

**МАГІСТЕРСЬКА РОБОТА**

**МР. ШМ - 41.00.00.000 ПЗ**

**Група ШМ-23-1**

**Дрогомирецький Володимир**

**2024**

**Івано-Франківський національний технічний університет нафти і газу**

Інститут інформаційних технологій

Кафедра інженерії програмного забезпечення

**Дрогомирецький Володимир Тарасович**

(прізвище, ім'я, по батькові)

УДК 004.942  
(індекс)

## **МАГІСТЕРСЬКА РОБОТА**

**Модель-базовані підходи проєктування цифрових образів для**

**віртуальних комунікацій**

(назва роботи)

**Інженерія програмного забезпечення**

(назва освітньої програми)

**121 - Інженерія програмного забезпечення**

(шифр і назва спеціальності)

**Дрогомирецький В.Т.**

(підпис, ініціали та прізвище здобувача освітнього ступеня)

**Науковий керівник Піх Володимир Ярославович, к.т.н., доцент**

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

**Допущено до захисту**

Завідувач кафедри

доц. Бандура В.В.

(посада) (підпис) (дата) (ініціали та прізвище)

**Нормоконтроль**

доц. Вовк Р.Б.

(посада) (підпис) (дата) (ініціали та прізвище)

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

Івано-Франківськ – 2024

**Івано-Франківський національний технічний університет нафти і газу**

Інститут інформаційних технологій

Кафедра інженерії програмного забезпечення

Освітній рівень магістр

Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Зав. кафедрою

ІПЗ

доц.

В.В. Бандура

“ 04 ” вересня 2024 р.

# ЗАВДАННЯ

## НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Дрогомирецькому Володимирі Тарасовичу

(прізвище, ім'я, по-батькові)

**1. Тема магістерської роботи “ Модель-базовані підходи проєктування цифрових образів для віртуальних комунікацій ”**

керівник проекту (роботи) Піх Володимир Ярославович, к.т.н., доцент

затверджені наказом закладу вищої освіти від “ 22 ” листопада 2024 р. № 781/7

**2. Строк подання студентом проекту (роботи) 15 грудня 2024 р.**

**3. Вихідні дані до проекту (роботи) Теоретичні концепції та формальні моделі побудови та функціонування інформаційних та програмних технологій проєктування цифрових образів**

**4. Зміст розрахунково - пояснювальної записки(перелік питань, які потрібно розробити)**

1. Дослідження предметної області проєктування цифрових образів для віртуальних комунікацій

2. Методологія розробки програмного забезпечення для систем віртуальних комунікацій

3. Представлення структури та аналіз вимог до системи розумних віртуальних комунікацій

4. Імплементация модель-базованої методології проєктування системи віртуальних комунікацій

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

1. Процес розробки (рис. 2.1)

2. Модульний підхід до розробки програмного забезпечення (рис. 2.2)

3. Приклад виконаної моделі засобами SysML (рис. 2.3)

4. Процедура тестування через різні етапи (рис. 2.4)

5. Етапи розробки проекту з використанням V-моделі (рис. 2.5)

## 6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата
Перевірка на плагіат	доц., к.т.н. Вовк Р.Б.	

7. Дата видачі завдання 04 вересня 2024 р.

Керівник \_\_\_\_\_

(підпис)

Завдання прийняв до виконання \_\_\_\_\_

(підпис)

## КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назви етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Підбір і вивчення літератури по темі магістерської роботи	15.09.2024	виконано
2	Аналіз концепцій та алгоритмів предметної області	29.09.2024	виконано
3	Дослідження предметної області проєктування цифрових образів для віртуальних комунікацій	15.10.2024	виконано
4	Методологія розробки програмного забезпечення для систем віртуальних комунікацій	08.11.2024	виконано
5	Представлення структури та аналіз вимог до системи розумних віртуальних комунікацій	20.11.2024	виконано
6	Імплементация модель-базованої методології проєктування системи віртуальних комунікацій	01.12.2024	виконано
7	Затвердження пояснювальної записки роботи завідувачем кафедри	15.12.2024	виконано

Студент – магістр \_\_\_\_\_

(підпис)

Керівник роботи \_\_\_\_\_

(підпис)

## АНОТАЦІЯ

**Магістерська робота:** 77 с., 41 рис., 52 джерела.

**Тема:** Модель-базовані підходи проектування цифрових образів для віртуальних комунікацій

**Об'єкт дослідження:** системи та моделі побудови цифрових двійників для віртуальних комунікацій.

**Мета роботи:** розробка методології та інструментарію для проектування цифрових образів на основі Digital Twin для систем віртуальних комунікацій з використанням багаторівневої архітектури.

**Предмет дослідження:** модель-базовані методи та підходи для проектування, моделювання і впровадження цифрових образів у системах віртуальних комунікацій.

### **Результати дослідження**

В роботі запропонована методологія розробки та використання моделі цифрового двійника в реальному часі на платформі Unity забезпечує підвищення інтерактивності та функціональності віртуальних комунікацій.

### **Висновок**

Впроваджено систему Digital Twin в Unity Game Engine, включаючи розробку 3D моделей та багаторівневу архітектуру, яка охоплює всі необхідні етапи обробки й відображення даних. Це дозволяє точно моделювати фізичні процеси у віртуальному середовищі, що сприяє вдосконаленню системи цифрових комунікацій.

**ВІРТУАЛЬНІ КОМУНІКАЦІЇ, МОДЕЛЮВАННЯ, РУШІЙ  
UNITY, БАГАТОРІВНЕВА АРХІТЕКТУРА, ІНТЕРАКТИВНІ ЦИФРОВІ  
ОБРАЗИ, СИСТЕМА ЦИФРОВОГО ДВІЙНИКА, МОДУЛЬНИЙ  
ПІДХІД, V-MODEL**

## ABSTRACT

**Master Thesis:** 77 pp., 41 fig., 52 sources.

**Thesis Subject:** Model-based approaches to designing digital images for virtual communications

**Research object:** systems and models of building digital doubles for virtual communications.

**The purpose of the work:** development of a methodology and toolkit for designing digital images based on Digital Twin for a virtual communications system using a multi-level architecture.

**Research subject:** model-based methods and approaches for designing, modeling and implementing digital images in virtual communication systems.

### **Research results**

The work proposes a methodology for the development and use of a real-time digital double model on the Unity platform, which provides increased interactivity and functionality of virtual communications.

### **Conclusion**

Implemented Digital Twin system in Unity Game Engine, including 3D model development and multi-level architecture, which completes all additional stages of data processing and display. This allows accurate modeling of physical processes in a virtual environment, which improves the improvement of the digital communication system.

**VIRTUAL COMMUNICATIONS, SIMULATION, UNITY ENGINE, MULTILEVEL ARCHITECTURE, INTERACTIVE DIGITAL IMAGES, DIGITAL DOUBLE SYSTEM, MODULAR APPROACH, V-MODEL**

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ .....	9
ВСТУП.....	10
<b>РОЗДІЛ 1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ПРОЄКТУВАННЯ ЦИФРОВИХ ОБРАЗІВ ДЛЯ ВІРТУАЛЬНИХ КОМУНІКАЦІЙ .....</b>	<b>13</b>
1.1. Опис предмету дослідження.....	13
1.2. Опис концепції “цифрових двійників” (Digital Twin).....	15
1.3. Методи моделювання системи цифрового двійника (DTS).....	17
1.3.1. Архітектури Digital Twin.....	18
Висновки до розділу .....	23
<b>РОЗДІЛ 2. МЕТОДОЛОГІЯ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ СИСТЕМ ВІРТУАЛЬНИХ КОМУНІКАЦІЙ ТА DIGITAL TWIN.....</b>	<b>24</b>
2.1. Представлення модульного підходу до розробки програмного забезпечення.....	24
2.2. Етапи розробки створення системи віртуальних комунікацій .....	25
2.2.1. Модель у циклі (MIL) .....	25
2.2.2. Програмне забезпечення в циклі (SIL) .....	27
2.2.3. Цифровий двійник у циклі (Digital-Twin-in-Loop - DTIL) .....	27
2.3. Метод розробки проекту з використанням V-Model.....	28
2.4. Представлення структури та аналіз вимог до системи розумних віртуальних комунікацій .....	30
2.4.1. Система HVAC.....	32
2.4.2. Безпека та підсистема доступу.....	33
2.4.3. Підсистема старт освітлення .....	34
2.4.4. Аудіосистема .....	35
2.5. Проектування розумної системи віртуальних переговорів .....	35
2.5.1. Режими роботи системи .....	35

2.5.2. Поведінка підсистем (пристроїв і драйверів).....	39
2.6. Багаторівнева архітектура для цифрового образу для розумної системи віртуальних комунікацій .....	43
2.6.1. Рівень абстракції даних .....	43
2.6.2. Рівень передачі даних .....	44
2.6.3. Рівень цифрового моделювання.....	44
2.6.4. Рівень інтеграції даних/моделі.....	45
2.6.5. Рівень програми/сервісу .....	45
2.6.7. Вимоги до системи .....	45
Висновки до розділу .....	47
РОЗДІЛ 3. ІМПЛЕМЕНТАЦІЯ МОДЕЛЬ-БАЗОВАНОЇ МЕТОДОЛОГІЇ ПРОЄКТУВАННЯ ЦИФРОВИХ ОБРАЗІВ ДЛЯ ВІРТУАЛЬНИХ КОМУНІКАЦІЙ .....	49
3.1. Впровадження моделі .....	49
3.2. Реалізація взаємодії віртуальної моделі з Unity .....	53
3.2.1. Моделювання введення користувача та відображення .....	53
3.3. Впровадження системи Digital Twin в Unity Game Engine .....	57
3.3.1. Розробка 3D моделей.....	58
3.3.2. Реалізація багаторівневої архітектури Digital Twin .....	60
3.4. Демонстрація проекту розумної системи віртуальних комунікацій .....	62
Висновки до розділу .....	69
ВИСНОВКИ .....	71
ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	73

## **ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ**

AR – Augmented Reality  
VR – Virtual Reality  
MR – Mixed Reality  
HCI – Human-Computer Interaction  
AI – Artificial Intelligence  
DT – Digital Twin  
CGI – Computer-Generated Imagery  
XR – Extended Reality  
IoT – Internet of Things  
CAD – Computer-Aided Design  
VRI – Virtual Reality Interface  
DVA – Digital Virtual Avatar  
CNS – Communication in Networked Spaces  
DED – Digital Environment Design  
FBS – Function-Behavior-Structure  
SL – Second Life (virtual world platform)  
VPL – Virtual Presence Level  
AGV – Autonomous Guided Vehicle

## ВСТУП

### **Актуальність теми.**

Актуальність теми роботи зумовлена швидким розвитком цифрових технологій, які змінюють підходи до віртуальної взаємодії, забезпечуючи більш реалістичну, інтерактивну та персоналізовану комунікацію. Концепція Digital Twin (цифрового двійника) стає інноваційною технологією, яка дозволяє створювати цифрові копії реальних об'єктів, систем і процесів з можливістю відтворення їхньої поведінки в реальному часі. Це відкриває нові горизонти для розробки віртуальних середовищ, де користувачі можуть взаємодіяти з реалістичними цифровими аналогами об'єктів, що особливо важливо для симуляційних платформ, управлінських систем, навчальних і тренувальних програм.

У сучасному світі багато сфер, від освіти і промисловості до медицини і розумного міста, потребують створення інтерактивних та динамічних віртуальних середовищ, що здатні відтворювати функціональність реальних об'єктів. Це обумовлює необхідність у нових підходах до розробки таких середовищ, які б враховували потребу в точності відображення, високій інтерактивності та адаптивності до різних сценаріїв застосування. Digital Twin дозволяє здійснювати постійний моніторинг і аналіз даних з реальних об'єктів, що покращує можливість прогнозування і підтримку прийняття рішень у віртуальних системах.

Розвиток методологій та інструментів для створення цифрових двійників є важливим аспектом для формування ефективних систем віртуальної комунікації, які стають основою для організації віддаленої роботи, навчання та комунікацій у віртуальному просторі. Запропоноване дослідження дозволяє створити багаторівневу архітектуру цифрових двійників, що забезпечує гнучкість у реалізації різних функцій, полегшує інтеграцію нових компонентів та підтримує адаптацію під вимоги

користувача. Тому результати роботи можуть знайти широке застосування в різноманітних сферах, які потребують якісної віртуальної інтерактивності.

Отже, актуальність даного дослідження обумовлена необхідністю підвищення якості та ефективності віртуальних комунікацій через використання цифрових двійників для створення високофункціональних, адаптивних та реалістичних цифрових образів.

**Мета дослідження** – розробка методології та інструментарію для проєктування цифрових образів на основі Digital Twin для систем віртуальних комунікацій з використанням багаторівневої архітектури.

**Об'єкт дослідження** – системи та моделі побудови цифрових двійників для віртуальних комунікацій.

**Предмет дослідження** – модель-базовані методи та підходи для проєктування, моделювання і впровадження цифрових образів у системах віртуальних комунікацій.

**Задачі дослідження:**

- Проаналізувати концепцію Digital Twin та визначити її ключові елементи для віртуальних комунікацій.
- Розробити багаторівневу архітектуру для цифрових образів в системах віртуальних комунікацій.
- Дослідити методологію V-Model для структурованої розробки цифрових двійників.
- Імплементувати модель цифрового двійника на платформі Unity з використанням 3D-моделей.
- Протестувати систему з точки зору інтерактивності та реалістичності взаємодії користувача з цифровими образами.

**Методи дослідження:**

- Аналіз і синтез для визначення предметної області Digital Twin.
- Моделювання для створення цифрових двійників об'єктів.
- Прототипування та імплементация для розробки інтерактивних 3D-моделей на платформі Unity.

- Тестування для оцінки функціональності системи.

**Наукова новизна роботи** полягає в розробці багаторівневої архітектури для інтеграції Digital Twin у системи віртуальних комунікацій, що дозволяє ефективніше моделювати складні цифрові образи.

#### **Практичне значення результатів**

Результати дослідження можуть бути використані для створення інтерактивних середовищ з підтримкою Digital Twin у системах управління, симуляційних тренажерах, навчальних платформах, що сприятиме підвищенню якості та ефективності віртуальних комунікацій. Запропонована архітектура та реалізація дозволяють адаптувати рішення до широкого спектра завдань у реальному часі.

**Структура магістерської роботи.** Робота складається зі вступу, трьох розділів та висновків. Загальний обсяг роботи становить 77 сторінок, і містить 41 рисунок, список використаних джерел із 53 найменувань.

# РОЗДІЛ 1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ПРОЄКТУВАННЯ ЦИФРОВИХ ОБРАЗІВ ДЛЯ ВІРТУАЛЬНИХ КОМУНІКАЦІЙ

## 1.1. Опис предмету дослідження

Потреба в інтелектуальному середовищі завжди була на піку з моменту початку революції технологій та Інтернету речей (IoT). Переосмислити навколишній світ, який поводить ся як пов'язаний і інтегрований організм, який вирішує більшість повсякденних механічних і повторюваних завдань, може бути чимось дуже цікавим. У цьому контексті передові інтелектуальні будівлі, безумовно, є корисною сферою досліджень. «Розумна будівля — це та, яка реагує на потреби мешканців, задовольняє цілі організації та довгострокові прагнення суспільства» [1].

Для визначення ефективної інтелектуальної будівлі життєво важливі різні процеси життєвого циклу, включаючи планування, проектування, будівництво, введення в експлуатацію та управління об'єктами. Усі ці процеси зазвичай трудомісткі та дорогі. Таким чином, дуже важливо розробити деякі загальні методології та стандарти для прийняття, щоб встановити економічно ефективні способи виконання процесів життєвого циклу інтелектуальної будівлі. Проектування та тестування контролерів, які використовуються в таких програмах, є першим і головним кроком. У розробці контролера існують різні рівні тестування. Однак моделі, створені традиційними методами, не завжди відповідають практичній роботі обладнання. Проблема розбіжних моделей становить велику проблему. Тому необхідно включити додатковий метод, який вирішує проблему розбіжних моделей. Метою цього проекту є створення моделі в контурі, програмного забезпечення в контурі та цифрового подвійного контролера в циклі для програми «розумна кімната», яка служить методом тестування для перевірки контролера.

Ця робота включає в себе проект з проектування та дослідження системи Digital Twin для Smart Meeting Room (SMR) на основі підходу системної інженерії, керованої моделлю.

Тому проект розділений на чотири етапи. По-перше: моделювання поведінки всіх підсистем SMR і аналіз всіх атрибутів, які взаємодіють із залученими наступними підсистемами. По-друге: створити центральний інтелект і логіку для всієї системи, а також створити програмний код для всіх підсистем і контролера. По-третє: розробити модель Digital Twin у 3D віртуальному середовищі. По-четверте: інтеграція моделей для спостереження за різними сценаріями під час тестування поведінки контролера в змодельованих віртуальних випадках.

Цей проект зосереджується на кожному аспекті проектування системи SMR, який розглядається, зроблених припущеннях, встановленні загальних цілей проекту, створенні моделей для кожної підсистеми, визначенні атрибутів системного зв'язку та обговоренні відповідного методу впровадження для кожної з етапи проекту разом із покроковим посібником із впровадження. Крім того, було проведено дослідження літератури щодо відповідних цифрових подвійних систем. Обговорюється призначення, ефективність та футуристичний вигляд таких систем.

У цьому проекті, по-перше, відповідна система SMR розробляється в інструменті розробника IBM Rhapsody. Для цього було розглянуто кілька підсистем та зроблено початкові припущення. Згодом ці міркування та припущення моделюються як системні вимоги. Програма SMR моделюється за допомогою підходу SysML в інструменті IBM Rhapsody. Програма SMR працює як автономна система, а моделювання перевіряє роботу типової програми SMR, маючи графічний інтерфейс користувача в інструменті. По-друге, проект передбачає проектування та розробку системи цифрових двійників для інтеграції в систему SMR за допомогою інструменту Unity Game Engine. Для цього використовується багат шарова архітектура цифрових двійників для розробки сучасного інтер'єру офісу та інтеграції його

з поведінкою системи програми SMR, яка паралельно працює в інструменті IBM Rhapsody. Однак, щоб забезпечити зв'язок між двома різними процесами, встановлюється мережевий сокетний зв'язок. Нарешті, модель розширюється для тестування HIL шляхом перенесення програми SMR на контролер Raspberry Pi. Однак, щоб забезпечити виконання проекту на віддаленому пристрої, були виконані певні обмеження, такі як створення профілю на основі Linux в інструменті IBM Rhapsody та автоматизація процесу.

Отже, програма SMR, що працює в інструменті IBM Rhapsody, разом з віртуальною моделлю в Unity Game Engine служить методом перевірки, який демонструє роботу системи цифрових двійників для програми SMR.

## **1.2. Опис концепції “цифрових двійників” (Digital Twin)**

Digital Twin служить методом тестування в різних галузях техніки. У цьому розділі обговорюються останні події на ринку Digital Twin, огляд літератури щодо Digital Twin System та системна архітектура Digital Twin System [5].

Digital Twin — це провідна технологія, яка привертає увагу в таких сферах, як виробництво, автомобільна промисловість, охорона здоров'я, роздрібна торгівля, розумні міста та промисловий Інтернет речей. Це інтеграція всіх сучасних інтелектуальних технологій, включаючи Big Data, штучний інтелект, машинне навчання та IoT, які використовуються для прогностного аналізу будь-якої системи чи обладнання [3]. Digital Twin поєднує дані в реальному часі та прогностні дані з існуючих програмних продуктів. Іншим важливим елементом є технологія, необхідна для візуалізації інформації, яка надходить від Digital Twin Systems. Він допомагає простим і ефективним способом підвищити економічну ефективність, оптимізувати енергоспоживання, дискретні датчики швидкості та часу в різних місцях, збирає дані та консолідує їх в одному концентраторі, що

дозволяє кожному отримати доступ до даних, які діють як фізичне середовище для цифрового простору [4].

Цифровий Twining має можливість віртуально відображати елементи та динаміку функціонування та роботи Інтернету речей. Поєднання фізичних елементів і віртуальних елементів сприяє кращому виробництву, високій толерантності, визначенню ефективності продукту через інше середовище. У Digital Twins фізичні дані впливають на дизайн продукту для кращої продуктивності та кращого виробництва перед тим, як дані передаються в цифровий простір, динаміка інформації аналізується за допомогою різноманітних аналізів, таких як обробка відео, обробка природної мови, акустичний аналіз, відео аналіз тощо, щоб перевірити дисперсію та допуски. Якщо коротко, то Digital Twin — це складна модель, керована даними, яка має цілі майже в усіх сферах.

Ось ключові аспекти концепції Digital Twin:

- Віртуальне представлення: Digital Twin створює цифровий аналог фізичного об'єкта, відтворюючи його геометрію, властивості, поведінку та навіть оточення.

- Зв'язок з реальним світом: Digital Twin отримує дані з датчиків, встановлених на фізичному об'єкті, що дозволяє йому відображати стан об'єкта в реальному часі.

- Симуляція та аналіз: Digital Twin дозволяє проводити симуляції різних сценаріїв, тестувати різні умови та аналізувати вплив змін на фізичний об'єкт без ризику для реального об'єкта.

- Оптимізація та прогнозування: Завдяки аналізу даних та симуляціям, Digital Twin може допомогти оптимізувати роботу фізичного об'єкта, прогнозувати його поведінку та попереджати про потенційні проблеми.

Останні досягнення та розробки в Digital Twins включають:

- Digital Twin в дизайні продукту.
- Digital Twin в інтелектуальному виробництві.
- Робота-магазин Digital Twin.

- Моделювання та імітація Digital Twin.
- Розумне підключення та взаємодія Digital Twin.
- Цифровий двійник у співпраці людини і машини.
- Digital Twin в управлінні даними життєвого циклу продукту.

### **1.3. Методи моделювання системи цифрового двійника (DTS)**

Контекст роботи Digital Twin включає в себе приладовий випробувальний стенд, в якому інструменти системної інженерії на основі моделей (MBSE) (наприклад, інструменти моделювання та верифікації системи) та моделювання операційних сценаріїв (наприклад, дискретне моделювання та моделювання на основі агентів) використовуються. Інсайти операційного середовища використовуються для модифікації системних моделей, що використовуються у віртуальному прототипі [5]. Дані, що надаються фізичною системою (PS), використовуються віртуальним прототипом для створення екземпляра цифрової подвійної системи (DTS) [2].

Набір інструментів MBSE надає засоби моделювання процесів системи. Методи моделювання включають матрицю структури дизайну, матрицю структури залежностей процесу, ймовірнісні моделі, такі як частково спостережуваний марковський процес прийняття рішень (POMDP), моделювання дискретних подій, моделювання на основі агентів, розповідь на основі моделі, база знань MBSE та системна інженерія моделі процесів життєвого циклу [5].

