School of Computer Science and Engineering Faculty of Engineering
The University of New South Wales

Enhancing Buyer Engagement through AI-Driven VR Real Estate Exploration

by

Kyle Wu

Thesis submitted as a requirement for the degree of

Bachelor of Engineering in Software Engineering

Submitted: November 2024 Supervisor: Dr. Samad Sepasgozar

Student ID: z5363490

Abstract

The real estate industry has increasingly adopted advanced technologies to enhance customer engagement and streamline property marketing. However, existing solutions need an integrated approach that combines immersive experiences with intelligent, context-aware AI support. This thesis presents the development of VIVA (Virtual Interactive Viewing Agent), a real estate tour featuring an AI-powered real estate agent. VIVA offers users a seamless and interactive experience for property exploration.

The project was built on a thorough literature review examining the state of property tours and AI adoption in real estate, identifying fundamental limitations such as agent bias and poor user engagement. The system architecture includes an interactive UE5 simulation integrated with a Convai chatbot for real-time interaction. The methodology leverages cutting-edge frameworks and tools to ensure usability, reliability, and scalability.

Evaluation of the prototype demonstrated its feasibility and effectiveness in addressing the identified gaps, though further user testing is required for comprehensive validation. The outcomes of this research provide a robust foundation for future advancements in AI-enhanced applications in real estate, with potential extensions including automated virtual reconstructions using 3D scanners.

Acknowledgements

I would like to thank my supervisor, Dr Samad Sepasgozar, for his expertise and commitment to my success. His guidance and insightful feedback throughout my thesis have been invaluable.

I would also like to thank my assessor, Dr Ali Darejeh, for his helpful feedback on my project.

This work has been inspired by the academic papers cited in the Bibliography.

Abbreviations

AI Artificial Intelligence

VR Virtual Reality

3D Three Dimensional

MVP Minimum Viable Product

UE5 Unreal Engine 5

Contents

Table of Contents

Chapter 1

Introduction

The built environment touches upon all aspects of our lives, from the buildings we live in to the systems we use. However, it has historically been restricted by the physical limitations of our world. This is until recently, with the emergence of immersive technologies like Virtual Reality and the rapid developments in Artificial Intelligence, the landscape of the built environment has begun to shift [1], as we are given digital ways to interact with the world. This is especially true in the case of the real estate sector, which due to increasing levels of funding into technology, has experienced a meteoric rise in the applications of these technologies [2].

A prominent example of a technological innovation was the creation of the 3D VR property tour, which gained significant traction as an alternative to physical property tours during the COVID-19 pandemic [3]. COVID-19 restrictions heavily motivated the rapid adoption of these virtual tours as a way for real estate businesses to survive the pandemic [3]. Whilst physical tours had historically presented geographical barriers for foreign buyers [4], who were unable to view properties due to physical limitations, the addition of new barriers to local buyers [5], led to the rapid local adoption of these tours. These two barriers were supposedly minimized through the introduction of the 3D VR property tour, however, later studies identified that once differences in image quality were accounted for, these tours were shown to ultimately offer marginal benefits, stemming from their lack of user engagement [6]. As such, this report describes the execution of a software solution which has bridged this gap in the market, by leveraging an AI-powered real estate agent, to improve upon 3D virtual real estate tours.

This report consists of 5 chapters that together justify the importance of this solution. Chapter 2 provides a detailed literary review of the key themes that underpin this thesis, by summarising the current state-of-the-art technologies, identifying gaps within the literature, and establishing the problem statement, research aims and report outcomes. Chapter 3 delivers an extensive methodology for the recreation of the project and its deliverables. Chapter 4 provides a comparative analysis of this project and similar existing systems. Finally, Chapter 5 concludes this thesis and examines possible directions for future work.

Chapter 2

Literature Review

In this section of the report, we will decompose the project into two central themes (**2.1**), and thematically review the associated literature (**2.2, 2.3**). From this review, we will identify gaps in the existing literature to develop an informed problem statement, appropriate research aim, and a complete list of expected outcomes (**2.4, 2.5**).

2.1 Project Themes

For this project, the two central themes are:

1. VR Integrated Real Estate Tours
2. The Adoption of AI Chatbots in Real Estate

2.2 VR Integrated Real Estate Tours

In recent years, we have witnessed a surge in the research and development of VR integrated real estate tours. This growing interest appears to stem from the potential of VR to revolutionise the property viewing experience. Through reviewing past simulations, we will evaluate the current state of these tours and provide evidence that whilst VR real estate tours can become an effective alternative to physical viewing, there are currently unable to.

2.2.1 Virtual Reality for Real Estate its evolution in Bluemind Software

In mid-2017, following the completion of the VR4RE (Virtual Reality for Real Estate) project by Bluemind Software, Dr Deaky et al. discussed the imminent evolution of the real estate industry, as the use of modern technologies, primarily referring to VR, could be used to save time and money for both real estate buyers and sellers [7]. VR4RE aided real estate sellers by transporting viewers directly to the property, allowing them to view a range of properties from afar and only physically visit the most compelling ones [7]. Whilst this project represented a major innovation in real estate exploration, as one of the earlier applications of VR, this project more so provided a starting point for future applications, as it notably suffered from low visual fidelity and poor user engagement caused by a lack of user interactivity [7].

2.2.2 Matterport: Virtual Tour as A New Marketing Approach in Real Estate Business During Pandemic COVID-19

Amidst the COVID-19 pandemic, the real estate industry was reported as one of the most critically affected economic sectors, since travel restrictions prevented potential buyers from being able to physically visit properties [3]. This led to the mainstream adoption of the 3D VR tour, as an alternative marketing solution [3]. This led to the creation of the Matterport 3D VR tour, which greatly improved on its predecessors, namely the VR4RE project, by introducing major visual enhancements and the ability to make accurate measurements within the tour.

The Matterport 3D VR tour also introduced the ability to display sections of information for select items, providing users with the ability to gain detailed information by selecting an item in the tour [3]. These additions meant that the new tour was able to improve upon some of the shortcomings of past applications.

2.2.3 Beyond the Hype: Unveiling the Marginal Benefits of 3D Virtual Tours in Real Estate

Despite the improvement and apparent benefits of the 3D VR tour, in mid-2023, Dr Zhang et al. posited that the actual benefits were far smaller than they were advertised to be and, that in their current state, virtual tours were not suitable replacements for traditional ones [6]. Using the sales data of 75,178 houses sold in the greater Los Angeles area from March 2019 to March 2021, Dr Zhang et al. established that after controlling for photo quality and listing descriptions, virtual tours did not significantly increase sale prices, suggesting that that the benefits were marginal at best [6]. They identified that ‘beyond the hype’, virtual tours had the same impact as video showcases, as they failed to meaningfully engage with the users.

2.2.4 Current Limitations of VR Integrated Real Estate Tours

From the review of this theme, we see that there exists a gap in the current literature, resulting from the fact that whilst VR integrated real estate tours offer a wealth of benefits to foreign and local buyers alike, because of their current lack of user interactivity and engagement, they are ultimately redundant.

2.3 The Adoption of AI Chatbots in Real Estate

Similar to VR, AI has also experienced a rapid adoption in the real estate industry, caused by its potential to improve the efficiency and reduce the cost of many real estate processes [8]. One such use case for AI specifically is the creation of AI chatbots and virtual assistants, which are able to support customer service providers and contribute to lead generation for property listings [8]. This section of the review will evaluate the potential of using an AI chatbot as a solution to the gap identified in section 2.2.4.

2.3.1 Real Estate App Development Based on AI/VR Technologies

In 2023, Dr. Miljkovic et al. completed an investigation on the development of a real estate application that would use VR and AI technologies to improve upon common real estate processes, such as automating property inspections and showcases [9]. This was done by utilising AI to analyse large volumes of market data and produce insights and prediction on property prices, rental rates, and other important market trends, improving the ability for real estate professionals to make informed business decisions [9].

During their investigation, they also highlighted the significant benefits of AI-powered real estate apps for professionals and consumers, citing the current success of chatbots in customer-facing general inquiry services [9]. They noted that AI-powered chatbots are able to continuously evolve, providing customers with a far more personalised and responsive service than what a human could provide [9]. However, whilst these chat bots appear to be promising, most existing implementations are not actually able to personalise their responses and consequently have poorer outcomes than human support providers [9]. As such, from this investigation, we can conclude that AI chatbots with the addition of NLP-based user profiling, can offer a suitable solution to the gap identified in the previous section, as their scalability, once paired with user profiling can offer a more engaging virtual tour.

2.4 Problem Statement

As identified through the literature review, the current problem lies in the gap between the potential of VR real estate tours to replace traditional tours and the limitations of their current, non-interactive solutions. By lacking in interactivity and engagement, these solutions fail to provide a truly immersive experience comparable to that of a physical property visit.

2.5 Research Aim and Outcomes

To solve the above problem statement, this thesis aimed to produce a feasible, novel application of VR and AI in the real estate sector, using these technologies to enhance the interactivity and engagement of the buyer during their virtual exploration of a new property.

To achieve this aim, the following objectives must be met:

1. Develop a prototype for an immersive and interactive 3D VR compatible real estate tour.
    - Current virtual tours lack interactive elements [6], such as light switches, doors, windows, or furniture, so the prototype will contain these interactive elements to enhance user engagement.
2. Integrate an AI chatbot inside of the prototype, which customises it responses based on the user.
    - Current implementation of AI chatbots exist as general support agents [8], providing general assistance to users on real estate websites, so the prototype will contain an AI chatbot within the simulation environment, capable of offering user-tailored advice.
3. Document the virtual construction of the property and the automation of property-based data collection.
    - The process of virtual construction and data collection must be replicable and scalable by future engineers to ensure that the thesis has a lasting impact on the research space.

As such, following the completion of all objectives, the project will produce the following outcomes:

1. A prototype interactive simulation of a 3D VR compatible real estate tour led by an AI-powered real estate agent.
2. Complete documentation for the recreation and data collection of a new property.

Chapter 3

Solution and Execution

3.1 Solution

The *Virtual Interactive Viewing Agent* (VIVA) is a greenfield project that enhances the engagement of virtual real estate exploration through the adoption of a real-time virtual agent. The main purpose behind this project is to provide potential buyers and renters with 24-hour, on-demand access to personalised and informed property tours.

The application consists of two central milestones:

1. Property Exploration
2. Real-Time Virtual Agent

3.1.1 Property Exploration

This is a central component of the VIVA system, allowing users to explore a pre-constructed property in a virtual environment. The workflow begins by loading a high-fidelity model of the sample property, allowing users to navigate through various rooms and outdoor spaces in a fully immersive and interactive environment. To enhance engagement, interactive elements are added in the virtual environment, including interactable doors, life-like interior lighting, accurately scaled property dimensions and fine-tuned player collision.

3.1.2 Real-Time Virtual Agent

During the property exploration, users can be led or followed (as requested by the user) by a real-time virtual agent that provides them with in-depth, up-to-date information about the features of the property, including its amenities, price history and neighbourhood. In addition, the virtual agent is capable of moving towards specified areas of the property and provide detailed descriptions of important features, such as open-floor designs.

