

MetaHouse

Real-estate application that enables prospective buyers to virtually tour, customise and buy a house in real-time

Group 24

cjh162@student.bham.ac.uk 2293161
jxs1535@student.bham.ac.uk 2268599
lxc141@student.bham.ac.uk 2198934
vxb284@student.bham.ac.uk 2364583
axs1949@student.bham.ac.uk 2310480
sxb1669@student.bham.ac.uk 2348094
hxp155@student.bham.ac.uk 2309553
mxa1311@student.bham.ac.uk 2201621

Introduction	3
Scope	3
Assumptions	4
Functional Requirements	4
REQ1: User Authentication	4
REQ2: House selection	5
REQ3: Buying a house	5
REQ4: Virtual Tour	5
REQ5: Properties of a house	6
REQ6: Customising a house	6
Non - Functional Requirements	7
NFREQ1: Reliability	7
NFREQ2: Efficiency	7
NFREQ3: Usability	8
NFREQ4: Security	8
NFREQ5: Legislative	8
NFREQ6: Ethical	8
NFREQ7: Implementation	8
Use Case Diagram	9
Use Cases	10
Join Virtual Tour	10
Scenario 1	11
Scenario 2	12
Process of Buying A House	12
Scenario 1	13
Scenario 2	14
Viewing House Properties	14
Scenario 1	15
Scenario 2	15
Customising a House	16
Scenario 1	17
Scenario 2	17
Activity Diagram	18
Class Analysis	19
Noun/Verb Analysis:	19
CRC Cards:	20
First cut class diagram:	22
Detailed class diagram:	23
Object Diagram	24

Object Diagram for Viewing a House	24
Sequence Diagrams	25
Sequence diagram for the process of buying a house	25
Sequence Diagram for Viewing Details	26
State Diagrams	27
State diagram for virtual tour	27
State diagram for Customising a house	27
Architecture Style, Modelling and Evaluation	28
Architecture Style 1:	28
Component Diagram	28
Deployment Diagram	29
Architecture Style 2:	30
Component Diagram	30
Deployment Diagram	30
Architecture Comparison	32
Testing	32
Introduction	32
Assumptions	33
Test Items	33
Features to be Tested	33
Features not to be Tested	34
Strategy	34
Pass/Fail Criteria	34
Exit Criteria	37
Usability and Prototyping	37
Ethics & Professional Practice	40
ACM1: General Ethical Principles	40
ACM2: Professional Responsibilities	41
ACM3: Professional Leadership Principles	42
ACM4: Compliance With Code	42

Introduction

We aim to create the architecture for a virtual viewing system hosted by a real-estate company. It will allow a user to remotely tour a selected house and navigate around it within the Metaverse. This new application renovates the old system of real-estate by allowing easier access to view real-estate, giving it a broader audience and giving buyers a better opportunity to find their perfect home.

It will present the user with the option of selecting a house based on its address or from a list of previously favorited houses. During a tour of a house, there will be functionality for creating a custom design for the interior of the house. This includes changing the surfaces, furniture and adding accessories. As well as this, key information about the property will be available to the user, such as bedrooms and bathrooms as well as security measures like CCTV cameras and intercoms.

The proposed system, named MetaHouse, will also offer a way for users to purchase a house using a credit card, which will be a large abstraction of the actual process of buying a property. However, this will still allow for remote purchases and in future this project could be improved by properly implementing this.

Scope

Our system scope includes the ability to view a virtual representation of a house in the form of a 3D render. It is assumed that this 3D render already exists and is provided by the real-estate company that operates this system. To virtually view this render the user must have a VR headset connected in the appropriate way to allow access to the Metaverse.

To allow for useful implementation of the house customisation feature, the 3D render should have no furniture included. As well as this, every wall, ceiling and floor should already be defined as an object so that its features can be changed.

We will also assume that the user already has a collection of their own furniture to be used. The functionality for creating a digital representation of a real piece of furniture is not handled by our system but could be developed as a future goal.

To enter the environment provided by our system, the user must already have an authenticated Metaverse account. The information that is assumed to be available to the system beforehand includes:

- Metaverse account name
- A collection of all houses being sold by the real-estate company
- Corresponding list of details for each house

- A list of flooring patterns/types
- A list of wall and ceiling colours
- A collection of the user's furniture used for customisation

A possible flaw in the design of our system is the situation where multiple users may try and purchase the same house at the same time. As there will be a very low volume of purchases through the system this is very unlikely, but should still be planned for. When a house is purchased it is immediately removed from the collection of houses for sale. This means that if two users try and buy the same house, the first request will be fulfilled whereas the second will fail. The transaction for buying the house for the second user will not complete as this will require the house to be on the market before it verifies payment details and completes the purchase.

Assumptions

For all purposes of our system we will assume that:

- The user is in the Metaverse with a valid Metaverse account
- The user has already set up their VR headset for use in the Metaverse
- There is a list of houses on the market provided by the company that runs this system
- Each house provided has had approval of the owner to be used in our system
- The 3D render of each house has been created with no furniture inside to ensure the customisation features can be used
- Every property has an associated list of features about the property that is displayed to the user
- Payment for a house is handled externally by a separate system
- The system can be scaled up to handle large volumes of tours and hold the data for many houses

Functional Requirements

Key:

- M - Must be included
- S - Should be included
- C - Could be included

REQ1: User Authentication

REQ1.1 The system shall require the user to either login or sign up before accessing the website M

REQ1.2 If the user selected to login, the system shall allow the user to type the email address and password and check if they are valid M

REQ1.3 The system shall transfer the user to the user account management subsystem if the user chose to sign up M

REQ1.4 Once the user account management have signed up the user, he can no longer change his profile details to avoid any kind of fraud M

REQ2: House selection

REQ2.1 The system shall require the user to select a house before accessing the other features of the system M

REQ2.2 The system shall allow the user to search for a house by address, city or postcode using the searching subsystem S

REQ2.3 The user can favourite a house which gives a marker on the house they like S

REQ3: Buying a house

REQ3.1 The system shall show the user the total price of the house M

REQ3.2 Once the user selects pay now, the system shall ask the user to enter credit card details M

REQ3.3 The system shall transfer the credit card details to the payment subsystem M

REQ3.4 If the payment has been done, the system shall record the details of the new owner and send him an invoice via email S

REQ4: Virtual Tour

REQ 4.1 The system shall allow the user to start virtual Tour using AR launcher subsystem S

REQ 4.2 The system shall ask the user to connect a VR for a virtual tour headset M

REQ 4.3 The system shall allow the user to move inside the house using the VR controller S

REQ 4.4 The system shall allow the user to invite a real estate agent to the virtual tour S

REQ5: Properties of a house

REQ 5.1 The system shall provide the user with the number of bedrooms and bathrooms if asked S

REQ 5.2 The system shall allow the user to see the local facilities using the google maps subsystem M

REQ 5.3 The system shall allow the user to know the securities installed in the house: locks, CCTV, intercom... S

REQ 5.4 The system shall allow the user to see the last price the house was sold for M

REQ6: Customising a house

REQ 6.1 The system shall allow the user to customise the house using AR Editor subsystem M

REQ 6.2 The system shall allow the user to change the design of the surfaces of the house (walls/ floors ..) S

REQ 6.3 The system shall allow the user to change the colour and style of furniture (beds/ sofas/ desks ...) S

REQ 6.4 The system shall allow the user to add or remove any accessories to the design C

REQ 6.5 The system shall allow the user to add a design agent to the editing model S

REQ 6.6 The system shall allow the user to export the 3d model M

Non - Functional Requirements

Key:

M - Must be included

S - Should be included

C - Could be included

NFREQ1: Reliability

NFREQ 1.1 The system shall be up 99.9% of the time (three-nines availability) M

NFREQ 1.2 The databases containing all the house models, house information & users are regularly backed up on a weekly basis M