Останні галузеві тенденції підвищили обсяг спроб досліджень у сфері розробки Digital Twin на рівнях будівель і міст. У цьому розділі обговорюються існуючі визначення, пов'язані з DTS. Нові ідеї можна отримати за допомогою огляду поточної літератури, в якій обговорюються обмеження, пов'язані з дослідницькими зусиллями щодо концепцій DTS в архітектурі, інженерії та будівництві та промисловості управління об'єктами

(сектор АЕС/FM). Метою цього розділу є ознайомлення з системною архітектурою, яка використовувалася під час розробки Digital Twins.

### *1.3.1 Архітектуру Digital Twin*

Існують різні багаторівневі архітектури для підтримки неоднорідних середовищ (наприклад, багатофункціональні та великий обсяг даних). Його можна класифікувати як кіберфізичні системи (CPS), архітектури платформи IoT та архітектури розумних міст. Digital Twins може підтримувати багато різних програм, таких як управління енергією, безпека та моніторинг стану. Вимоги до даних відрізняються для кожної програми. Таким чином, це стає проблемою, коли дані надходять з різних систем, оскільки система може використовувати ці дані по-різному, що не повністю відповідає вимогам усіх цих програм.

У цьому розділі обговорюється системна архітектура, що використовується для DTs, які спеціально розроблені як на рівні будівлі, так і на рівні міста, на основі демонстратора DT, розробленого для сайту West Cambridge (Developing a Digital Twin at Building and City Levels: A Case Study of West Cambridge Campus - Qiuchen Lu) [6]. На рисунку представлена модель DT для рівня будівлі.

Місто — це комплексна система, яка поєднує різні фізичні, соціальні та бізнес-аспекти. Місто можна розглядати як актив, який об'єднує різні підактиви, такі як будівлі, мости, комунікації, інфраструктура та люди (субцифровий двійник). Типовий Digital Twin рівня будівлі складається з кластера датчиків, платформи керування датчиками, доступу менеджера/користувача, служб, платформи керування активами та з'єднання з хмарною базою даних. Зображення моделі Digital Twin для рівня будівлі зображено на рисунку 1.1. Подібним чином у цьому проекті різні підсистеми, такі як система HVAC, аудіосистема, система безпеки, система доступу та безпеки та система освітлення, можна розглядати як суб-цифрові двійники.

Кожен суб-цифровий двійник має зв'язок цифрового близнюка на різних рівнях.

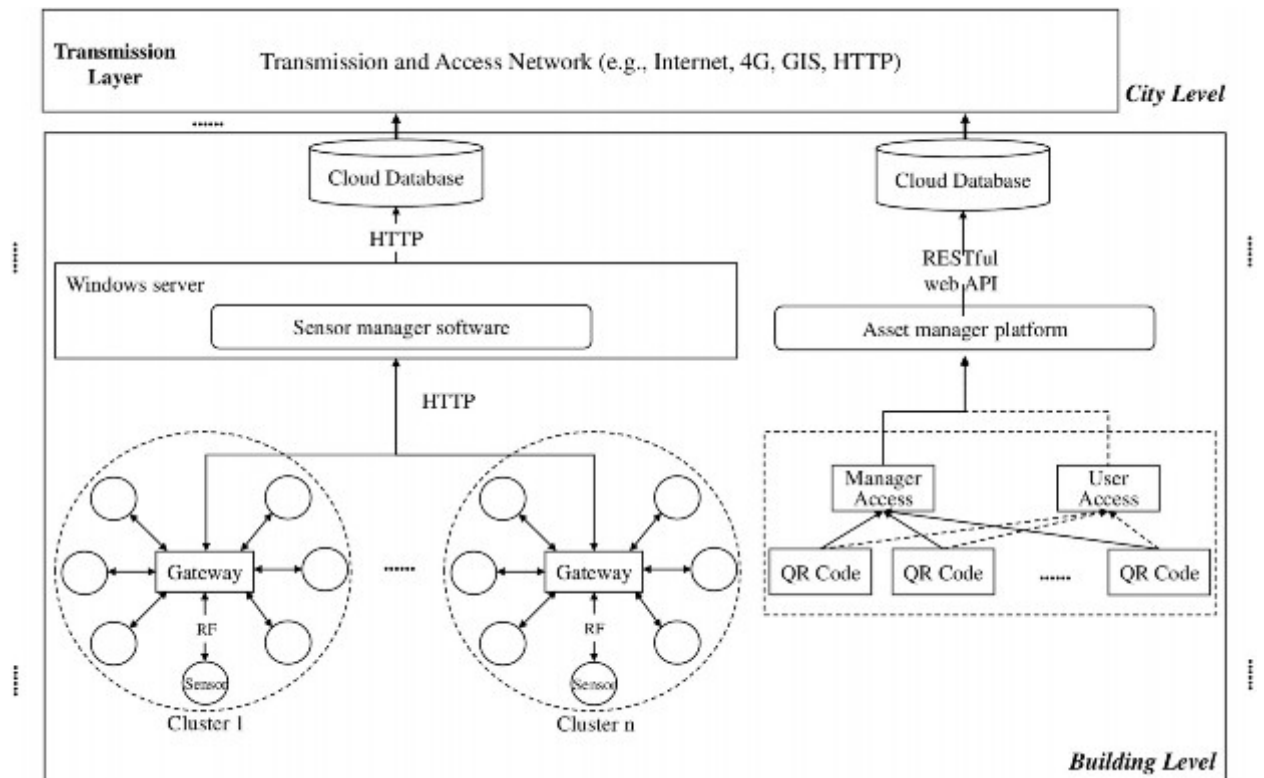


Рис. 1.1. Модель Digital Twin для рівня будівлі

На рисунку 1.1 представлена модель Digital Twin для рівня будівлі, яка демонструє архітектуру системи збору, передачі та обробки даних для створення та управління цифровим двійником. Розглянемо її детальніше:

#### 1. Рівень будівлі:

- Кластери сенсорів: Будівля обладнана кількома кластерами сенсорів (Cluster 1, Cluster n), кожен з яких містить:
  - Сенсори: Збирають дані про різні параметри будівлі (температура, вологість, освітлення, рух тощо).
  - Шлюз (Gateway): Приймає дані від сенсорів та передає їх на вищий рівень через RF (радіочастотний) зв'язок.
  - QR-коди: Використовуються для ідентифікації обладнання та отримання доступу до інформації про нього через систему Digital Twin.

- Доступ: Система передбачає різні рівні доступу до інформації (Manager, User).

## 2. Рівень передачі:

- Transmission and Access Network: Забезпечує передачу даних між рівнем будівлі та міським рівнем. Можуть використовуватися різні технології (Інтернет, 4G, GIS, HTTP).

## 3. Міський рівень:

- Хмарні бази даних: Зберігають дані, зібрані з сенсорів у будівлі, а також інформацію про активи будівлі.

- Програмне забезпечення для управління сенсорами: Обробляє дані з сенсорів, контролює їх роботу та забезпечує їх інтеграцію з іншими системами.

- Платформа управління активами: Надає інструменти для управління активами будівлі, включаючи їх моніторинг, аналіз та оптимізацію.

- RESTful web API: Забезпечує доступ до даних та функціональності системи Digital Twin через веб-інтерфейс.

- Windows server: Сервер, на якому розміщено програмне забезпечення для управління сенсорами та платформа управління активами.

- HTTP: Протокол зв'язку між сервером та хмарними базами даних.

Загалом, ця модель Digital Twin для рівня будівлі дозволяє:

- Збирати та аналізувати дані про стан будівлі в реальному часі.
- Моніторити роботу обладнання та систем.
- Оптимізувати використання ресурсів та енергоспоживання.
- Покращувати управління активами будівлі.
- Забезпечувати доступ до інформації про будівлю для різних користувачів.

Ця модель може бути адаптована до різних типів будівель та застосувань, залежно від конкретних потреб.

Системна архітектура розвитку Digital Twin у місті є багаторівневою архітектурою. Ця архітектура складається з 5 рівнів:

- рівня збору даних,
- рівня передачі,
- рівня цифрової моделі,
- рівня інтеграції даних/моделі,
- рівня обслуговування.

Кожен рівень архітектури системи зображено на рисунку 1.2.

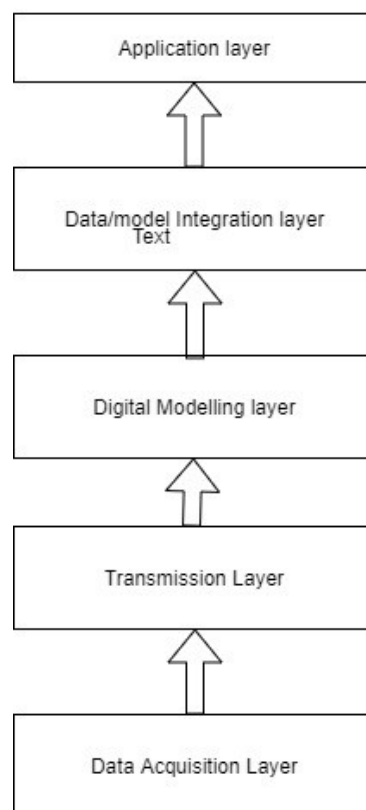


Рис. 1.2. Архітектура системи Digital Twin (DT)

Рівень збору даних є основою кожного Digital Twin. Розробка механізму збору даних і підходу є головним і складним завданням через великий обсяг і неоднорідний характер даних на рівнях міст. Приклади методів збору даних включають безконтактний збір даних (RFID, методи на основі зображень), розподілені сенсорні системи, бездротовий зв'язок і мобільний доступ (середовище WiFi).

Рівень передачі спрямований на передачу отриманих даних на вищі рівні для моделювання та аналізу. Можна використовувати комунікаційні технології, такі як покриття малого радіусу дії, технології мереж доступу (наприклад, WiFi, Zigbee, зв'язок ближнього поля (NFC), M2M і Zwave), а також деякі з ширшого покриття (тобто 3G, 4G, 5G). в цьому шарі. Для розробки DT на рівнях будівель і міст, точність світла (Li-Fi) і LP-WAN розглядалися як перспективні засоби для широкого покриття.

Рівень цифрового моделювання - рівень фактично містить усі цифрові моделі фізичних активів (наприклад, BIM, інформаційне моделювання міста (CIM)) і доповнює інформацію (наприклад, інформацію про погоду, культурне походження), яка підтримує верхні рівні. Різні моделі/типи моделей можна використовувати для різних цілей у Digital Twins. Прикладами цього є: стан/контроль у реальному часі, управління активами (наприклад, модель управління активами), планування інфраструктури/міст (наприклад, CIM), сценарії моделювання та підтримка прийняття рішень.

Рівень інтеграції даних/моделі - рівень спрямований на інтеграцію всіх ресурсів даних на основі розробленої структури даних. Цей рівень також містить функції, необхідні для даних і моделювання, маніпулювання, зберігання, аналізу, обробки. У цій архітектурі функції аналізу та обробки даних у режимі реального часу оновлюватимуть поточний стан міських активів (включаючи транспортні умови та споживання енергії). Тут можна використовувати хмарне сховище та обчислення, а також візуалізацію даних/моделей для досягнення динамічного та ефективного управління даними на рівні міста та будівлі. Крім того, інтелектуальні функції можуть постійно оновлювати свої вбудовані алгоритми та підтримувати безперервні програми в майбутньому розвитку.

Сервісний рівень є верхнім рівнем реалізації архітектури Digital Twin, який інтерпретує знання з ядра та забезпечує взаємодію між людьми/суспільством і рівнем інтеграції даних/моделі. Цей рівень надає

послуги для різних функцій, оцінює продуктивність для побудованих цифрових двійників.

### **Висновки до розділу**

У першому розділі проведено дослідження предметної області, пов'язаної з проєктуванням цифрових образів для віртуальних комунікацій, що є актуальним для підвищення якості та інтерактивності віртуальної взаємодії. Визначено, що цифрові двійники (Digital Twin) забезпечують реалістичне відображення фізичних об'єктів та середовищ у цифровому форматі, сприяючи створенню високоточної віртуальної моделі. Концепція Digital Twin дозволяє розширити можливості комунікації завдяки створенню інтерактивних, динамічних образів, здатних до взаємодії в реальному часі.

Розглянуті методи моделювання системи цифрового двійника (DTS) забезпечують багатоаспектний підхід до створення цифрових образів для віртуальних середовищ. Використання DTS дозволяє відображати зміни та адаптуватися до нових даних, що підвищує гнучкість і достовірність моделі. Це забезпечує широкий спектр можливостей для застосування в різних сферах, включаючи освіту, маркетинг, промисловість і міські цифрові екосистеми.

## РОЗДІЛ 2. МЕТОДОЛОГІЯ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ СИСТЕМ ВІРТУАЛЬНИХ КОМУНІКАЦІЙ ТА DIGITAL TWIN

### 2.1. Представлення модульного підходу до розробки програмного забезпечення

Цілі проекту включають демонстрацію процесу розробки через Model-in-Loop, Software-in-Loop, Digital-Twin-in-Loop та Hardware-in-Loop [8]. Ми знаємо, що створювати моделі, які ідеально відповідають фактичному тестуванню апаратного забезпечення, є складним завданням. Digital-Twin-in-Loop діє як новий інтерфейс, який вирішує проблему розбіжних моделей. Крім того, більшість компаній після тестування програмного забезпечення в циклі переходять до тестування апаратного забезпечення, яке є дорожчим, повільним і схильним до помилок. Тому виникає необхідність включити моделі Digital Twin. Таким чином, новий процес розробки через різні етапи можна побачити на рисунку 2.1. Різні етапи розробки та методи розробки проекту обговорюються в розділах нижче.

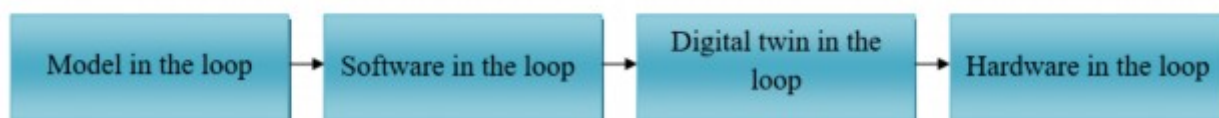


Рис. 2.1. Процес розробки

У цьому проекті розробка Model-in-Loop (MIL) виконується в IBM Rhapsody. Після розробки моделей пристроїв і драйверів генерується програмний код і створюється метамодель програми SMR для керування функціями розумної кімнати. Потім додаток SMR зв'язується з механізмом Unity Game за допомогою сценаріїв C#. На етапі HIL сигнали від програмного контролера будуть додатково перевірені контролером raspberry

рі на відповідність. Далі контролер можна моделювати в Simulink і взаємодіяти з моделями IBM Rhapsody. Як показано на рисунку 2.1, процес розробки перевіряється послідовно на кожному з чотирьох етапів. Питання дослідження тут полягає в тому, щоб з'ясувати шляхи взаємодії між 3D-моделями, розробленими в ігровому движку Unity, і профілюванням SysML, виконаним через IBM Rhapsody за допомогою пакета програмного забезпечення Prespective.

На рис. 2.2 показано модульний підхід до проектування програмного забезпечення [8]. Модульний підхід передбачає відокремлення вузлів введення/виведення для зв'язку між різними блоками. Один функціональний рівень поверх інших функціональних рівнів дозволяє нам змінювати логіку та функціональність системи [8]. Функціональний рівень в основному відповідає за прийняття рішень, обробку даних, динамічне масштабування для виконання функціональних вимог системи. У разі наявності будь-яких майбутніх оновлень окремий блок можна модифікувати та протестувати, що дозволяє уникнути зайвої перевірки моделі та оновлень.

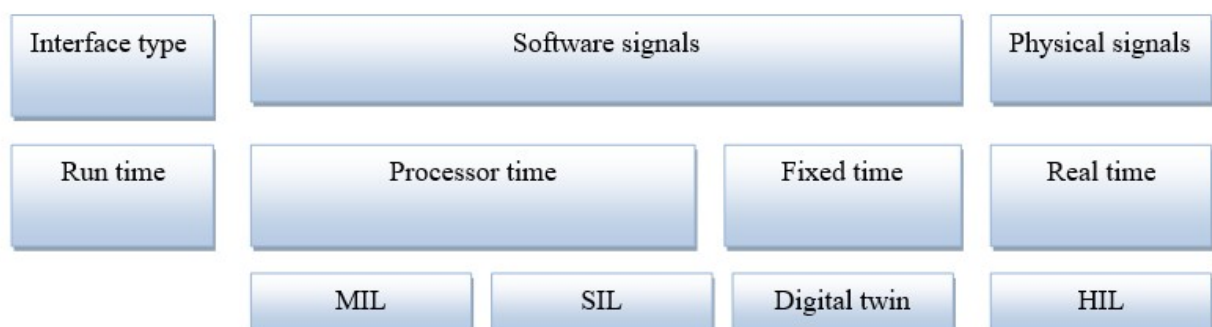


Рис. 2.2. Модульний підхід до розробки програмного забезпечення

## 2.2. Етапи розробки створення системи віртуальних комунікацій

### 2.2.1. Модель у циклі (MIL)

На цьому етапі проектується модель контролера та окремих пристроїв. Розробка MIL виконується в IBM Rhapsody і не потребує жодних фізичних

сигналів. Основною метою МІЛ є створення алгоритму керування та забезпечення того, щоб взаємодія між контролером і моделями не порушувала системні вимоги.

Вимоги визначені як функціональні, загальні та специфічні для кожної з підсистем. Виконання всіх похідних вимог підтверджує валідність тестування МІЛ.

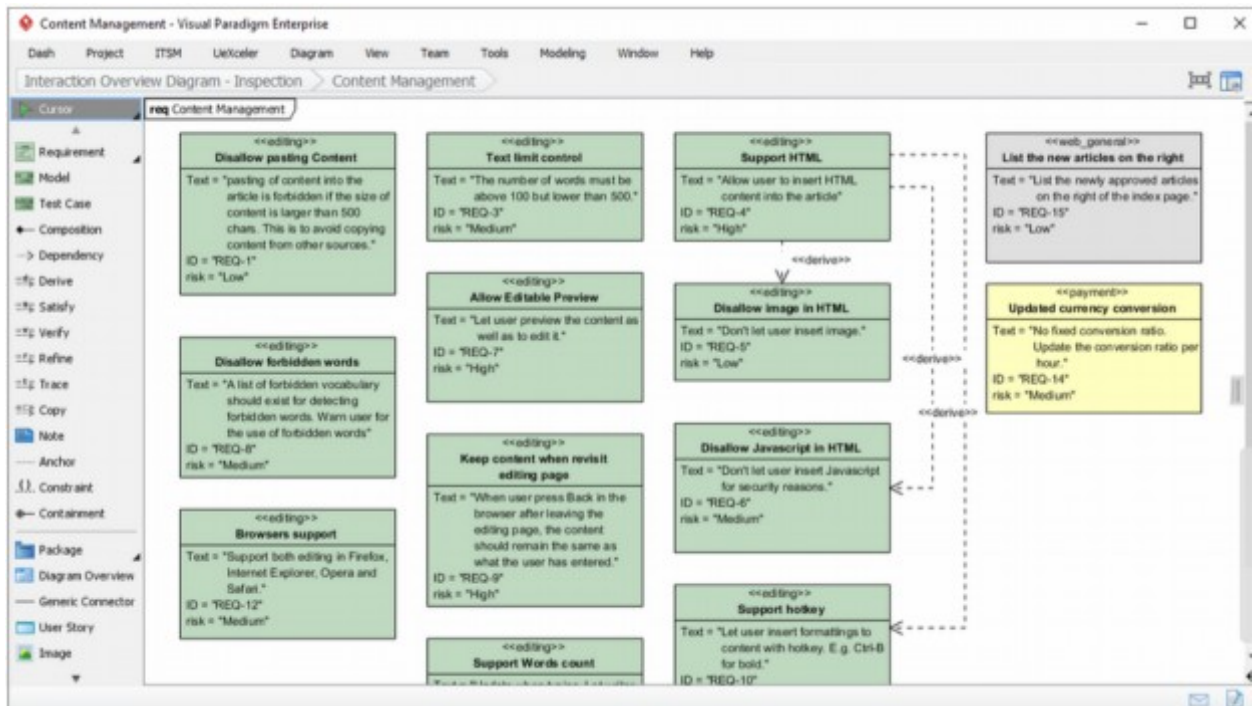


Рис. 2.3. Приклад виконаної моделі засобами SysML

SysML — це базова технологія для системної інженерії на основі моделей (MBSE). SysML є діалектом UML 2 і визначається як профіль UML 2. Профіль UML — це діалект UML, який налаштовує мову за допомогою трьох механізмів: стереотипів, позначених значень і обмежень.

Підхід SysML є економічно ефективним методом застосування систем моделювання для дослідження та тестування характеристик системи. Раннє тестування та валідація характеристик системи сприяють своєчасному отриманню інформації про властивості та поведінку, уможливаючи швидкий зворотний зв'язок щодо вимог і проектних рішень.

### 2.2.2. Програмне забезпечення в циклі (SIL)

Після завершення метамодельовання створюються моделі для генерації коду C# для програмного контролера. Програмний код поєднується зі сценаріями Unity для передачі даних у 3D-середовище, розроблене в Unity. Крім того, сигнали програмного забезпечення є важливими для подальшого тестування НІЛ. Програмні сигнали з IBM Rhapsody переносяться в середовище Raspbian через інтерфейс Python, щоб тестувати контролер Raspberry Pi для тестування НІЛ. Програмне забезпечення також діє як інтерфейс для підключення до механізму Unity за допомогою хеш-сценаріїв C. Це допомагає в обслуговуванні Digital-Twin.

### 2.2.3. Цифровий двійник у циклі (Digital-Twin-in-Loop - DTIL)

DTIL - це концепція, яка розширює ідею цифрових двійників, інтегруючи їх безпосередньо в процес розробки, тестування та управління фізичними системами.

Основна ідея DTIL:

- Цифровий двійник використовується не тільки для моделювання та аналізу, але й для активної взаємодії з фізичною системою в режимі реального часу.

- Цикл означає, що дані з фізичної системи постійно передаються до цифрового двійника, а результати симуляцій та аналізу використовуються для управління фізичною системою

Цифровий двійник віртуально представляє фізичний об'єкт, який можна використовувати для аналізу продуктивності розробленої системи. Проектування для цифрового моделювання Twin має виконуватися за допомогою програмного пакету Prespective, який має вбудовані моделі. Розумна кімната для переговорів, яка складається з системи освітлення, аудіо- та візуальних систем, систем безпеки та доступу, а також систем безпеки, моделюється в сценаріях єдності. Взаємодія системи та поведінка контролера, які були змодельовані в IBM Rhapsody, тепер передаються у

віртуальне 3D-середовище. Таким чином, це дає нам змогу фактично спостерігати за реакцією розумної кімнати на різні сценарії, варіанти використання та дії користувачів.