3.2 Scope and Requirements

Whilst the VIVA system has near-endless possibilities for additional features and enhancements, such as advanced user analytics, automated property creation and real-life fidelity, the time and resource constraints of this thesis limit the current implementation to the two milestones detailed above. This scope of work was chosen to account for the additional effort required to complete relevant training, documentation and testing.

3.2.1 Functional Requirements

The functional requirements of this project were drawn from the aim, objectives and outcomes identified in **Section 2.5**. These functional requirements are shown below in **Table 3.2.1** and are expressed as user stories and acceptance criteria.

Table 3.2.1 User stories and acceptance criteria of the VIVA system.

Requirement	User Story	Acceptance Criteria
1	• **As a** potential property buyer or renter, • **I want to** explore properties virtually, • **So that** I can save time when looking through my options.	• The user loads into the virtual property. • The virtual exploration allows the user to navigate through the entire property, including all rooms and outdoor spaces. • The user can enter and exit rooms smoothly without delays or glitches. • The virtual environment is visually accurate and representative of the actual property.
2	• **As a** potential property buyer or renter, • **I want to** be able to interact with the environment, • **So that** I can fully immerse myself inside the virtual environment.	• The user faces a door and presses the E key. • The door the user is facing opens. • The door creates the sound of a door opening. • The user walks through the door. • The user turns around to face the door and presses the E key. • The door the user is facing closes. • The door creates the sound of a door closing.
3	• **As a** potential property buyer or renter, • **I want to** have access to up-to-date information on a property through an agent, • **So that** I can make an informed decision on the value of a property.	• The user asks the virtual agent a specific question about the amenities of the property. • The virtual agent answers the question with up-to-date and accurate information about the property's amenities. • The user asks about the historical price data of the property. • The virtual agent answers the question with up-to-date and accurate information about the property's price history.
4	• **As a** potential property buyer or renter, • **I want to** have access to up-to-date information on the neighbourhood of a property, • **So that** I can see if the surrounding community matches my needs and values.	• The user asks the virtual agent to describe the neighbourhood. • The system responds with an overview of the neighbourhood. • The system highlights specific points of interest, such as schools, parks and public transport. • The user asks the virtual agent about a point of interest. • The virtual agent explains the relevant point of interest, with details such as a description of the location and its distance from the property.
5	• **As** a user, • **I want to** provide	• The user provides feedback to the virtual agent about how they want more detailed

	feedback on my experience, • **So that** the VIVA system can improve its usability, features and functionalities.	descriptions. • The virtual agent stores the feedback. • The user asks a question about the amenities of the property. • The virtual agent answers the question with a highly detailed response of each amenity.
6	• **As a** potential property buyer or renter, • **I want to** have access to a completely guided tour of the property, without any prior knowledge about the property • **So that** I can efficiently explore all key aspects of the property.	• The user starts the simulation and tells the agent that they want to be led. • The virtual agent completes the tour without any further instructions on where to go. • The virtual agent listens for questions after described every room.

3.2.2 Non-Functional Requirements

The non-functional requirements of this project were chosen to ensure the system is user-friendly, robust and maintainable whilst also meeting the performance and usability expectations of the potential end-user. These requirements aim to complement the functional requirements by focusing on the overall quality attributes of the system. These non-functional requirements are shown below in **Table 3.2.2**.

Table 3.2.2: Non-Functional Requirements of the VIVA system and their descriptions.

Requirement	**Description**
Consistent UI and UX	Both user interfaces and their respective workflows for the VIVA system should be easy to use, and all interactions within and between user sessions should be consistent.
Good application performance	The VIVA system must run with an average of at least 30 frames per second on a mid-tier computer.
Well-documented	The VIVA system should have comprehensive documentation on its design and creation so that future developers may maintain and build upon the current implementation.
Well-tested	The VIVA system should contain a series of manual and automated tests, which verify that the system reliably fulfils the acceptance criteria detail in **Table 3.2.1**.

3.3 System Architecture

The following software systems were selected as part of the software technology stack of the VIVA system. The uses of and relationships between these technologies are shown through the software architecture diagram in **Figure 3.3** and show groups of technologies and frameworks used to create specific modules of the system.

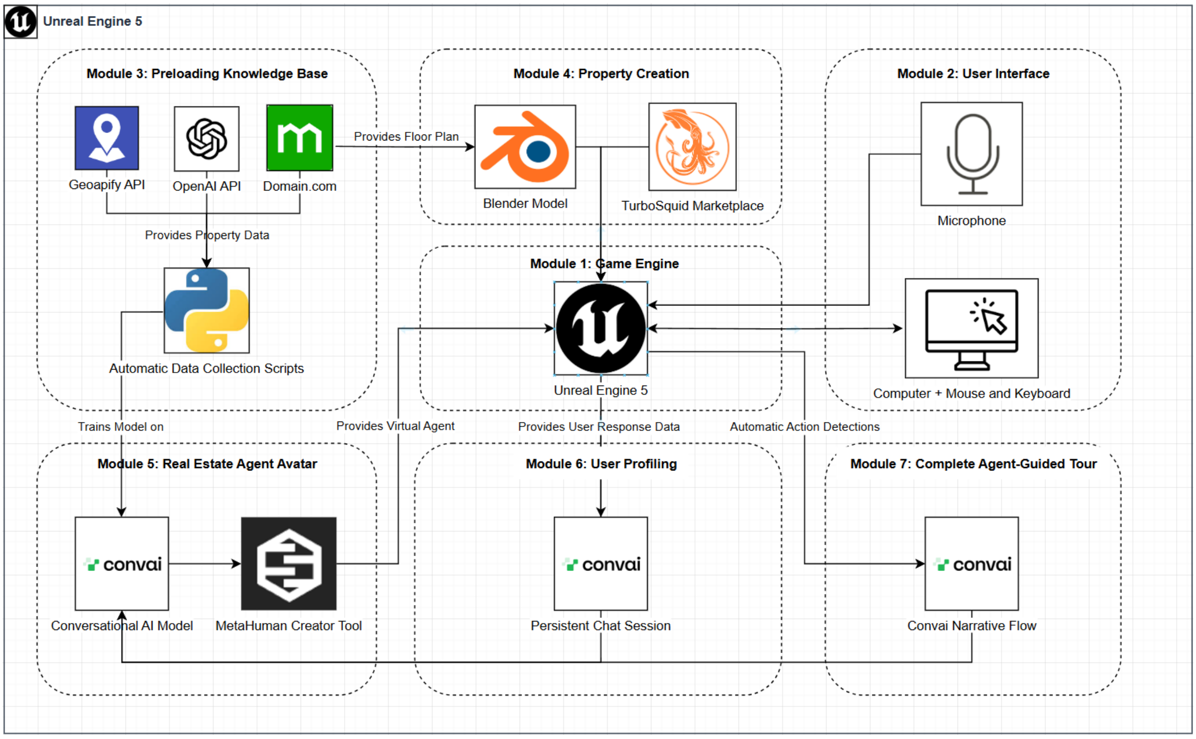

Figure 3.3 Software Architecture Diagram of the VIVA system

3.3.1 Module 1: Game Engine

Table 3.3.1 Comparison Table for Game Engines.

Features	**Unreal Engine 5**	**Unity**
Performance	High	Moderate
Graphics Fidelity	High	Moderate
Learning Curve	Moderate	Low
Customisability	High	Moderate
Community Support	High	Moderate

Table 3.2.2 was formed based on a comparative analysis of game engines conducted by Dr. Christopoulou et al. [10]. From the table, it's clear that UE5 is the best game engine for this project, as it offers better performance, graphics fidelity, customisability, and community support at the cost of a steeper learning curve, which was quickly resolved through upskilling and training. Additionally, UE5 provides essential frameworks such as the Metahuman Creator module, which provides high-fidelity, custom-made, human game models.

3.3.2 Module 2: User Interface

To use the VIVA system, the user must have a Microphone as well as a Mouse and Keyboard. The choice to remove the VR-compatible was made to reduce the workload of the thesis and lower the bar of entry for running the system. Despite this consideration, the immersion provided by a VR headset is far superior and should be added in future enhancements.

3.3.3 Module 3: Preloading Knowledge Base

The knowledge base of the real estate agent is collected through a single python script which collects data from the Geoapify Place API and the Realestate.com API. From these two sources, the system is able to provide information on the following types of data:

- **General property data**: **N**umber of bedrooms, bathrooms and key features as well as the floor plan of the property.
- **Room specific data**: Key features of each room within the property.
- **Construction-based data**: Structural information like building materials, foundation type, roof type and other mechanical systems like plumbing.
- **Neighbourhood data**: Places of interest, community values and crime rates.

3.3.4 Module 4: Property Creation

The virtual property was imported from a Blender model purchased from the TurboSquid marketplace. Due to time constraints, the custom reconstruction of a property would take far too long and so instead this sample model was used. The model was, however, trained from the real-world data of a property which was similar to the sample model and so that data was fairly accurate.

3.3.5 Module 5: Real Estate Agent Avatar

The real estate agent avatar was created through integrating the conversational Convai AI with a Metahuman UE5 model. This results in a high-fidelity virtual agent, capable of moving within the virtual environment seamlessly and answering all of the users' questions. The conversational model was trained from all data sources identified in **Module 3** and is capable of answering most questions about the property including the features of its rooms and the points of interest in its surrounding neighbourhood.

3.3.6 Module 6: User Profiling

User profiling was completed through the persistent chat feature of Convai AI, and required little additional changes as the existing system supported this functionality. This module allows the user to maintain a persistent chat session so that the identified qualities they value in a property are prioritised by the agent.

3.3.7 Module 7: Complete Agent-Led Tour

A completely agent-led tour was achieved using the Narrative Design workflow offered by Convai. By adding a decision tree and pre-defined path, agents are able to autonomously lead the user through the property without direct instructions.

3.4 Execution of the Project

The VIVA project was executed using an Agile approach, instead of the Waterfall method, as Agile allows for faster development and continuous verification through testing, due to its short iterative development windows, known as '**sprints**' [11]. Each of the sprints represents two central requirements and their respective user stories and acceptance criteria, as described in **Table 3.2.1**. The complete timeline for the completion of the project is shown below in **Figure 3.4**, in the form of a Gantt Chart.