NFREQ 1.3 The system shall have a load balancer to distribute requests across servers S

NFREQ2: Efficiency

NFREQ 2.1 When wanting to do a virtual tour, the model of the house shall take no longer than 10 seconds to load M

NFREQ 2.2 When a payment goes through after buying a house, the house owner shall immediately have legal ownership of the house M

NFREQ 2.3 There shall be a sufficient amount of servers to be able to initially serve 100,000 people S

NFREQ 2.4 The system shall allow for horizontal scaling for increased scalability as more and more people use the system M

NFREQ 2.5 The system shall allow for multiple users to access the same data at the same time, without too much delay S

NFREQ3: Usability

NFREQ 3.1 The system Shall provide instructions at every step of the stage so someone who is unfamiliar with the system can pick it up quickly C

NFREQ 3.2 The system shall be usable by any piece of applicable hardware, to allow for the widest scope of users for the system. S

NFREQ4: Security

NFREQ 4.1 All information stored on external databases/servers shall be properly encrypted to allow only the necessary parties to access it. M

NFREQ 4.2 The system shall require appropriate validation for buyers and sellers and their ownership of the property. M

NFREQ 4.3 The user shall be able to have a unique identifying attribute that can be linked to the system to ensure security. M

NFREQ5: Legislative

NFREQ 5.1 The system shall store user data in a manner that complies with current GDPR legislation. M

NFREQ6: Ethical

NFREQ 6.1 For a property to be put on our system the current owner of it shall give written consent to prevent possible breaches of privacy M

NFREQ 6.2 The 3D render of a house shall only include the buildings owned by the seller, this means blurring or removing neighbouring houses S

NFREQ7: Implementation

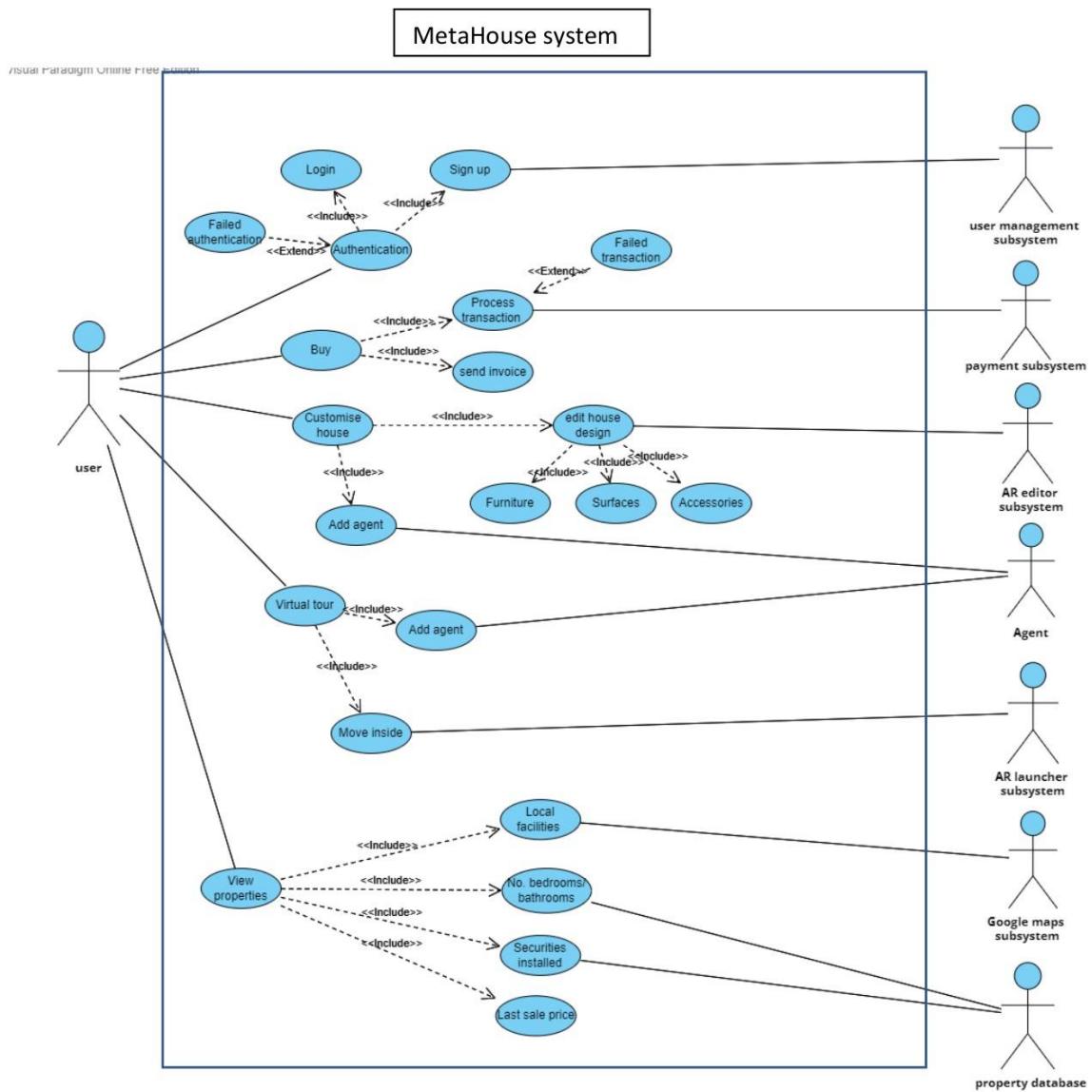
NFREQ 7.1 The system shall be written in a way that should be able to be run across multiple devices. S

NFREQ 7.2 The system shall be written in a modular way, in order to provide better re-usability, expansion and repairability. S

NFREQ 7.3 The system shall be written in a manner that encourages efficiency to allow for a smoother user-experience. C

NFREQ 7.4 The system shall be written in a language that can be supported across multiple platforms, i.e. C or Java. C

Use Case Diagram



Use Cases

Join Virtual Tour

Use Case: Join Virtual Tour	
Actors	User, estate agent, metaverse, MetaHouse system
Preconditions	<ol style="list-style-type: none"> 1. User logged into metaverse and system 2. User and estate agent is in VR 3. User details successfully authenticated 4. The property has already been chosen to be viewed 5. Systems virtual tour is available and ready to be viewed 6. User is loaded into main menu of our system in VR
Flow of events	<ol style="list-style-type: none"> 1. User clicks on property that they want to tour 2. User clicks on 'Enter Virtual Tour' button 3. User enters virtual tour 4. User is loaded into model of property 5. User explores the property using their controller 6. If user wants estate agent to join: <ol style="list-style-type: none"> a. User presses options button on controller to bring up menu b. User clicks invite button c. User invites estate agent d. Estate agent joins in VR e. User exits menu 7. If user wants to access property information: <ol style="list-style-type: none"> a. User presses options button on controller to bring up menu b. User click information button c. Floor plan, number of likes and information about the house comes on a virtual page which is attached to user's controller (user can still explore property) d. If user wants to like property they just click the favourite button e. When user wants to remove information: <ol style="list-style-type: none"> i. User presses options button ii. User clicks remove information button (replaces information button when in use) 8. If user wants to leave property: <ol style="list-style-type: none"> a. User presses options button on controller to bring up menu b. User clicks leave property button c. User leaves property and is loaded back into main menu 9. If user wants to change property another property is chosen and point 1 starts
Post-conditions	<ol style="list-style-type: none"> 1. User is put back into main menu 2. User's view of property is recorded and added to property view count

Scenario 1

John and his husband Joseph have just gotten married and have adopted a kid and now are looking to find their first home. Due to the limited free time they both have, they find travelling to properties very difficult. John enters the metaverse and opens MetaHouse to the main menu. He has already seen a house he and Joseph like so he presses the enter virtual tour button and is loaded into the virtual house. John wants assistance from an estate agent, so he presses the options button and opens up the menu and clicks the invite button and invites the estate agent. He uses the VR controller to move around the house and look around. Then he presses the menu button again and presses the information button which shows him: the price of the house, the postcode, the floorplan, the number of views on the house, and a more information button. John and Joseph decide they have seen enough and want to leave so they press options, and then click the leave property button and they are loaded back into the main menu.