Структура загального процесу розробки зображена на рисунку 2.4. Для полегшення тестування на основі сценаріїв блоки ідентифікуються як драйвер, контролер і віртуальний.

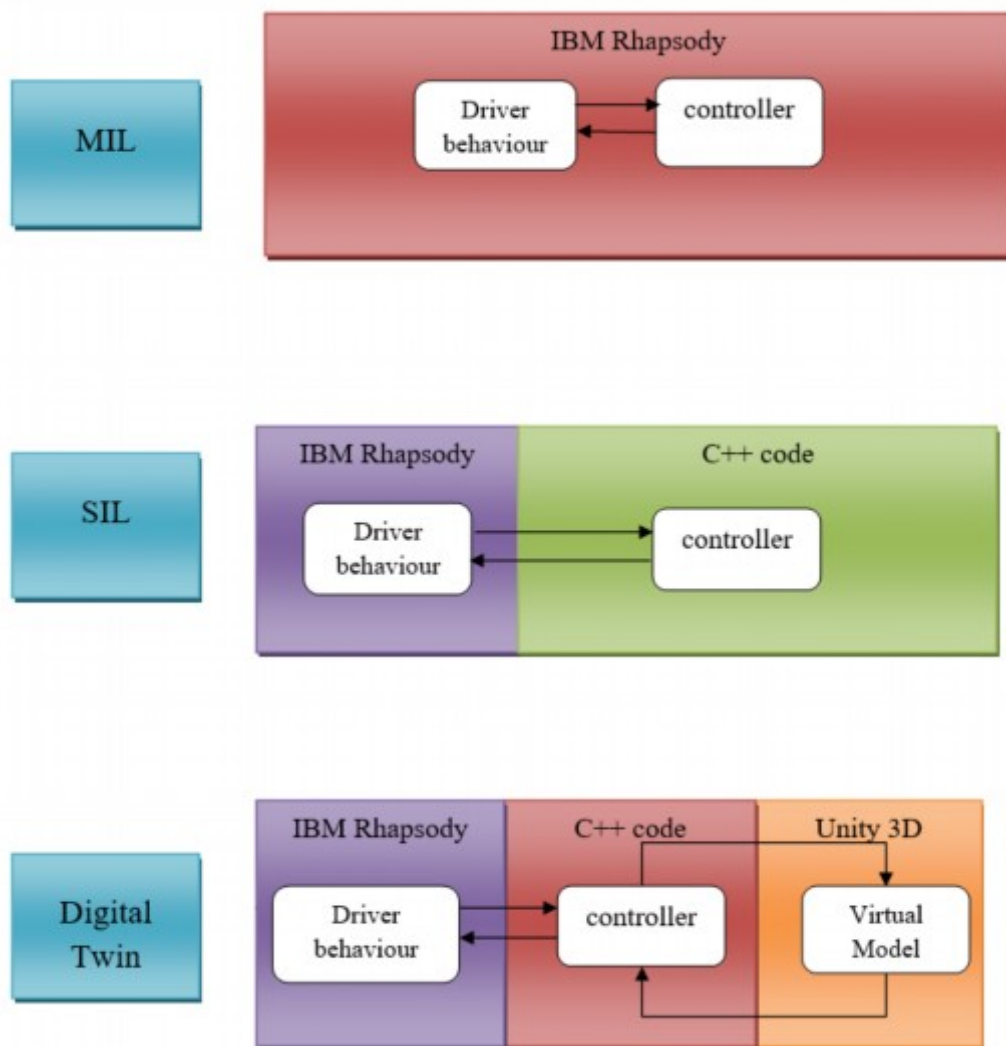


Рис. 2.4. Процедура тестування через різні етапи

### 2.3. Метод розробки проекту з використанням V-Model

Методом дослідження та розробки, який планується використовувати в цьому проекті, є безперервний процес розробки V-моделі, зображений на

рисунку 2.5. У цьому методі проводиться ретельне дослідження, щоб зрозуміти доцільність цього проекту. Вимоги аналізуються та фіксуються на діаграмах вимог відповідно до методу розробки моделі SysML. Аналіз і проектування системи складаються зі створення варіантів використання для різних сценаріїв системи Smart Meeting Room System (SMR). На етапі віртуального аналізу та інтеграції 3D-моделі, створені за допомогою сценаріїв Unity, поєднуються з профілюванням SysML і перевіряються на функціональність і випадки використання. Фізичні сигнали від контролера потім інтегруються з імітованою системою SMR. На етапі впровадження система додатково перевіряється на прийняття. Після того, як модуль інтегровано, протестовано та реалізовано на контролері Raspberry Pi, він тепер готовий до розгортання.

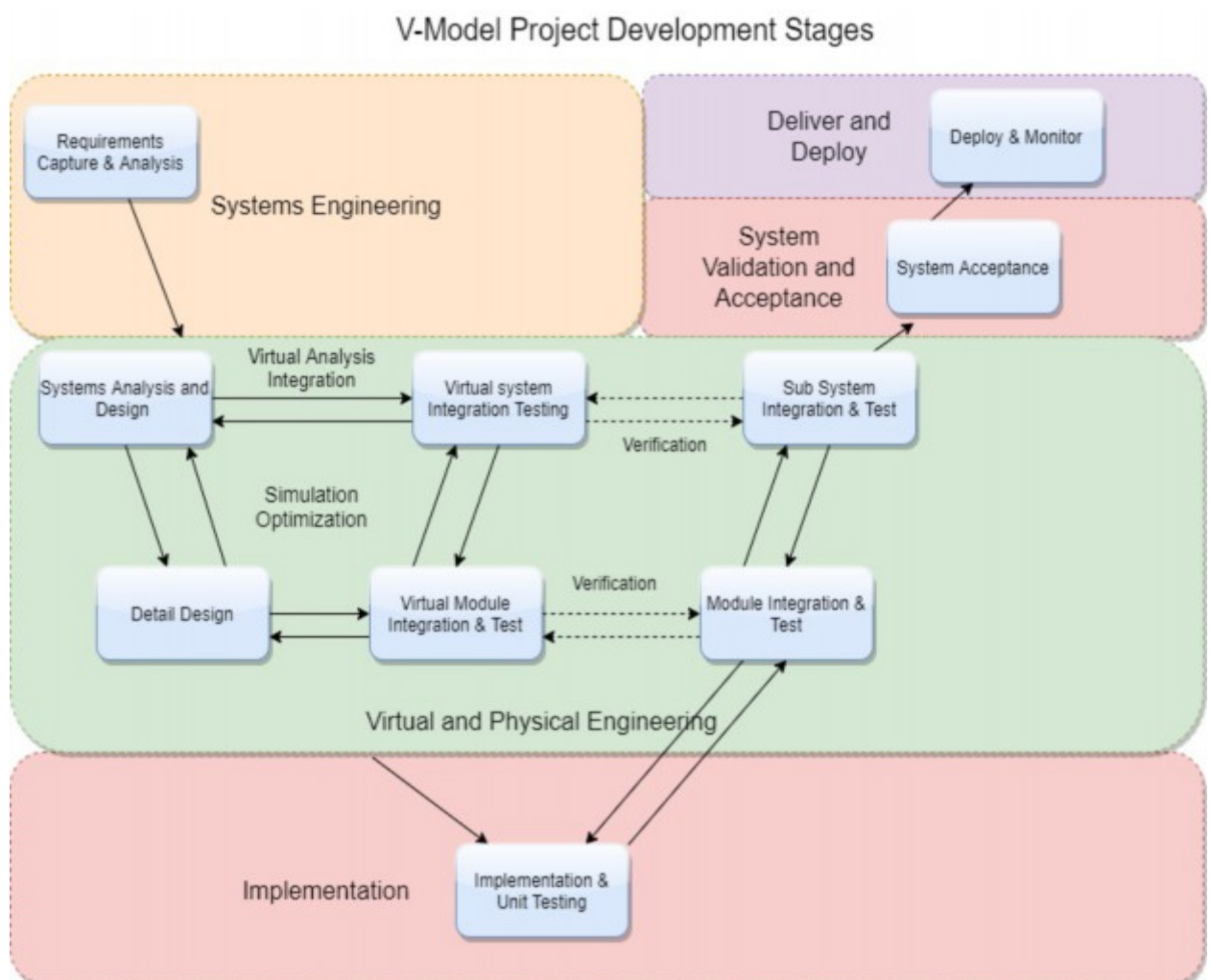


Рис. 2.5. Етапи розробки проекту з використанням V-моделі

## 2.4. Представлення структури та аналіз вимог до системи розумних віртуальних комунікацій

Основна увага моделі системи віртуальних комунікацій (Smart Meeting Room) буде зосереджена на вираженні та записі вимог, дизайні, аналізі та верифікації інформації.

Підхід SysML до моделювання Smart Meeting Room складається з наступних кроків:

а) Класифікація вимог на основі варіантів використання та відображення їх у діаграмі вимог.

б) Реалізація діаграм варіантів використання для подальшого аналізу вимог.

в) Розробка діаграм послідовності з можливістю моделювання та діаграм станів для моделювання динамічної поведінки системи

г) Реалізація пакетних схем для чіткого визначення структури системи. Система SMR може бути представлена на контекстній діаграмі, як показано на рисунку 2.6.

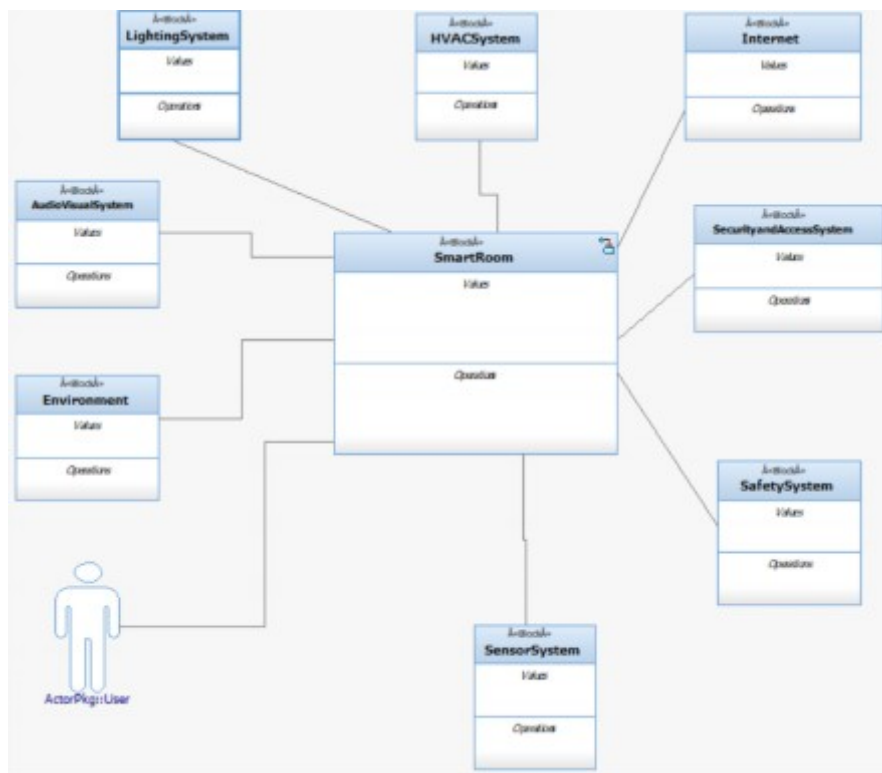


Рис. 2.6. Контекстна діаграма для системи SMR з її підсистемами

Компоненти, необхідні для моделі системи віртуальних комунікацій (SMR):

- Інтелектуальні лампи, система HVAC, пожежні датчики, датчики CO2 і датчики присутності.
- Аудіосистема, динаміки якої ввімкнено для різних налаштувань групи.
- Система безпеки та доступу, яка надає доступ за допомогою 3-значного PIN-коду (використовує 3 - значний числовий код для автентифікації користувачів та надання їм доступу до певного ресурсу або приміщення). Важливо пам'ятати, що 3-значний PIN-код не забезпечує високого рівня безпеки. Для критично важливих систем рекомендується використовувати більш надійні методи автентифікації.

Послуги, необхідні в моделі SMR

- Система повинна контролювати та контролювати температуру в кімнаті інтелектуальним способом.
- Температура і освітлення в приміщенні повинні бути з урахуванням присутності людини в приміщенні.
- Користувач може віддалено отримати доступ до всіх послуг, які надає Smart Meeting Room Classification of Requirements для моделі SMR.

Виходячи з цілей і варіантів використання, вимоги можна класифікувати наступним чином. Усі вимоги, що відповідають наведеній нижче класифікації, додані в розділ додатка.

- Загальні вимоги
- Вимоги до комунікацій
- Вимоги безпеки
- Вимоги щодо безпеки та доступу
- Вимоги до контролю температури та повітря

Окремі підсистеми, їх склад датчиків і системи можна зобразити за допомогою діаграми моделі домену, як показано на рисунку 2.7.

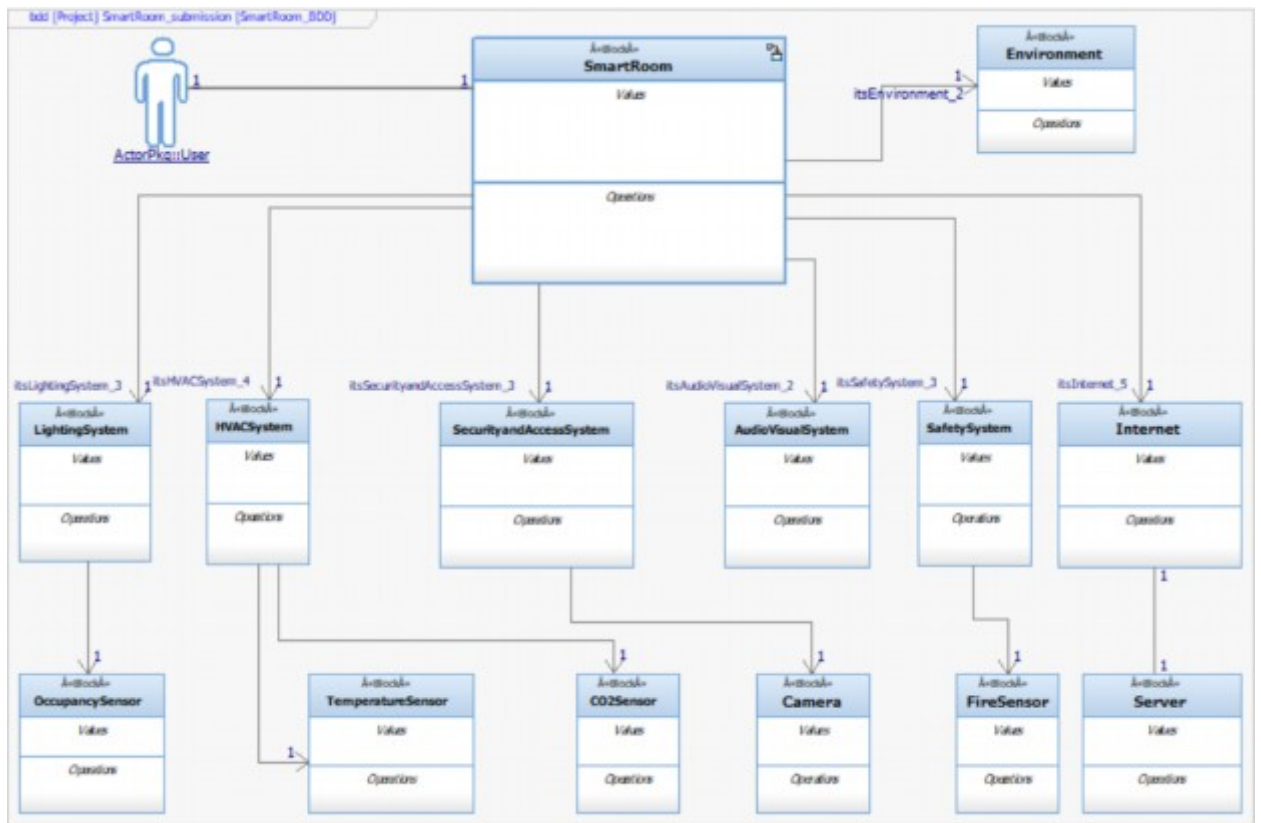


Рис. 2.7. Діаграма моделі домену системи SMR

#### 2.4.1. Система HVAC

Система HVAC (Heating, Ventilation, and Air Conditioning) призначена для контролю середовища, в якому вона працює. Це досягається шляхом контролю температури приміщення за допомогою нагріву та охолодження. Він також контролює рівень вологості в цьому середовищі, контролюючи рух і розподіл повітря в кімнаті.

Досконаліші моделі пропонуватимуть змінну швидкість вентилятора для зменшення споживання електроенергії, однак вони залишаються неефективними порівняно з багатоступневими системами, і, отже, дорожчі в довгостроковій перспективі. Системи опалення, вентиляції та кондиціонування також можуть бути побудовані для забезпечення контролю вологості, і як зволожувачі, так і осушувачі можуть бути додані як варіанти систем опалення та охолодження. Люди, які живуть у дуже сухому середовищі або в тропіках, вважають ці доповнення до системи необхідними. Зважаючи на це, деякі люди вважають за краще встановлювати окремі

системи зволожувача або осушувача повітря, щоб вони могли контролювати вологість навколишнього середовища без потреби вмикати кондиціонер.

Три основні функції системи опалення, вентиляції та кондиціонування: опалення, охолодження та вентиляція. Мета системи HVAC полягає в тому, щоб середовище кімнати для переговорів було безпечним і комфортним для людей. Безпека в основному стосується якості повітря в приміщенні (IAQ). Повітря в приміщенні має бути насичене киснем і не містить шкідливих газів, і це досягається за допомогою вбудованих сенсорних систем. Датчики, які зазвичай використовуються в системах HVAC, це датчики CO<sub>2</sub>, датчики температури, датчики вологості тощо. Спеціально застосовуючи технологію DT і можливості для світу HVAC, Digital Twins може бути корисним [9]:

- Відстеження продуктивності активів у режимі реального часу для планування технічного обслуговування та заміни;
- Навчання персоналу набагато швидше вивчати ключові аспекти експлуатації та обслуговування;
- Прийняття рішень у режимі реального часу з використанням повторного моделювання та внесення коригувань параметрів ;
- Ресурси виведення з експлуатації з використанням детальних записів продуктивності для аналізу наприкінці терміну експлуатації.

#### *2.4.2. Безпека та підсистема доступу*

Оскільки інформація з навколишнього середовища та різноманітні мультимедіа об'єднані разом із такими туристичними технологіями, як «розумні» міста та автономні автомобілі, зростає попит на вміст просторової інформації з вищою доданою вартістю та потреба в технології захисту просторової інформації. Однак існує проблема з ненадійною або дорогою зміною політики безпеки. Такі проблеми часто виникають у процесі координації або інтеграції систем управління інформацією, які використовуються в державних установах і приватних компаніях. Тому необхідно підтримувати оновлену систему безпеки та доступу, яка може

адаптуватися до майбутніх модифікацій. Система безпеки та доступу, яка розглядається в цьому проекті, має наступні функції для належного обслуговування та забезпечення безпеки. Вхід і вихід з кімнати визначається датчиком присутності. Він авторизує користувачів після входу в особистий кабінет. Камера відеоспостереження допомагає записувати дані з метою безпеки.

#### *2.4.3. Підсистема старт освітлення*

Елементи керування освітленням є засобом забезпечення функціональності освітлення та економії енергії освітлення. Згадані нижче стратегії (або комбінація стратегій) часто включаються для реалізації сумісного дизайну [10].

Освітлення в певній зоні вмикається, вимикається або тьмяніє відповідно до попередньо визначеного розкладу, який можна налаштувати.

У кожній частині будівлі освітлення вмикається, вимикається або досягає заданого рівня залежно від виявленого рівня заповненості. З «ваканцією» контрольні лампи автоматично вимикаються, коли звільняється місце. Користувачам більше не потрібно вручну вмикати світло. Часто кімнатні умови можуть не вимагати електричного освітлення, як у випадку з денним освітленням офісу.

У такому типі ситуації датчики присутності здатні економити енергію, яка в іншому випадку була б автоматично спожита, якби датчик присутності примусово вмикав світло. З цієї причини датчики вакансій зазвичай вважаються більш ефективними, ніж, отже, кращими, ніж датчики присутності, що стимулює включення приписувальних датчиків вакансій у багато версій оновлених електричних кодів.

Рівень освітлення регулюється вручну та/або автоматично залежно від кількості природного світла в просторі. Відповідні рівні освітлення забезпечуються для функціональних цілей, а загальне освітлення підтримується у всьому просторі. Різні способи збирання денного світла

включають безперервне затемнення систем освітлення (найефективніший варіант), дворівневе або багаторівневе зональне затемнення або перемикання, або прості елементи керування увімкненням/вимкненням для зон освітлення, де очікується достатньо денного світла. Типова система пожежної сигналізації має низку пристроїв, які працюють синхронно, щоб попередити та оповістити людей за допомогою візуальних та звукових пристроїв у разі виявлення диму, вогню, чадного газу чи інших надзвичайних ситуацій. Систему можна активувати автоматично від детекторів диму та тепла або також можна активувати альтернативно за допомогою пристроїв ручної активації пожежної сигналізації, таких як ручні сповіщувачі або тягові станції.

#### *2.4.4 Аудіосистема*

Аудіосистема для кімнати для переговорів повинна бути розроблена таким чином, щоб задовольнити користувачів і, крім того, відповідати вимогам комфорту та ефективності. Автоматичне керування динаміками доступне за допомогою центрального інтелекту, який підходить для різних групових налаштувань. Крім того, є ручне керування для ввімкнення/вимкнення, збільшення/зменшення гучності, здійснення відеодзвінків і показу екрана. У кімнаті є засоби для проведення зустрічей, які вимагають відеоконференції для невеликої групи людей. Для цього потрібне підключення до Інтернету.

## **2.5. Проектування розумної системи віртуальних переговорів**

### *2.5.1. Режими роботи системи*

Розумна кімната для переговорів як система працює в різних режимах, які відповідають умовам навколишнього середовища, автоматично підлаштовуючись під атмосферу.

Контроль системи має бути чітко визначений, щоб гарантувати, що система відповідає потребам безпеки та комфорту користувачів. Різні підсистеми взаємодіють і функціонують разом розумним і ефективним способом. Щоб визначити центральний інтелект і визначити характер взаємодії між кількома функціональними можливостями та підсистемами, важливо ідентифікувати та розділяти різні стани системи на основі потреб користувачів.

Виходячи з вимог, стани системи SMR можна класифікувати як очікування, сон, зустріч, пожежна тривога. У цьому розділі кожен із станів пояснюється та зображено графічно. На рисунках у кожному стані системи показано відповідну поведінку підсистеми/компонента.