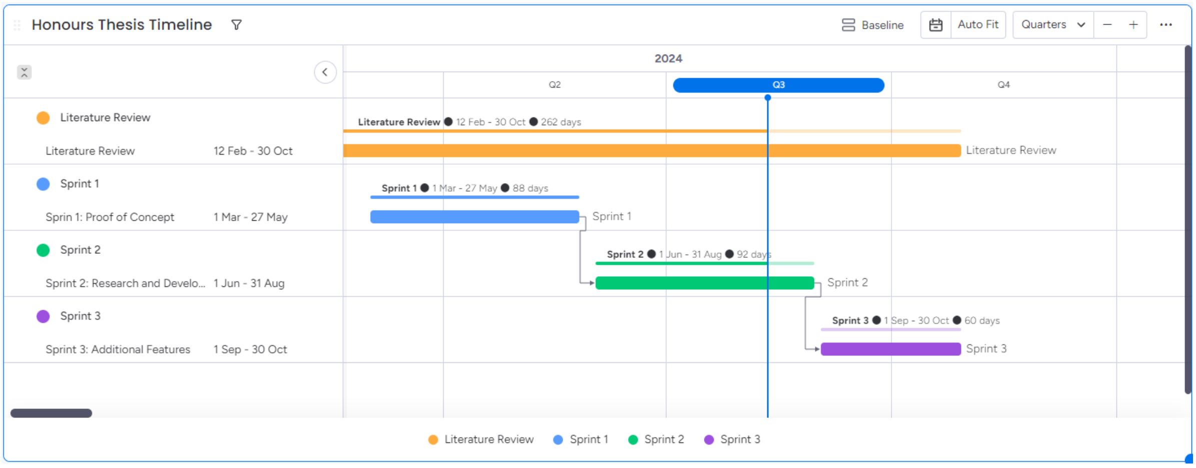

Figure 3.4 Honours Thesis Timeline Gantt Chart.

The following subsections (**3.4.1, 3.4.2**) will provide an in-depth and precise methodology on the execution of each milestone and their respective tasks.

3.4.1 Milestone 1: Property Exploration

Pre-tasks: General Setup and Software Downloads

1. Install the latest LTS version of Blender.
    a. This thesis uses Blender version 4.2.3 LTS.
2. Install the latest version of Unreal Engine 5.
    a. This thesis uses Unreal Engine version 5.4.4.
3. Inside of Unreal Engine 5, install the following Libraries:
    - AI for NPC, MetaHuman - Dialog, actions and general intelligence - by Convai
    - Quixel Bridge

Task 1: Import and Configure Property Model

1. **Source and prepare the 3D Model using Blender**
   - Using a 3D model marketplace such as TurboSquid or UE5 Marketplace, obtain a high-fidelity FBX property model.
     - To replicate the model used in this thesis, download the following model:
       1. https://www.turbosquid.com/3d-models/classic-style-porch-house-with-furniture-1637050
   - Open the FBX model in Blender.
     - Merge overlapping vertices:
       1. Enter Edit Mode by pressing the Tab key.
       2. Select all elements by pressing the A key.
       3. Open the merge options by pressing the M key.
       4. Select "**By Distance**" (**Figure A.1.a**).
       5. If merged correctly, 1221 vertices will be removed (**Figure A.1.b**).
2. **Export Model for Unreal Engine**
   - In Blender, select **File** > **Export** > **FBX (.fbx)**:
     - Set the export settings to the following:
       1. Include **Empty, Camera, Lamp, Armature, Mesh and Other**.
       2. Scale: **1.0**
       3. Forward Axis: **-Z**
       4. Up Axis: **Y**
       5. Leave everything else as the default value (**Figure A.1.c**).
3. **Import to Unreal Engine 5**
   - Open Unreal Engine 5.
     - Start a new project using the First-Person template
     - Select **File** > **New Level** > **Basic** > **Create.**
     - After the new level is created, remove the checkered mesh.
     - Open the **Content Drawer** and create a subfolder named **PropertyModel** inside of **Content.**
     - Import the exported FBX model from **Step 2** into the subfolder using the default import settings (**Figure A.1.d**).
     - Select the Static Mesh filter and select all meshes using CTRL + A.
     - Drag and drop the meshes into the UE5 environment.
     - You should now have the property model inside of the UE5 virtual environment (**Figure A.1.e**).

4. **Player Setup**
   - Inside your **Content Drawer**, select **Content** > **FirstPerson** > **BP_FirstPersonCharacter.**
   - Drag and drop the blueprint into the virtual property.
   - If you preview the first-person player, you will now be able to move around the property (**Figure A.1.f**).
5. **Collision Setup**
   - Select **Edit > Project** Settings.
   - Search for "**Default Shape Complexity**".
   - Set this value to "**Use Complex Collision As Simple**" (**Figure A.1.g**).
   - Show the collision meshes by using **ALT + C**.
   - You should notice that the collision now matches the meshes (**Figure A.1.h**).
6. **Test Property Model**
   - Run the simulation using **ALT + P.**
   - Using WASD walk throughout the property, ensuring all textures look correct and collision is working as expected.

Task 2: Create an Interactive Environment (Doors)

1. **Blueprint Actor for Doors**
    - Download a sound clip of an opening and closing door from platform like Pixabay.
        - This thesis uses the following sound clip: https://pixabay.com/sound-effects/open-and-closed-door-156814/.
    - Cut the sound clip into two audio files and import both sounds clips into your Content **Browser**.
    - Create a new Blueprint Actor named **BP_Door**.
        - Add a **Static Mesh Component** inside of **BP_Door** and set the Static Mesh option to be a door inside of your property model.
        - Select the Event Graph tab and add the combination of events shown in **Figure A.1.i**.
        - Open the **OpenDoorTimeline** component:
            1. Select **+ Track** and create a **Float Track**.
            2. Set the length of the track to be 0.7.
            3. Right click on the track and add two keys:
                1. Time = 0.0, Value = 0.0
                2. Time = 0.7, Value = 1.0
            4. This will manage the transition from open to close for the door meshes (**Figure A.1.j**).
2. **Implement Door Interaction Logic**
    - Open the **BP_FirstPersonCharacter** Blueprint:
        - Add the combination of events shown in **Figure A.1.k.**
        - If you preview the simulation, you will notice that when you press E, a line trace will be produced that will open and close a door.
3. **Implement Door Interaction Prompt Logic**
    - In your **Content Browser**, select Add > User Interface > Widget.
        - Inside of the widget add a **Canvas Panel** inside of the **Widget** and a **Text Box** inside of the Panel.
        - Select the **Text Box** and change the text to “Press 'E' to Interact”
        - Set the settings to the ones shown in **Figure A.1.M**.
    - Open the **BP_FirstPersonCharacter** Blueprint:
        - Add the combination of events shown in **Figure A.1.l.**
        - If you preview the simulation, you should notice that when you look at a door, the newly added **Interaction Prompt** will appear and when you look away it

will disappear.

4. **Adjusting Door Logic**
    - For every door in the property, add a new tag to inherited **BP_Door**.
        - For doors you want to rotate 90 degrees clockwise, name the tag **Door90**.
        - For doors you want to rotate 90 degrees anticlockwise, name the tag **Door**.
5. **Test Door Functionality**
    - Check that when you look at any door from a close distance, the interaction prompt appears.
    - Check that all doors can be opened using the player character.
    - Ensure no clipping occurs when the door is moving and that the collision is working as expected during the animation.

3.4.2 Milestone 2: Real-Time Virtual Agent

Task 1: Property Data Collection

1. **Data Collection Setup**
   - Clone the following repository: https://github.com/Kydars/Honours-Thesis-Data-Collection-Scripts.
   - Run the following command to install all requirements:
     - *pip install -r requirements.txt*
   - If the Python scripts fail to run on your system due to a permission issue run the following command in your terminal to give the scripts execution permissions.
     - chmod +x <FILE_NAME>
2. **property_data_collection.py**
   - This script is used to collects key property data using Selenium and Domain.com.
     - The script generates a file called **property_data.txt** containing this data.
   - To alter the property address, search for a different property on Domain.com and change the URL in the script with the URL of that new property.
3. **room_data_collection.py**
   - This script is a semi-automated data collection script used to generate descriptions for rooms within the property.
     - The script generates a file called **room_data.txt** containing this data.
   - Create an API key for the **OpenAI** API.
     - Instructions: https://platform.openai.com/docs/api-reference/authentication
   - Replace the value of **openai_key** with your newly created API key.
   - Open the URL present in **property_data_collection.py** script and enter the URL of an image of the property on Domain.com.
   - Enter the type of room or space that image is showing.
   - Continue until you have queried for all necessary rooms.
4. **neighbourhood_data_collection.py**
   - This script is used to collect a list of points of interest within 500 metres of the target location.
     - The script generates a file containing this data called **neighbourhood_data.txt**.
   - Create an API key on the **Geoapify** platform.
     - Instructions: https://apidocs.geoapify.com/docs/places/
   - Replace the value of **geoapify_key** with your newly created API key.
   - To alter the property address, open the URL in your web browser and enter your

address, then once the new URL is loaded, copy it into the script.

Task 2: Integrate the Real-Time Virtual Agent

1. **Convai API Integration**
    - Go to https://www.convai.com/.
    - Sign up and select "**Create a new character**".
        - This thesis uses the settings shown in **Table A.2.a** for its agent **John.**
    - Inside of your UE5 project:
        - Select **Edit** > **Plugins.**
        - Search for **"Convai"** and enable the plugin.
        - After reloading your project, select **Edit** > **Project Settings.**
        - Search for **"Convai"** and input your Convai API key.
2. **Metahuman API Integration**
    - Select **Window** > **Quixel Bridge**.
    - Click on the **MetaHumans** menu and then select one of the presets:
        - This thesis uses the MetaHuman character **Myles**.
    - After the character has been added to your **Content Browser**, drag and drop the Blueprint into your UE5 environment.
3. **Integrating Metahuman Character with Convai Conversational AI**
    - Inside of the Metahuman Character Blueprint Components:
        - Set the **Body** > **Anim Class** to be **Convai_MetaHuman_BodyAnim**.
        - Set the **Body** > **Face** > **Anim Class** to be **Convai_MetaHuman_FaceAnim**.
        - Select **Class Settings** and change the **Parent Class** to be **ConvaiBaseCharacter**.
        - Now go back to the UE5 viewer and select the character inside the environment and change the value of its **Char ID** to be the same as the Convai character you created initially.
    - Inside of the FirstPerson Character Blueprint Components:
        - Select **Class Settings** and change the **Parent Class** to be **ConvaiPlayerCharacter**.

4. **Integrating Real-Time Virtual Agent with Virtual Environment**
    - Using the collected room data file:
        - Create a new instance of a **Plane** at key locations within each room you want your agent to visit.
            1. For example, if you wanted the agent to path to the kitchen, place an invisible **Plane** inside of the **Kitchen** room (**Figure A.2.b**).
        - Next, add an object reference to the **Plane** and make sure you add a name and description, which matches the key used inside of the room_data.txt file.
            1. For example, if the **Plane** is located inside of the **Kitchen**, and the key within the room_data.txt file is 'kitchen', make sure you add that the 'kitchen' as the name and description inside of the object reference (**Figure A.2.c**).
5. **Adding a Narrative Design workflow to your Virtual Agent**
    - Inside of the Convai character creator, select the **Narrative Design** tab.
    - Create a unique section for the **Introduction**, **Agent-Led** and **Self-Led** scenarios.
    - Then create a new section for each location within your property.
    - Connect the introduction to the Agent-Led and Self-Led section based on whether the user wants to lead the tour or be led (**Figure A.2.d**).
    - Now using a similar style connect all of the room sections based on the order you want the agent to visit them **(Figure A.2.e)**.
6. **Test the Real-Time Virtual Agent Functionality**
    - Check that you are able to communicate with the virtual agent using both voice and text commands.
    - Check that the virtual agent will follow you after you send the following message:
        - **"Please follow me"**
    - Check that the virtual agent will move to the living room after you send the following message:
        - **"Please move to the living room"**
    - Complete a full run through of an agent-led tour and ensure that all generated responses are accurate based on the training data.
    - Check that during movement, the virtual agent does not collide with objects.

