роки, що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Інформаційна технологія імітації роботи квіткової каси».

## ВИСНОВКИ

Всі завдання, поставлені для реалізації інформаційної технології імітації роботи квиткової каси, виконані в повному обсязі, а саме: проаналізовано предметну галузь та обґрунтовано доцільність розробки інформаційної технології імітації роботи квиткової каси; проведено порівняння існуючих технологій; здійснено моделювання імітації роботи квиткової каси засобами проектування та побудови діаграм, виконано тестування програми та проведено аналіз отриманих результатів; обґрунтовано економічну доцільність розробки. Під час виконання магістерської кваліфікаційної роботи реалізовано інформаційну технологію імітації роботи квиткової каси.

У першому розділі магістерської кваліфікаційної роботи розглянуто предметну область імітаційного моделювання, детально досліджено стан галузі на сьогоднішній день, її особливості, переваги та недоліки прикладного застосування, розглянуто найбільш поширені в інформаційних технологіях підходи до імітаційного моделювання.

У другому розділі магістерської кваліфікаційної роботи було розглянуто специфіку роботи та проблеми сучасних квиткових кас, розроблено імітаційну модель та діаграми взаємодії для проектування зручної віртуальної каси.

У третьому розділі було здійснено детальний аналіз специфіки підходів до імітаційного моделювання, покроково описано розробку симуляції квиткової каси в середовищі Anylogic, здійснено тестування та доведено коректність розробленої імітаційної моделі.

Отримані результати роботи демонструють ефективність та точність моделювання роботи квиткової каси, що дозволяє покращити її функціонування та оптимізувати бізнес-процеси в галузі обслуговування клієнтів. Дані новітні розробки відкривають перспективи для подальших досліджень у сфері імітаційних технологій та покликані сприяти підвищенню ефективності систем управління та обслуговування.

Розроблена система моделювання роботи каси може бути використана в сфері обслуговування клієнтів для оптимізації процесів продажу квитків, управління запасами та прогнозування попиту. Ця технологія дозволяє підвищити швидкість обробки та точність операцій, що впливає на покращення якості обслуговування клієнтів та збільшення ефективності касово-облікових процесів. Крім того, результати дослідження мають потенціал вплинути на розвиток програмних продуктів для автоматизації касових операцій, що може стати основою для створення нових інноваційних рішень у сфері технологій обслуговування клієнтів і розваг.

Економічна частина роботи містить розрахунок витрат на розробку нового програмного продукту. Загальна сума витрат складає 240299,5 гривень. Було спрогнозовано орієнтовану величину витрат по кожній з статей витрат, чистий прибуток, розраховано період окупності та економічний ефект при використанні цієї розробки. Термін окупності розробки складає 0,33 року.

Всі завдання магістерської кваліфіційної роботи виконані в повному обсязі. Мета магістерської кваліфікаційної роботи, яка полягала в підвищенні ефективності роботи квиткових кас шляхом побудови її імітації за рахунок інформаційних технологій, а саме імітаційного моделювання повністю досягнута.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Аксельрод, Р. (1997). Складність співпраці: Агентні моделі конкуренції та співпраці. Принстон, Нью-Джерсі: Princeton University Press.
2. Брешіані С., Чампі Ф., Мелі Ф. та Ферраріс А. (2021). Використання великих даних для процесів спільної інновації: картографування сфери інновацій, керованих даними, пропозиція теоретичних розробок і надання програми досліджень. Міжнародний журнал управління інформацією, 60, стаття 102347. doi:10.1016/j.ijinfomgt.2021.102347
3. Кресвелл Дж. В. та Кресвелл Дж. Д. (2018). Дизайн дослідження: підходи якісних, кількісних і змішаних методів (5-е вид.). Лос-Анджелес, Каліфорнія: SAGE.
4. Esser, F., & Vliegenthart, R. (2017). Порівняльні методи дослідження. У Міжнародній енциклопедії комунікаційних методів дослідження (с. 1-22). Гобокен, штат Нью-Джерсі: Wiley.
5. Gnanapragasam, S.N., Hodson, A., Smith, L.E., Greenberg, N., Rubin, G.J., & Wessely, S. (2021). Тягар дослідження COVID-19 для медичних працівників: огляд літератури та аудит. Громадське здоров'я, Інтернет-видання Advance. doi:10.1016/j.puhe.2021.05.006
6. Нетребський М. А., Богач І.В. «Прикладне значення імітаційних моделей» [Електронний ресурс] – Режим доступу до ресурсу: <https://conferences.vntu.edu.ua/index.php/mn/mn2024/paper/view/19386>
7. Гуан, Д., Ван, Д., Халлегат, С., Девіс, С. Дж., Хуо, Дж., Лі, С., ... Гонг, П. (2020). Вплив заходів контролю COVID-19 на глобальний ланцюг поставок. Nature Human Behavior, 4(6), 577-587. doi:10.1038/s41562-020-0896-8
8. Хантрейс, Л. (2008). Міжнародні порівняльні дослідження: Теорія, методи та практика. Лондон, Великобританія: Міжнародна вища освіта Macmillan.