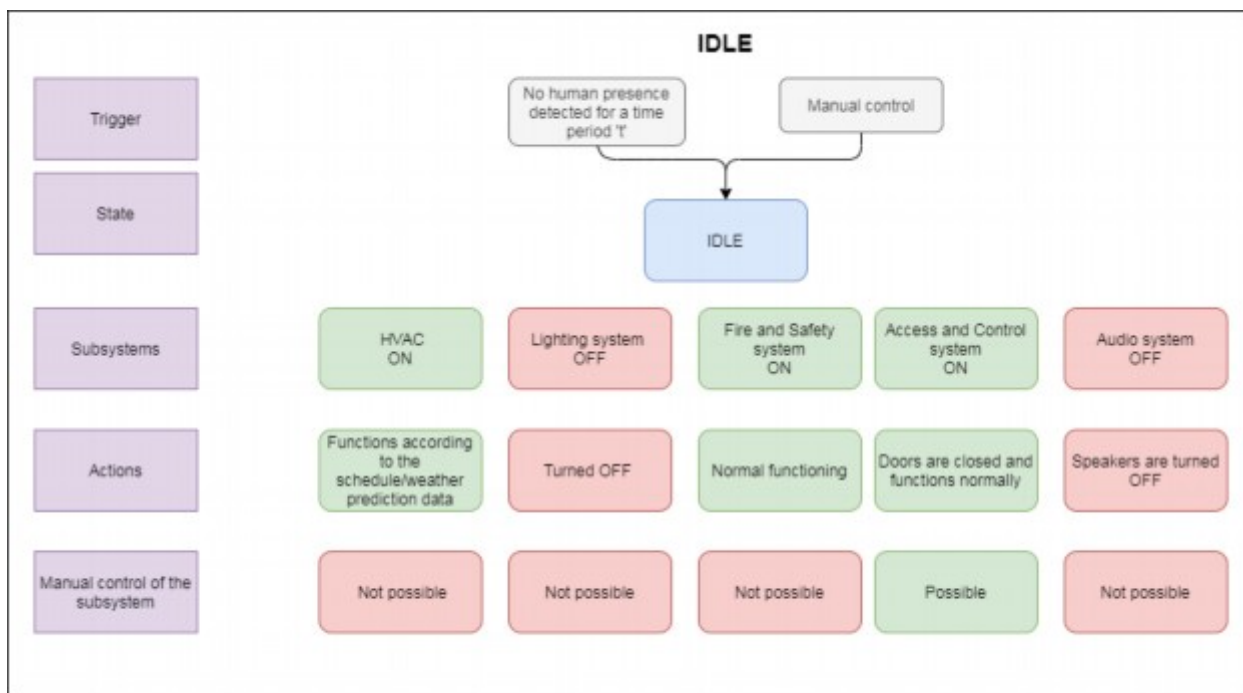


Рис. 2.8. Стан/поведінка підсистем у режимі очікування

Коли кімната для нарад перебуває в стані очікування, світло, динаміки, камери відеоспостереження вимикаються й залишаються вимкненими, доки стан не зміниться. У цей період двері залишаються зачиненими. Однак система опалення, вентиляції та кондиціонування повітря має працювати, щоб підтримувати рівень температури в кімнаті. У цьому стані неможливо

керувати системою опалення, вентиляції, вентиляції та кондиціонування, отже, функціонування системи опалення, вентиляції та кондиціонування буде базуватися на попередньо визначеному розкладі або відповідно до даних прогнозу погоди. Система автоматично переходить у режим очікування, якщо протягом певного періоду часу не виявлено присутності людини. Крім того, кімнатою для нарад можна керувати вручну, щоб перевести її в режим очікування за допомогою пульта дистанційного керування. Загальна поведінка та стан підсистем у режимі очікування зображено на рисунку 2.8.

Коли кімната для нарад перебуває в режимі сну, усі пристрої та підсистеми вимикаються, включаючи систему HVAC. Якщо користувачі вирішать заблокувати кімнату для нарад, вимкнувши всі її функції, вони можуть зробити це вручну за допомогою пульта дистанційного керування. Кімната для нарад автоматично перейде в режим сну, якщо вона буде залишатися в стані очікування протягом тривалого часу. Температуру в кімнаті більше не потрібно підтримувати, виходячи з припущення, що кімната не потребує охолодження/опалення/вентиляції, і в кімнаті залишається ідеальна температура. Загальна поведінка та стан підсистем у режимі сну зображено на рисунку 2.9.

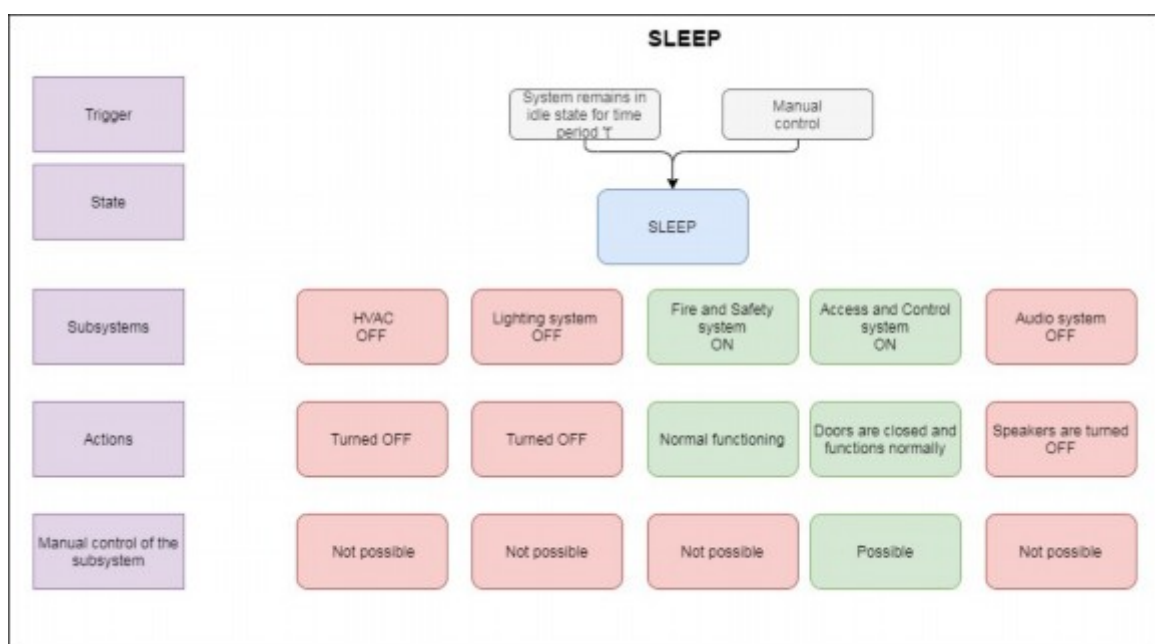


Рис. 2.9. Стан/поведінка підсистем у режимі керування: сон

Якщо кімнату потрібно підготувати до зустрічі, користувачі можуть вручну перейти в стан зустрічі. Щоб полегшити проведення зустрічей, освітлення кімнати потрібно налаштувати відповідним чином. За замовчуванням налаштування освітлення переходить у режим зустрічі. Однак його можна перемикає між інтелектуальним, звичайним і зустрічним. Система HVAC працює відповідно до розкладу/даних про прогноз погоди, двері залишаються зачиненими, камери відеоспостереження залишаються ввімкненими, динаміки та динаміки ввімкнені для будь-якого аудіо/відеовиходу. Загальна поведінка та стан підсистем у режимі зустрічі зображено на рисунку 2.10.

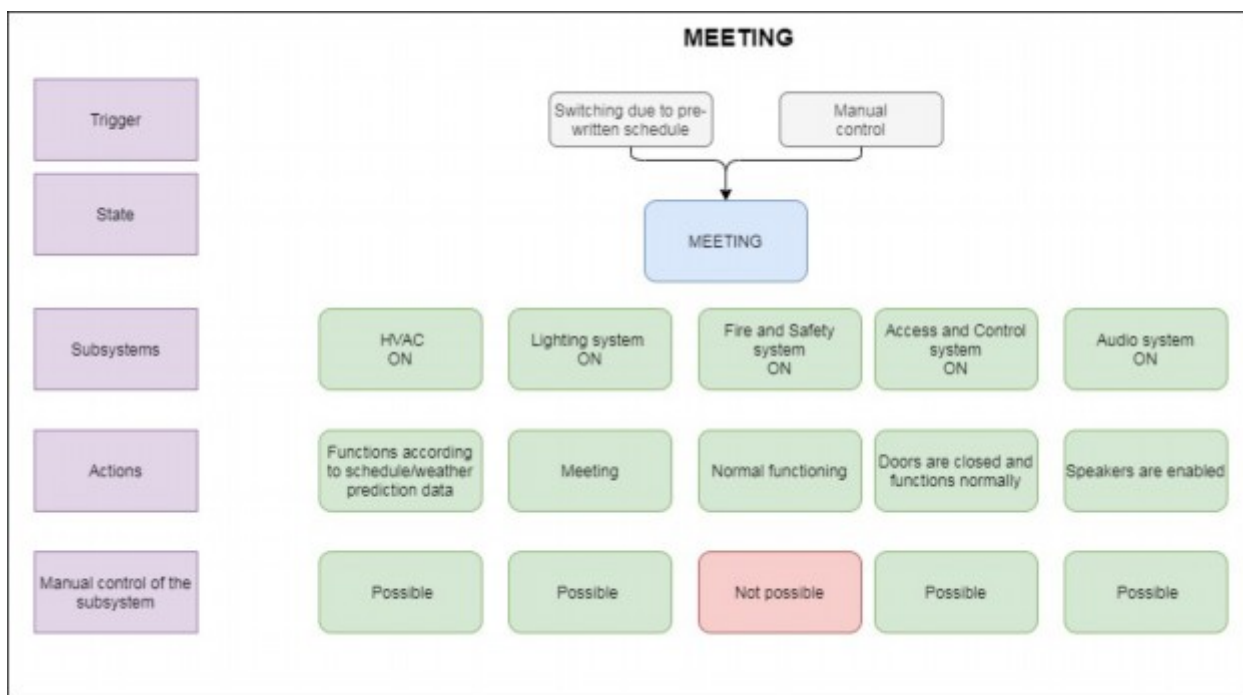


Рис. 2.10. Стан/поведінка підсистем у режимі керування: зустріч

Коли пожежні датчики/датчики CO<sub>2</sub> виявляють надзвичайну ситуацію, кімната переговорів переходить у стан пожежної тривоги. У цьому стані попереджувальне повідомлення оголошується через динаміки як будильник, а потім лунає сигнал. Двері залишатимуться відкритими, система опалення, вентиляції та кондиціонування буде вимкнена, система освітлення буде вимкнена через короткий проміжок часу, щоб дозволити користувачам

залишити кімнату. Камери відеоспостереження та сенсорні панелі вимкнені. У цьому стані всі підсистеми будуть негайно вимкнені. Стан пожежної тривоги кімнати для нарад не можна активувати вручну. Слід зазначити, що пожежна сигналізація та датчики CO<sub>2</sub> залишатимуться онлайн протягом усіх станів системи. Загальна поведінка та стан підсистем у режимі пожежної сигналізації зображено на рисунку 2.11.

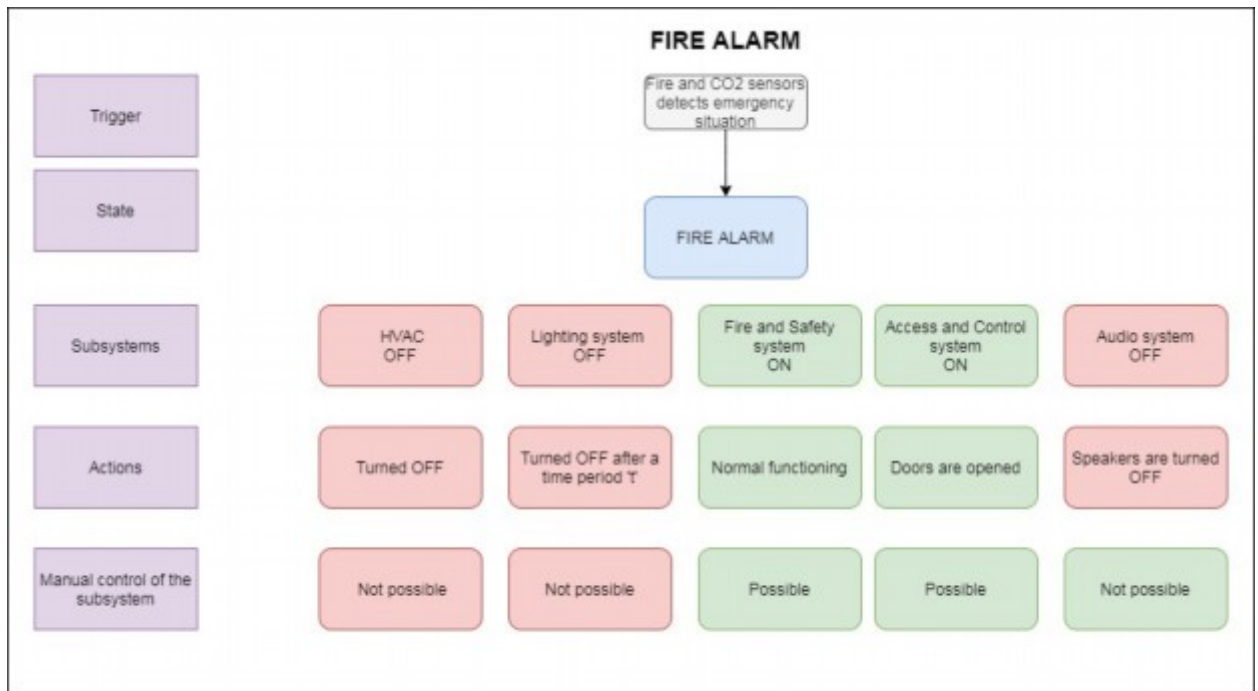


Рис. 2.11. Стан/поведінка підсистем у режимі керування: пожежна тривога

### 2.5.2. Поведінка підсистем (пристроїв і драйверів)

Представлення поведінки системи HVAC, системи освітлення, системи безпеки, системи безпеки та доступу та аудіовізуальної системи зображено як служби за допомогою діаграм варіантів використання. Це частина гнучкої розробки моделей SysML для Smart Meeting Room. Деякі зі згаданих нижче підсистем об'єднані разом, щоб створити загальний варіант використання, який служить багатьом цілям.

На рисунку 2.12 показано діаграму варіантів використання послуги HVAC і системи безпеки разом. Користувач може вмикати/вимикати систему

HVAC та вручну регулювати температуру в приміщенні. Система HVAC має функції опалення, кондиціонування повітря та забезпечення вентиляції приміщення. Центральна система, яка має дані від датчика присутності, виявляє підвищену присутність людей і автоматично забезпечує вентиляцію та кондиціонування повітря. Щоразу, коли система безпеки виявляє пожежу за допомогою пожежного датчика, вона спрацьовує пожежну тривогу, і тому система HVAC автоматично вимикається. Датчики вуглекислого газу в системі безпеки перевіряють допустимі межі CO<sub>2</sub> і видають попереджувальне повідомлення щоразу, коли досягається порогове значення.

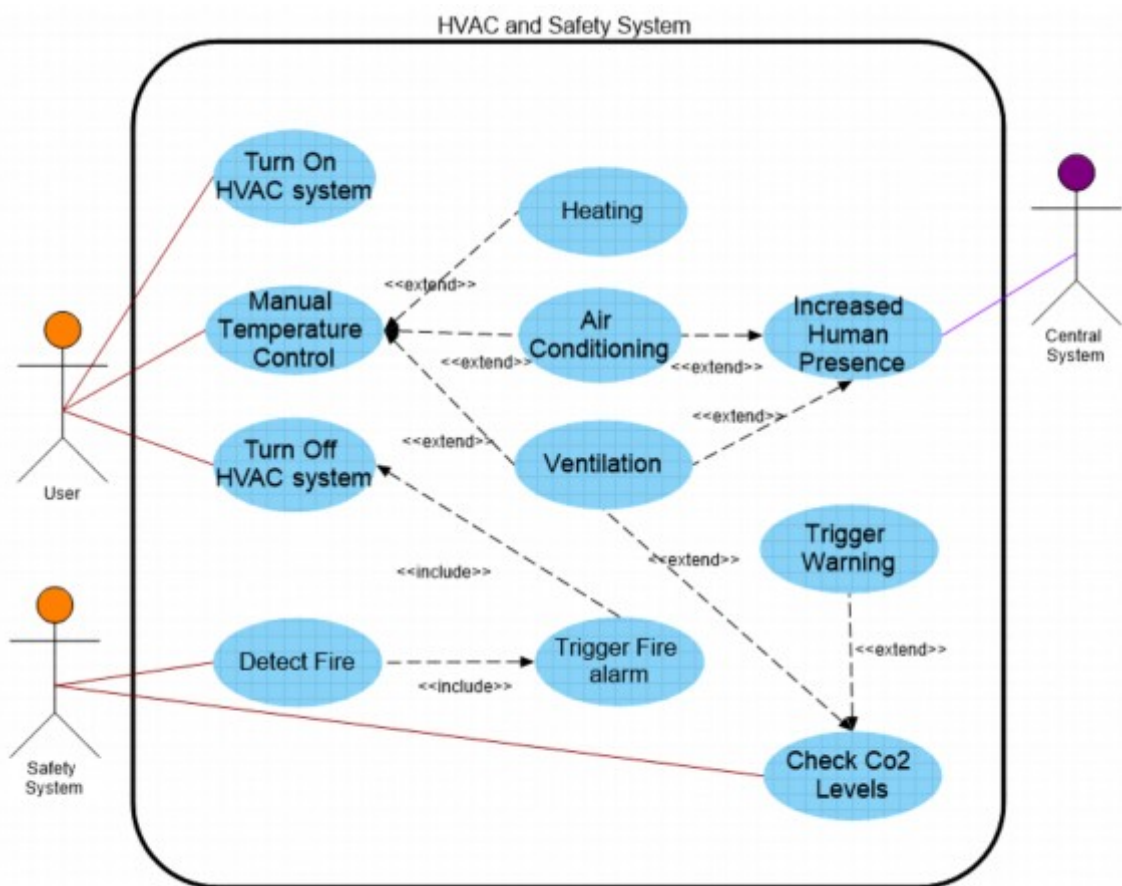


Рис. 2.12. HVAC та система безпеки

Послуга системи освітлення представлена на діаграмі варіантів використання на рисунку 2.13. Користувач може вручну вмикати світло та регулювати яскравість, або центральна система може виявляти вхід і вихід

біля дверей за допомогою датчика присутності та вмикати/вимикати світло, коли це необхідно. Щоразу, коли аудіо- та візуальна система виконує функцію спільного перегляду екрана, яскравість кімнати має зменшуватися, щоб забезпечити кращий відображення та перегляд для мешканців.

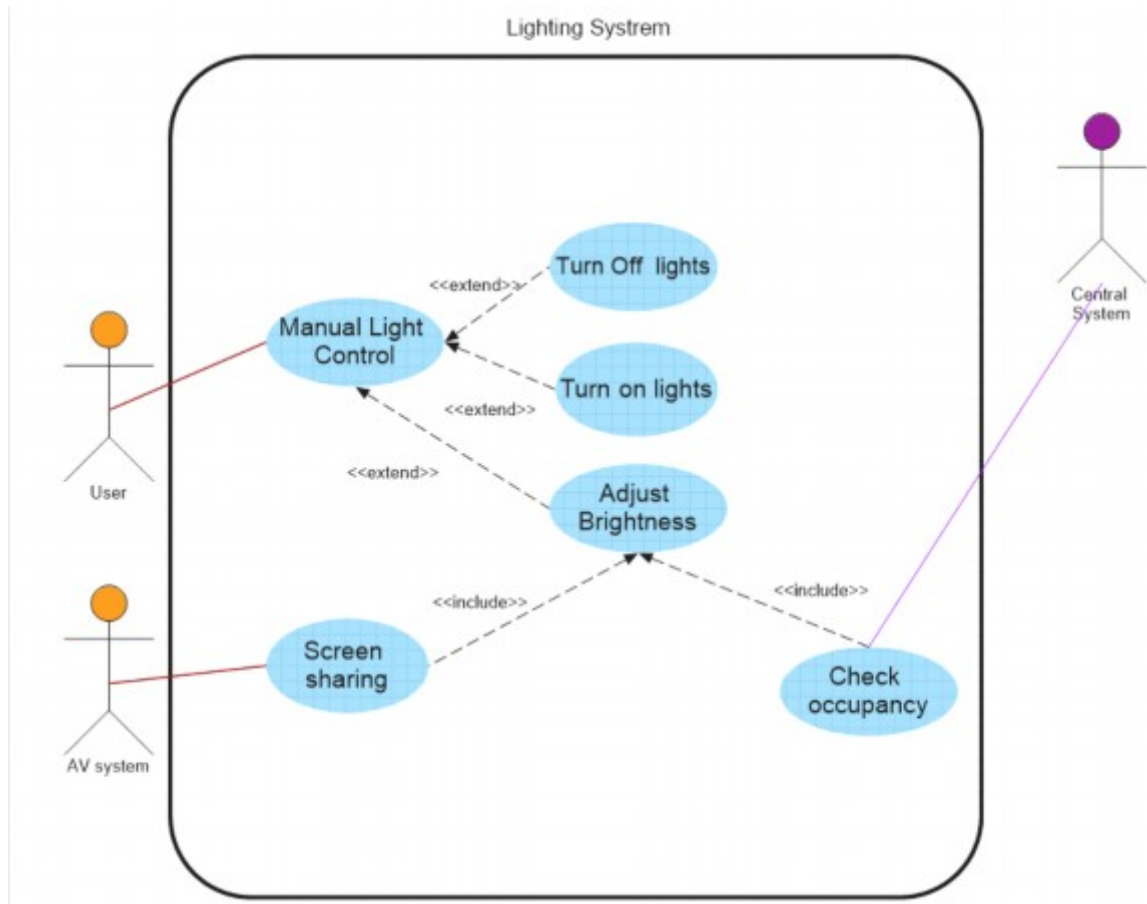


Рис. 2.13. Система освітлення

Система безпеки та доступу розумної кімнати надає користувачам можливість входу за допомогою пароля. Вхід до кімнати може бути обмежений для обмеженої кількості осіб. Центральна система постійно стежить за входом біля дверей.

Щоразу, коли є вхід у двері, камера відеоспостереження вмикається автоматично. Це робиться для моніторингу подій у кімнаті з метою безпеки. Служба діаграми безпеки та доступу представлена як діаграма варіантів використання на рисунку 2.14.