Chapter 4

Evaluation

This chapter evaluates the VIVA system through a comparative analysis with existing technologies (**4.1**) and the identification of key limitations (**4.2**).

4.1 Comparative Analysis

To determine the value of the VIVA project, we will compare it against existing platforms like Matterport (**4.1.1**), Zillow 3D Home Tours (**4.1.2**) and Redfin's AI-Powered Recommendations (**4.1.3**), which are all industry leaders in virtual property tours and recommenders. Each of these systems represent a unique approach to enhancing the real estate market and analysing them provides valuable context for understanding VIVA's strengths and weaknesses.

4.1.1 Matterport

Matterport is a leading platform for virtual tours, known for its automated 3D property scanning technology [3]. As previously discussed in **Section 2.2.2**, Matterport greatly improved on its predecessors through the utilisation of advanced hardware like the Pro2 Camera and the integration of photogrammetry and LiDAR to create highly detailed 3D property models [3]. As such, Matterport's primary strength lies in its scalability, allowing agents to quickly scan properties and generate accurate models. In contrast, VIVA relies on pre-constructed virtual property models, limiting its scalability and increasing its relative setup time. VIVA could benefit from adopting similar automated spatial mapping technologies to address this limitation.

In spite of this, VIVA's conversational AI offers a more interactive and easier to use workflow than the low-level item descriptions of Matterport. A primary gap identified when reviewing Matterport, was the inherent lack of engagement [6] due to these descriptions, which has been addressed through VIVA's more interactive virtual environment and more engaging real-time virtual agent.

4.1.2 Zillow 3D Home Tours

Zillow's solution emphasizes accessibility by enabling property scanning through a smartphone app, removing the need for expensive hardware [12]. Integrated directly with Zillow's real estate listings platform, it allows agents to generate 3D static tours, showcasing that scalability, like Matterport is one of its key strengths. In comparison, VIVA is again limited by its use of preconstructed models and as such could greatly benefit from adopting a similar smartphone-based scanning application.

However, Zillow's solution suffers from the same lack of engagement created from a static application, meaning that VIVA's capacity for movement and AI-driven tours provides a less scalable but far more novel and useful solution.

4.1.3 Redfin's AI-Powered Recommendations

Redfin utilizes AI to enhance the process of searching for property by analysing user preferences and providing tailored property recommendations [13]. Its strength lies in its ability to leverage large datasets to predict user needs and streamline the property discovery [13]. Due to the efficacy, in the future, VIVA should implement a similar workflow prior to exploration, allowing the agent to showcase multiple properties and identify what the user's key property requirements.

4.2 Limitations

The implementation of the VIVA system revealed several key limitations during its execution as well as its comparison with more established systems. These limitations provide a framework for understanding the current constraints of VIVA whilst also offering pathways for future enhancements.

4.2.1 Virtual Property Recreation

The process of creating virtual representations of real-world properties posed a significant challenge, particularly when compared to the automated scalability offered by Matterport and Zillow’s systems. VIVA's reliance on pre-constructed 3D models introduces two major constraints:

- **Time-Consuming Workflow:** Preparing 3D property models involves extensive cleaning, optimisation, and configuration. This process cannot scale effectively for a larger number of properties, bottlenecking its ability to scale in the future.
- **Dependence on External Resources:** The current approach relies heavily on pre-existing 3D models available from third-party platforms such as TurboSquid. While these models provide a visual approximation, they require a great deal of modification before they are able to accurately reflect real-world properties.

Proposed Solution: The use of 3D spatial mapping cameras or photogrammetry techniques could automate property recreation, creating realistic digital twins directly from real-world scans, similar to Matterport’s automated pipeline. These technologies were identified early on into this thesis, but they were excluded due to time and budget constraints.

4.2.2 Absence of User Testing

While VIVA's AI-powered tours and conversational interactions address the static nature of Zillow and Matterport tours, the absence of formal user testing limits the system's ability to adapt to real-world needs. Without iterative feedback from diverse user groups, the following challenges arise:

- **Limited Usability Insights**: VIVA's current evaluation relies on heuristic and cognitive walkthroughs, which provide valuable insights but fail to replicate the diversity of user behaviours encountered in real-world scenarios.
- **Missed Edge Cases**: Without broader testing, it is difficult to identify and resolve edge cases that could impact accessibility or functionality.

Proposed Solution: Conduct iterative usability tests with a diverse group of potential users so that we can identify pain points, improve accessibility, and ensure the system meets user expectations.

Chapter 5

Conclusion

This thesis has explored the limitations of existing virtual real estate tours and identified a significant gap in the current market, focusing on the lack of engagement and the need for a more interactive and user-driven experience. The development of the VIVA system addresses this gap by incorporating a conversational AI agent that guides users through virtual property tours, creating a more dynamic and engaging experience compared to traditional static tours. Through a comparative analysis with established platforms like Matterport, Zillow, and Redfin, this thesis has highlighted both the strengths and limitations of VIVA, providing valuable insights for future development.

5.1 Future Work

Following on from this project, there are numerous possibilities and directions to enhance the VIVA system. The most salient of these include:

- **Automated 3D Property Scanning.** Implement LiDAR and photogrammetry to automate 3D model creation, streamlining virtual property creation or explore more affordable method's like to Zillow smart phone application for wider accessibility.
- **User Testing and Feedback:** Conduct formal usability tests, gather user insights, and refine the system to improve accessibility, usability, and real-world robustness.
- **AI-Powered Property Recommendations:** Develop an AI-driven recommendation engine that analyses user preferences and refines suggestions based on location, budget, and user-specific needs, similar to system created by Redfin.

Bibliography

[1] Naeem, N., Rana, I.A. and Nasir, A.R. (2023) Digital real estate: a review of the technologies and tools transforming the industry and society. Smart Constr. Sustain. Cities 1, 15. https://doi.org/10.1007/s44268-023-00016-0

[2] CB-Insights. (2020) How Real Estate Tech Investment Is Trending. Available online: https://www.cbinsights.com/research/real-estate-tech-funding-trends/ (accessed on 20 March 2024).

[3] Mohamad Zaidi Sulaiman, Mohd Nasiruddin Abdul Aziz, Mohd Haidar Abu Bakar, Nur Akma Halili, Muhammad Asri Azuddin (2020) Matterport: Virtual Tour as A New Marketing Approach in Real Estate Business During Pandemic COVID-19. Available at https://doi.org/10.2991/assehr.k.201202.079

[4] Feldberga, E. and Dick, E. (2022) Attracting foreign direct investments through augmented and virtual reality, RSS. Available at: https://itif.org/publications/2021/08/09/attracting-foreign-direct-investments-through-augmented-and-virtual-reality/ (Accessed: 1 April 2024).

[5] Devine, A. (2021) Long open home queues a warning for buyers ahead of summer auctions, Search for Real Estate, Property & Homes. Available at: https://www.realestate.com.au/news/long-open-home-queues-a-warning-for-buyers-ahead-of-summer-auctions/ (Accessed: 01 April 2024).

[6] Zhang, Mengxia and Troncoso, Isamar (2023) Beyond the Hype: Unveiling the Marginal Benefits of 3D Virtual Tours in Real Estate. Available at SSRN: https://ssrn.com/abstract=4517728 or http://dx.doi.org/10.2139/ssrn.4517728

[7] B. A. Deaky and L. Parv (2017) Virtual Reality for Real Estate: Its evolution in Bluemind Software, 2017 4th Experiment@International Conference (exp.at'17), Faro, Portugal, 2017, pp. 83-86, doi: 10.1109/EXPAT.2017.7984408.

[8] Alsawan, N.M.; Alshurideh, M.T. The Application of Artificial Intelligence in Real Estate Valuation: A Systematic Review. In International Conference on Advanced Intelligent Systems and Informatics 2023; Springer: Cham, Switzerland, 2023; pp. 133–149.

[9] Miljkovic, Ivana, Olena Shlyakhetko, and Solomiia Fedushko (2023) Real Estate App Development Based on AI/VR Technologies, Electronics 12, no. 3: 707. https://doi.org/10.3390/electronics12030707

[10] Christopoulou, E. and Xingogalos, S. (2017) Overview and Comparative Analysis of Game Engines for Desktop and Mobile Devices, International Journal of Serious Games, 4(4). Available at: https://doi.org/10.17083/ijsg.v4i4.194

[11] Mishra, A., Alzoubi, Y.I. (2023) Structured software development versus agile software development: a comparative analysis. Int J Syst Assur Eng Manag 14, 1504–1522. https://doi.org/10.1007/s13198-023-01958-5

[12] Hutchcroft, W. (2022). How the Zillow Indoor Dataset Facilitates Better 3D tours and Advances the Science of Indoor Spaces. [online] Zillow. Available at: https://www.zillow.com/tech/zillow-indoor-dataset-facilitates-better-3d-tours/ [Accessed 10 Nov. 2024].

[13] Osgood, E. (2023). Introducing Redfin Redesign, an AI-Powered Tool to Redesign Listing Photos. [online] Redfin Real Estate News. Available at: https://www.redfin.com/news/introducing-redfin-redesign-ai-tool/ [Accessed 10 Nov. 2024].