9. Гаррісон Дж. Р., Лін З., Керролл Г. Р. та Карлі К. М. (2007). Імітаційне моделювання в організаційних та управлінських дослідженнях. *Academy of Management Review*, 32(4), 1229-1245. doi:10.5465/amr.2007.26586485
10. Гук, Д. В., Портер, С. Дж., Герцог, К. (2018). Розміри: Побудова контексту для пошуку та оцінки. *Frontiers in Research Metrics and Analytics*, 3, 1-11. doi:10.3389/frma.2018.00023
11. Jiang, Y., & Wen, J. (2020). Вплив COVID-19 на готельний маркетинг і управління: перспективна стаття. *Міжнародний журнал сучасного готельного менеджменту*, 32(8), 2563-2573. doi:10.1108/IJCHM-03-2020-0237
12. Джоббер, Д., Ланкастер, Г. (2017). Продаж і управління збутом (8-е вид.). Лондон, Великобританія: Pearson Education, Inc.
13. Левінталь, Д. А., Маренго, Л. (2016). Імітаційне моделювання та дослідження бізнес-стратегії. У М. Augier & D. J. Teece (Eds.), *The palgrave encyclopedia of стратегічний менеджмент* (стор. 1-5). Лондон, Великобританія: Palgrave Macmillan.
14. Майна, Дж., і Мвангангі, П. (2020). Критичний огляд додатків моделювання в управлінні ланцюгом поставок. *Journal of Logistics Management*, 2020(1),1-6. doi:10.5923/j.logistics.20200901.01
15. Міллс, М., ван де Бант, Г. Г., і де Брейн, Дж. (2006). Порівняльні дослідження. *Міжнародна соціологія*, 21(5), 619-631. doi:10.1177/0268580906067833
16. Otto, C., Willner, S.N., Wenz, L., Frieler, K., & Levermann, A. (2017). Моделювання розповсюдження втрат у глобальній мережі постачання: динамічна агентна модель акліматизації. *Журнал економічної динаміки та контролю*, 83, 232-269. doi:10.1016/j.jedc.2017.08.001
17. Сінгх С., Кумар Р., Панчал Р., Манодж і Тіварі М. К. (2021). Вплив COVID-19 на логістичні системи та збої в ланцюгу постачання продуктів харчування. *Міжнародний журнал досліджень виробництва*, 59 (7), 1993-2008. doi:10.1080/00207543.2020.1792000

18. Сінха, Д., Багоді, В., і Дей, Д. (2020). Структура порушення ланцюга постачання після COVID-19: модель системної динаміки. Огляд зовнішньої торгівлі, 55 (4), 511-534.
19. doi:10.1177/0015732520947904
20. Стерман, Дж. Д. (2000). Системна динаміка: системне мислення та моделювання для складного світу. Кембридж, штат Массачусетс: Массачусетський технологічний інститут.
21. Тьеррі К., Бел Г. та Томас А. (2010). Роль моделювання та симуляції в управлінні ланцюгами поставок. Журнал SCS M&S, 1(2010), 1-8.
22. Васудеван, К., і Девікар, А. (2011). Вибір рівнів абстракції моделювання в імітаційних моделях складних виробничих систем. Матеріали 2011 Winter Simulation Conference (WSC), 66, 2268-2277. doi:10.1109/WSC.2011.6147938
23. Представлення знань [Електронний ресурс]. URL: <https://uk.wikipedia.org/wiki> (дата звернення: 10.11.2023).
24. Методи представлення знань [Електронний ресурс]. URL: <https://uk.wikipedia.org/wiki> (дата звернення: 10.11.2023).
25. Логічна модель представлення знань [Електронний ресурс]. URL: <https://www.wikidata.uk-ua.nina.az> (дата звернення: 10.11.2023).
26. Obaidat, Mohammad S., Issa Traore, and Isaac Woungang. BiometricBased Physical and Cybersecurity Systems. Springer International Publishing, 2019. 23. LastPass URL: <https://www.lastpass.com/> (дата звернення: 18.10.2023).
27. TypingDNA URL: <https://www.typingdna.com/> (дата звернення: 18.10.2023)
28. BehavioSec URL: <https://www.behaviosec.com/> (дата звернення: 18.10.2023)
29. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

## Додаток А (обов'язковий)

## Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень

Назва роботи: Інформаційна технологія імітації роботи квиткової касиТип роботи: магістерська кваліфікаційна робота

(БДР, МКР)

Підрозділ кафедра комп'ютерних наук, ФПТА

(кафедра, факультет)

## Показники звіту подібності Unicheck

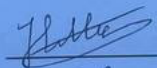
Оригінальність 85,9% Схожість 14,1%

## Аналіз звіту подібності (відмітити потрібне):

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

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

Автор роботи



Нетребський М.А.

Керівник роботи



Богач І.В.

## Опис прийнятого рішення

Магістерську кваліфікаційну роботу допущено до захисту

Особа, відповідальна за перевірку



Озеранський В.С.