Scenario 2

Stacy and her Mum are selling their house and looking to buy a smaller house and have found 2 possible houses. Due to Stacy's mum being unfamiliar with modern technology, Stacy logs in to the metaverse and lets her mum look around the first house. They think the second house might be better so they bring up the menu and exit the tour, then enter the second house. Stacy's mum likes what is going on with the second house and decides she wants to favourite the house, so she opens the menu, clicks open property information and clicks the favourite button. They then exit the property and the second house is favoured on the menu.

Process of Buying A House

Use Case: The process of buying a house	
Actors	User, Estate Agent, Metaverse, Users bank account, third party maps (for searching), MetaHouse
Preconditions	<ol style="list-style-type: none">1. User is logged into the metaverse and on our system2. The estate agent is in the metaverse and in our system3. The user has sufficient funds in their bank account to complete the transaction4. The property the user wants to buy is registered in our system5. The user should be able to have a virtual tour of the house if they would like to (guided or non guided)
Flow Of Events	<ol style="list-style-type: none">1. The user selects the 'Buy a house' option in the main menu2. A page showing the list of all available houses will be shown, with the following search/filter options<ol style="list-style-type: none">a. Ascend/Descend by priceb. Search for price within a rangec. Search by number of floors

	<ul style="list-style-type: none"> d. Search by number of bedroom e. Search by type of house (e.g Detached/Semi-Detached) f. Search for properties within a certain radius/area g. Search for properties to a close distance to certain facilities (e.g schools, transport) h. Search for properties with certain security features (e.g CCTV, House Alarms etc..) <p>3. A user will have the option to have a house viewing of the following types on a selected house:</p> <ul style="list-style-type: none"> a. A virtual reality tour of the property in the metaverse, which can either be: <ul style="list-style-type: none"> I. A guided tour of the property with a qualified estate agent II. A non-guided tour of the property b. A non virtual reality tour from a computer for users who don't have a VR headset <p>4. Users will be presented with following additional details on the page whilst displaying the property:</p> <ul style="list-style-type: none"> a. Past sell prices of the property b. Any modifications to the property from when it was first put on sale c. Any damages to the property d. Blueprints of the property e. The inventory of all the items in the property f. The current owners metaverse name & previous house owners <p>5. Once viewing a house, a user will have the option to 'favourite' a property</p> <p>6. If more than one property is favorited, in the the user can open the 'favourites' tab to view a comparison of all the potential properties</p> <p>7. Once a user has found a property they would like to buy, the user can click on:</p> <ul style="list-style-type: none"> a. Buy Property <ul style="list-style-type: none"> I. The system transfers the credit card details to the payment subsystem II. The funds get taken out of the users account and transferred to the seller III. The user is now the owner of the property IV. The property is now taken of the market
Post Conditions	<ol style="list-style-type: none"> 1. If a property isn't bought, any favorited properties will be saved for the user 2. If a property is bought: <ol style="list-style-type: none"> a. The property is taken of the market/viewings b. The user is sent back to the main menu

Scenario 1

Mark, who is currently living with his parents, has decided that he would like to move out and find a place of his own. He has been saving up money from work and his parents have agreed to help with the payment. Mark has already made the choice that he wants to live in a new city, and as an avid football player, he would like to live near a park/football field. Mark logs in to our system, and he clicks on the buy house option where he is sent to a new page where he is able to see all the available properties. Mark has decided that he wants to move to London, so he filters the properties by Location. As London is very expensive, Mark filters the search to see properties furthest away from the central to keep the prices low, as well as searching for a house close to a park/football field so he can continue his hobbies. Mark then finds a potential house which matches against his requirements, where he can choose to go on a virtual tour (self-guided or with a qualified real-estate agent). Mark likes the house, so he then adds it to his favourites. After favoriting several houses, Mark then goes to his favourites tab, where he compares all the houses he has favourited, and makes a final informed choice. Mark has now found a potential house, so he clicks on the 'Buy Property' option. Once Mark clicks on buy now, his bank details are sent to the payment subsystem, where his funds get transferred over to the respective seller. If the payment goes through successfully, Mark's new home is taken off the market and he is sent back to the main menu. Mark then receives an invoice detailing the transaction.

Scenario 2

Mary, an interior design graduate already has a couple of houses in mind which happened to already be on the marketplace. As she has a working knowledge of interior design from her studies, she wishes to see what her perfect home would look like under her own configurations in each of the rooms of the house before buying her preferred property. For her process of buying a house, she would like to go on a non-guided virtual tour, trying out the feature of being able to place different types of furniture around to see the design of the house before buying it and designing it in real life. Mary logs in to the system and searches for the houses she already had in mind. For each of the houses she goes on a non-guided virtual tour of the house, and she starts designing each room in VR to see what it would look like in real life had she bought the house and designed it. Mary does this for each of the houses she had in mind and favourites the ones that had the most potential. After favouriting several houses, Mary then goes to his favourites tab, where she compares all the houses she has favourited, and makes a final informed choice. Mary has now found a potential house, so she clicks on the 'Buy Property' option. Once Mary clicks on buy now, her bank details are sent to the payment subsystem, where her funds get transferred over to the respective seller. If the payment went through successfully, Mary's new home is taken off the market and she is sent back to the main menu. Mary then receives an invoice to her email detailing the transaction.

Viewing House Properties

Use case: Viewing House Properties	
Actors	User, mapping software, house database, metaverse, MetaHouse.
Preconditions	<ol style="list-style-type: none"> 1. The user is logged into the system. 2. The user has selected a house they wish to view the details of. 3. The house has a relevant entry in the database. 4. The house has a valid address linked to it.
Flow of events	<ol style="list-style-type: none"> 1. This use case will start when the user wishes to view the details of a property. 2. The system will send a request to the database to retrieve the attributes of the property, these attributes include (but are not limited to): <ol style="list-style-type: none"> a. Number of bedrooms and bathrooms. b. The address of the property. c. Any extra items installed in the property. d. The number of views of the property. 3. The attribute for the number of views of the property will then be incremented by one on the property database. 4. The number of bedrooms in the house, as well as address and items installed in the property will be displayed to the user. 5. The user can then choose to then enter some sub-menus: <ol style="list-style-type: none"> a. They can choose to select the house price sub-menu, which will give the user the price that the house last sold for, as well as prices of other houses in the area for reference. b. The user can choose to view a list of facilities in the local area, where the system will request information from the mapping system about the local facilities, and display this information as a list. 6. Once the user has finished with the sub-menus, they can choose to exit if they wish. 7. The user will then exit this menu, once they have finished.
Postconditions	<ol style="list-style-type: none"> 1. The user has been returned to the previous menu. 2. The number of views has been updated correctly. 3. Requests have been sent to the appropriate helper servers.

Scenario 1

Dave is currently looking at possible options for houses to purchase, and decides there is one he would like to take a closer look at. He decides he would like to know more about this property, so he selects the “View Properties” button on the property. Dave then sees that the house has 2 bedrooms and 1 bathroom, and also has CCTV cameras installed. Dave is interested in this property so he decides to view the previous sale price, so selects this option in the menu. The price of £200,000 is displayed, which Dave hopes the house will sell around the same amount this time. Since Dave has seen all the information he wishes to, he exits the price sunscreen and then decides to look at some other housing options, so chooses to exit the detail view menu.