Appendix A

A.1 Milestone 1: Property Exploration Figures

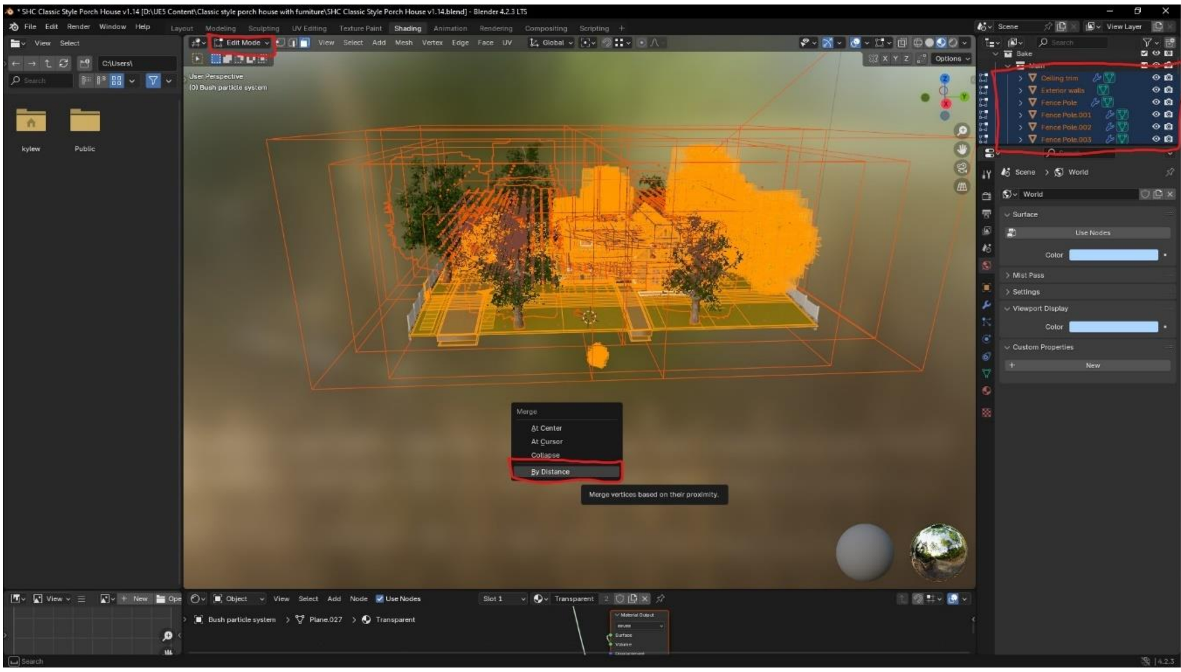

Figure A.1.a Merging overlapping vertices in Blender.

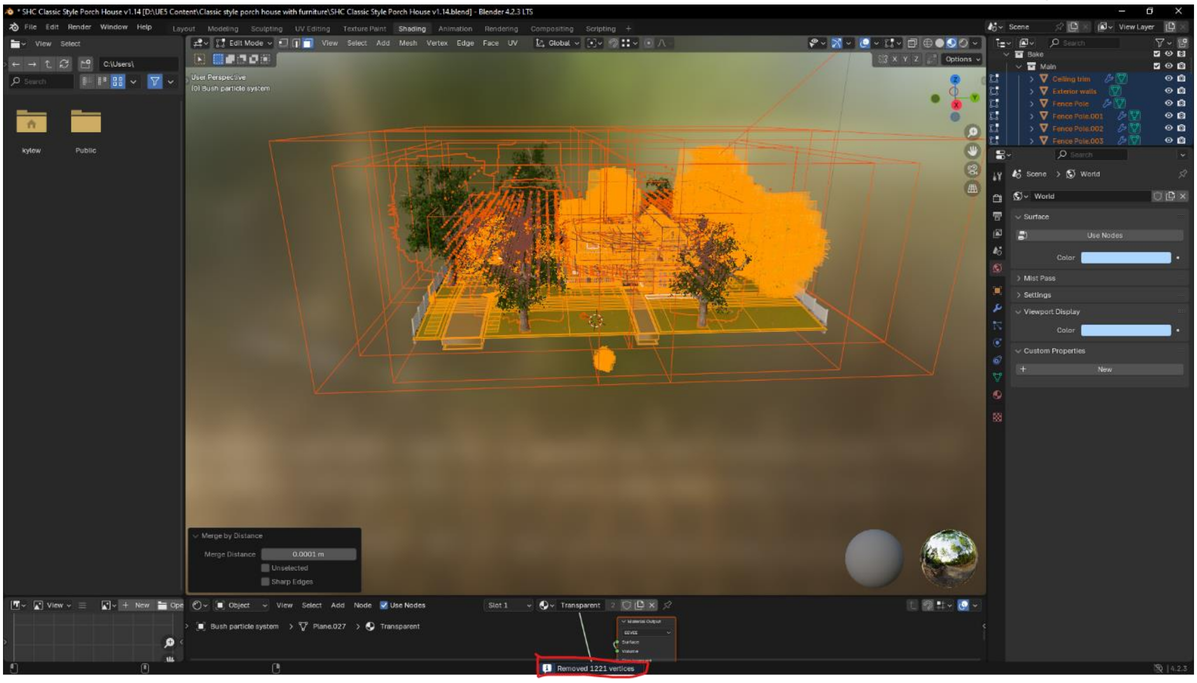

Figure A.1.b Result of merging overlapping vertices in Blender.

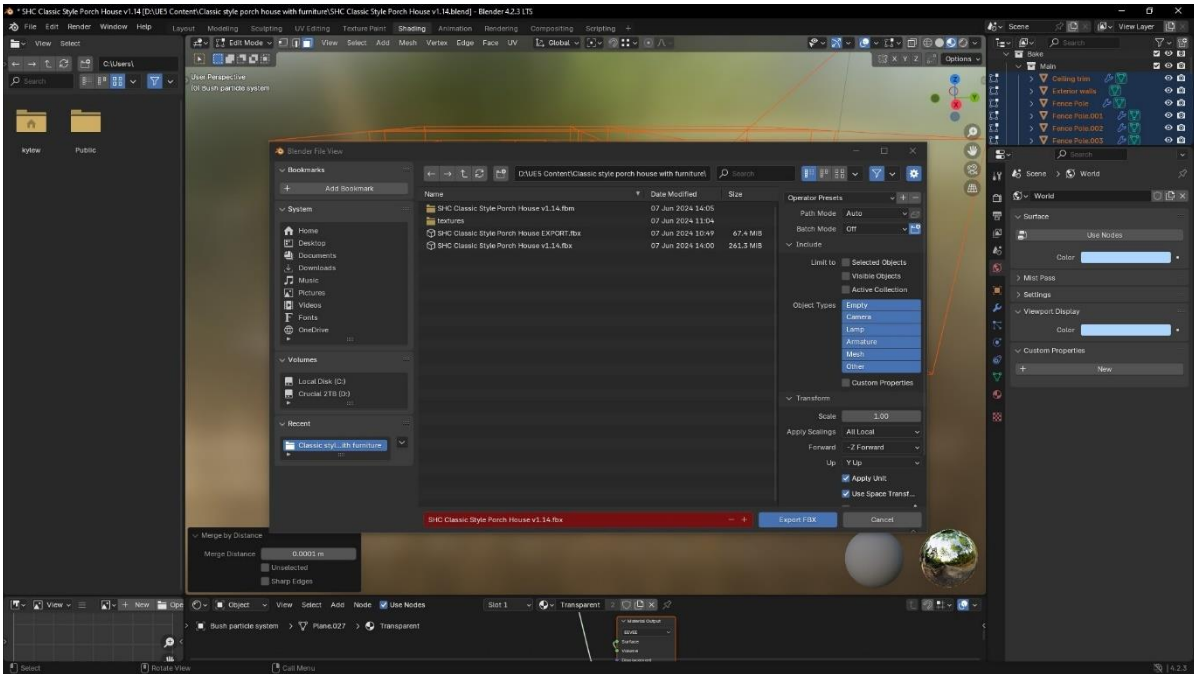

Figure A.1.c Export settings in Blender.

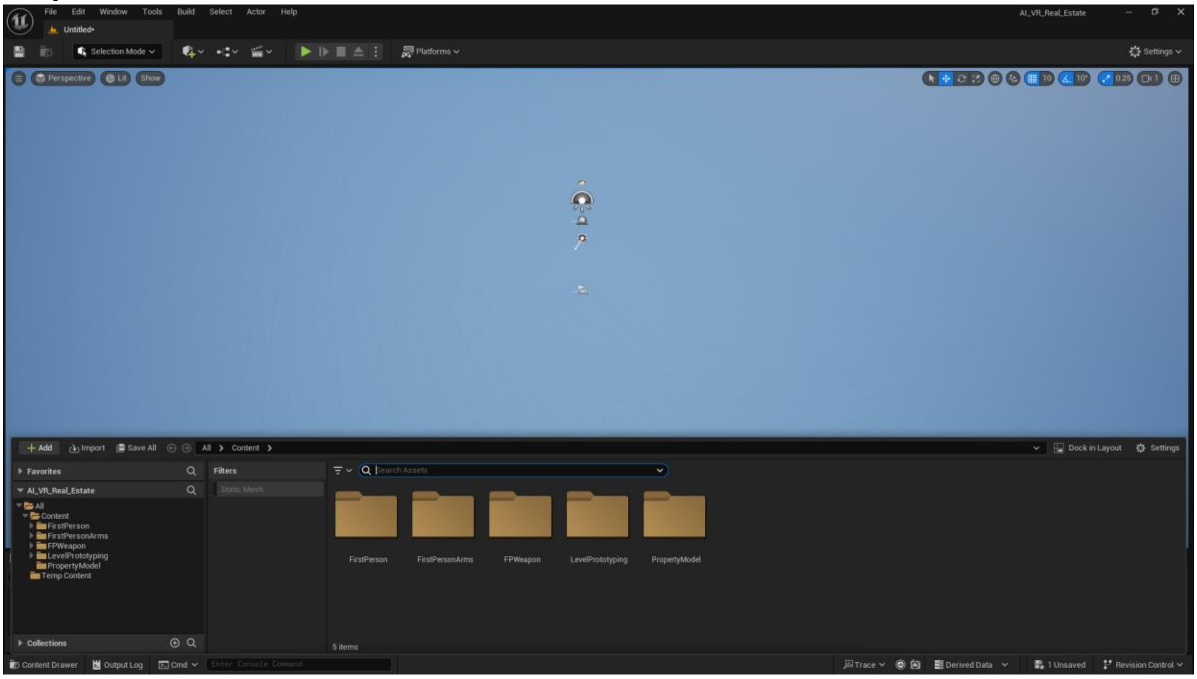

Figure A.1.d New level created from Basic template.

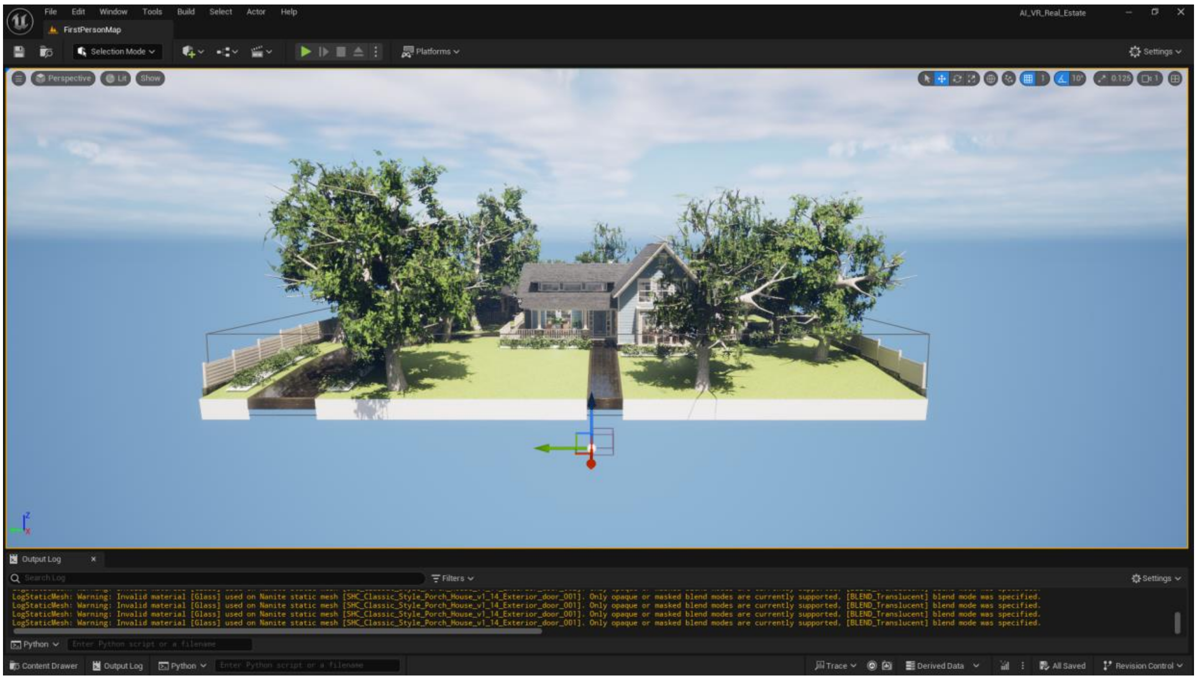

Figure A.1.e Imported property model inside of UE5 viewer.