**Додаток Б (обов'язковий)****Лістинг програми**

```
import com.anylogic.engine.*;
import com.anylogic.libraries.processmodeling.*;

public class TicketOffice extends Engine {

    public static void main(String[] args) {

        // Create model
        Engine engine = new Engine("Ticket Office");
        Agent owner = engine.getEnvironment();

        // Define blocks
        Source ticketSource = new Source(owner, "Ticket Customers");
        Queue ticketQueue = new Queue(owner, "Queue");
        Service ticketService = new Service(owner, "Ticket Service");
        Sink ticketsSold = new Sink(owner, "Tickets Sold");

        // Set source parameters
        ticketSource.setInterarrivalTime(Exponential.getInstance(50));

        // Set service parameters
        ResourcePool cashiers = new ResourcePool(owner, "Cashiers");
        cashiers.setCapacity(3);
        ticketService.setServiceTime(Triangular.getInstance(1, 2, 3));
        ticketService.setQueueCapacity(100);
        ticketService.setResources(cashiers);
```



```
// Connect blocks
ticketSource.out().to(ticketQueue.in());
ticketQueue.out().to(ticketService.in());
ticketService.out().to(ticketsSold.in());

// Counter
Counter ticketsSoldCount = new Counter(owner, "Tickets Sold");
ticketService.out().stats(ticketsSoldCount);

// Run model
engine.run();

// Output results
System.out.println("Tickets sold: " + ticketsSoldCount.getCount());

}

}

// Additional features
// Log queue length over time
QueueHistogram queueLengthStats = new QueueHistogram(owner, "Queue Length");
ticketQueue.stats(queueLengthStats);
// Output results
System.out.println("Tickets sold: " + ticketsSoldCount.getCount());

// Display queue length statistics
System.out.println("Average queue length: " + queueLengthStats.getAverage());
```

```
System.out.println("Maximum queue length: " + queueLengthStats.getMax());
import com.anylogic.engine.*;
import com.anylogic.libraries.processmodeling.*;

public class TicketOffice extends Engine {

    public static void main(String[] args) {
        // Create model
        Engine engine = new Engine("Ticket Office");
        Agent owner = engine.getEnvironment();

        // Define blocks
        Source ticketSource = new Source(owner, "Ticket Customers");
        Queue ticketQueue = new Queue(owner, "Queue");
        Service ticketService = new Service(owner, "Ticket Service");
        Sink ticketsSold = new Sink(owner, "Tickets Sold");

        // Set source parameters
        ticketSource.setInterarrivalTime(Exponential.getInstance(50));

        // Set service parameters
        ResourcePool cashiers = new ResourcePool(owner, "Cashiers");
        cashiers.setCapacity(3);
        ticketService.setServiceTime(Triangular.getInstance(1, 2, 3));
        ticketService.setQueueCapacity(100);
        ticketService.setResources(cashiers);

        // Connect blocks
        ticketSource.out().to(ticketQueue.in());
```

```
ticketQueue.out().to(ticketService.in());
ticketService.out().to(ticketsSold.in());

// Counter
Counter ticketsSoldCount = new Counter(owner, "Tickets Sold");
ticketService.out().stats(ticketsSoldCount);

// Additional features
// Log queue length over time
QueueHistogram queueLengthStats = new QueueHistogram(owner, "Queue Length");
ticketQueue.stats(queueLengthStats);

// Log service time for each transaction
Histogram serviceTimeStats = new Histogram(owner, "Service Time");
ticketService.stats(serviceTimeStats);

// Log cashier utilization
ResourcePoolHistogram cashierUtilizationStats = new
ResourcePoolHistogram(owner, "Cashier Utilization");
cashiers.stats(cashierUtilizationStats);

// Run model
engine.run();

// Output results
System.out.println("Tickets sold: " + ticketsSoldCount.getCount());
System.out.println("Average queue length: " + queueLengthStats.getAverage());
System.out.println("Maximum queue length: " + queueLengthStats.getMax());
System.out.println("Average service time: " + serviceTimeStats.getAverage());
```

```

System.out.println("Cashier Utilization: " + cashierUtilizationStats.getAverage());
}
}
// Додати паузу після кожної транзакції для кращого візуального спостереження
ticketService.out().doAfterExit(transaction -> engine.pause());
// Запустити симуляцію з певного моменту часу (наприклад, після 1000 одиниць
часу)
engine.start(1000);
// Збільшення кількості касирів під час симуляції
schedule.scheduleRepeat(() -> cashiers.setCapacity(cashiers.getCapacity() + 1), 100,
500);
// Зміна параметрів сервісу під час симуляції (наприклад, збільшення часу
обслуговування)
schedule.scheduleRepeat(() ->
ticketService.setServiceTime(Triangular.getInstance(1, 2, 4)), 500, 500);
import com.anylogic.engine.*;
import com.anylogic.libraries.processmodeling.*;

public class TicketOffice extends Engine {

public static void main(String[] args) {
// Create model
Engine engine = new Engine("Ticket Office");
Agent owner = engine.getEnvironment();

// Define parameters
int initialCashierCount = 3; // Початкова кількість касирів
double cashierServiceTimeMin = 1.0; // Мінімальний час обслуговування касира
double cashierServiceTimeMax = 3.0; // Максимальний час обслуговування касира
int maxQueueCapacity = 100; // Максимальна ємність черги

```

```
double interarrivalTimeMean = 50.0; // Середній інтервал між клієнтами

// Define blocks
Source ticketSource = new Source(owner, "Ticket Customers");
Queue ticketQueue = new Queue(owner, "Queue");
Service ticketService = new Service(owner, "Ticket Service");
Sink ticketsSold = new Sink(owner, "Tickets Sold");

// Set source parameters
ticketSource.setInterarrivalTime(Exponential.getInstance(interarrivalTimeMean));

// Set service parameters
ResourcePool cashiers = new ResourcePool(owner, "Cashiers");
cashiers.setCapacity(initialCashierCount);
ticketService.setServiceTime(Triangular.getInstance(cashierServiceTimeMin,
cashierServiceTimeMax));
ticketService.setQueueCapacity(maxQueueCapacity);
ticketService.setResources(cashiers);

// Connect blocks
ticketSource.out().to(ticketQueue.in());
ticketQueue.out().to(ticketService.in());
ticketService.out().to(ticketsSold.in());

// Counter
Counter ticketsSoldCount = new Counter(owner, "Tickets Sold");
ticketService.out().stats(ticketsSoldCount);

// Additional features
```

```
// Log queue length over time
QueueHistogram queueLengthStats = new QueueHistogram(owner, "Queue Length");
ticketQueue.stats(queueLengthStats);

// Log service time for each transaction
Histogram serviceTimeStats = new Histogram(owner, "Service Time");
ticketService.stats(serviceTimeStats);

// Log cashier utilization
ResourcePoolHistogram cashierUtilizationStats = new
ResourcePoolHistogram(owner, "Cashier Utilization");
cashiers.stats(cashierUtilizationStats);

// Run model
engine.run();

// Output results
System.out.println("Tickets sold: " + ticketsSoldCount.getCount());
System.out.println("Average queue length: " + queueLengthStats.getAverage());
System.out.println("Maximum queue length: " + queueLengthStats.getMax());
System.out.println("Average service time: " + serviceTimeStats.getAverage());
System.out.println("Cashier Utilization: " + cashierUtilizationStats.getAverage());
}
}
```


**Додаток В (обов'язковий)****ІЛЮСТРАТИВНА ЧАСТИНА  
«ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ІМІТАЦІЇ  
РОБОТИ КВИТКОВОЇ КАСИ»**

Виконав: студент 2-го курсу, групи  
2КН-22м

спеціальності 122 – «Комп'ютерні  
науки»

Нетребський М. А. 