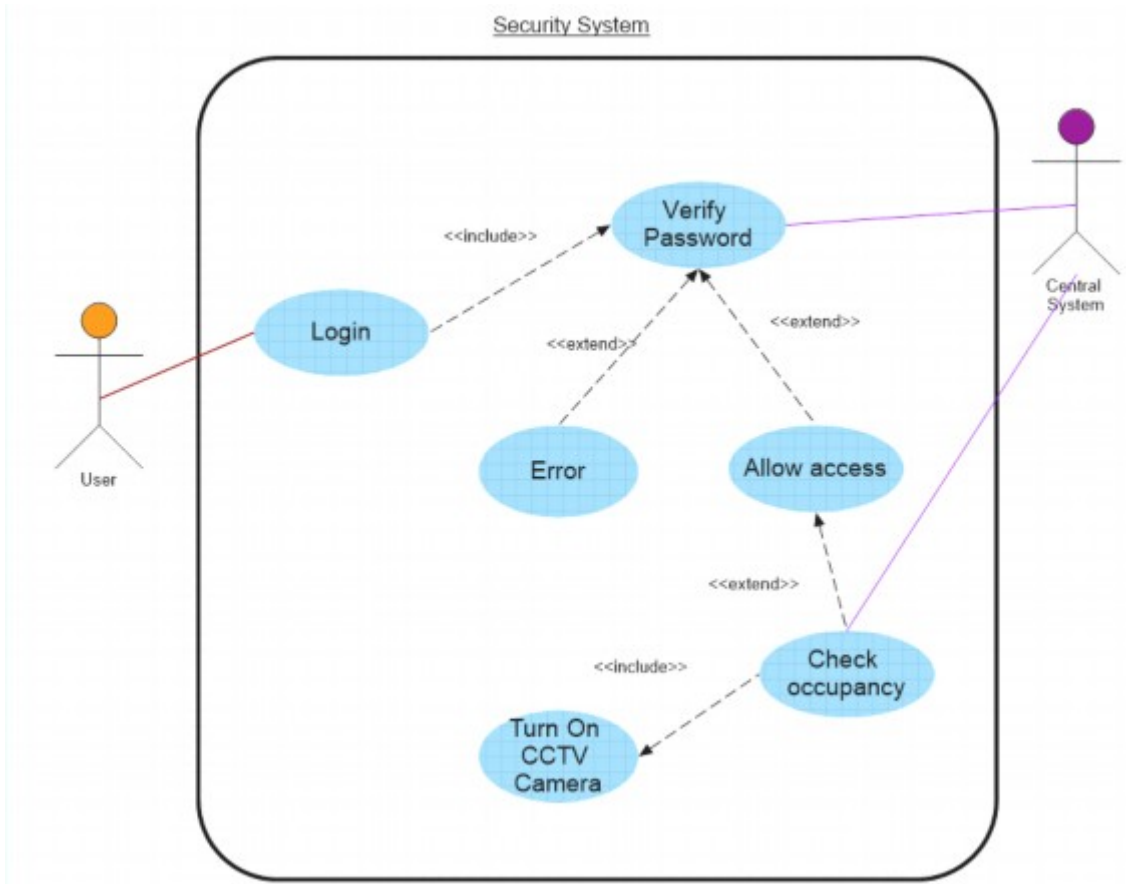


Рис. 2.14. Система безпеки та доступу

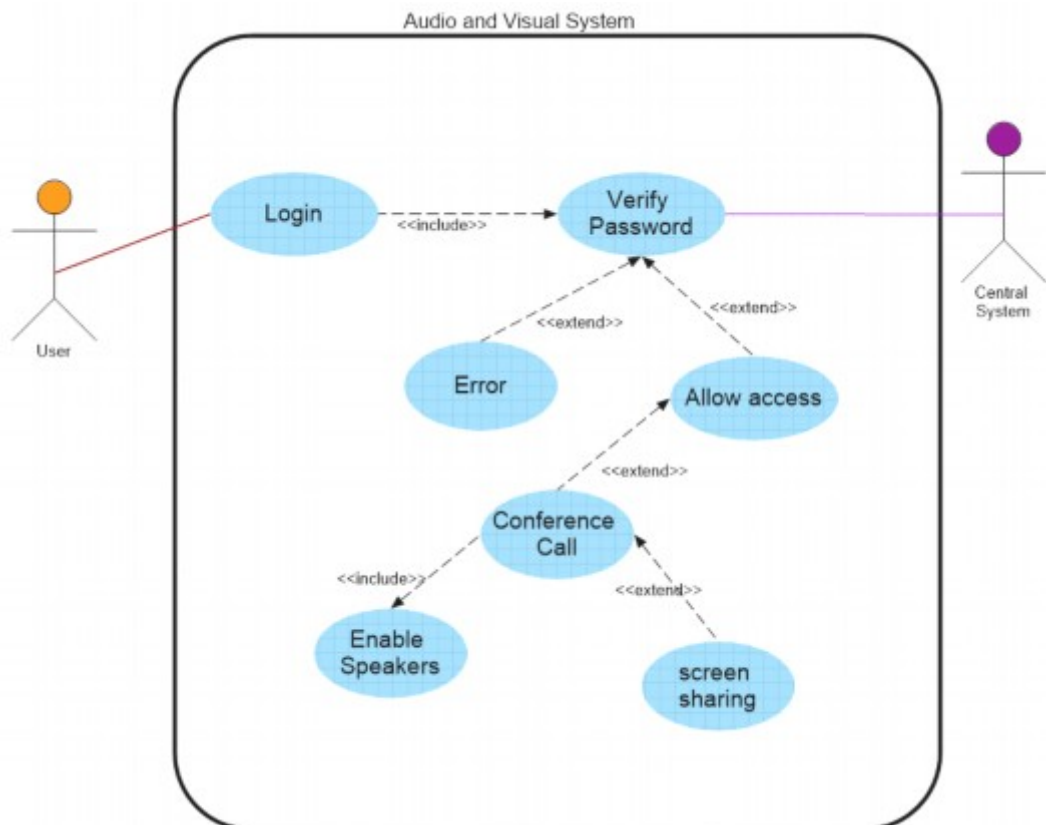


Рис. 2.15. Аудіо та відеосистема

На рисунку 2.15 зображено послугу аудіо- та відеосистеми. Сенсорна панель в розумній кімнаті дозволяє користувачам взаємодіяти, увійшовши в особистий кабінет. Сенсорна панель має опції для здійснення конференц-дзвінків і надання спільного доступу до екрана.

## **2.6. Багаторівнева архітектура для цифрового образу для розумної системи віртуальних комунікацій**

Багатошаровий архітектурний підхід, який раніше обговорювався, може бути використаний у цьому проєкті. У цьому контексті розробка віртуального простору в Unity Game Engine може бути виконана за допомогою принципів багаторівневої архітектури Digital Twin. У цьому розділі обговорюється адаптація системи SMR до кожного рівня архітектури Digital Twin.

### *2.6.1. Рівень абстракції даних*

Усі дані, зібрані з різних компонентів і їхніх датчиків, інкапсульовані на цьому рівні. Моделі SysML, створені за допомогою IBM Rhapsody, містять усі дані цього рівня.

Загальні змінні вхідних даних моделі SMR

- Прапор виявлення руху
- Прапор виявлення зайнятості

Змінні вхідних даних із системи ОВК

- Поточна кімнатна температура
- Поточна зовнішня температура
- Поточний рівень CO<sub>2</sub>
- Поточний рівень вологості
- Дані прогнозу погоди

Змінні вхідних даних із системи безпеки

- Прапор пожежної сигналізації

- сигнал тривоги CO<sub>2</sub>

Змінні вхідних даних із системи освітлення

- Показник освітленості приміщення
- Показник зовнішньої освітленості
- Регулювання віконної панелі

Змінні вхідних даних із системи безпеки та доступу

- PIN-код
- Прапор відеоспостереження

Змінні вхідних даних із аудіо та відеосистеми

- Прапор конференції
- Прапор спікера
- ID користувача

### *2.6.2. Рівень передачі даних*

Усі дані від датчиків і підсистем, які збираються на рівні збору даних, тепер передаються в домен контролера. У цьому домені дані маніпулюють, щоб пропонувати послуги системи SMR. Незважаючи на те, що фізичні вхідні дані від датчиків моделюються за допомогою моделей SysML і введених користувачами даних, існує потреба в рівні передачі, оскільки він діє як міст між моделлю IBM Rhapsody і моделлю 3D Virtual. Значення даних з моделі IBM передаються до Unity Game Engine за допомогою відповідного протоколу зв'язку. Тому для організованого керування системою SMR необхідний окремий сервіс, який передає вхідні дані з моделі SysML.

### *2.6.3. Рівень цифрового моделювання*

Розумна кімната для переговорів з усіма її компонентами моделюється на цьому рівні. Для цього використовується сховище безкоштовних ресурсів в ігровому движку Unity. 3D-моделі фону та компонентів кімнати для нарад розробляються на основі наявних шаблонів. Крім того, сценарії C#

допомагають розробити ігровий сценарій для генерації рухів гравця в системі SMR.

#### 2.6.4. Рівень інтеграції даних/моделі

На цьому рівні сценарії C# для віртуальної моделі в Unity розроблені відповідно до поведінки системи та пов'язані з кодом C++, який автоматично генерується з моделей IBM Rhapsody. Інтеграція має здійснюватися таким чином, щоб вона забезпечувала гнучкість і можливість адаптації до коду.

#### 2.6.5. Рівень програми/сервісу

Після того як модель SysML інтегровано з 3D-моделлю, даними потрібно маніпулювати та отримувати доступ розумним способом. Послідовність послуг має бути належним чином, щоб сприяти реальним сценаріям.

#### 2.6.7. Вимоги до системи

На наступних рисунках 2.16 – 2.19 представлено вимоги до системи.

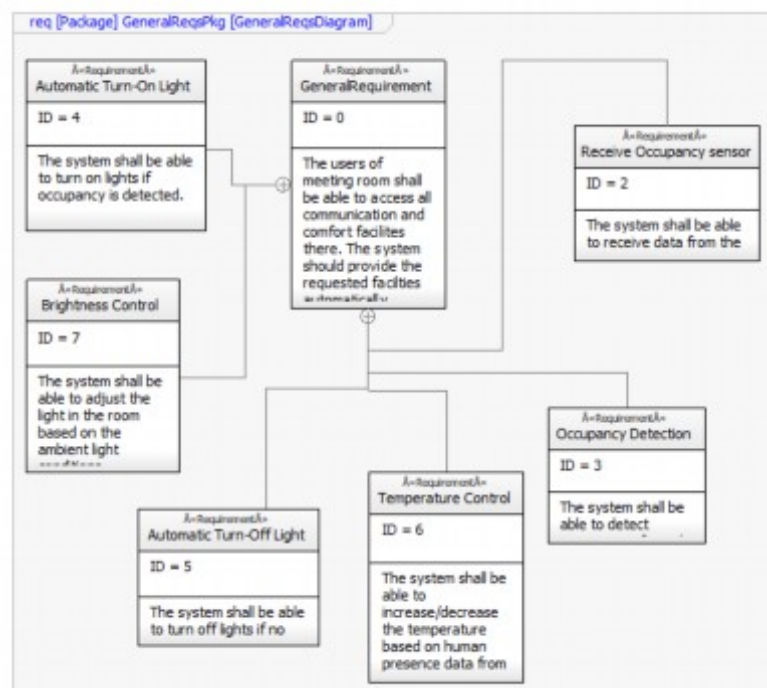


Рис. 2.16. Діаграма загальних вимог

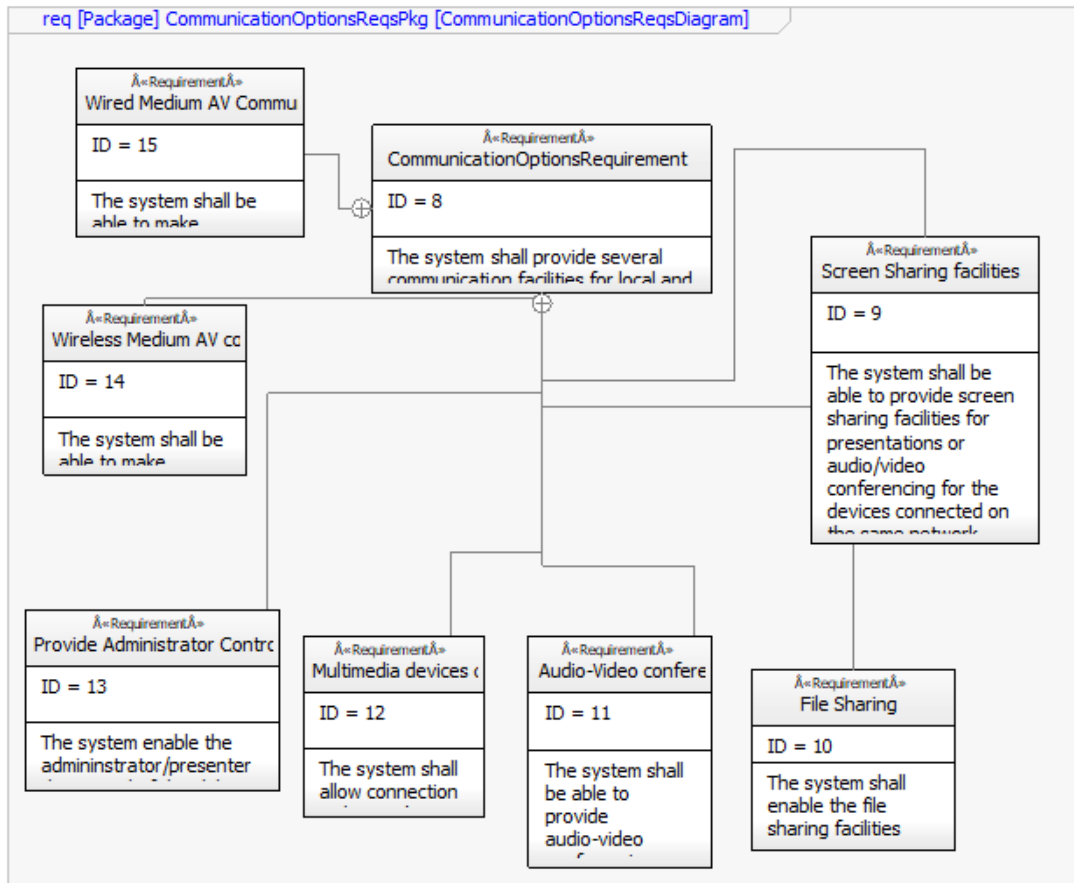


Рис. 2.17. Діаграма вимог до пакета системи зв'язку

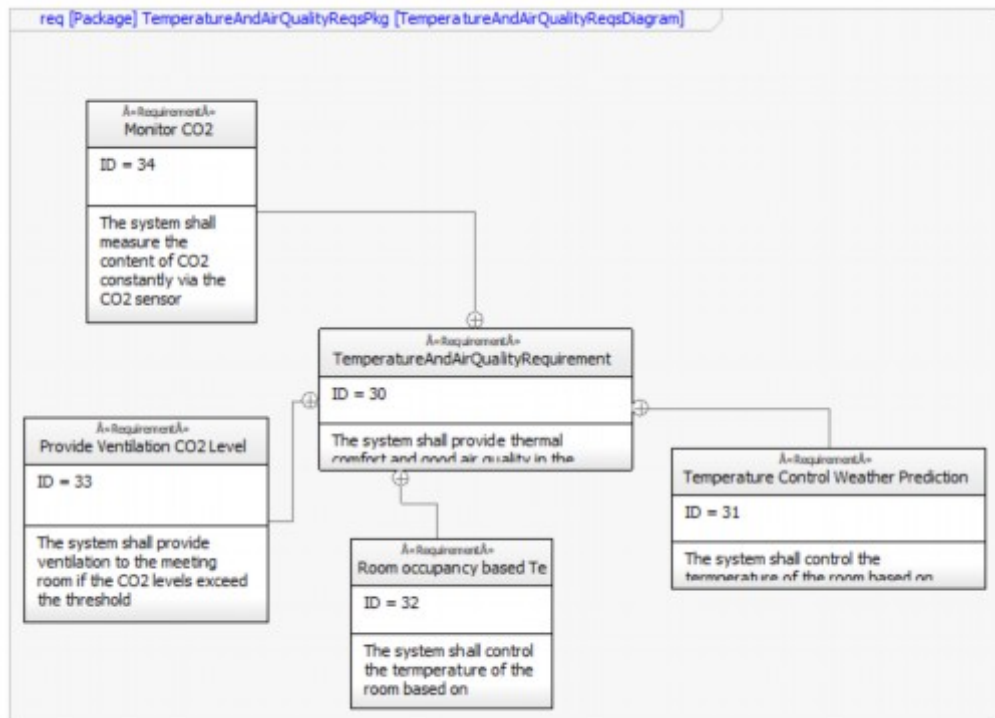


Рис. 2.18. Діаграма вимог до системи контролю температури та повітря

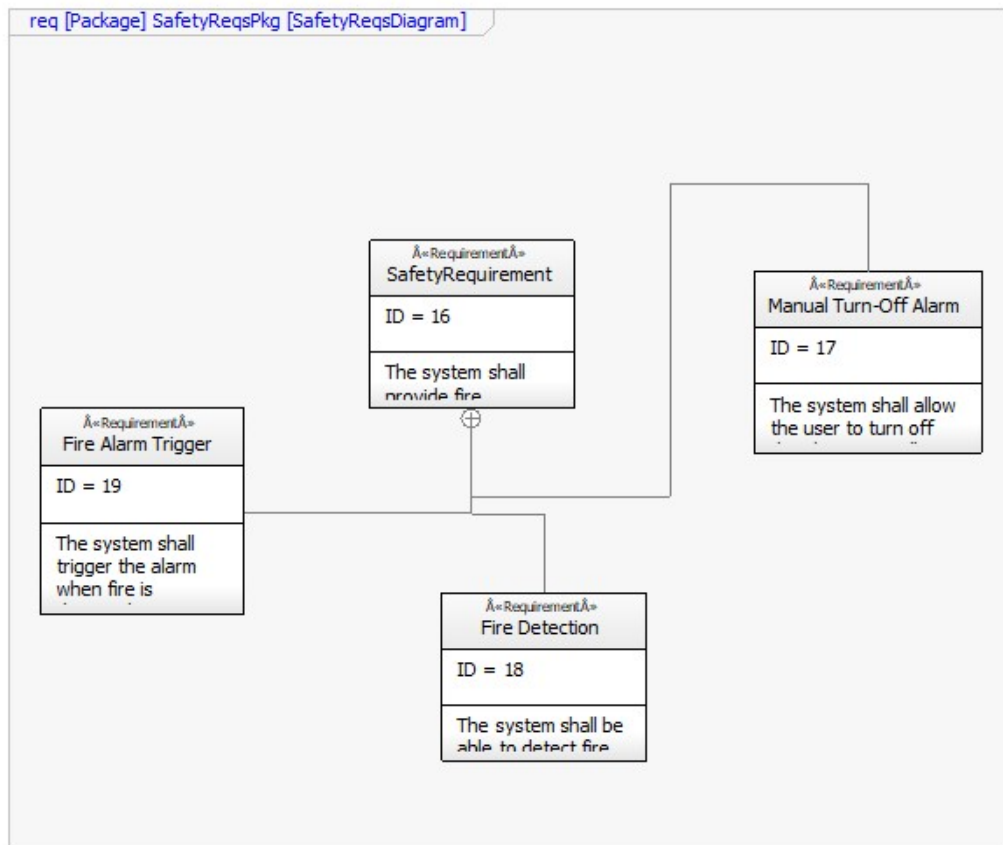


Рис. 2.19. Діаграма вимог до пакету системи безпеки

### Висновки до розділу

Другий розділ роботи присвячений розробці методології для створення програмного забезпечення, призначеного для систем віртуальних комунікацій і Digital Twin. Представлено модульний підхід до розробки, який забезпечує гнучкість, масштабованість та легкість інтеграції різних функціональних модулів, що важливо для комплексних цифрових систем.

У межах методології описано ключові етапи розробки: модель у циклі (MIL), програмне забезпечення в циклі (SIL) та цифровий двійник у циклі (DTIL). Ці етапи дозволяють відтворити процеси розробки та тестування, створюючи більш ефективну та адаптивну систему. Особливо значущим є використання підходу DTIL, що дає змогу інтегрувати цифровий двійник у процес розробки для поліпшення тестування й ітеративного вдосконалення системи.

Метод V-Model забезпечує структурований підхід до розробки, дозволяючи здійснювати аналіз вимог і їх узгодження на кожному етапі розробки, що сприяє створенню надійної та стабільної системи.

Також розглянуто архітектуру для розумної системи віртуальних комунікацій, зокрема багаторівневу структуру, яка складається з рівня абстракції даних, рівня передачі даних, рівня цифрового моделювання, рівня інтеграції та рівня сервісів. Кожен з рівнів виконує специфічну функцію, що сприяє ефективному збору, передачі, моделюванню та обслуговуванню цифрових даних, створюючи інтегровану екосистему для віртуальних комунікацій

## РОЗДІЛ 3. ІМПЛЕМЕНТАЦІЯ МОДЕЛЬ-БАЗОВАНОЇ МЕТОДОЛОГІЇ ПРОЄКТУВАННЯ ЦИФРОВИХ ОБРАЗІВ ДЛЯ ВІРТУАЛЬНИХ КОМУНІКАЦІЙ

### 3.1. Впровадження моделі

Проект реалізується у 3 етапи. По-перше, система SMR моделюється в IBM Rhapsody, а потім інтегрується віртуальна модель в Unity Game Engine. Нарешті, пов'язану модель можна переносити в середовище Linux для віддаленої реалізації. Спочатку описується реалізація моделі IBM, а потім її зв'язок з Unity та впровадження Digital Twin в Unity Game Engine у наступних розділах.

Впровадження моделі IBM є першим і головним кроком до реалізації проекту. Опис проекту, поданий у третьому розділі, надає нам необхідну інформацію для розробки відповідної системи SMR. Випуск IBM Rhapsody Developer разом із Visual Studio 2017 Community використовується як інструмент моделювання для цього проекту. Крім того, в інструменті IBM створюється новий проект C++ із стандартними конфігураціями проекту та інструкціями компіляції, характерними для системи Windows, яка використовує процесор x86.

По-перше, вимоги моделюються в окремому пакеті під назвою «RequirementAnalysisPkg» – вимоги до системи SMR. По-друге, системний контекст і діаграма визначень блоків разом із діаграмою класів моделі SMR створюються, як показано раніше на рисунках 2.6 і 2.7. Ці діаграми допомагають визначити блоки та компоненти, необхідні для типової системи SMR. Крім того, діаграми встановлюють зв'язок між підсистемами та їх відповідними компонентами. На наступному етапі ми підтримуємо структуру файлу так, щоб окремі компоненти створювалися як блоки об'єктів для пакета під назвою «default». Це показано на рисунку 3.1. кожен із цих об'єктів перераховано. Таким чином, щоб використати об'єктно-орієнтоване

програмування в C++, підсистеми та її компоненти стають доступними як об'єкти, які додатково містять асоціації, атрибути, операції, діаграми переходів станів і властивості значень для виконання та імітації програми для різного використання. випадкові сценарії.