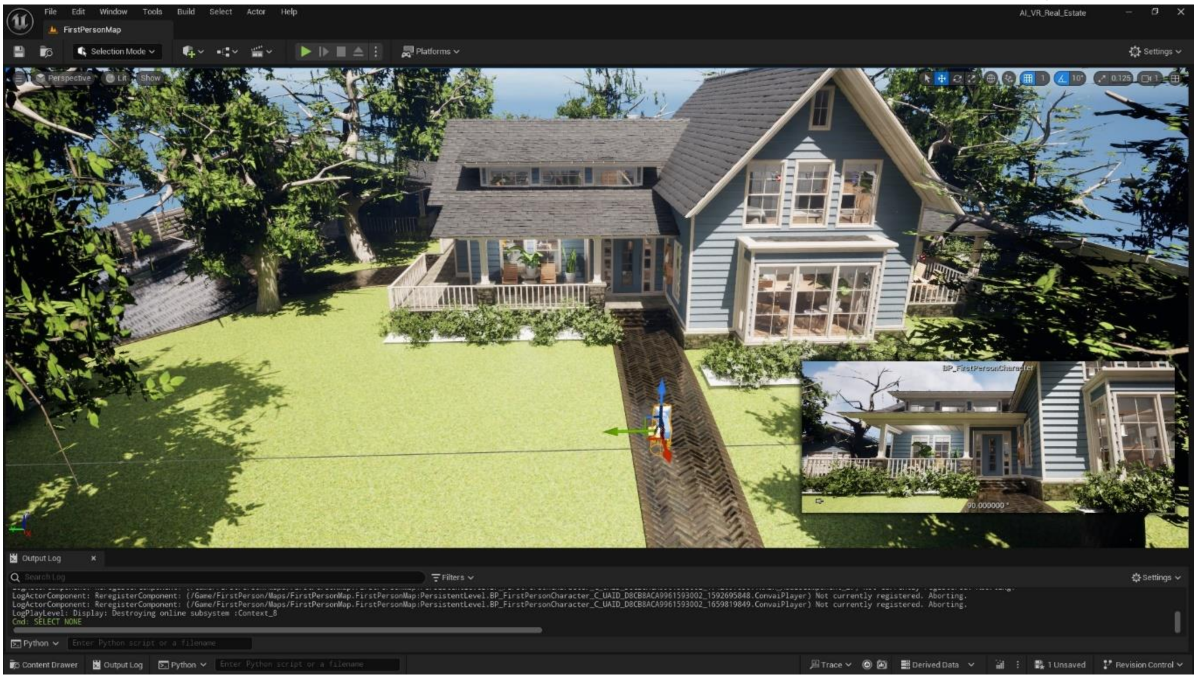

Figure A.1.f Player blueprint inside of UE5 viewer.

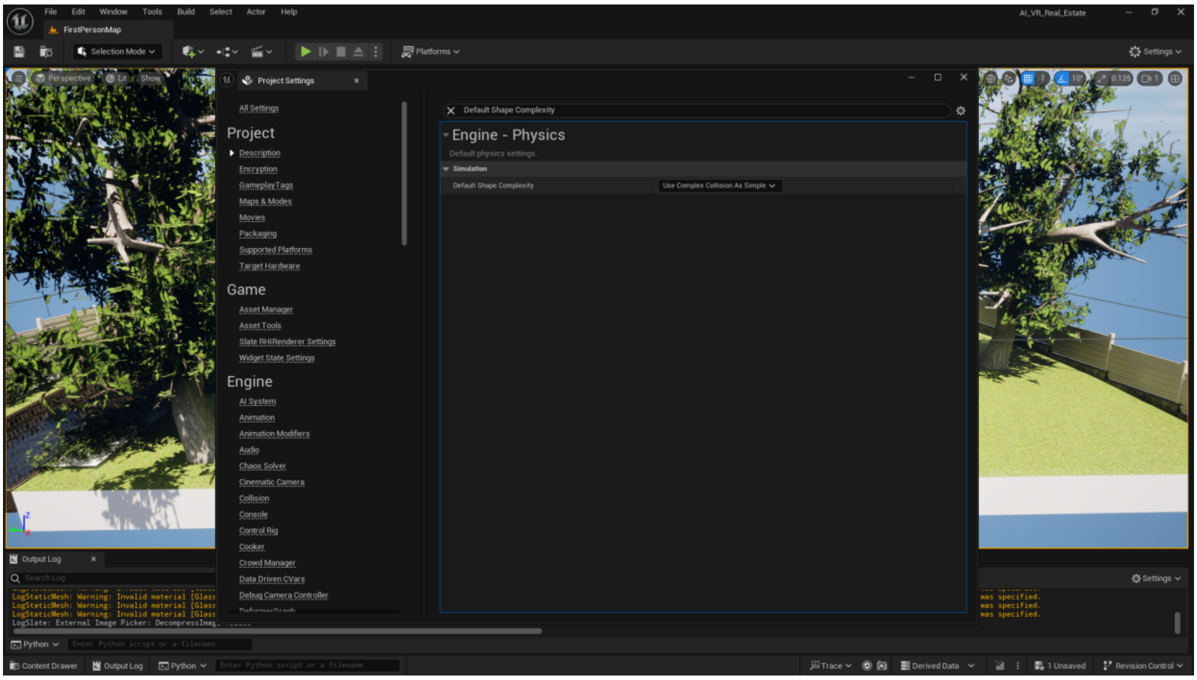

Figure A.1.g Default collision settings

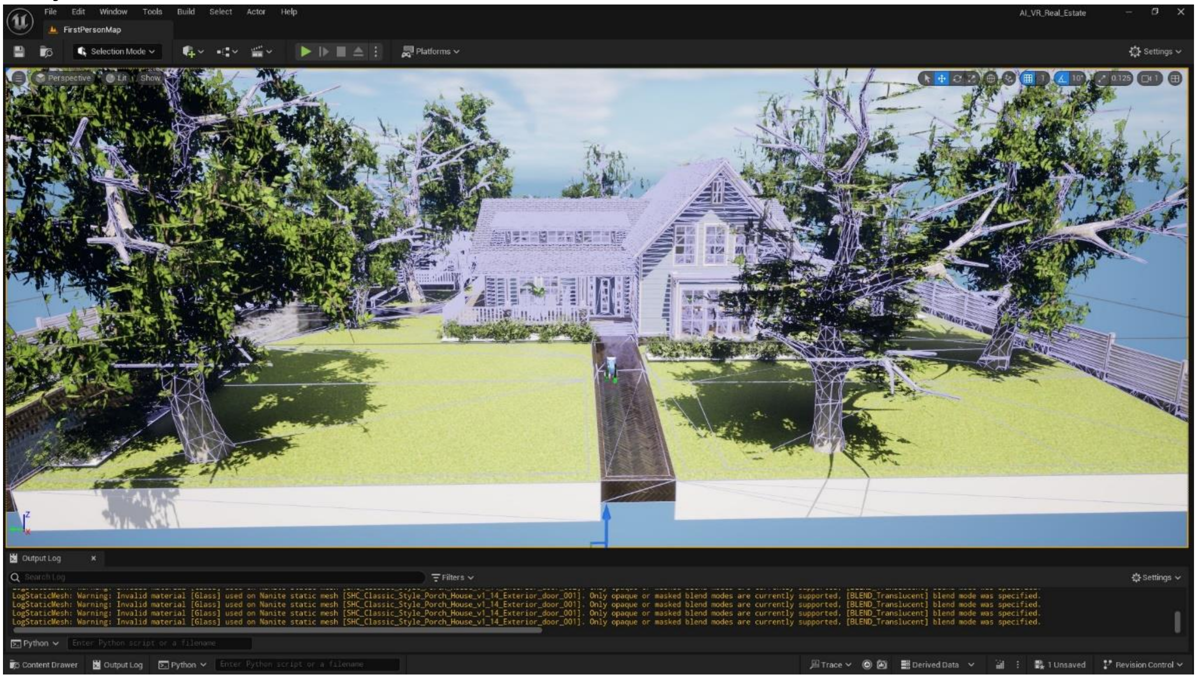

Figure A.1.h Result of changing default collision settings.

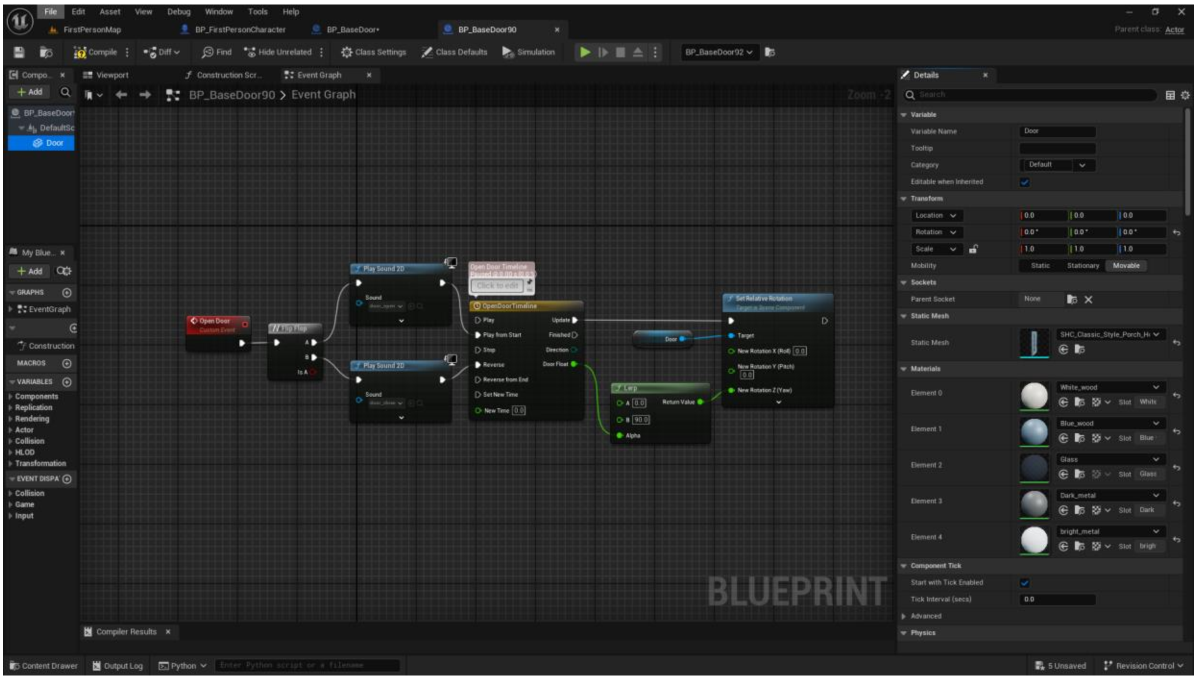

Figure A.1.i Opening a door event graph.