Керівник: к.т.н., доц. каф. КН

Богач І. В. 

« 07 » 12 2023 р.

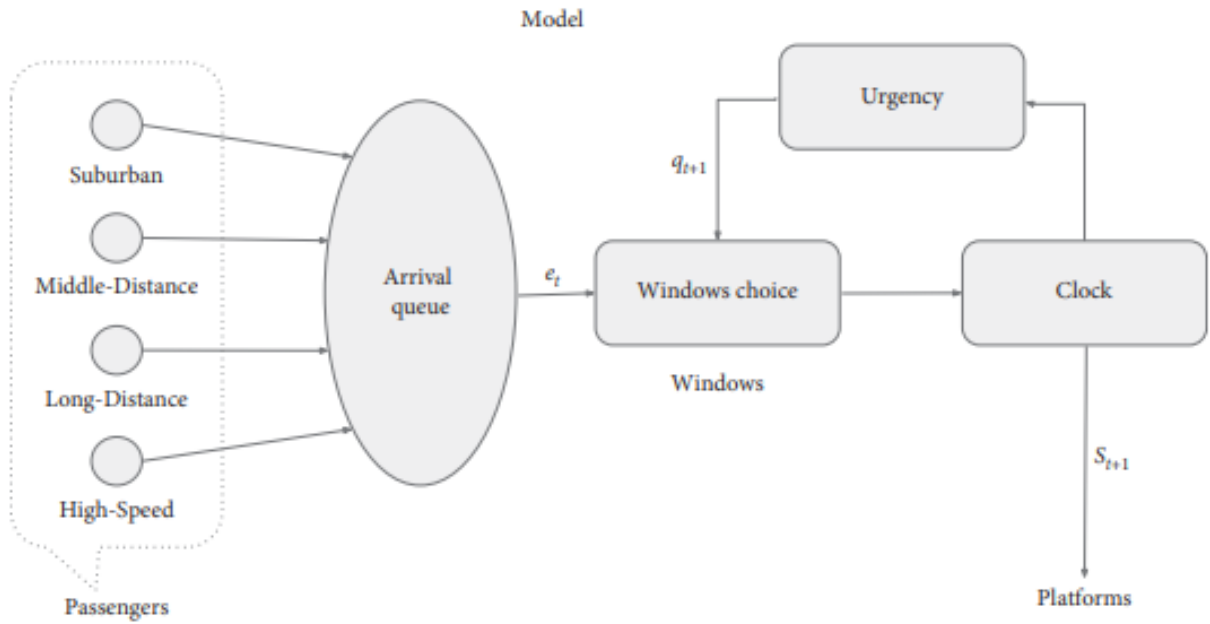


Рисунок В.1 - Діаграма взаємодії агентів системи

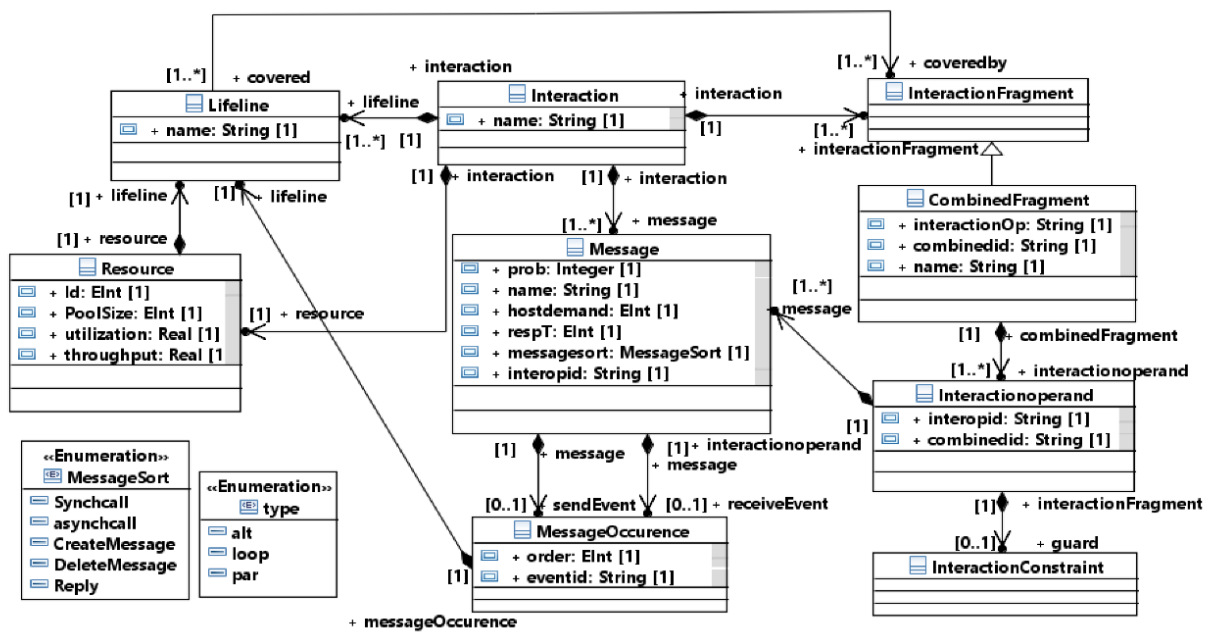


Рисунок В.2 - Метамодель UML діаграми послідовностей



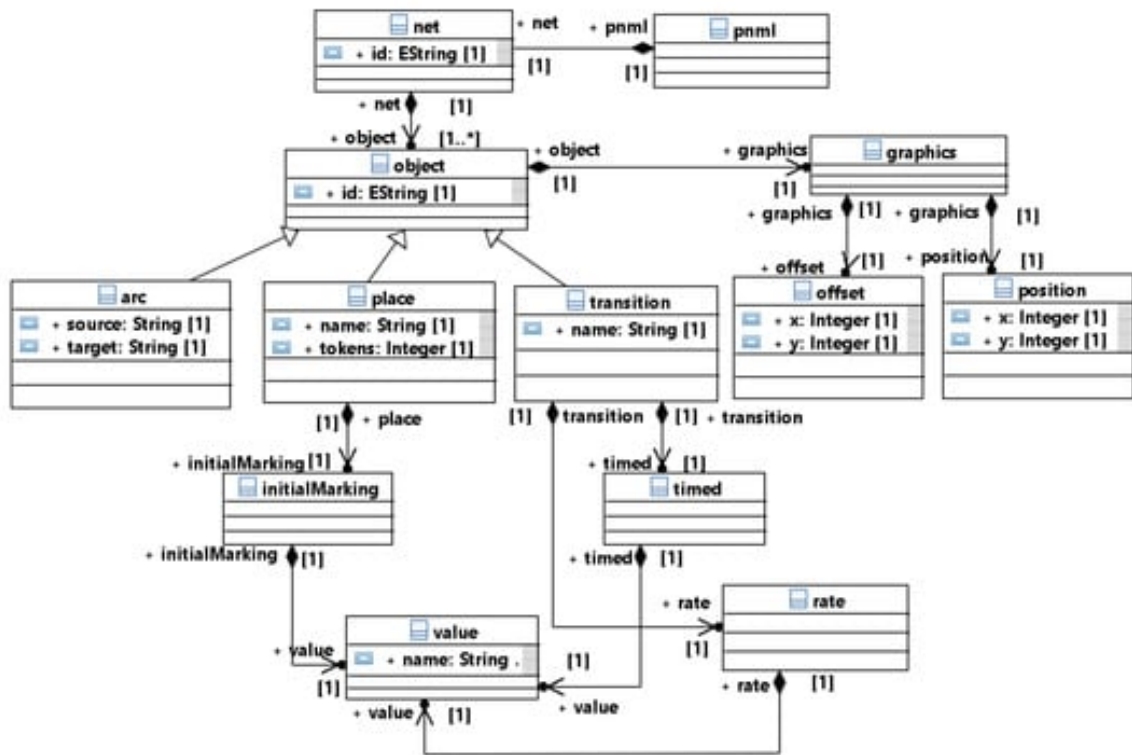


Рисунок В.3 - Метамодел ь PNML (модифікована UML діаграма класів)