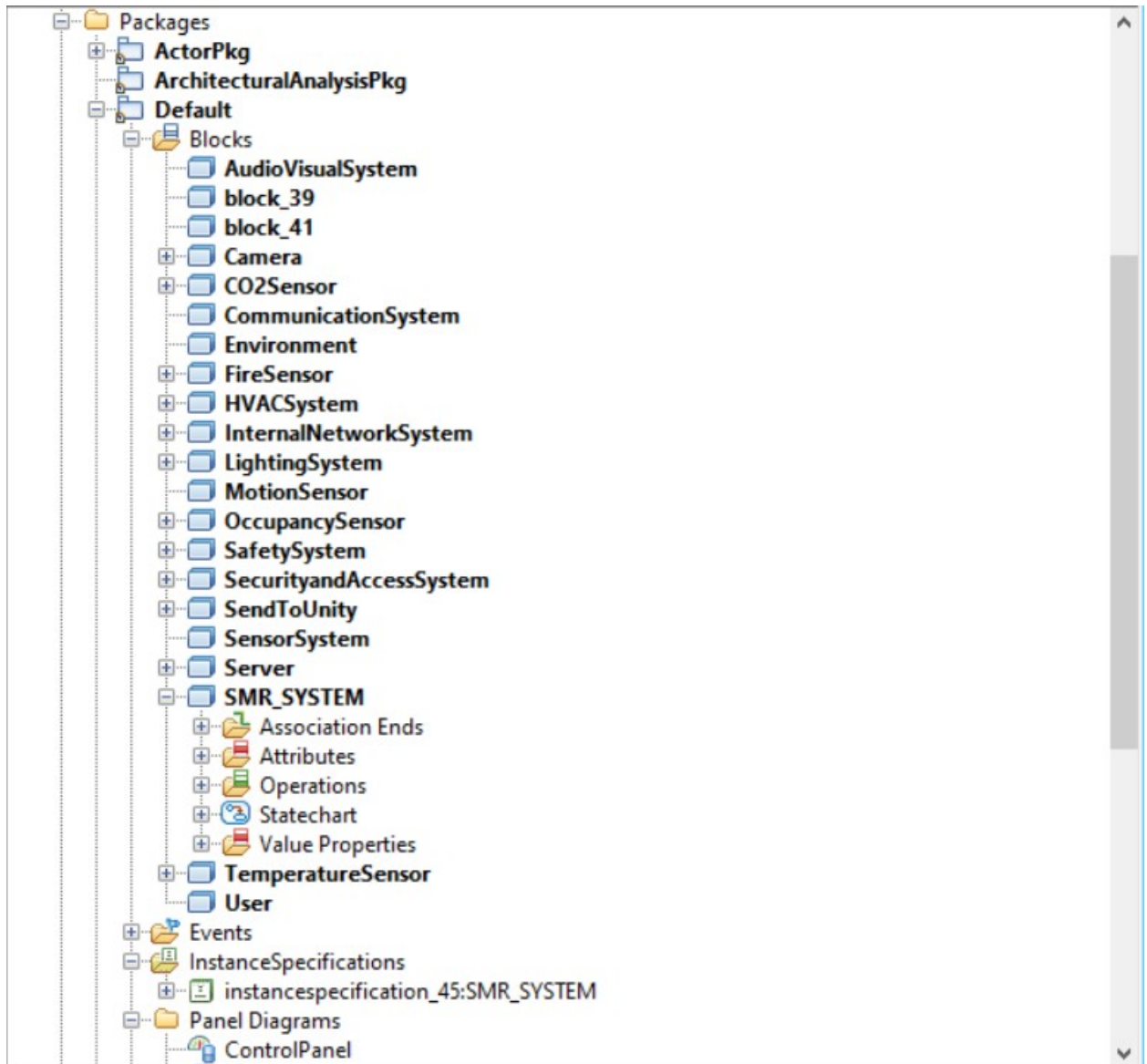


Рис. 3.1. Структура файлу проекту в IBM Rhapsody Tool

Після того, як об'єкти разом із відповідними атрибутами та операціями визначені, ми переходимо до впровадження системної динаміки та контролю. Система SMR працює в п'яти режимах, а саме: режим очікування, онлайн, режим сну, зустріч, пожежа/надзвичайна ситуація, як зазначено в другому розділі. Однак для реалізації цих режимів важливо використовувати діаграми

переходів станів і події. Наприклад, у разі відсутності людей у приміщенні протягом періоду часу «t» датчики виявлення руху в системі мають переходити у стан очікування. Аналогічно, кожен із режимів має власний тригер і набір дій для виконання, тому відповідний дизайн діаграми переходів станів показано на рисунку 3.2.

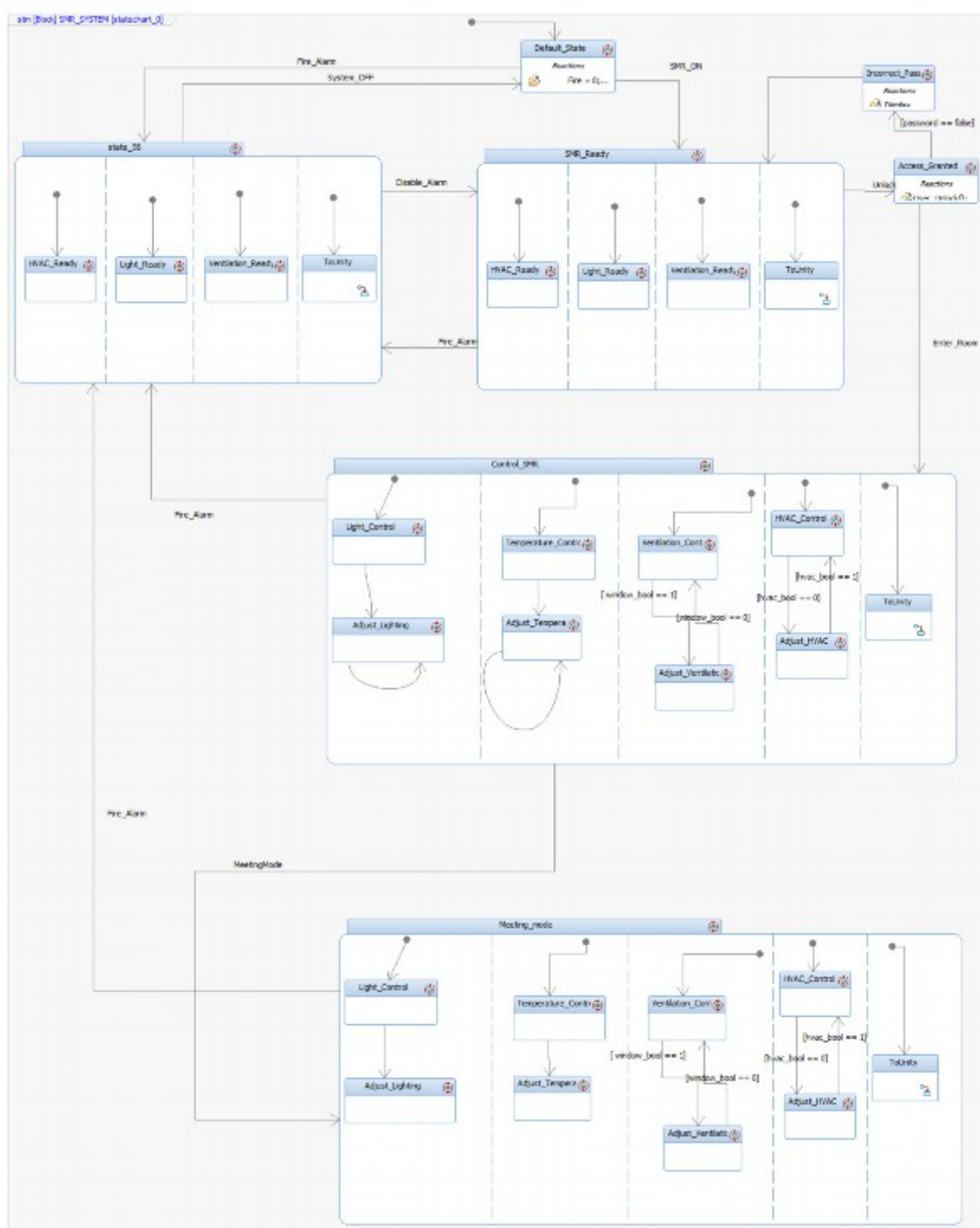


Рис. 3.2. Діаграма станів системи "Розумна кімната для переговорів" (SMR)

Крім того, зміна стану має бути розроблена таким чином, щоб не призводити до тупикової ситуації. Крім того, загальний процес системи SMR можна зобразити за допомогою діаграми послідовності, як показано на рисунку 3.3.

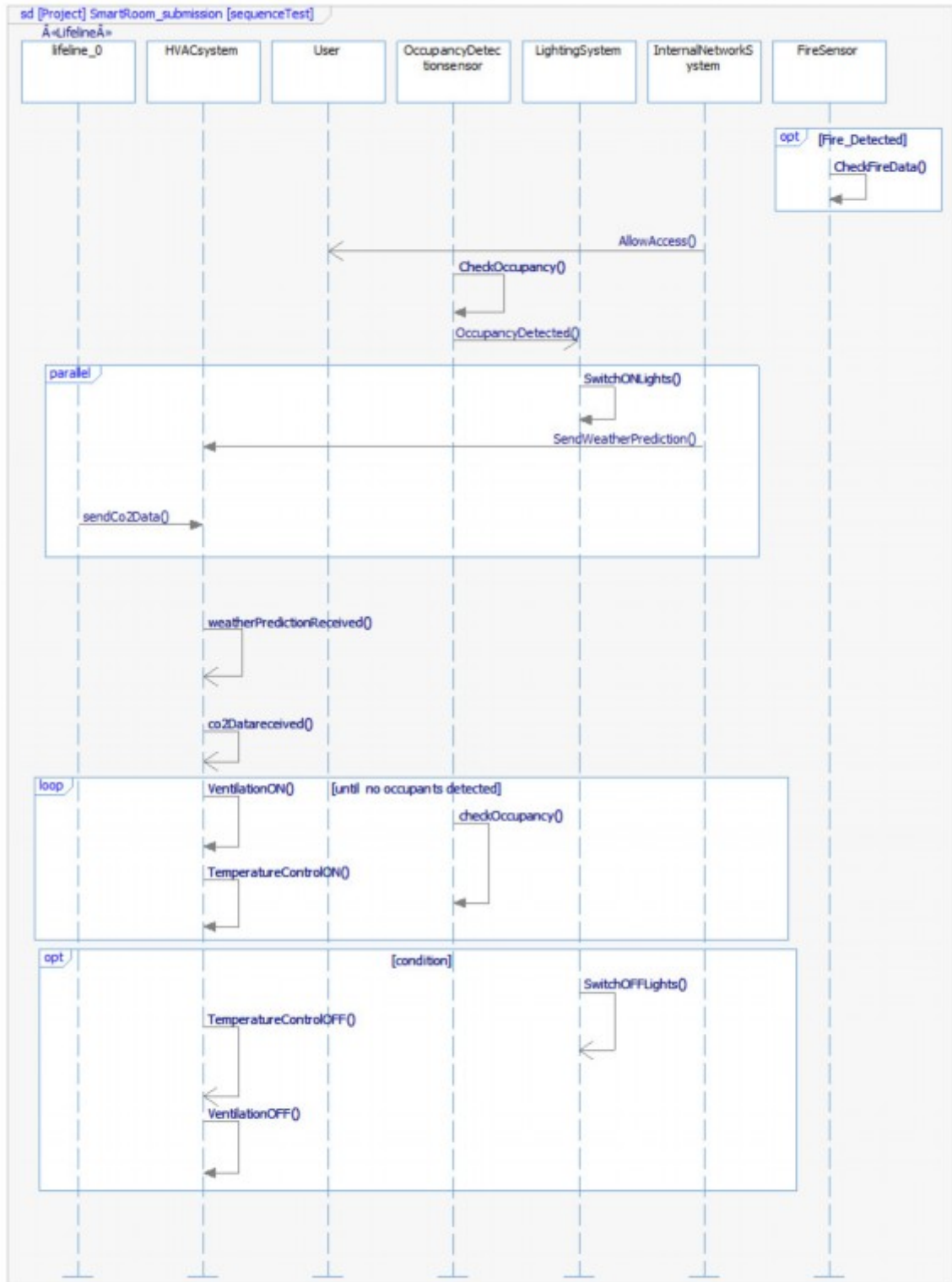


Рис. 3.3. Діаграма послідовності системи "Розумна кімната для переговорів" (SMR)

## 3.2. Реалізація взаємодії віртуальної моделі з Unity

Модель SMR, реалізована в IBM Rhapsody, повинна взаємодіяти з віртуальною моделлю системи SMR в Unity Game Engine. Для цього окремі підсистеми, такі як система HVAC, система протипожежної безпеки, система освітлення, система доступу та контролю, а також аудіо- та візуальна система, повинні оновити свою змінну стану та активувати набір дій у Unity Game Engine. Цей набір дій базується на введенні користувача, що визначає сценарій і контрольний стан, у якому працює SMR. Що стосується методу зв'язку, програмування мережевого сокета забезпечує достатній і гнучкий обмін необробленими даними між програмами. У цьому розділі детально обговорюються графічний інтерфейс системи SMR, блок, який реалізує зв'язок сокетів у моделі SMR – «SendToUnity», а також атрибути та значення даних, необхідні для успішного зв'язку.

### 3.2.1. Моделювання введення користувача та відображення

Сенсорна панель діє як графічний інтерфейс для системи SMR. усі кнопки, ручки, дисплей і перемикачі на панелі керування прив'язані до події або атрибуту в специфікації екземпляра системи SMR. Діаграма сенсорної панелі, яка приймає вхідні дані моделюється, як показано на рисунку 3.4.

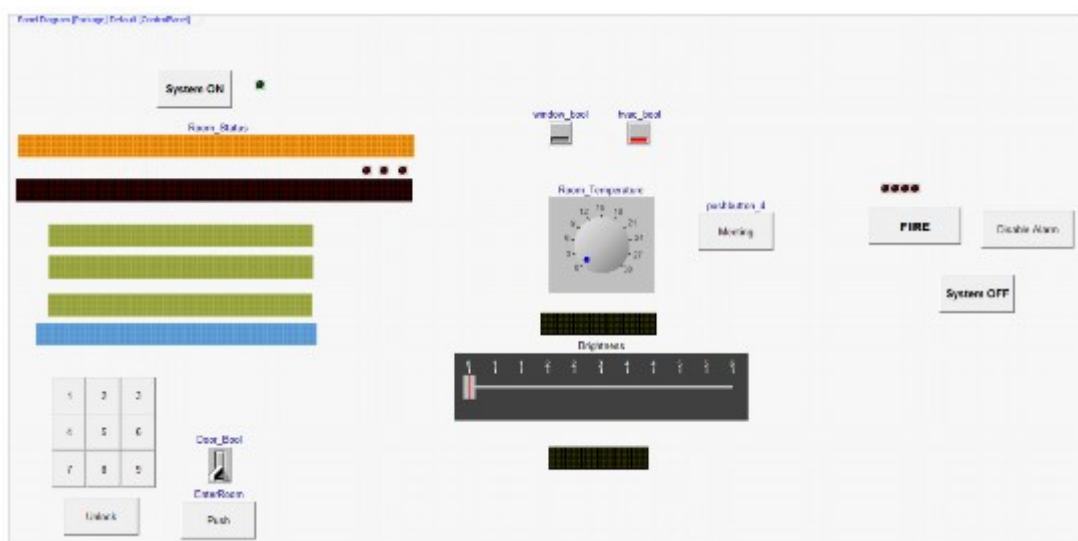


Рис. 3.4. Діаграма сенсорної панелі, що діє як дисплей GUI для системи SMR

Властивості значення, які прив'язуються до графічного інтерфейсу користувача, показані на рисунку 3.5 нижче.

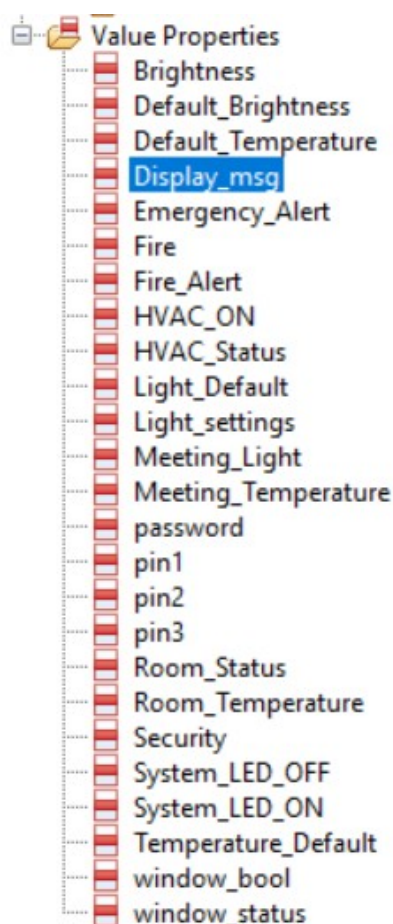


Рис. 3.5. Властивості значення, які прив'язуються до діаграми сенсорної панелі

Блок «SendToUnity» моделі SMR є найважливішою частиною цього проекту та допомагає досягти фундаментальної мети шляхом інтеграції системи SMR із Digital Twin. Оскільки блок «SendToUnity» є важливою функцією, яка взаємодіє з віртуальною моделлю туди й назад, це важливий крок у системному підході, керованому моделлю. Він пропонує активовані необхідні вихідні порти Unity Game Engine і діє як метод тестування для перевірки моделі SMR у віртуальному просторі. Діаграма переходу станів блоку «SendToUnity» показана на рисунку 3.6 нижче. Змінні в блоці автоматично оновлюються з інтервалом часу 5000 мс, як показано на рис. 3.7.

Однак функція «CallMain()», яка отримує та надсилає дані до інструменту Rhapsody та з нього, викликається рекурсивно із затримкою часу 1000 мс.

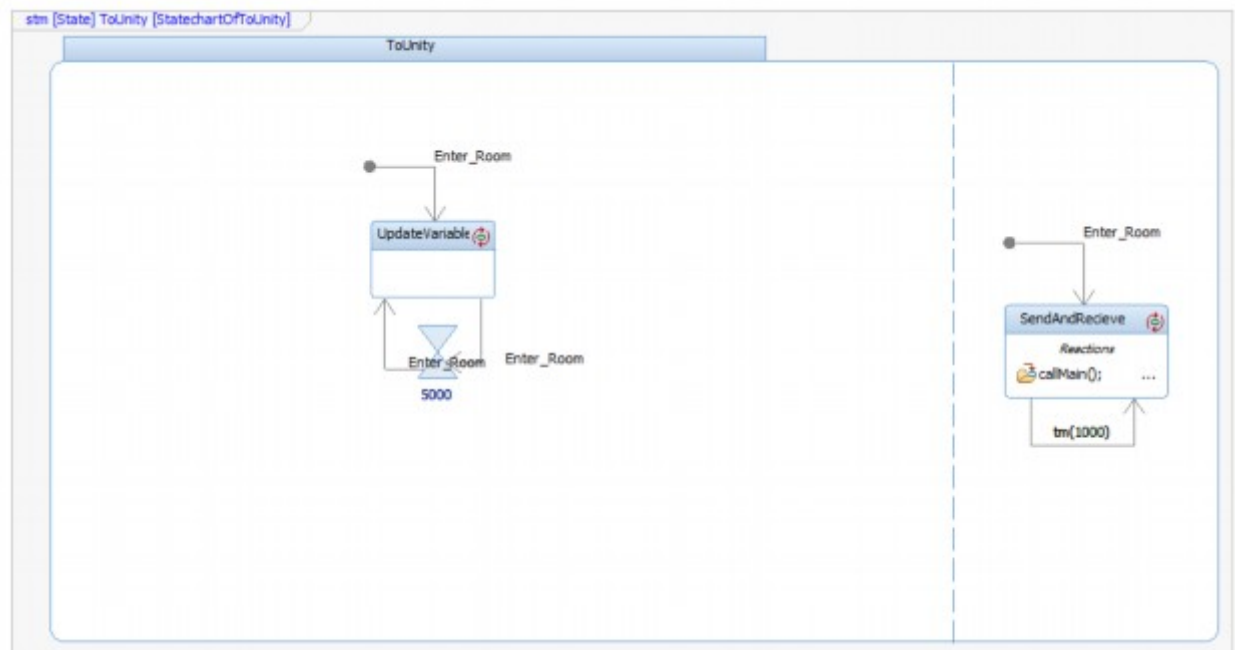


Рис. 3.6. Діаграма переходу стану блоку «SendToUnity»

Модель SMR у IBM Rhapsody та віртуальна модель у Unity Game Engine дійсно працюють на одній машині Windows у цій реалізації, яка була продемонстрована в цьому звіті. Однак важливо зазначити, що можна встановити такий зв'язок між двома процесами, навіть якщо вони працюють на двох різних машинах, якщо вони працюють в одній мережі.

Оскільки зв'язок через мережевий сокет використовує протокол зв'язку TCP/IP, стає легше реалізувати функціональні можливості в інструменті IBM Rhapsody. Бібліотека «WINSOCK», яка є в середовищі розробки програмного забезпечення C++, має структуру та попередньо визначені функції для забезпечення зв'язку Socket у інструменті IBM Rhapsody. Необроблені дані та рядкові значення, які потрібно надіслати/отримати, отримують окремі окремі порти. Крім того, блок «SendToUnity» по суті діє як клієнт, який хоче підключитися до певних портів на сервері, який працює на Unity Game

Engine. Атрибути, необхідні для встановлення зв'язку через мережевий сокет, показані на наступних рисунках 3.7 і 3.8.