Scenario 2

Alex and her husband Steve are looking to buy a new house to be able to start a family, they both have very good jobs so the price of the property isn't as much of a worry to them, but they do like having close access to shops, as well as having a nearby school for when their children eventually grow up. They have found a house they wish to see more about and then decide to select the “View Properties” option. The couple are then shown that the house they wish to purchase has 4 bedrooms and 2 bathrooms, which they are happy with. Alex decides to select the option to see the nearby facilities, as they would like to know what is nearby this new house. They are then shown a list of the nearby facilities, with a Pub ½ a mile away, a supermarket 1 mile away and a primary school ¾ of a mile away. They both agree that all of these would be very useful to them, especially with the prospect of having children, being able to walk them to school would be very helpful. They have decided that this is a property that they would like to purchase, so exit the detailed view and decide to go and request a virtual tour of the property.

Customising a House

Use Case: Customising a house	
Actors	User, Design Agent, Metaverse, MetaHouse
Preconditions	<ol style="list-style-type: none">1. User is logged into the system2. User is authenticated3. Property selected to customise4. User's items have been added to the system5. User in a virtual tour of the selected house
Flow of events	<ol style="list-style-type: none">1. User selects ‘Customisation Mode’ to be able to customise the house they are currently viewing2. The user has the options to add a design agent into the editing model3. The user will now be able to edit parts of the house:<ol style="list-style-type: none">a. Design of the surfaces including walls/floorsb. Colour / style of the furniturec. Add / remove any accessories4. When the user looks around the house, the system will

	<p>change the shade of the part currently being looked at to indicate that part is editable</p> <ol style="list-style-type: none"> 5. When the user selects an editable part of the house they are given the options: 'Remove', 'Move', 'Change colour', 'Change design' 6. If the user selects 'Remove': <ul style="list-style-type: none"> a. The item will be removed from the virtual tour 7. If the user selects 'Move': <ul style="list-style-type: none"> a. The user will now effectively be holding the item and can place it down anywhere in the tour 8. If the user selects 'Change colour' <ul style="list-style-type: none"> a. A menu will appear with a colour selector b. When the user hovers over the colour, the colour of the item will change in real time c. When the user selects a colour the colour of the item will change d. The menu will close 9. If the user selects 'Change design' <ul style="list-style-type: none"> a. A menu will appear with a list of all the designs available b. When the user hovers over a design the design of the item will change in real time c. When the user selects a design the design of the item will change d. The menu will close 10. If the user selects '+' <ul style="list-style-type: none"> a. A list of items already scanned in will show b. The user can select a item c. Once an item is selected they can place it down inside the virtual tour 11. Once the user is finished customising, the user will have a option to select 'exit' 12. Two options will appear: 'Save' and 'Export' 13. If the user selects 'Save': <ul style="list-style-type: none"> a. The system will ask for a name and save it 14. If the user selects 'Export' <ul style="list-style-type: none"> a. The model the user has been edited is exported as a 3D model b. The 3D model will be downloaded
Post-conditions	<ol style="list-style-type: none"> 1. The user if put back into virtual tour mode 2. The user is able to switch between the 'original' virtual tour and 'edited' virtual tour

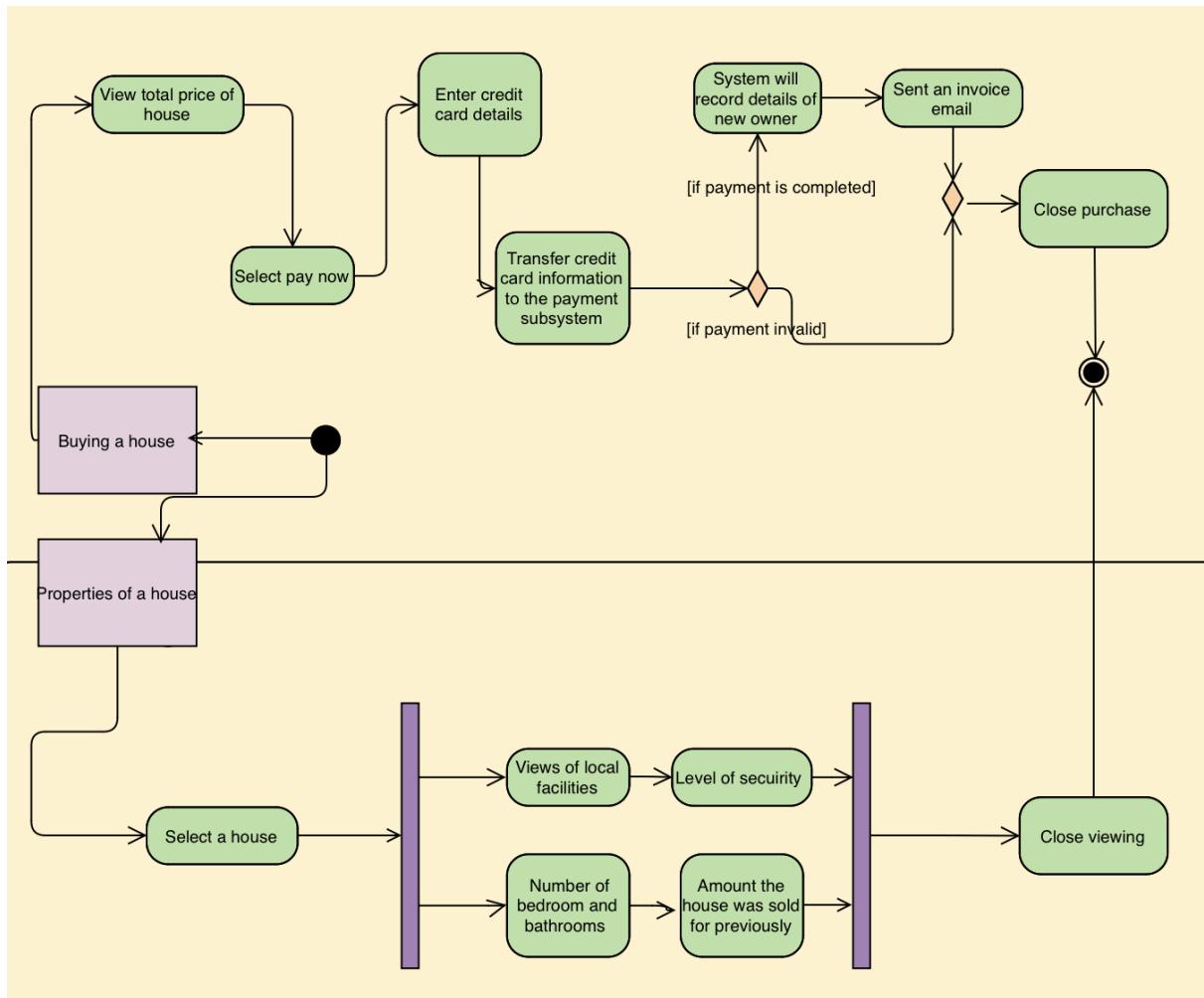
Scenario 1

John wants to see what the house looks like with his current furniture and his own design choices before making any real life changes. He loads up the system and logs in, selects the virtual tour for the house he wants to purchase and selects the 'Customization Mode'. He selects the options to add a design agent to assist him in the process. He then removes the current beds and sofas in the virtual tour and replaces them with the ones from his house that he has already added to the system. He likes how this looks, so he decides that he wants to change the design of the floor in the hallway, he removes the carpet and selects wood to replace it. He also likes how this looks. He then saves this edited virtual tour and exits the virtual tour. Being able to see what his current furniture looks like in this house and being able to edit things to how he likes them has convinced him that he likes this house. He is now confident that he wants to buy it.