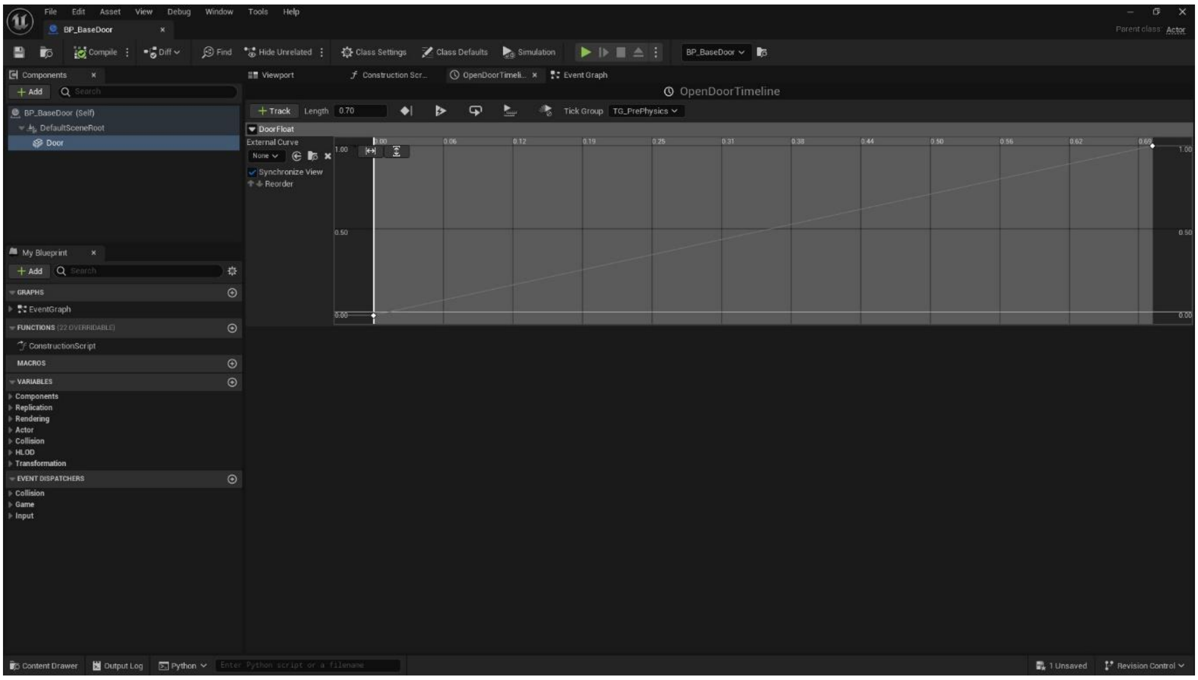

Figure A.1.j OpenDoorTimeline event track.

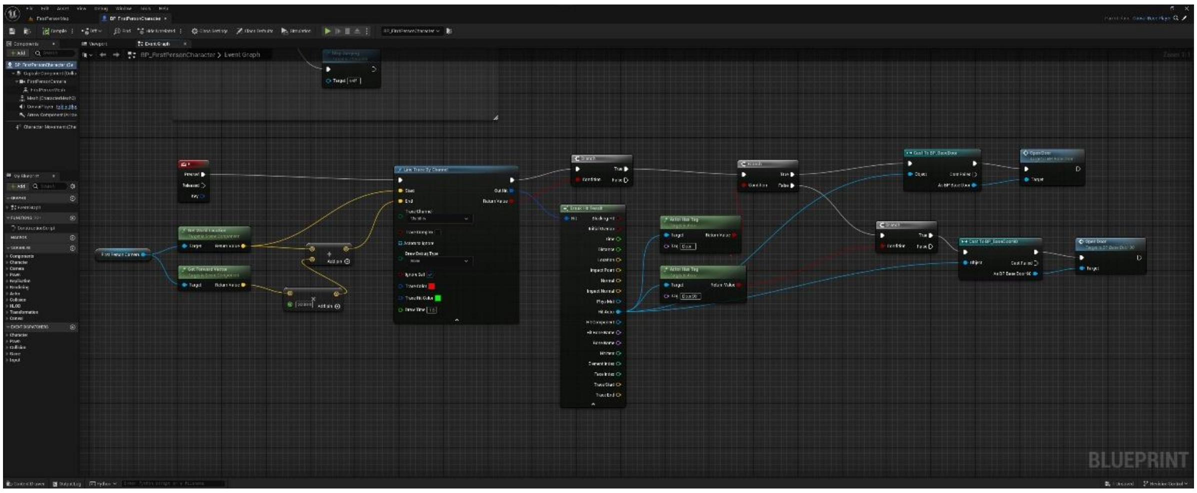

Figure A.1.k Player interaction with door event graph.

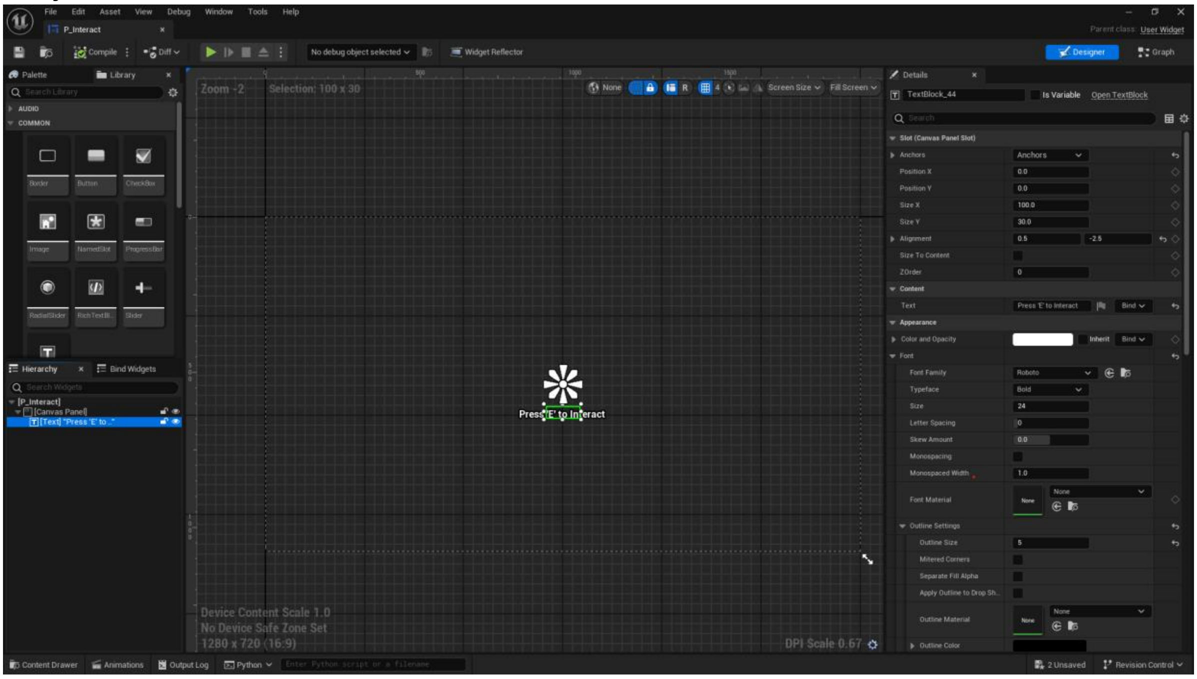

Figure A.1.m Player interact prompt viewport widget.

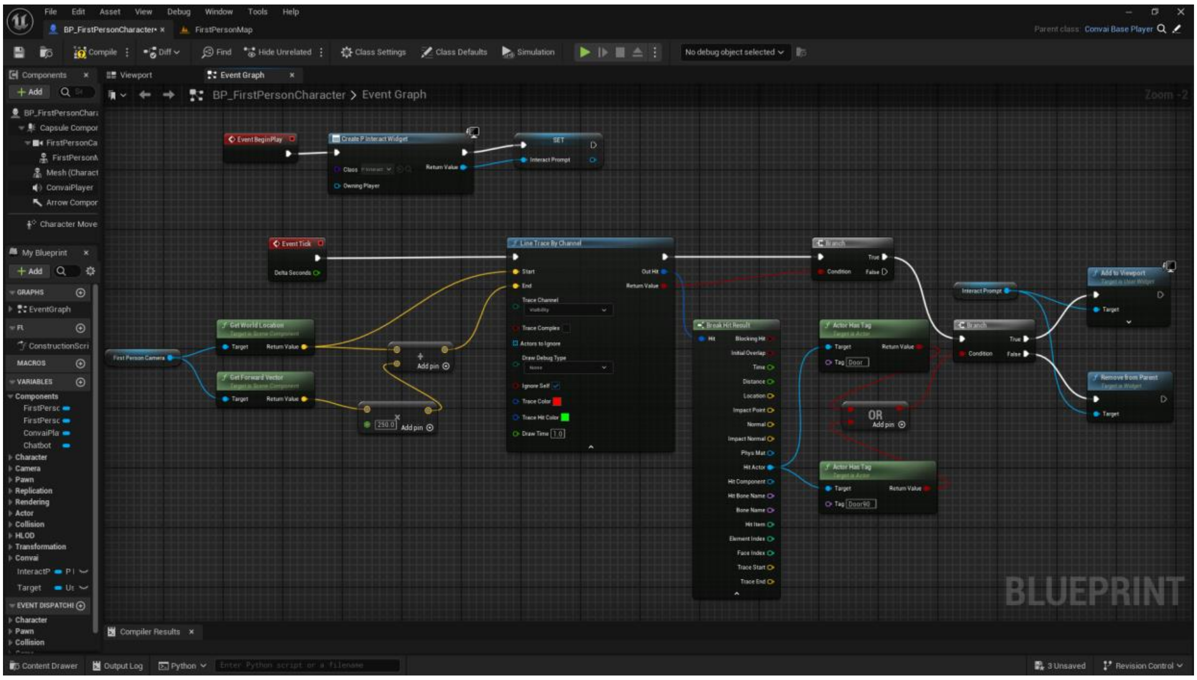

Figure A.1.m Player viewing door adds interact prompt door event graph.