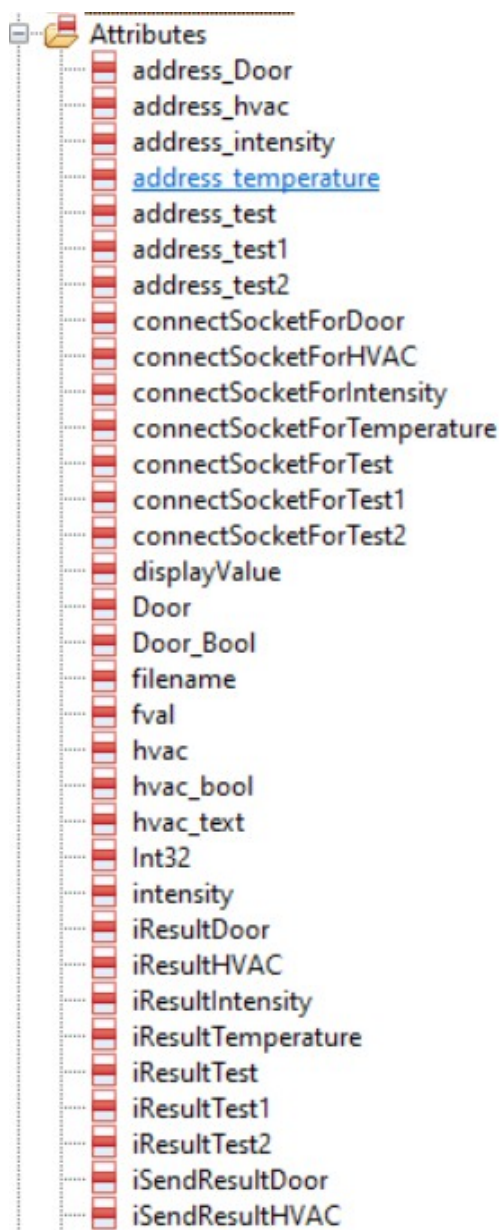


Рис. 3.7. Атрибути, необхідні для зв'язку в блоці «SendToUnity»

Коли модель SMR розроблена, необхідно перевірити модель на наявність помилок. Модель, вільна від усіх помилок, здатна автоматично генерувати вихідні файли C++, файли заголовків і make-файл, необхідні для компіляції та виконання проекту. Треба створити та скомпілювати структуру для конфігурації C++ за замовчуванням. Фреймворк створить файли

інформації про налагодження для бібліотеки. Запуск виконуваного файлу активує панель інструментів анімації.

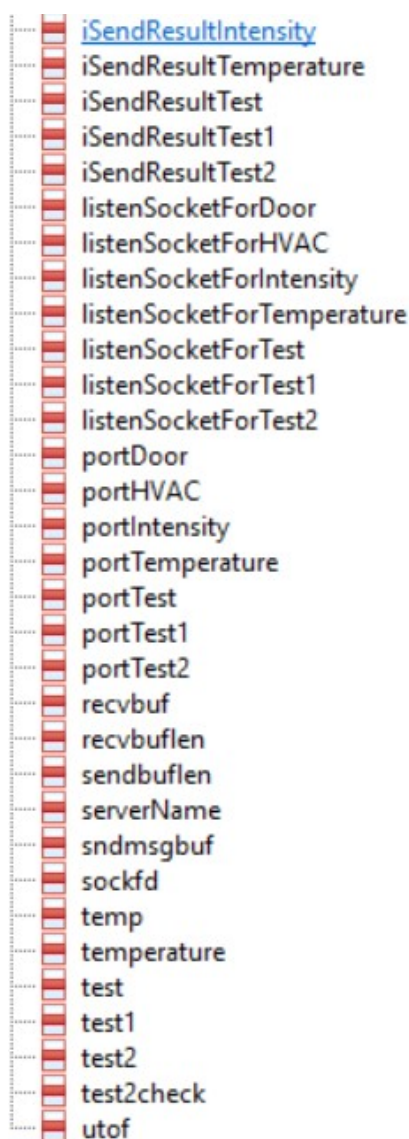


Рис. 3.8. Атрибути, необхідні для зв'язку в блоці «SendToUnity»

### 3.3. Впровадження системи Digital Twin в Unity Game Engine

Віртуальні моделі, необхідні для візуалізації системи SMR, розроблені в Unity Game Engine. Цей інструмент дозволяє користувачеві розробляти 3D-моделі з нуля або безпосередньо зі сховища ресурсів. Крім того, сховище активів у програмному забезпеченні Unity складається з багатьох 3D-моделей, які є у вільному доступі та мають відкритий код. Ресурси

імпортуються в проект як пакет, і можна імпортувати активи, які містять попередньо визначені анімації та поведінку. У наступних підрозділах описано деталі створення системи SMR з нуля, а саме розробку 3D-моделей в Unity та багат шаровій архітектурі Digital Twin.

### 3.3.1. Розробка 3D моделей

Пакет «ModernOfficeInterior» імпортується зі сховища активів і містить майже всі необхідні компоненти системи SMR. Як наслідок, сцена, яка показує тривимірну візуалізацію інтер'єру офісу після імпорту в проект, показана на рисунку 3.9. Як показано на рисунку, пакет складається з необхідних підсистем і компонентів, таких як датчики присутності, «WallVent» для символізації системи HVAC, телевізор із плоским екраном, стіни, підлога, вентилятор, стельові світильники тощо.



Рис. 3.9. Вид сцени пакета «ModernOfficeInterior» зі сховища ресурсів

Однак певні об'єкти і анімації потрібно додавати окремо. Наприклад, кімната для нарад потребує дверей, які відкриваються лише тоді, коли користувач вводить правильний PIN-код у графічному інтерфейсі користувача IBM Rhapsody.

Крім того, необхідна анімація пожежі, щоб зобразити надзвичайну ситуацію в кімнаті для зустрічей. Для цього використовується пакет «Tim'sAssets\_Door» і префаб «FlamesParticleEffect». Обидва вищезазначені компоненти вимагають подальшої деталізації своїх анімацій, щоб вони виглядали відповідними для середовища SMR. Таким чином, у випадку дверей, поведінка стану дверей визначається так, щоб оживити їх. Стан поведінки дверей зображено на рисунку 3.10. Перехід стану А не має тригерів або умов, отже двері залишаються в стані очікування без будь-яких дій. Однак для переходів між станами В і D потрібна умова, що змінна прапора має значення «true». нарешті, оскільки модель вибуху вогню має вбудовану попередньо визначену анімацію, її можна використовувати як таку. Однак розмір і колір полум'я можна змінити за допомогою панелі інструментів Inspector у unity Game Engine. Таким чином, 3D-моделі розробляються, анімуються та візуалізуються для системи SMR.

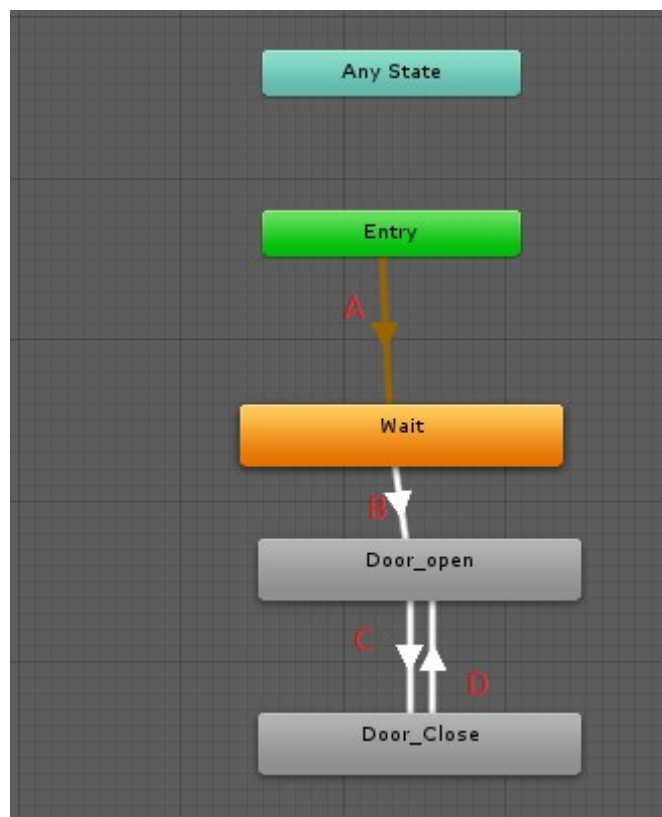


Рис. 3.10. Поведінка стану об'єкту (дверей), що використовуються в системі SMR

### 3.3.2. Реалізація багаторівневої архітектури Digital Twin

Як обговорювалося в другому розділі, цифровий двійник системи SMR розроблений у багаторівневій архітектурі. Перший і основний рівень — це рівень абстракції даних, за яким слідує рівень передачі, рівень цифрового моделювання, рівень інтеграції даних/режимів і, нарешті, рівень програми/сервісу. Що стосується цього проекту, то він не входить до сфери реалізації фізичного збору даних і абстрагування від апаратних датчиків і компонентів. Значення датчиків і дані користувача збираються з моделі IBM Rhapsody. Однак рівень передачі, рівень інтеграції даних/моделі та рівень програми/сервісу реалізовані в окремих сценаріях C# і детально обговорюються в розділах нижче.

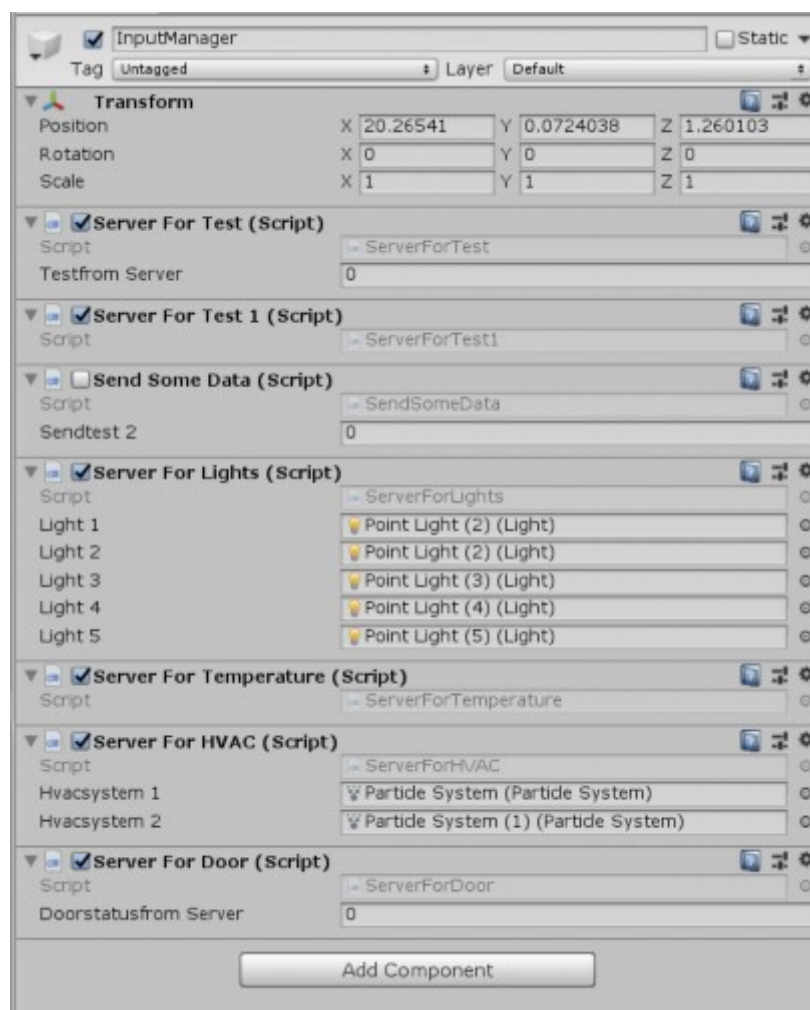


Рис. 3.11. Менеджер введення, який зв'язує серверні сценарії на рівні передачі Digital Twin

Рівень передачі допомагає передавати значення датчиків і дані користувача з моделі Rhapsody на ігровий движок Unity. Зв'язок клієнт-сервер із використанням концепції програмування сокетів Windows уже обговорювався в попередньому розділі. У той час як IBM Rhapsody діє як клієнт із відповідними даними датчиків і інформацією про користувача, ігровий движок Unity діє як сервер, який активує порти і, отже, встановлює зв'язок із IBM Rhapsody.

У рівні цифрового моделювання вигляд сцени системи SMR має бути змодельований у грі, схожій на моду. Для цього в цьому шарі розроблено рух гравця всередині офісного інтер'єру та відповідне управління камерою та позиціонування миші. Скрипти для керування рухом гравця, керування камерою та положенням миші прив'язані до ігрового об'єкта «Гравець» у проєкті unity. Інструкція з контролера персонажа допомагає реалізувати простий рух гравця [11]. Крім того, після тестування руху гравця всередині офісного інтер'єру остаточні параметри керування персонажем показані на рисунку 3.12.

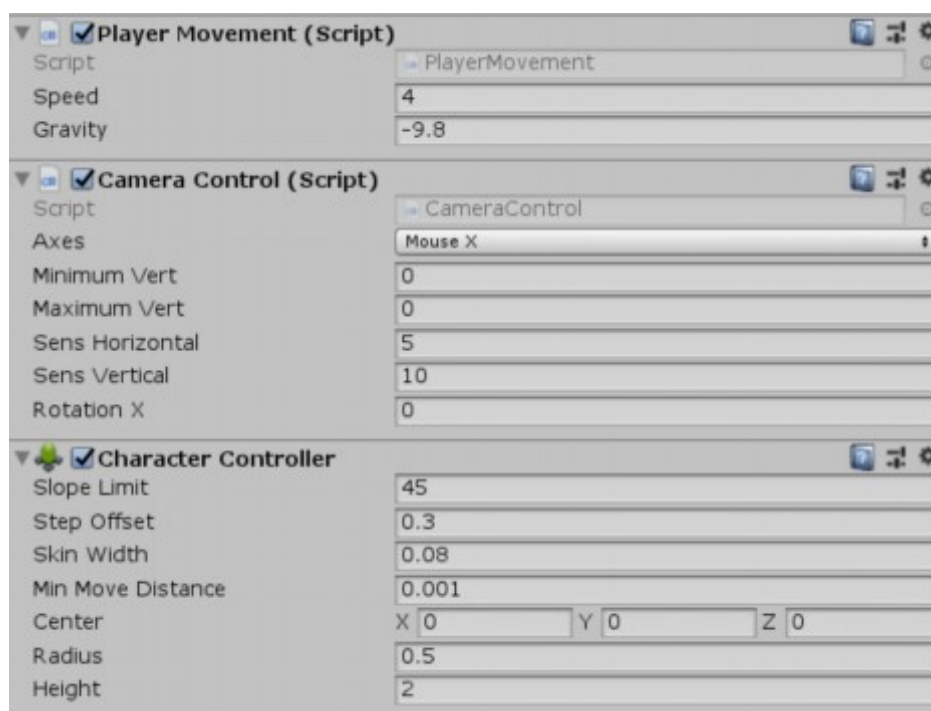


Рис. 3.12. Параметри керування камерою, гравцем і персонажем у Unity Game Engine

Що стосується рівня інтеграції даних/моделі, поведінка окремої підсистеми моделюється шляхом інтеграції з ігровими об'єктами всередині офісного інтер'єру. Наприклад, сценарій зіткнення та тригера допомагає реалізувати функцію датчика присутності, вмикаючи світло щоразу, коли гравець увійшов до кімнати. Подібним чином сценарій налаштування світла допомагає змінювати інтенсивність використовуваного світла. Важливо відзначити, що можна навіть змінити колір використовуваного світла. Крім того, сценарії «Flamechange» і «doomopening» допомагають реалізувати функціональність сценарію вибуху пожежі та анімації відкриття дверей відповідно.

На прикладному рівні встановлюється загальний контроль системи SMR. Кілька режимів системи SMR моделюються в сценарії під назвою "ChangeText". Системні значення, такі як статус HVAC, температура, стан освітлення, повідомлення кімнати для нарад, відображаються на екрані в текстовому форматі разом із аудіовиходом. Сценарій бере вхідне значення режиму з моделі IBM Rhapsody і відповідно змінює поведінку системи.

### **3.4. Демонстрація проекту розумної системи віртуальних комунікацій**

У цьому розділі детально описані результати, отримані після успішного завершення проекту. Наступна серія зображень демонструє роботу системи SMR у різних сценаріях, коли програма поєднана з Unity Game Engine. Результати, які можна спостерігати в кожному штаті, підтверджують роботу системи SMR за допомогою підходу Digital Twin. Для демонстрації відображених результатів було виконано наступні кроки з 1 по 8.

1. Спочатку треба запустити проект в Rhapsody. Панель інструментів анімації автоматично вмикається. Як перший крок, переведіть виконуваний файл у стан очікування, а потім натисніть Go на панелі інструментів анімації. Спочатку кімната вільна, як показано на рисунку 3.13. Щоб увімкнути

систему SMR, потрібно перемкнути кнопку «Увімкнути систему» на панелі керування.

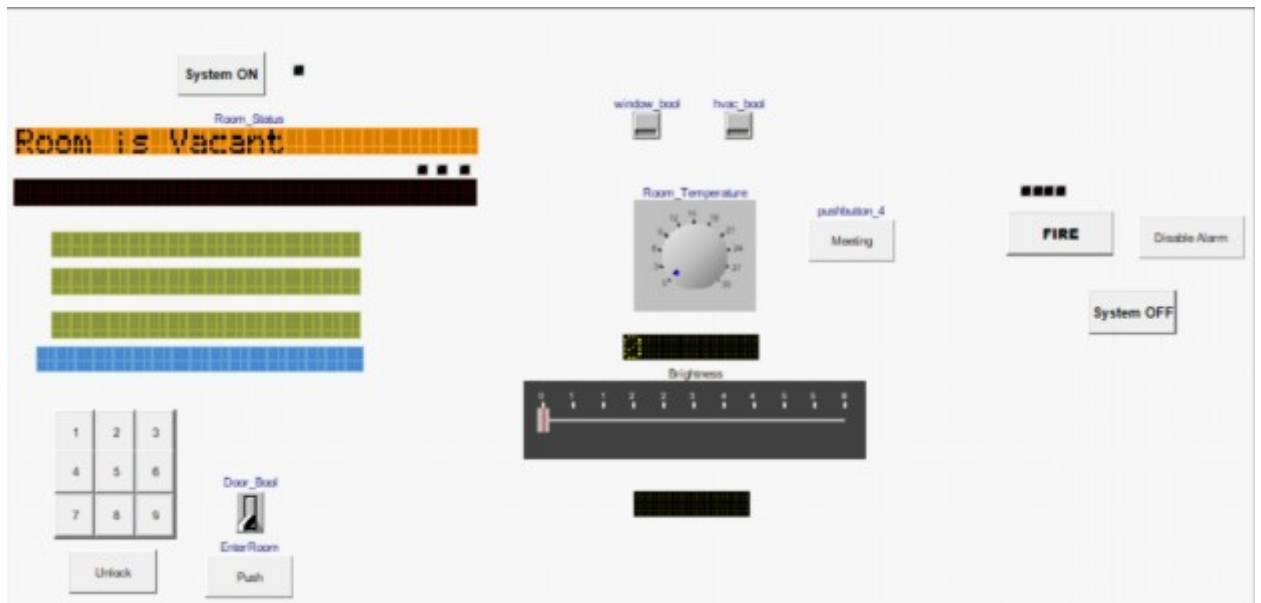


Рис. 3.13. Крок 1а: увімкнення Rhapsody GUI

2. Після ввімкнення системи SMR у графічному інтерфейсі користувача Rhapsody проект Unity можна виконувати. Відповідно до реалізації, дисплей на Unity вітатиме користувача в кімнаті SMR, як показано на рисунку 3.14. Тепер користувач може переміщатися по інтер'єру офісу, щоб перевірити рухи гравця та управління камерою.



Рис. 3.14. Крок 1б: Запуск послідовності в Unity

3. По-друге, користувач тепер може ввести PIN-код, щоб отримати доступ до кімнати. PIN-код «159» забезпечує доступ до кімнати, а повідомлення на дисплеї Rhapsody GUI свідчить про те, що двері зараз розблоковано. Панель керування має розроблені повідомлення, що допомагають у покроковій перевірці моделі, як показано на рисунку 3.15.

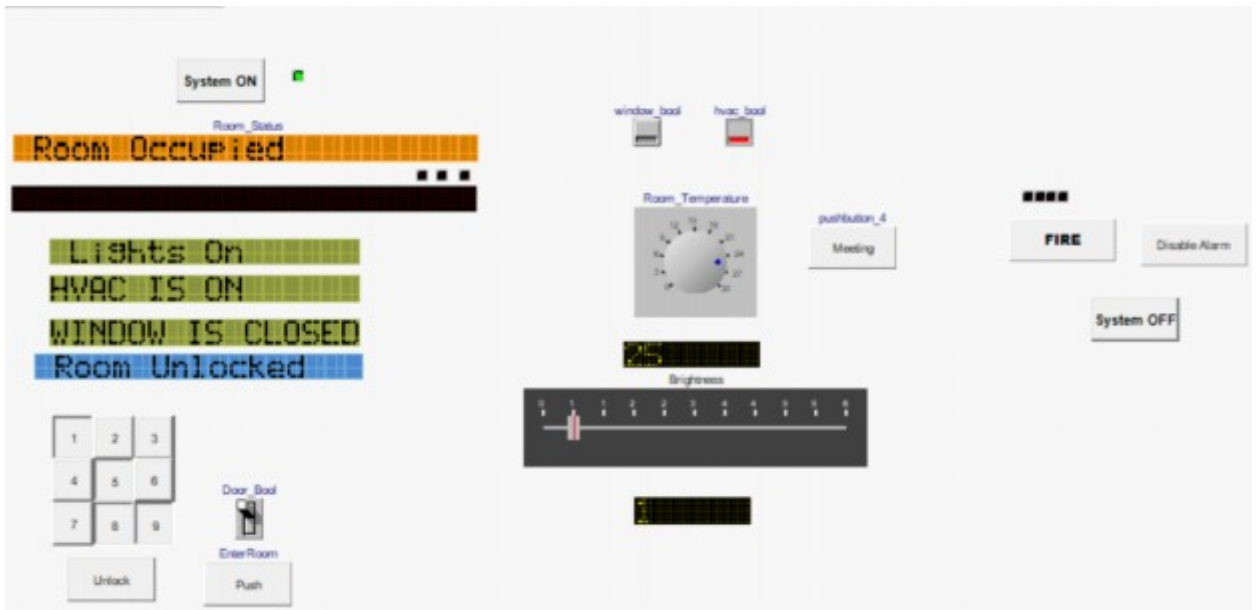


Рис. 3.15. Крок 2а: Надання доступу до кімнати SMR у Rhapsody GUI



Рис. 3.16. Крок 2б: Вхід у віртуальну кімнату переговорів в Unity

4. Після отримання доступу до кімнати шляхом надання правильних облікових даних у Rhapsody GUI двері для SMR в Unity автоматично

розблокуються та відкриваються. Тепер користувач може рухатися всередині кімнати, і система SMR виявляється в стані сну, як показано на рисунку 3.16. Рівень освітлення, статус HVAC і кімнатна температура також відображають належні значення відповідно до проекту.