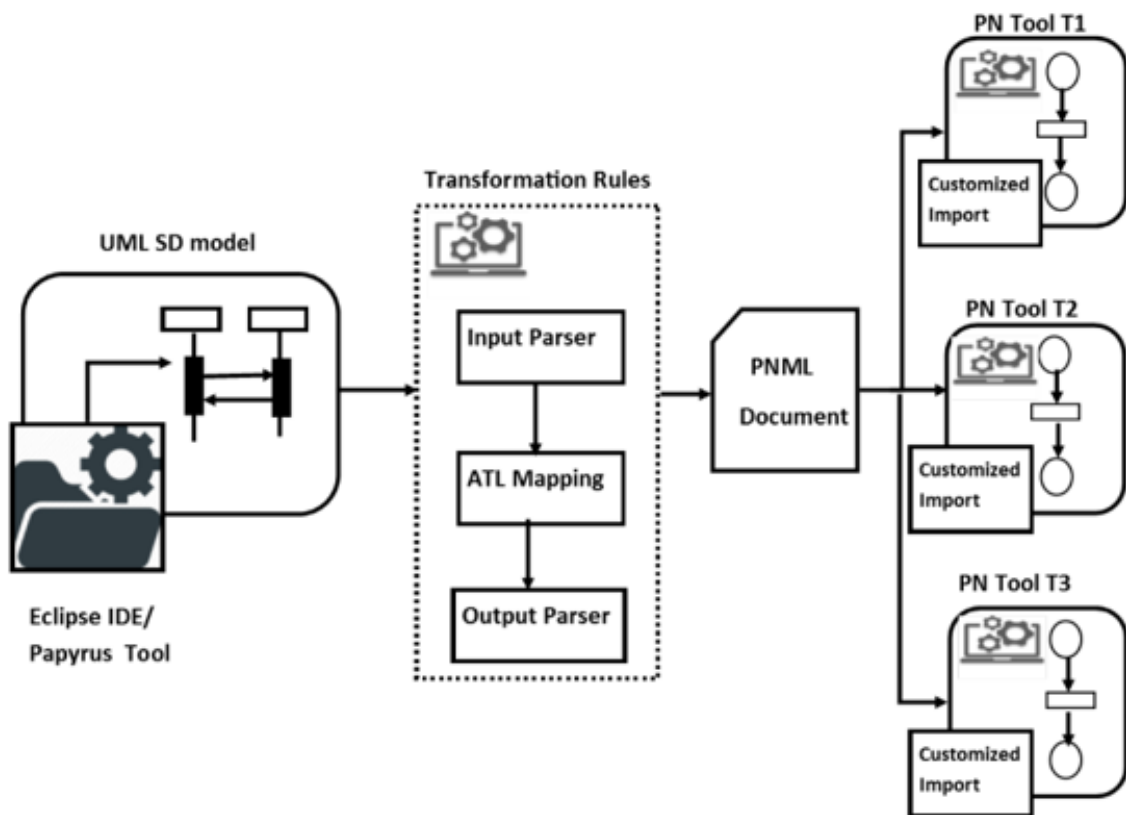


Рисунок В.4 – Діаграма компонентів

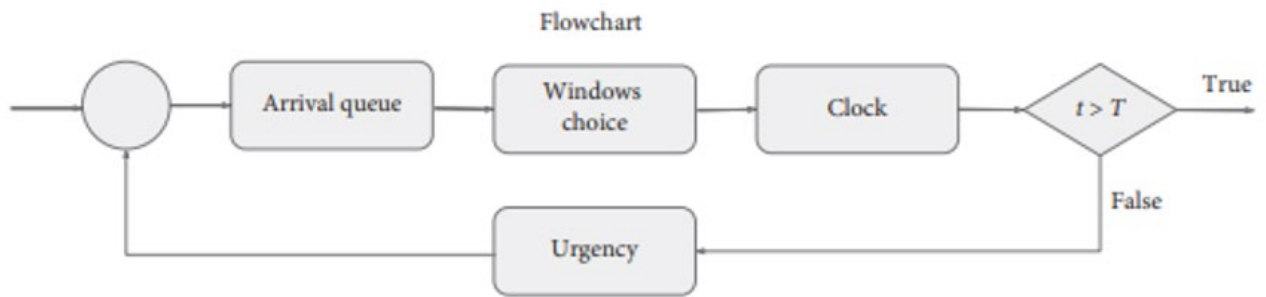


Рисунок В.5 – Діаграма взаємодії черги пасажирів

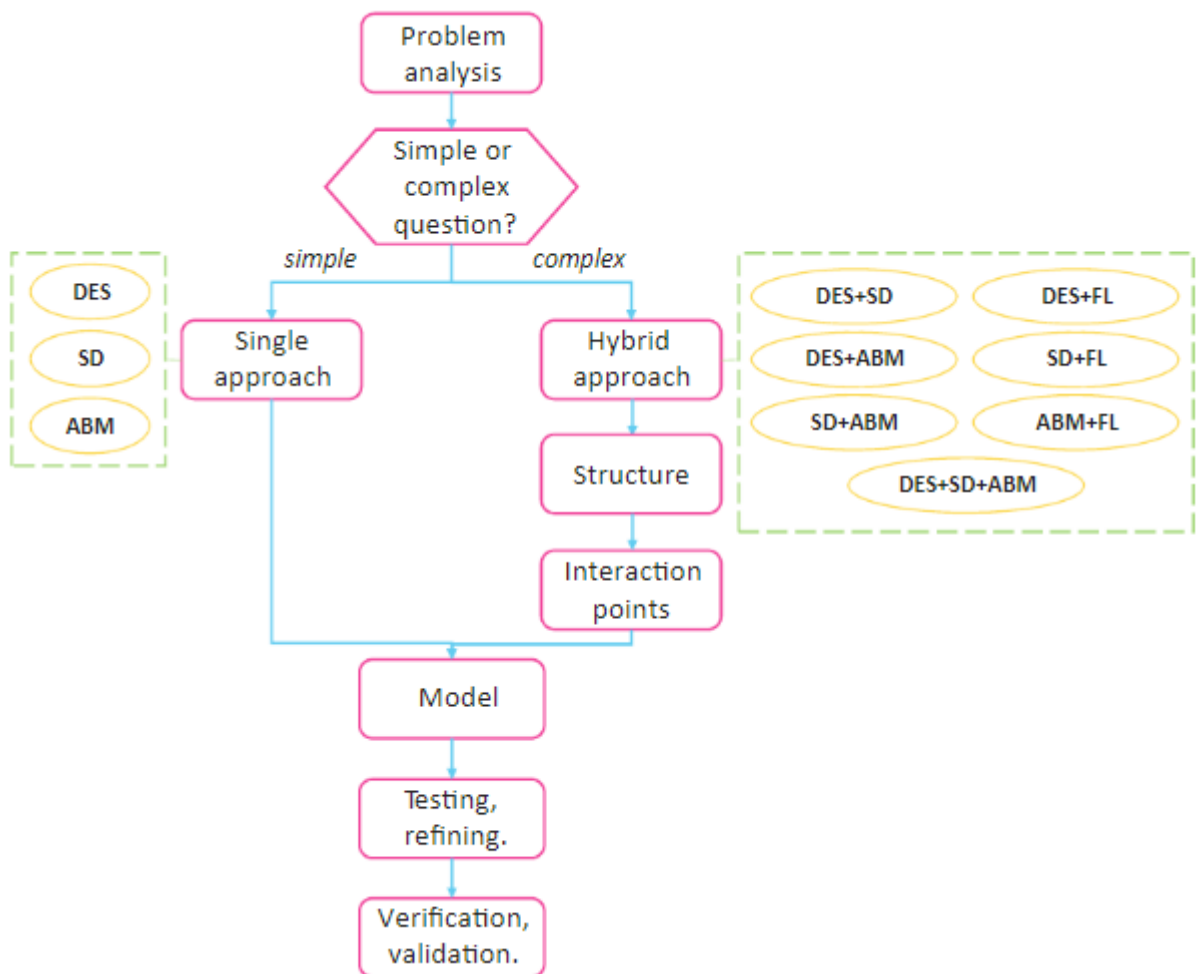


Рисунок В.6 – Схема побудови імітаційної моделі

## Додаток Г (довідниковий)

### Інструкція користувача

Для того щоб почати роботу з інформаційною технологією, потрібно мати персональний комп'ютер.