A.2 Milestone 2: Real-Time Virtual Agent Figure

Table A.2.a Settings and values for the creation of the Convai conversational agent.

Setting	**Value**
Character's Name	John
Character's Voice	American Middle-aged Determined Male
Character's Backstory	You are John, a virtual reality real estate agent who is an expert in real estate. With a sleek avatar and dynamic AI, you guide users through property exploration with empathy and knowledge. Trained extensively on the knowledge bank, you understand layouts, market trends, and aim to be a trusted companion when searching for the perfect home. Driven by a desire to have a meaningful impact, you approach each interaction with professionalism and virtual charm.
Knowledge Bank	property_data.txt, room_data.txt and neighbourhood_data.txt files generated from Milestone 2 Task 1.

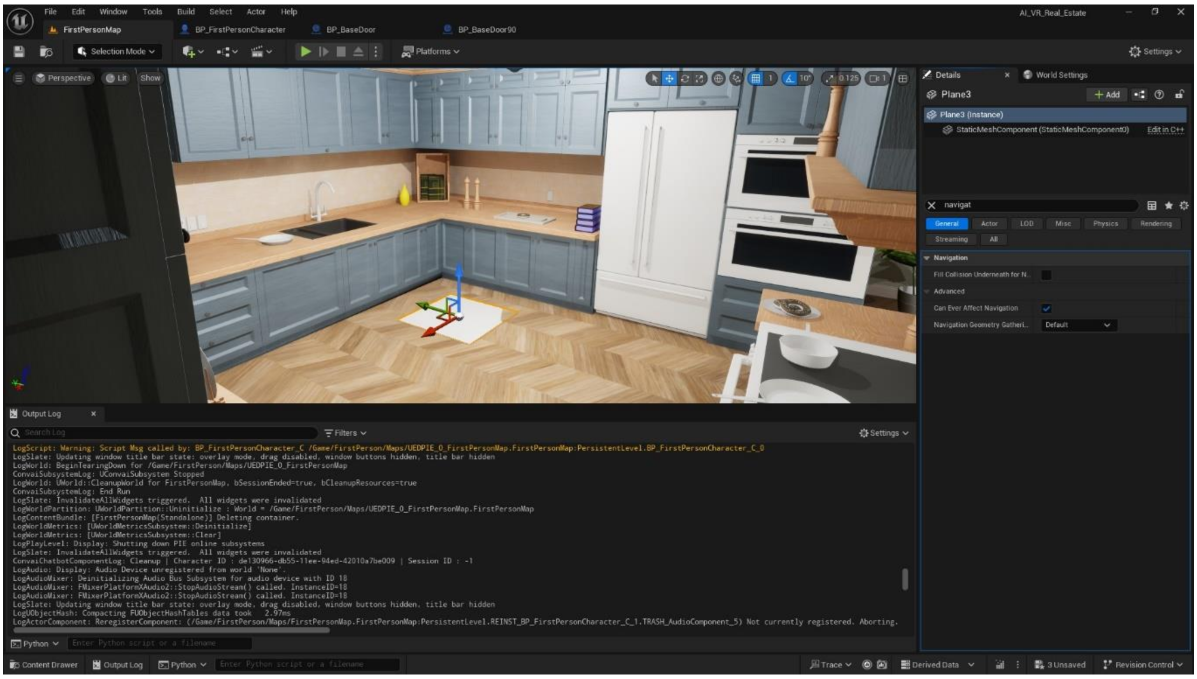

Figure A.2.b Invisible Plane object inside of the kitchen.

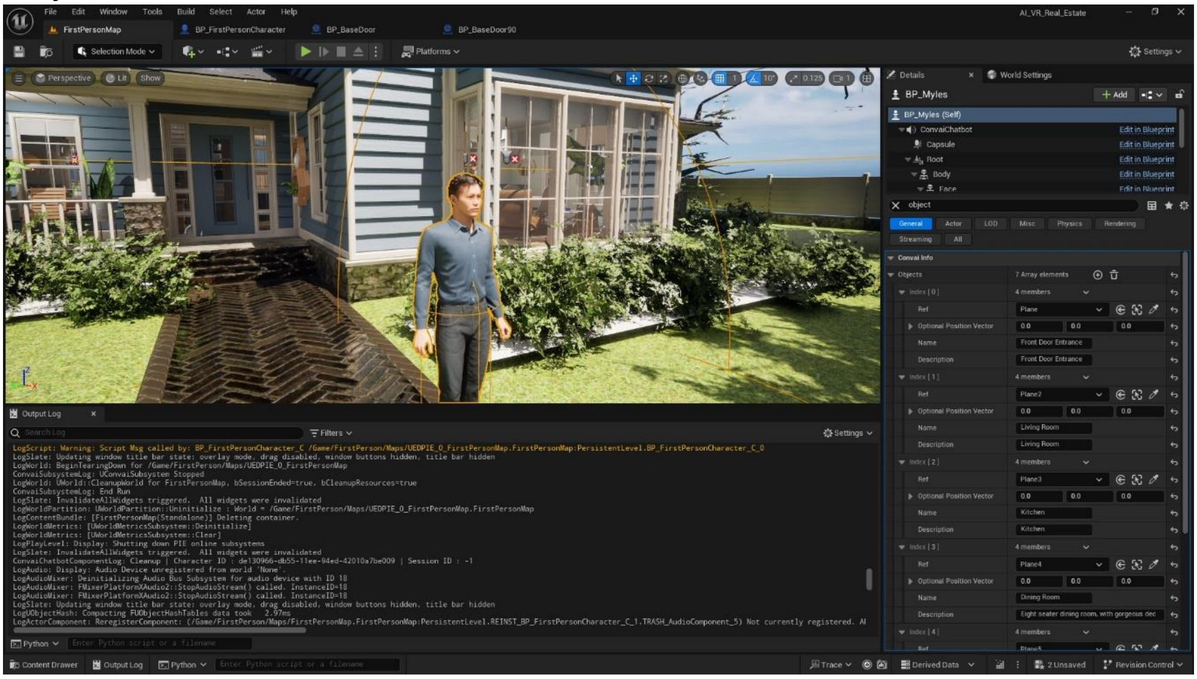

Figure A.2.c Real-Time virtual agent object references to rooms.

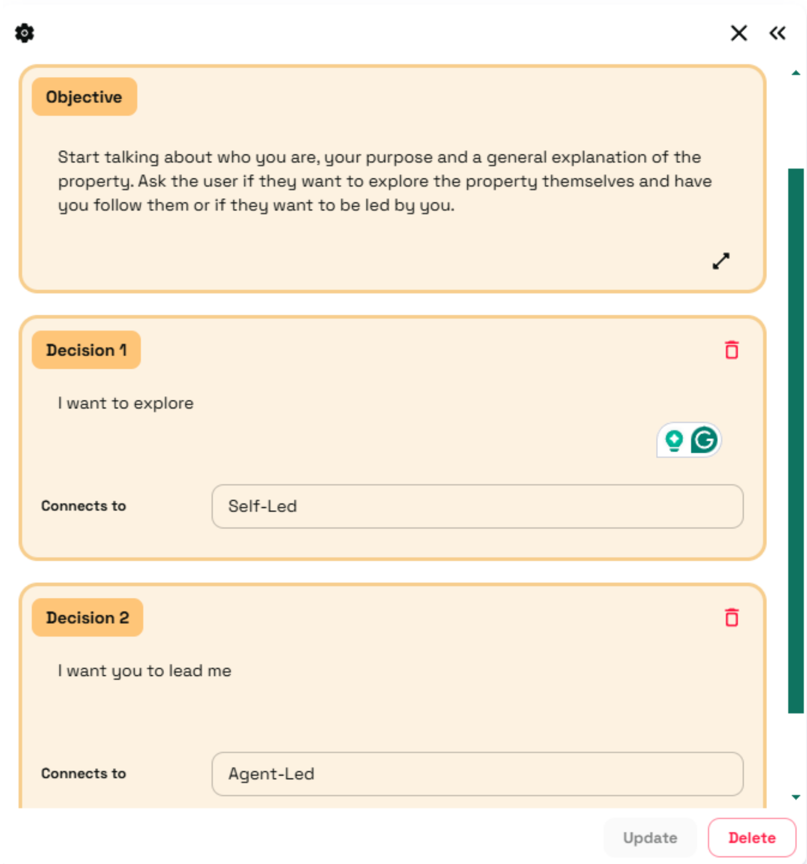

Table A.2.d Example objective and decisions for the Introduction section.

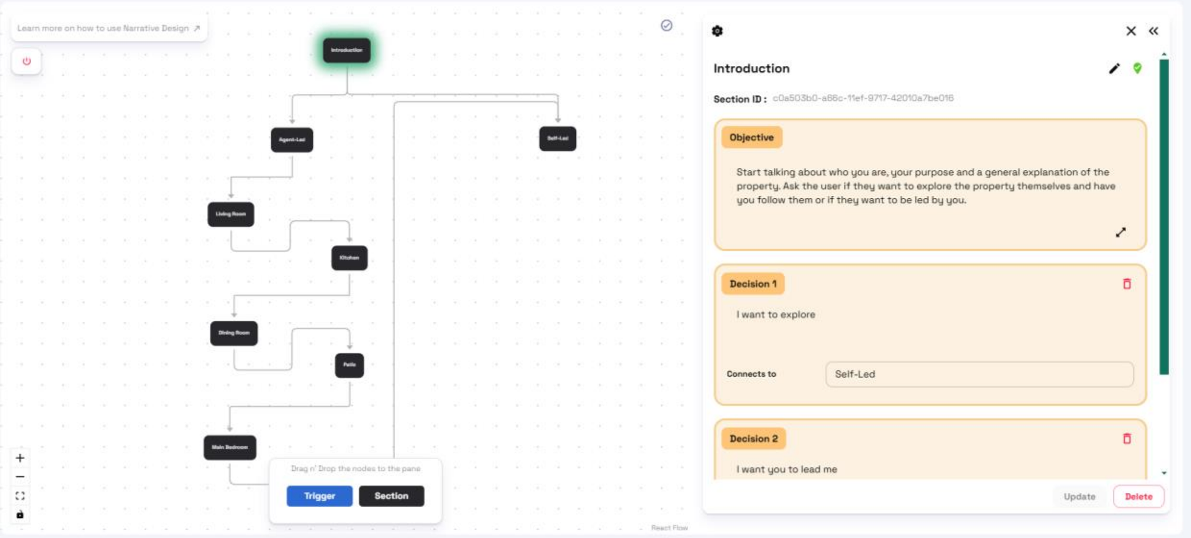

Table A.2.e Completed Narrative Design Flowchart.