5. Тепер системою можна керувати, змінюючи температуру та рівень освітлення користувачем. Крім того, користувач тепер може перейти в режим зустрічі. Відповідно до дизайну, режим зустрічі має попередньо визначений набір значень, які можна перевірити в графічному інтерфейсі Rhapsody. Рисунок 3.17 показує графічний інтерфейс Rhapsody, який відображає SMR у режимі зустрічі з відповідними станами підсистеми.

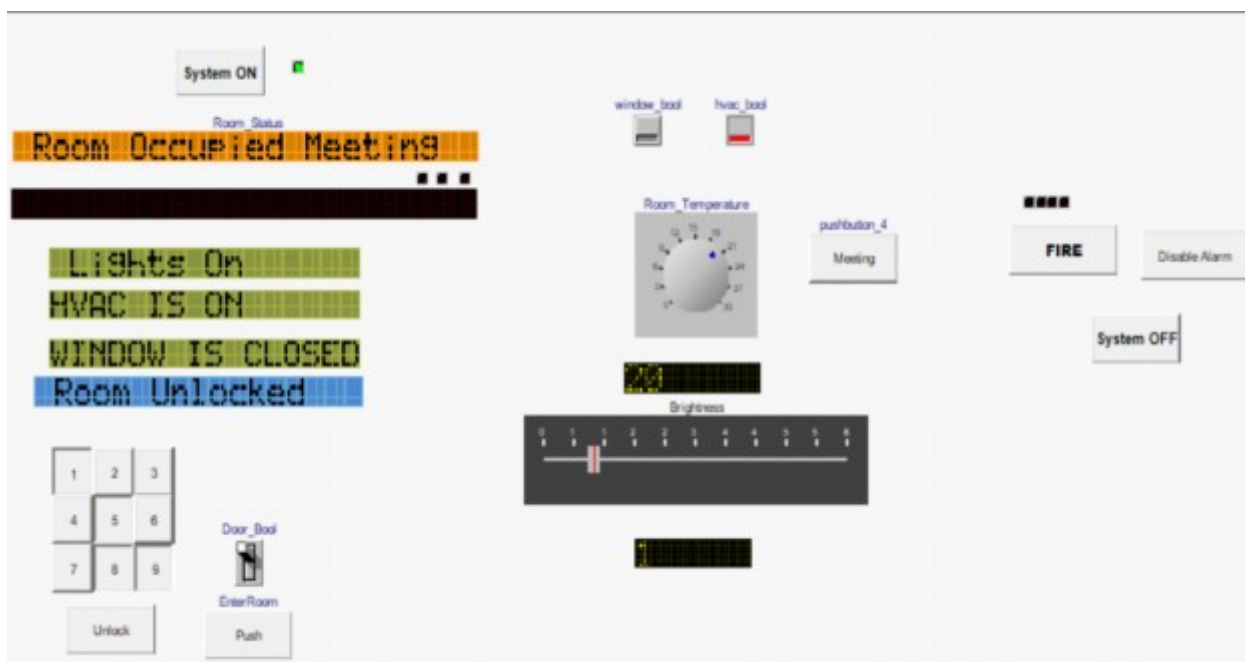


Рис. 3.17. Крок 3а: переключення режиму SMR на “Зустріч”

6. Режим зустрічі (переговорів) системи SMR в unity повністю синхронізується з додатком Rhapsody. Модель unity на рисунку 3.18 показує, що рівень освітлення та температура системи SMR змінюються відповідно до даних, надісланих додатком rhapsody. Крім того, динаміки також увімкнені, і їх вихід чути в цьому режимі. Це перевіряє роботу системи SMR в режимі зустрічі.



Рис. 3.18. Крок 3б: Перевірка зміни в unity

7. Сценарій виникнення пожежі можна змодельовати за допомогою кнопки «Пожежа», змодельованої на панелі керування. Натискання кнопки вогню імітує поведінку SMR і відображає результати на дисплеї, як показано на рисунку 3.19. Ми бачимо, що система HVAC, система освітлення вимкнені з повідомленням про евакуацію.

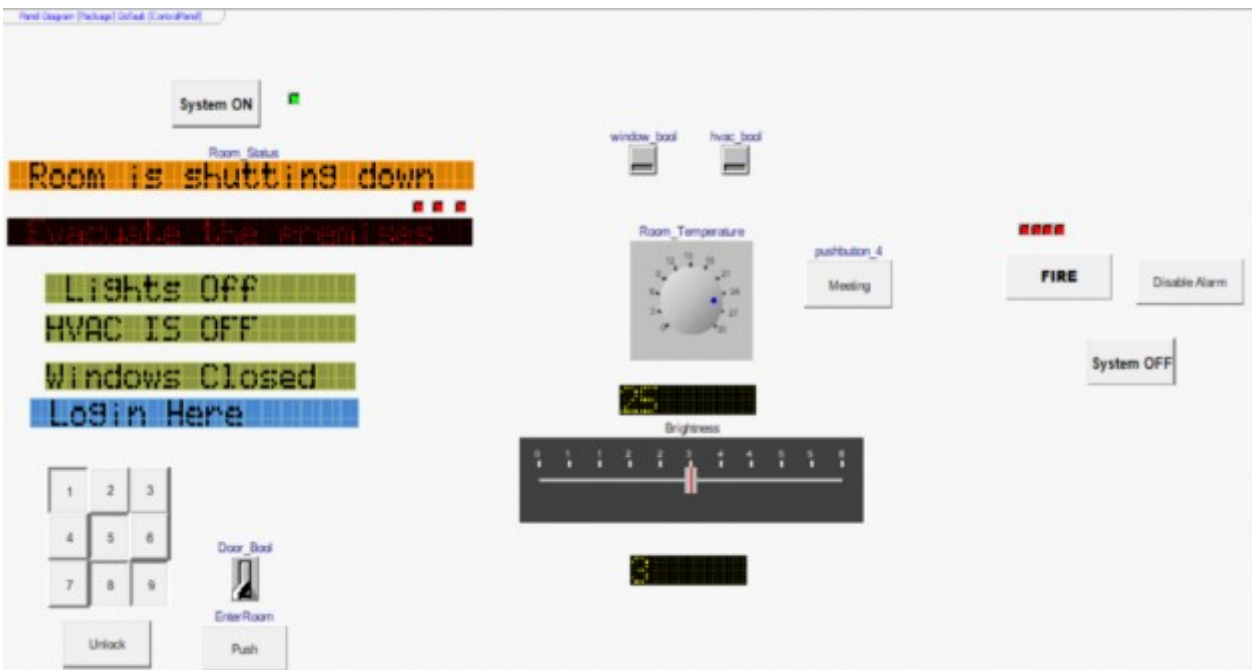


Рис. 3.19. Крок 4а: зміна режиму SMR на пожежну надзвичайну ситуацію в Rhapsody GUI

8. Симуляція пожежі дає такий результат у ігровому движку Unity, як показано на рисунку 3.20. Ми можемо помітити, що програма Rhapsody забезпечує єдину систему SMR, виконуючи необхідні елементи керування та генеруючи спостережувані результати системи. Можна вимкнути пожежну сигналізацію, і система SMR знову повернеться в сплячий стан.



Рисунок 3.20. Крок 4б: Перевірка зміни середовища в Unity

На завершення, розроблена програма розумної системи віртуальних комунікацій (SMR) задовольняє фундаментальні цілі цієї роботи та ефективно відповідає на запитання дослідження. Однак недоліки мінімальні та менш суттєві порівняно з іншими аспектами цього проекту.

По-перше, при оцінці впровадження системи SMR було виявлено, що вона задовольняє базовий набір вимог. Однак певні підсистеми не було реалізовано відповідно до проектних міркувань і специфікацій варіантів використання. Наприклад, робота камер відеоспостереження в системі безпеки, активація сенсорної панелі в аудіовізуальній системі, функції конференц-зв'язку та відеодзвінків у телевізорі, датчики виявлення руху та активація віконної панелі видалені з реалізації системи SMR. Головним чином це пов'язано зі складністю віртуальної реалізації акторів і компонентів ігрового движку Unity. Крім того, це не служить жодній додатковій меті,

щоб сприяти дослідницьким питанням. Таким чином, ці випадки використання та пов'язані з ними суб'єкти були видалені з реалізації системи SMR.

По-друге, щодо тестування варіантів використання системи SMR для зв'язку з Digital Twin не вдалося виконати. Це головним чином через те, що анімація, увімкнена в IBM Rhapsody, не дозволяє виконувати операції в портах Linux на контролері Raspberry Pi. Хоча систему SMR можна успішно запуснути на віддаленому пристрої, функціональні можливості системи SMR не були повністю перевірені на контролері Raspberry Pi. Однак, оскільки в реалізації моделі немає змін, можна стверджувати, що тестові приклади працюватимуть подібно до реалізації Windows. Крім того, оскільки тестування та оцінка впровадження HIL не є частиною питання дослідження, це можна залишити для майбутнього розвитку. Відповідним обхідним шляхом для цієї ситуації може бути експорт діаграми панелі керування до веб-інтерфейсу та доступ до веб-сторінки з контролера Raspberry Pi, щоб надати користувачеві дані для Digital Twin. Крім того, у наведених нижче розділах знайдено та обговорено нове розуміння та майбутній обсяг удосконалення, пов'язаний із проектуванням цифрового близнюка на основі підходу системної інженерії, керованого моделлю.

Програма SMR, розроблена в рамках проекту, має функціональність і перевагу взаємодії між різними процесами. Моделювання SysML в інструменті IBM Rhapsody допомагає автоматично генерувати вихідні файли C++, необхідні для успішного виконання проекту. У цьому проекті значною мірою використано перевагу IBM Rhapsody як інструменту взаємодії між різними програмами. Дійсно, віртуальна модель у 3D-просторі розроблена в Unity Game.

Двигун успішно поєднується з програмою SMR. Крім того, оскільки C++ пропонує переносимість коду в різні середовища ОС, простіше виконувати проект на віддалених пристроях за допомогою відповідного профілювання моделей для необхідного середовища ОС у інструменті IBM

Rhapsody. Це було продемонстровано за допомогою Linux-профілювання програми SMR і, таким чином, виконання проекту у віддалено підключеному контролері raspberry pi (Raspbian OS — ОС на базі Linux).

Крім того, додаток можна розширити, щоб мати додаткові контролери та автономні компоненти, які працюють паралельно на одній або різних хост-машинах. Наприклад, у програмному забезпеченні MATLAB/simulink можна застосувати додатковий рівень керування, щоб дозволити багатоагентне моделювання. Хоча це не було реалізовано в цьому проекті, це корисна концепція, яку можна використати в майбутньому.

Підхід системної інженерії, керований моделлю, має перевагу в тому, що він гнучкий для розробки на різних етапах. Під час проекту було виявлено, що гнучкість моделювання SysML пропонує нові доповнення до функцій, а також модифікацію існуючих атрибутів функцій і послуг простим і менш зайвим методом. Наприклад, додавання нової функції, такої як анімовані двері в 3D-простір, потребує сигналів програмного забезпечення для їх активації. Ці програмні сигнали мають бути згенеровані з програми SMR через моделі SysML. Таким чином, це вимагає додаткових портів для зв'язку сокетів і подальшого відповідного проектування на рівні інтеграції моделі, а також на рівні програми/сервісу. Однак ці зміни жодним чином не заважають роботі вже існуючих компонентів або служб, оскільки всі вони відокремлені окремими рівнями. Таким чином, у гнучкому системному інженерному підході, що керується моделлю, існує мінімум переробок, що покращує продуктивність і, таким чином, підвищує економічну ефективність розробки життєвого циклу.

### **Висновки до розділу**

У третьому розділі розглянуто імплементацію модель-базованої методології для проектування цифрових образів у віртуальних комунікаціях. Запропонована модель була успішно впроваджена, що підтвердило

ефективність підходу до створення інтерактивних цифрових образів із реалістичною поведінкою. Це підвищує якість та інтерактивність віртуальних комунікацій, роблячи їх більш залученими та динамічними.

Реалізація взаємодії віртуальної моделі з Unity дозволила створити інтерактивне середовище для цифрових двійників. Моделювання введення користувача та відображення в Unity забезпечує плавність і реалістичність взаємодії, що підвищує користувацький досвід та дає можливість отримати достовірне відтворення поведінки цифрових образів.

Також успішно впроваджено систему Digital Twin в Unity Game Engine, включаючи розробку 3D моделей та багаторівневу архітектуру, яка охоплює всі необхідні етапи обробки й відображення даних. Це дозволяє точно моделювати фізичні процеси у віртуальному середовищі, що сприяє вдосконаленню системи цифрових комунікацій.

Демонстрація проєкту розумної системи віртуальних комунікацій показала, як модель-базована методологія дозволяє створити повноцінне рішення для інтерактивної взаємодії в цифровому просторі, що підвищує потенціал Digital Twin у контексті віртуальних комунікацій.

## ВИСНОВКИ

У ході магістерського дослідження було проведено комплексний аналіз предметної області проектування цифрових образів для віртуальних комунікацій, акцентуючи увагу на концепції та архітектурі Digital Twin. В результаті дослідження зроблено такі висновки:

Проведено ґрунтовний аналіз Digital Twin як ключового елемента сучасних цифрових комунікацій, що дає змогу створювати інтерактивні та реалістичні копії об'єктів у віртуальному середовищі. Визначено його місце в процесі проектування та роль у покращенні інтерфейсу користувача.

Представлено методологію розробки програмного забезпечення для систем віртуальних комунікацій. Запропоновано модульний підхід, який структурує процес розробки програмного забезпечення та сприяє спрощенню інтеграції компонентів Digital Twin. Розглянуто використання V-Model у цьому контексті, що дає змогу чітко структурувати етапи проектування та забезпечує зручність верифікації на кожному з них.

Детально розроблено концепцію застосування циклів MIL (Model-in-Loop), SIL (Software-in-Loop) та DTIL (Digital-Twin-in-Loop) для забезпечення поступового тестування та інтеграції цифрових двійників в систему.

Побудовано п'ятирівневу архітектуру, що включає рівні збору даних, передачі, цифрового моделювання, інтеграції даних/моделей та обслуговування. Це рішення сприяє підвищенню масштабованості та функціональності системи, а також забезпечує ефективну обробку даних і доступ до різноманітних сервісів.

Виконано імплементацію системи Digital Twin на платформі Unity. Впровадження Digital Twin у Unity забезпечило можливість інтерактивного моделювання цифрових образів, зокрема розробку 3D моделей та відображення користувацьких сценаріїв. Досліджено питання інтеграції користувацьких дій та відгуків системи для підвищення реалістичності

віртуальної взаємодії. Демонстрація роботи розумної системи віртуальних комунікацій. Розроблена система показала свою ефективність у моделюванні та управлінні складними віртуальними середовищами, що в перспективі сприятиме розвитку нових технологій у сфері цифрових комунікацій.

Отримані результати можуть бути корисними для проєктування цифрових двійників у різних сферах, зокрема в інтелектуальних системах керування, симуляціях реального часу та навчальних середовищах.

## ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Azuma, R. T. (1997). "A survey of augmented reality." *Presence: Teleoperators & Virtual Environments*, 6(4), 355–385.
2. Bainbridge, W. S. (2007). "The scientific research potential of virtual worlds." *Science*, 317(5837), 472–476.
3. Burdea, G. C., & Coiffet, P. (2003). *Virtual reality technology*. Wiley-IEEE Press.
4. Cassidy, T. (2008). *Virtual Reality and Artificial Environments in Communication Studies*. Routledge.
5. Davis, M. (2003). *Computational Models of Creative Design in Interactive Digital Media*. MIT Press.
6. Egger, M., et al. (2020). "Design patterns for digital twins." *Future Internet*, 12(6), 102.
7. Gerbaud, S., et al. (2008). "Virtual reality and augmented reality for collaborative product design." *Computers in Industry*, 59(7), 732–741.
8. Gero, J. S., & Kannengiesser, U. (2004). "The situated function-behaviour-structure framework." *Design Studies*, 25(4), 373–391.
9. Grieves, M. (2014). "Digital twin: Manufacturing excellence through virtual factory replication." White paper, 1–7.
10. Gusikhin, O., & Meckl, P. H. (2007). "Model-based design approaches in the development of intelligent mechatronic systems." *IEEE Transactions on Industrial Electronics*, 54(4), 2342–2348.
11. Hall, E. T., & Braff, P. J. (2005). *Communication Theory and Virtual Reality*. Palgrave Macmillan.
12. Jain, P. (2013). "Augmented reality and virtual reality applications in design visualization." *Visual Computing for Industry, Biomedicine, and Art*, 1(1), 1-9.

13. Kasap, M., & Magnenat-Thalmann, N. (2007). "Intelligent virtual humans with interactive cultural protocols." *IEEE Computer Graphics and Applications*, 27(5), 32–39.
14. Kim, K., et al. (2019). "Designing user interfaces for virtual reality: A literature review and model-based design approach." *ACM Computing Surveys (CSUR)*, 52(6), 1–38.
15. Koehler, W., et al. (2010). "Using virtual characters in social communication." *International Journal of Human-Computer Studies*, 68(3), 141–149.
16. A. Barni, A. Fontana, S. Menato, M. Sorlini and L. Canetta, "Exploiting the Digital Twin in the Assessment and Optimization of Sustainability Performances," 2018 International Conference on Intelligent Systems (IS), Funchal - Madeira, Portugal, 2018, pp. 706-713.
17. F. Tao, H. Zhang, A. Liu and A. Y. C. Nee, "Digital Twin in Industry: State-of-the-Art," in *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405-2415, April 2019.
18. Leavitt, A., & Wise, K. (2009). "Creating avatars: Digital identities in virtual worlds." *New Media & Society*, 11(5), 769–788.
19. Lee, J., et al. (2017). "Cyber-physical systems for digital twin-driven manufacturing." *Journal of Manufacturing Systems*, 43, 381–391.
20. Levac, D., et al. (2010). "Virtual reality and motor learning in rehabilitation." *Cyberpsychology, Behavior, and Social Networking*, 13(5), 523–532.
21. S. Paladi, MIL, SIL, HIL for Testing Electronic Controls in Vehicle Engineering, Master Internship. 2019.
22. "The Advantages of Using Digital Twins in HVAC Systems: A Primer", *HPAC Engineering*, 2020. [Online]. Available: <https://www.hpac.com/iaq-ventilation/article/21133096/the-advantages-of-using-digital-twins-in-hvac-systems-a-primer>.

- 23.Li, C., & Tung, W. T. (2012). "Virtual reality-based social networks: The design of interactive communication in digital environments." *International Journal of Information Management*, 32(1), 100–107.
- 24.Lin, C. Y., & Duh, H. B. (2013). "A survey of digital twin research in manufacturing." *International Journal of Production Research*, 51(23–24), 7317–7342.
- 25.Lombard, M., & Ditton, T. (1997). "At the heart of it all: The concept of presence." *Journal of Computer-Mediated Communication*, 3(2).
- 26.Mangina, E., & O’Keefe, M. (2009). "Enhancing collaborative virtual reality using avatars." *Educational Technology & Society*, 12(4), 45–53.
- 27.Marr, B. (2017). *Data Strategy: How to Profit from a World of Big Data, Analytics and the Internet of Things*. Kogan Page Publishers.
- 28.McCrea, P. (2008). "Designing digital avatars for interactive environments." *Computers in Entertainment*, 6(4), 5.
- 29.Morreale, S. P., et al. (2010). *Human Communication: Motivation, Knowledge, and Skills*. Cengage Learning.
- 30.Müller, H., et al. (2012). "User-centered design and evaluation of a virtual reality application." *Human-Computer Interaction*, 27(1–2), 36–61.
- 31.Munzner, T. (2014). *Visualization Analysis and Design*. CRC Press.
- 32.Neff, M., et al. (2011). "Performance analysis of virtual characters in immersive environments." *IEEE Transactions on Visualization and Computer Graphics*, 17(6), 829–839.
- 33.North, C., & Shneiderman, B. (2000). "Snap-together visualization: A user interface for coordinated views." *International Journal of Human-Computer Interaction*, 12(3), 263–282.
34. "Lighting control system", [En.wikipedia.org](https://en.wikipedia.org), 2020. [Online]. Available: [https://en.wikipedia.org/wiki/Lighting\\_control\\_system](https://en.wikipedia.org/wiki/Lighting_control_system).
35. "Medium. 2020. How To Write A Simple 3D Character Controller In Unity." [online] Available at: <https://itnext.io/how-to-write-a-simple-3d-character-controller-in-unity>

36. Oliver, M., & Carr, D. (2009). "Learning in virtual worlds: Using communities of practice." *British Journal of Educational Technology*, 40(3), 444–457.
37. Park, S., et al. (2021). "Augmented reality digital twin for maintenance in industrial environments." *Procedia CIRP*, 97, 316–321.
38. Peeters, M. M., & Megens, C. J. (2014). "A digital twin framework for model-based development." *International Journal of Digital Engineering*, 4(3), 216–228.
39. Reitmayr, G., & Schmalstieg, D. (2004). "Collaborative augmented reality for outdoor navigation." *ACM SIGGRAPH 2004 Course Notes*.
40. Schilling, M. (2008). "Virtual worlds and augmented reality in communication training." *Journal of Communication*, 12(4), 379–388.
41. Schroeder, R. (2011). *Being There Together: Social Interaction in Shared Virtual Environments*. Oxford University Press.
42. Steuer, J. (1992). "Defining virtual reality: Dimensions determining telepresence." *Journal of Communication*, 42(4), 73–93.
43. Tang, A., & Hummels, C. (2014). "Designing for digital presence in virtual spaces." *Human-Computer Interaction*, 29(3), 235–255.
44. Tori, R., et al. (2006). "Using virtual reality in the classroom." *Educational Technology & Society*, 9(4), 202–213.
45. Warburton, S. (2009). "Second Life in higher education: Assessing the potential for and the barriers to deploying virtual worlds in learning and teaching." *British Journal of Educational Technology*, 40(3), 414–426.
46. White, M., & Le Cornu, A. (2011). "Digital natives, digital immigrants, and the digital divide." *Digital Creativity*, 22(3), 129–137.
47. Yee, N., & Bailenson, J. N. (2007). "The Proteus effect: The effect of transformed self-representation on behavior." *Human Communication Research*, 33(3), 271–290.
48. Zhang, Z., et al. (2018). "Real-time virtual reality and digital twin simulation for predictive maintenance." *IEEE Access*, 6, 43021–43029.

49. Zhao, S. (2003). "Toward a taxonomy of copresence." *Presence: Teleoperators & Virtual Environments*, 12(5), 445–455.
50. O'Connor, C. IBM IoT Platform. Available online: <https://www.ibm.com/blogs/internet-of-things/leaderiot-platforms/>.
51. Madni, A.M.; Madni, C.C.; Lucero, S.D. Leveraging Digital Twin Technology in Model-Based Systems Engineering. *Systems* 2019,
52. Mohammadi, N., Taylor, J. E. (2017). "Smart city digital twins." 2017 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, pp. 1-5.