Scenario 2

Alison is looking to buy a new house, she has narrowed it down to 3 choices by using the virtual tour system and was able to get a good idea about the house without having to travel. To try and decide, she uses the customisation feature in the virtual tour on one of the houses, she selects the 'Customisation Mode'. She starts by changing the colour of the walls to pink as it's her favourite colour, then changes the colour of the carpet to green. She loves the way the house looks now. She decides to see what it would look like with her furniture from her current home, so she removes the dining table and sofas and replaces them with her own from the '+' menu. After adding her own furniture she decides that her current furniture does not suit the house and she would have to buy different furniture if she were to move into this house. She then presses 'exit' and saves the edited virtual tour and selects the options to download the edited virtual tour as a 3D model, then exits the customisation mode.

Activity Diagram



Class Analysis

Noun/Verb Analysis:

Candidate Classes (Nouns):

Candidate Class	Use
User	Class
Tour	Class
House	Class
Real-estate	Reject - Duplicate of Properties
Metaverse	Reject - Out of Scope
Wallpaper	Attribute - Surfaces
Carpets	Attribute - Surfaces
Furniture	Class - Subclass of CustomDesign
Menu	Reject - Out of Scope
Sale	Class
Credit Card	Class - Subclass of Sale
Purchase	Reject - Duplicate of Sale
3D Render	Reject - Out of Scope
VR Headset	Reject - Out of Scope
Ceiling	Attribute - Surfaces
Inventory	Reject - Out of Scope
Authenticate	Class - Subclass of User
Property	Reject - Duplicate of House
VRController	Class
Agent	Class
Securities	Class - Subclass of Properties
Facilities	Attribute - Properties
Bedrooms	Attribute - Properties

Bathrooms	Attribute - Properties
Locks	Attribute - Securities
CCTV	Attribute - Securities
Intercomms	Attribute - Securities
Price	Attribute - Sale

Candidate Operations (Verbs):

Candidate Operation	Class
Tour	Reject - Too Broad
Select House	Tour
Navigate	Reject - Out of Scope
Creating Custom Design	Reject - Too Broad
Change Surface	CustomDesign
Change Furniture	CustomDesign
Add Accessories	CustomDesign
Purchase House	Sale
Buy	Reject - Duplicate
Authenticate	Authentication

CRC Cards:

User	
Holds details about the user Changes the user's password	House Sale Authentication
creditCard	
Holds details about a credit card	Sale
Sale	
Holds the house to be bought Gets information needed for purchase Outsources the payment to external system	User House creditCard
House	
Holds the key info about the house	Sale User CustomDesign
Authentication	
Registers or Logs in a user Verifies whether a user is authenticated	User
Tour	
Launches the VR tour Invites an agent for tour	VRController Agent House
VRController	
Lets the user move around the 3D render	Tour
Agent	
Holds details about an invited agent Requests an agent	Tour CustomDesign
CustomDesign	

Changes the customisation of the house Invites an agent for designing Adds furniture and accessories to house Saves created design	House Surfaces Accessories Furniture Agent
---	--

Surfaces	
Holds details about a surface	CustomDesign

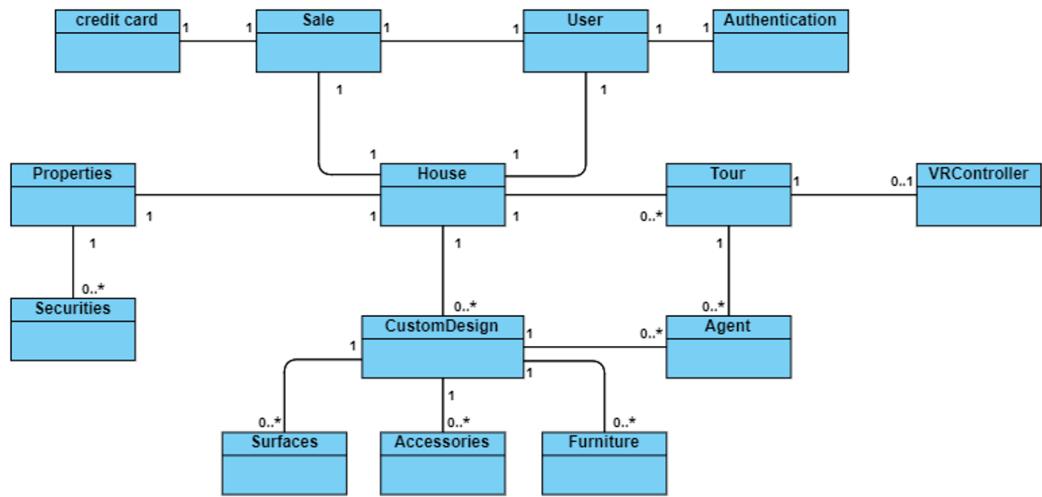
Accessories	
Holds details about an accessory	CustomDesign

Furniture	
Holds details about furniture	CustomDesign

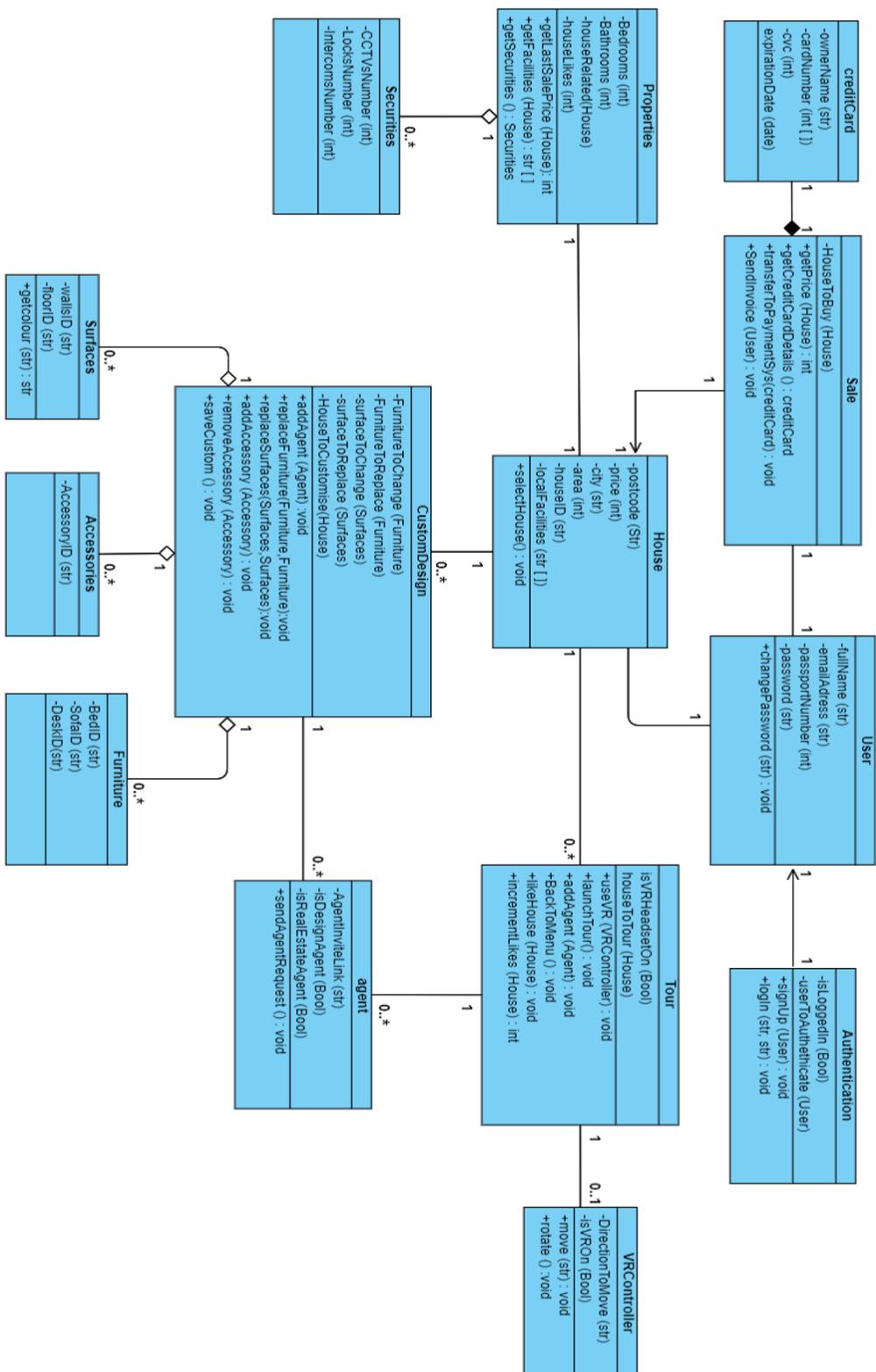
Properties	
Holds high level details about the property Gets the price of the house	House Securities