#### *Крок 1: Необхідно запуснути моделі*

Для цього потрібно відкрити програму AnyLogic на комп'ютері та завантажити імітаційну модель роботи квиткової каси (рисунок Г.1).

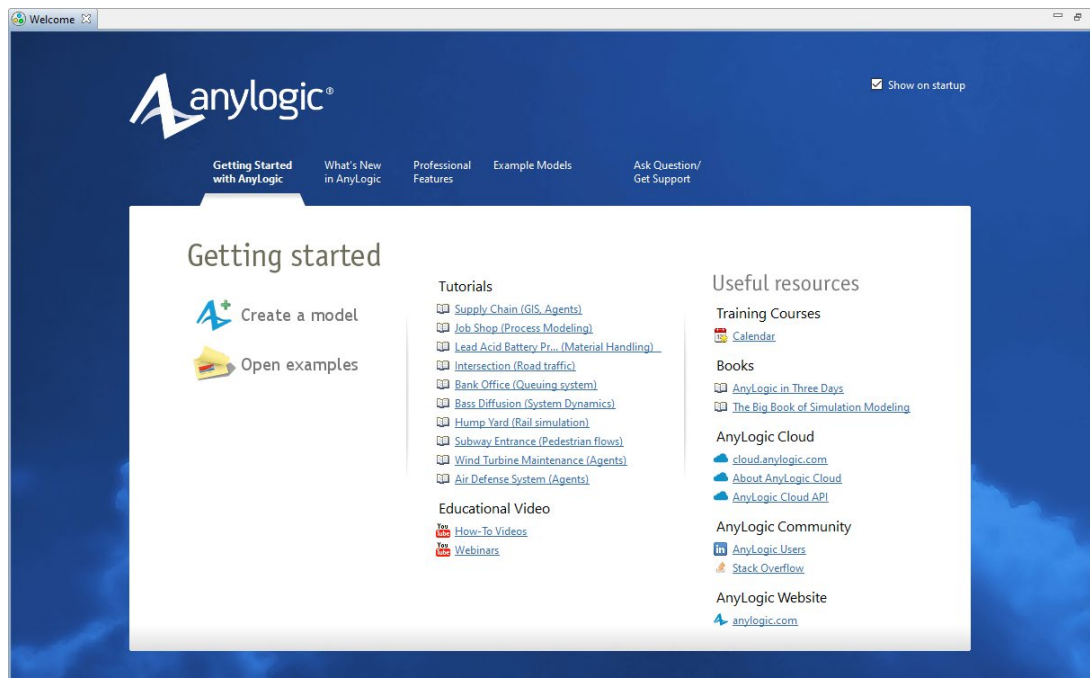


Рисунок Г.1 – Вікно запуску моделі

#### *Крок 2: Налаштування параметрів системи*

Блоки Моделі: Ознайомтеся з різними блоками моделі, такими як "Джерело" (вхід клієнтів), "Черга", "Обслуговування" та інші.

Параметри: Перегляньте та налаштуйте параметри, такі як час між клієнтами, кількість касирів тощо.

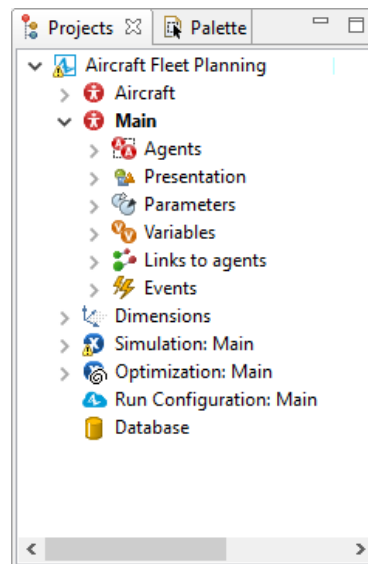


Рисунок Г.2 – Вікно налаштування параметрів

### Крок 3: Запуск Симуляції

Щоб почати симуляцію натисніть кнопку "Run" або "Пуск".

Спостерігайте за роботою квиткової каси, вивчайте статистику та результати.

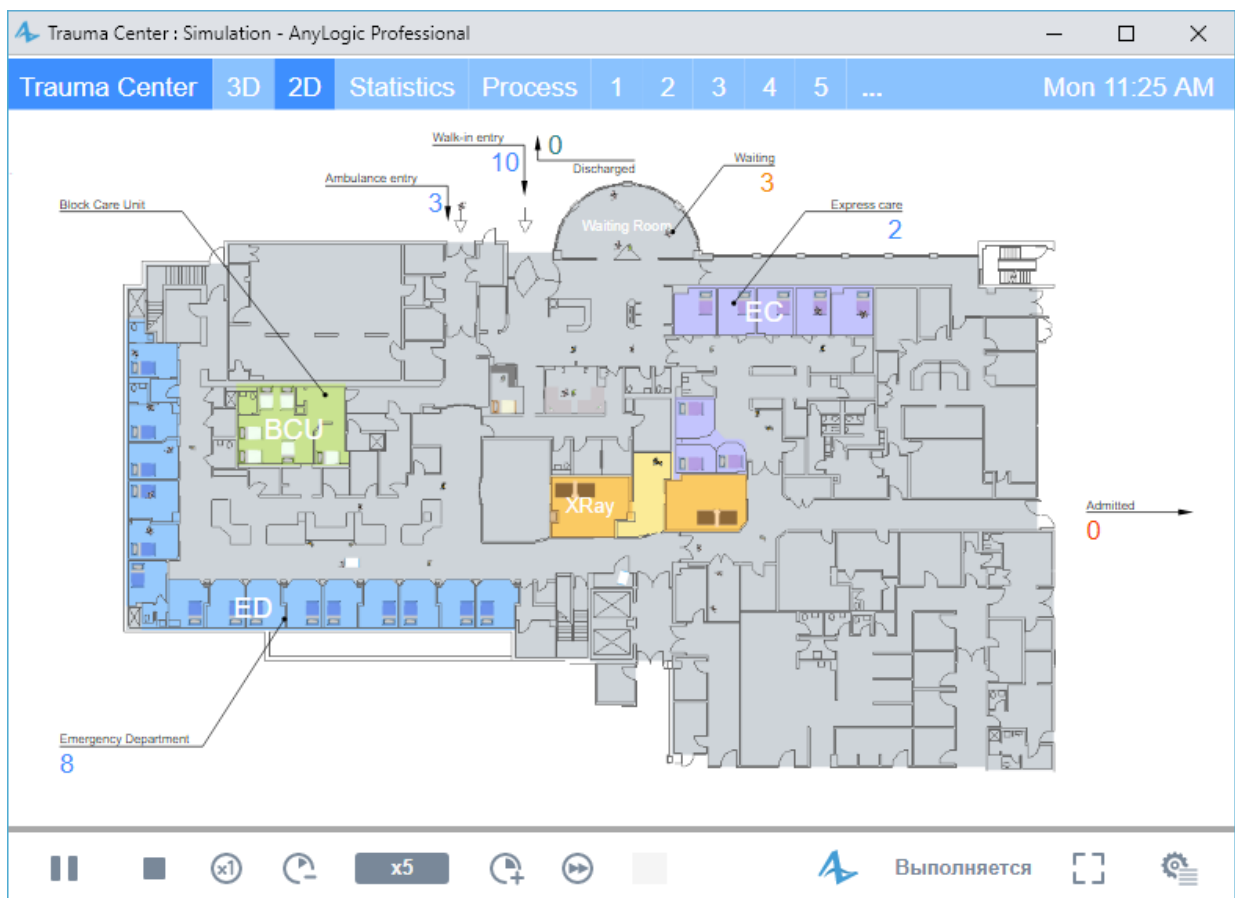


Рисунок Г.3 – Симуляція роботи квиткової каси

#### Крок 4: Аналіз результатів

Перегляньте дані, зібрані під час симуляції, такі як кількість проданих квитків, середній час очікування та інші параметри.

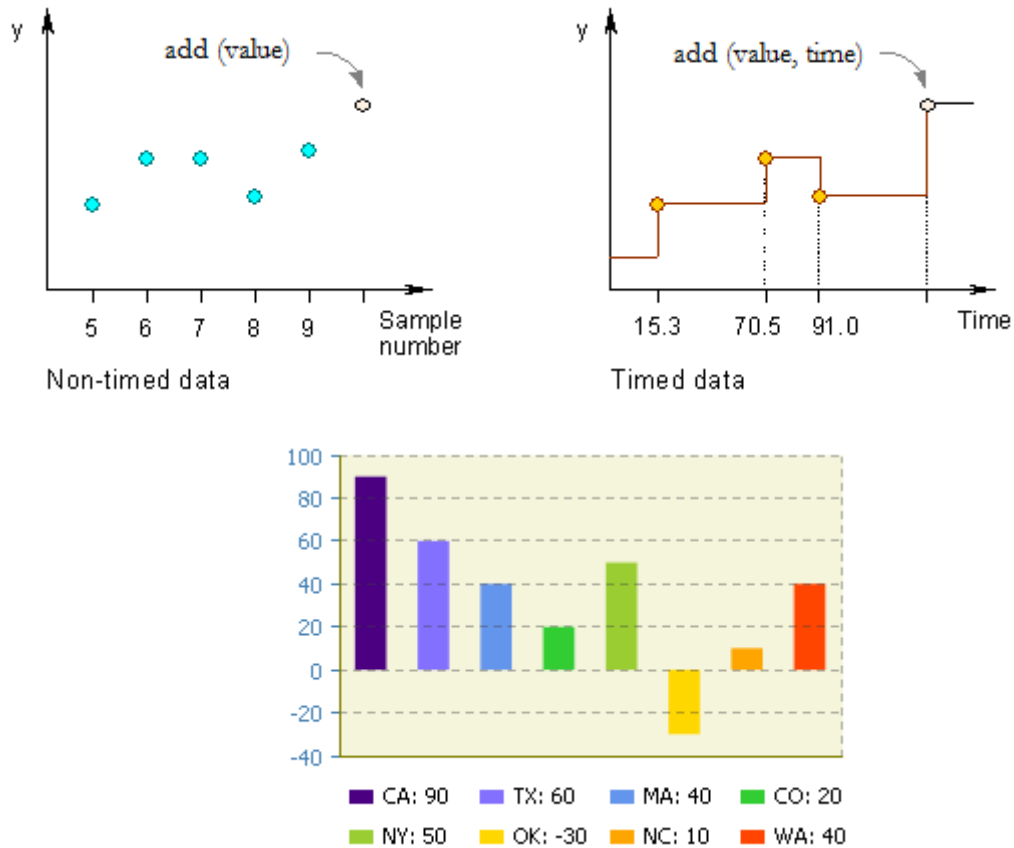


Рисунок Г.4 – Результати запуску моделі

Зверніть увагу на графіки та діаграми, що відображають динаміку системи.

#### Крок 5: Відшукування оптимізацій

Спробуйте змінити параметри системи та перезапустіть симуляцію, щоб визначити оптимальні значення.

Експериментуйте з кількістю касирів, часом обслуговування тощо.

#### Крок 6: Завершення роботи

Після завершення симуляції, збережіть результати або експортуйте графіки за необхідності.