Securities	
Holds details about the securities of the house	Properties

First cut class diagram:

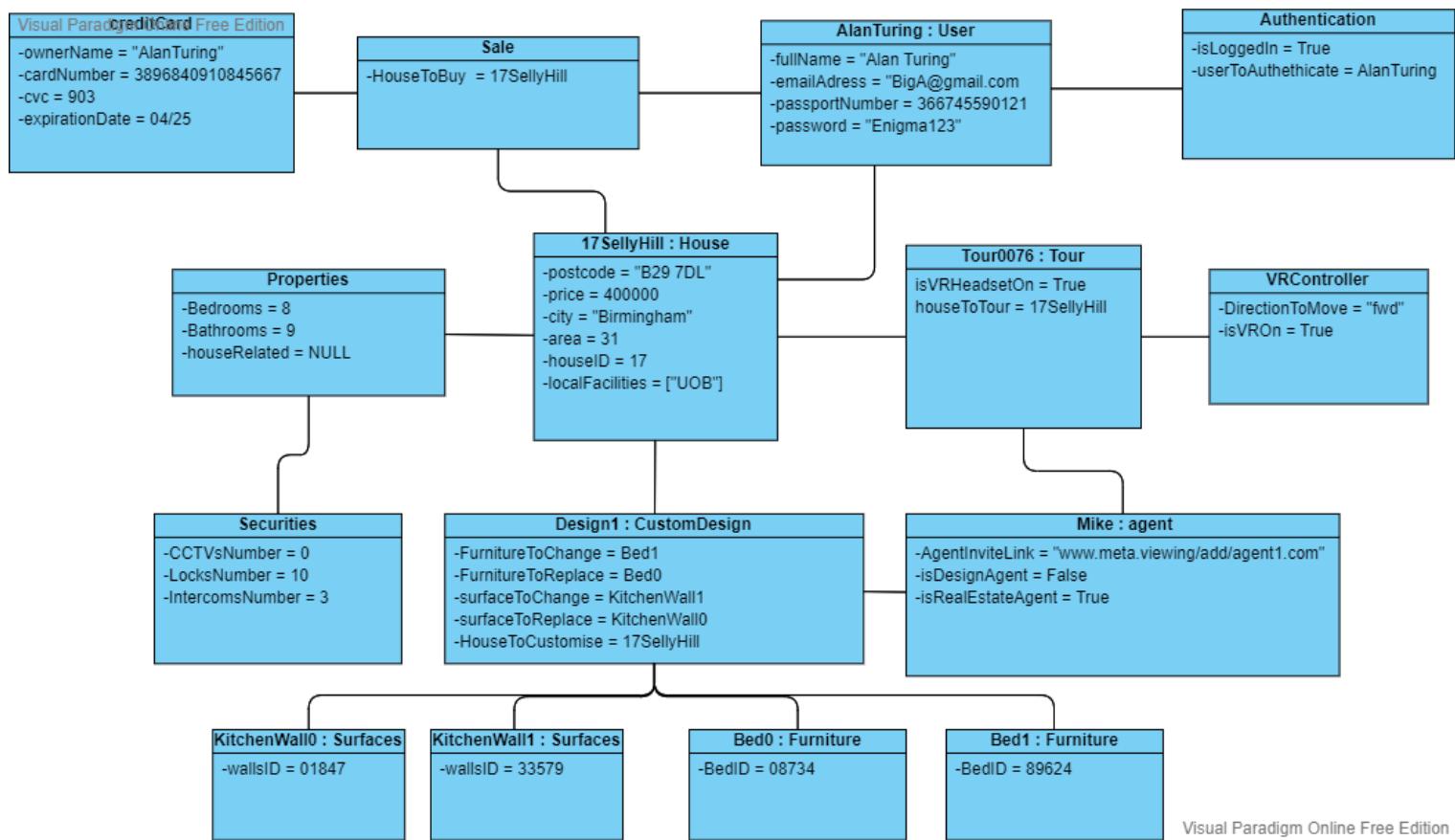


Detailed class diagram:



Object Diagram

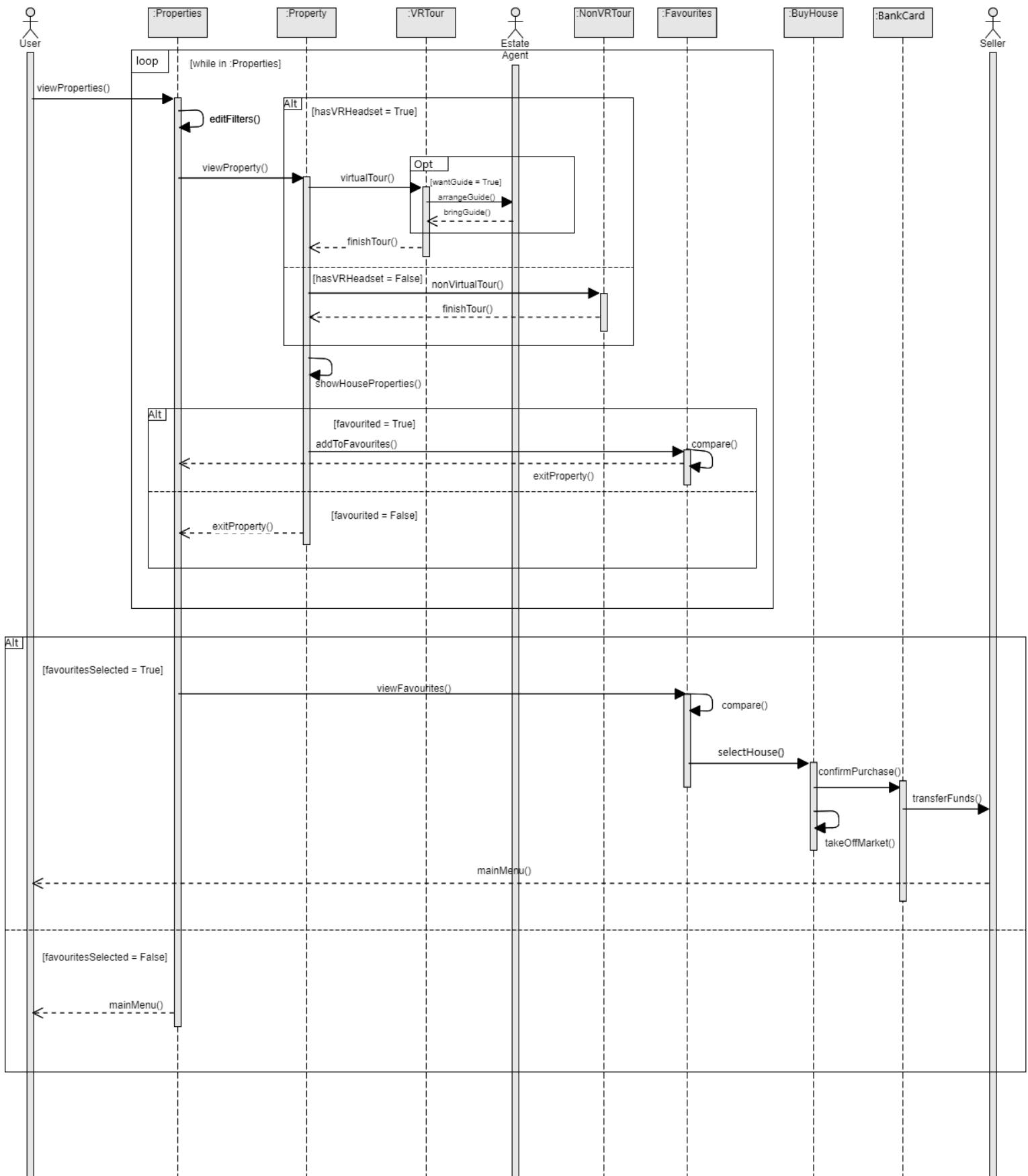
Object Diagram for Viewing a House



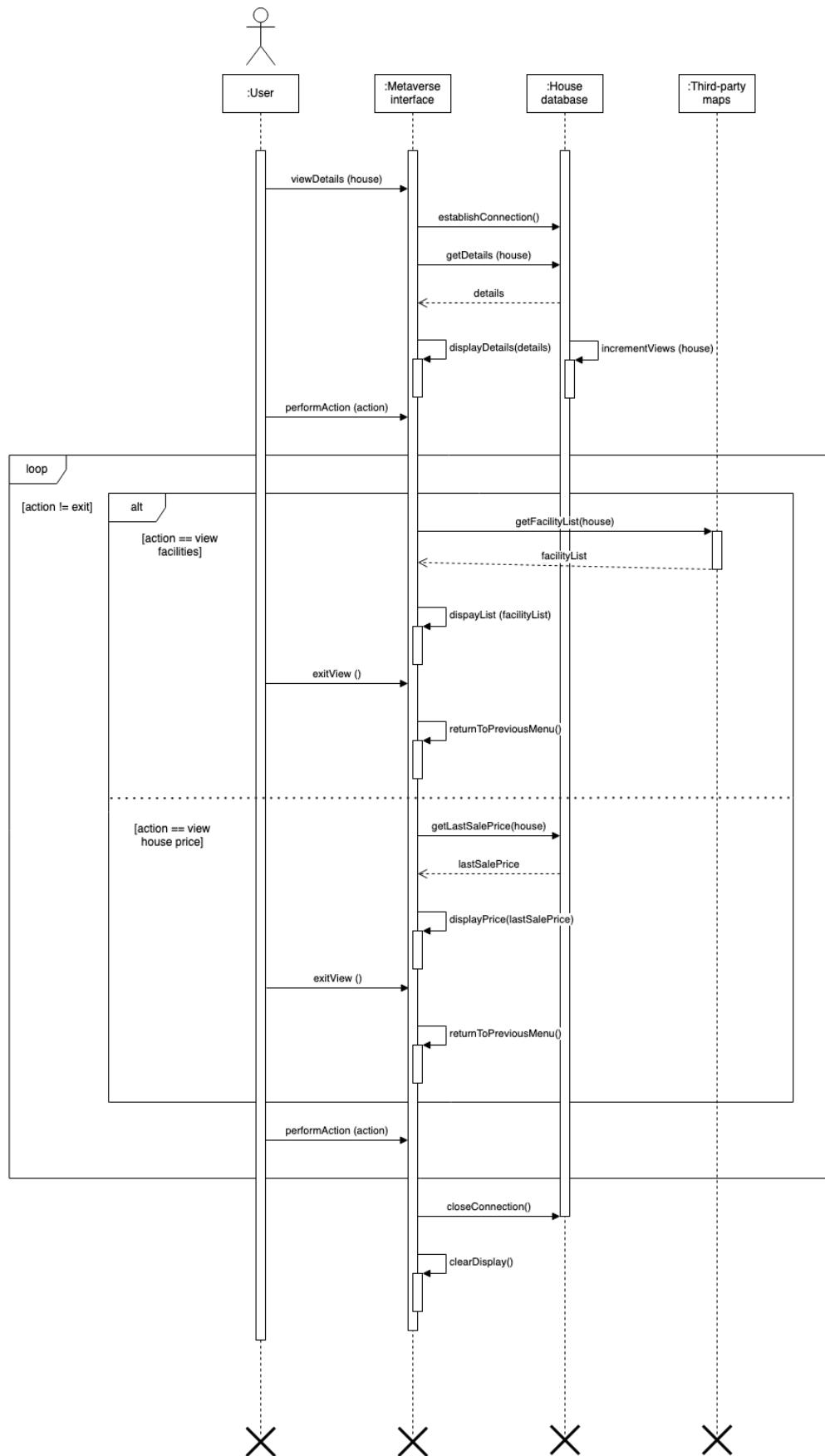
Visual Paradigm Online Free Edition

Sequence Diagrams

Sequence diagram for the process of buying a house

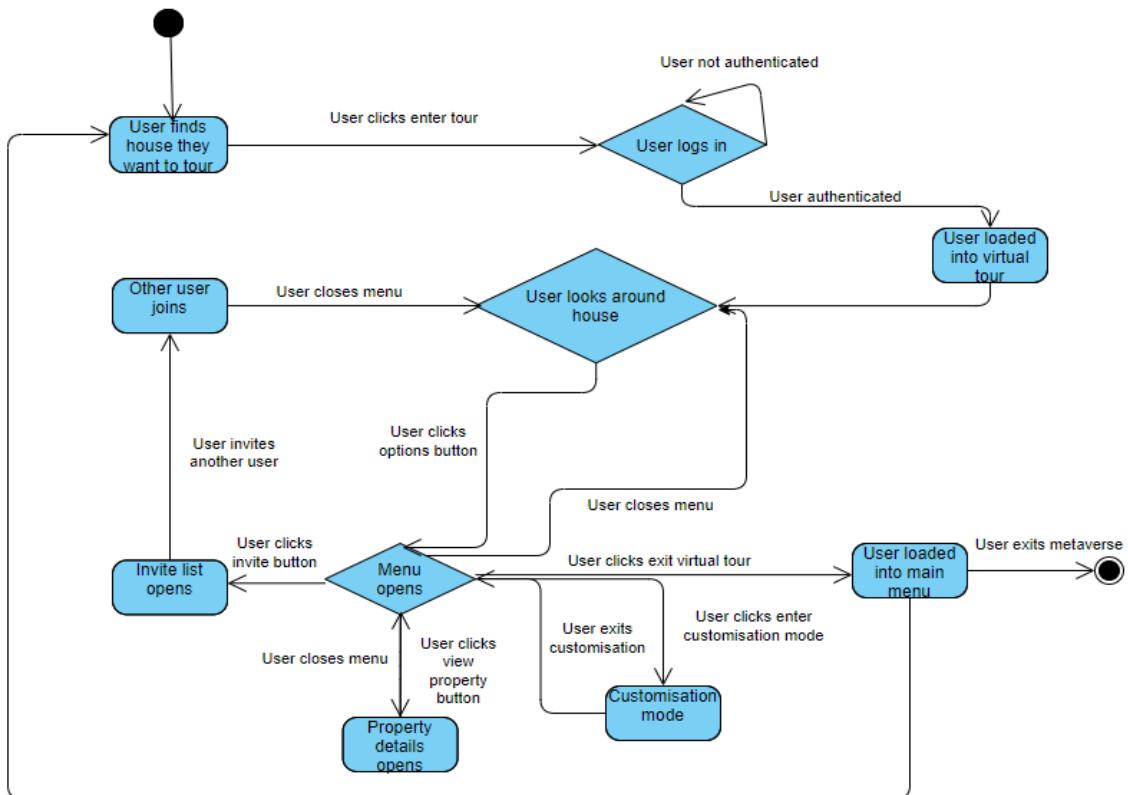


Sequence Diagram for Viewing Details

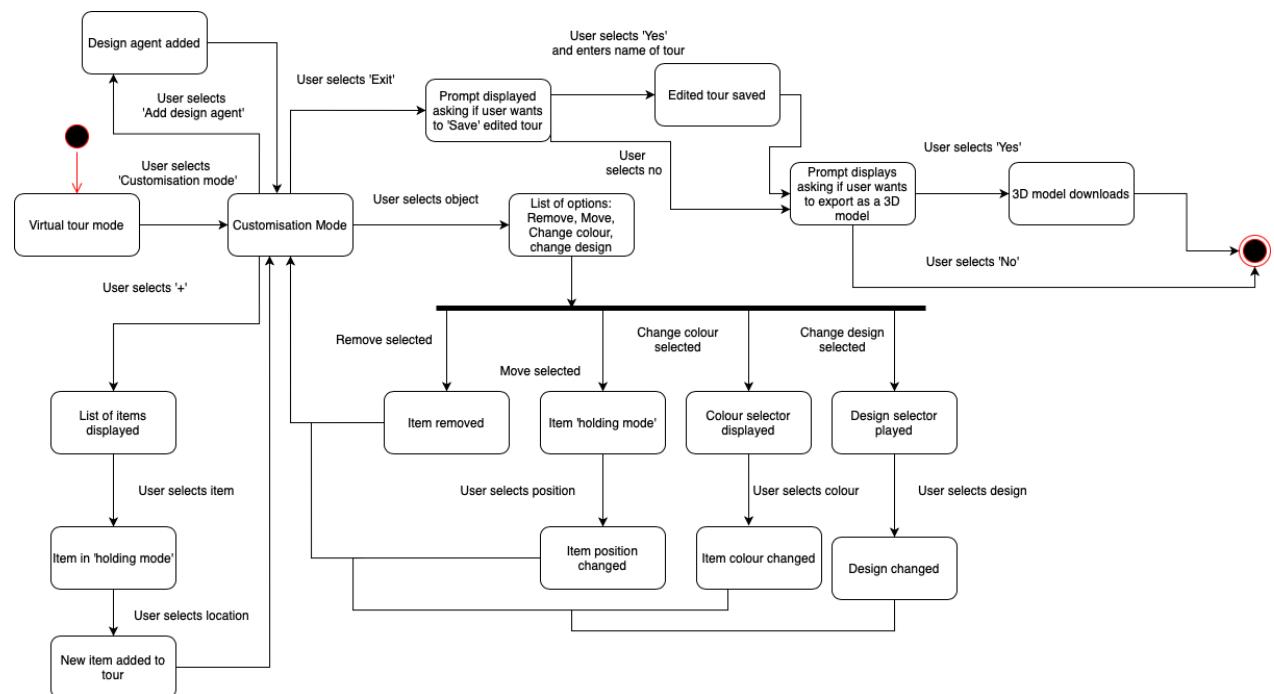


State Diagrams

State diagram for Virtual Tour



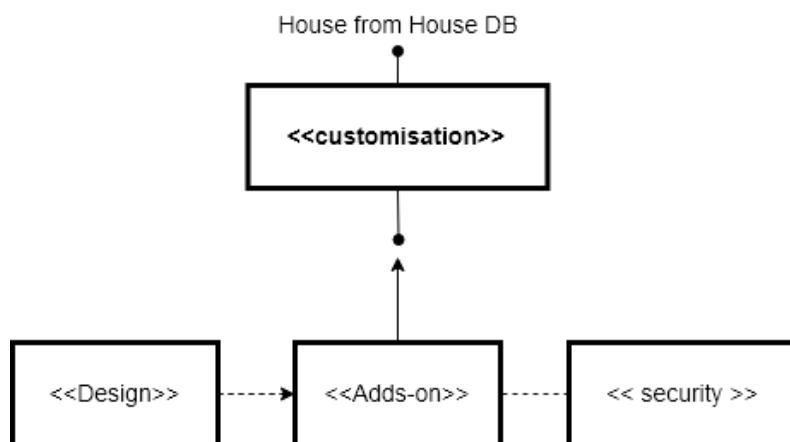
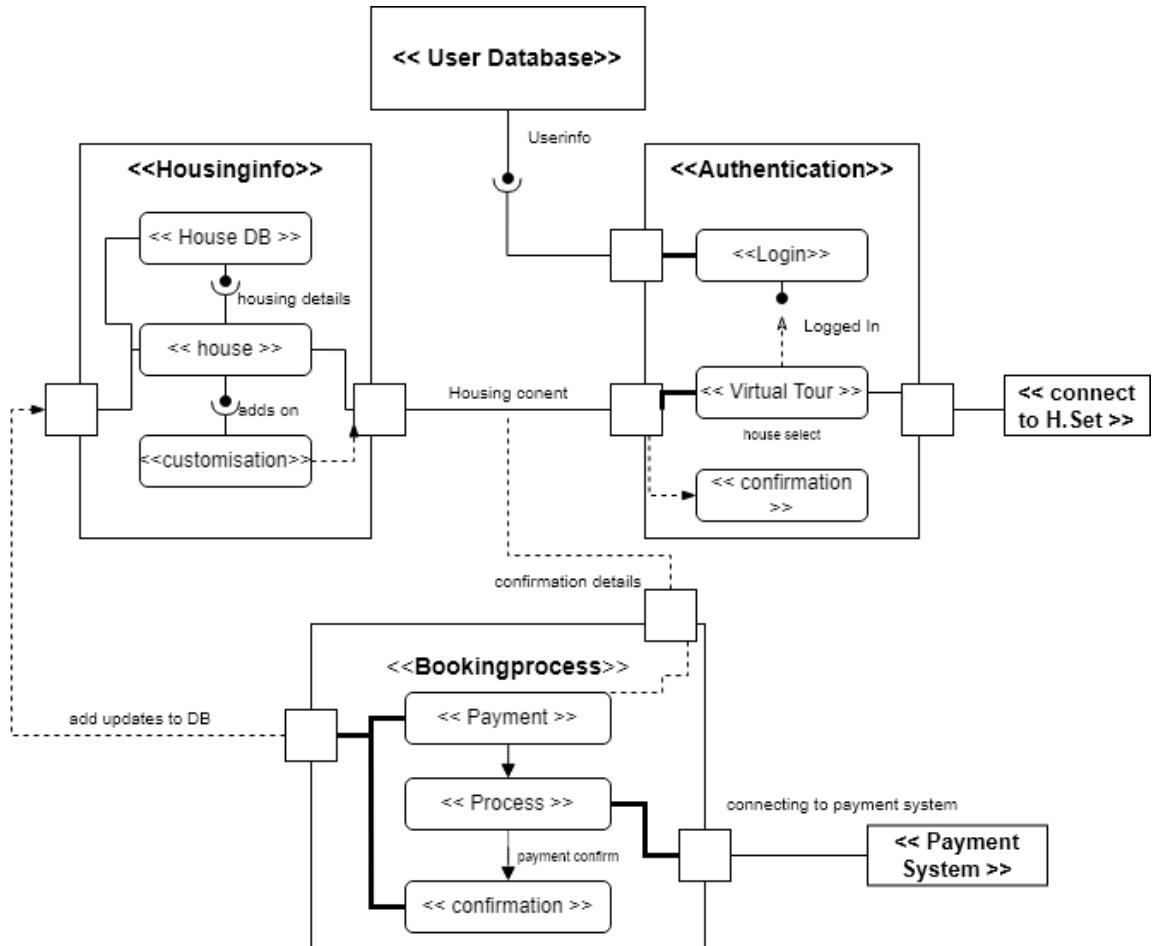
State diagram for Customising a House



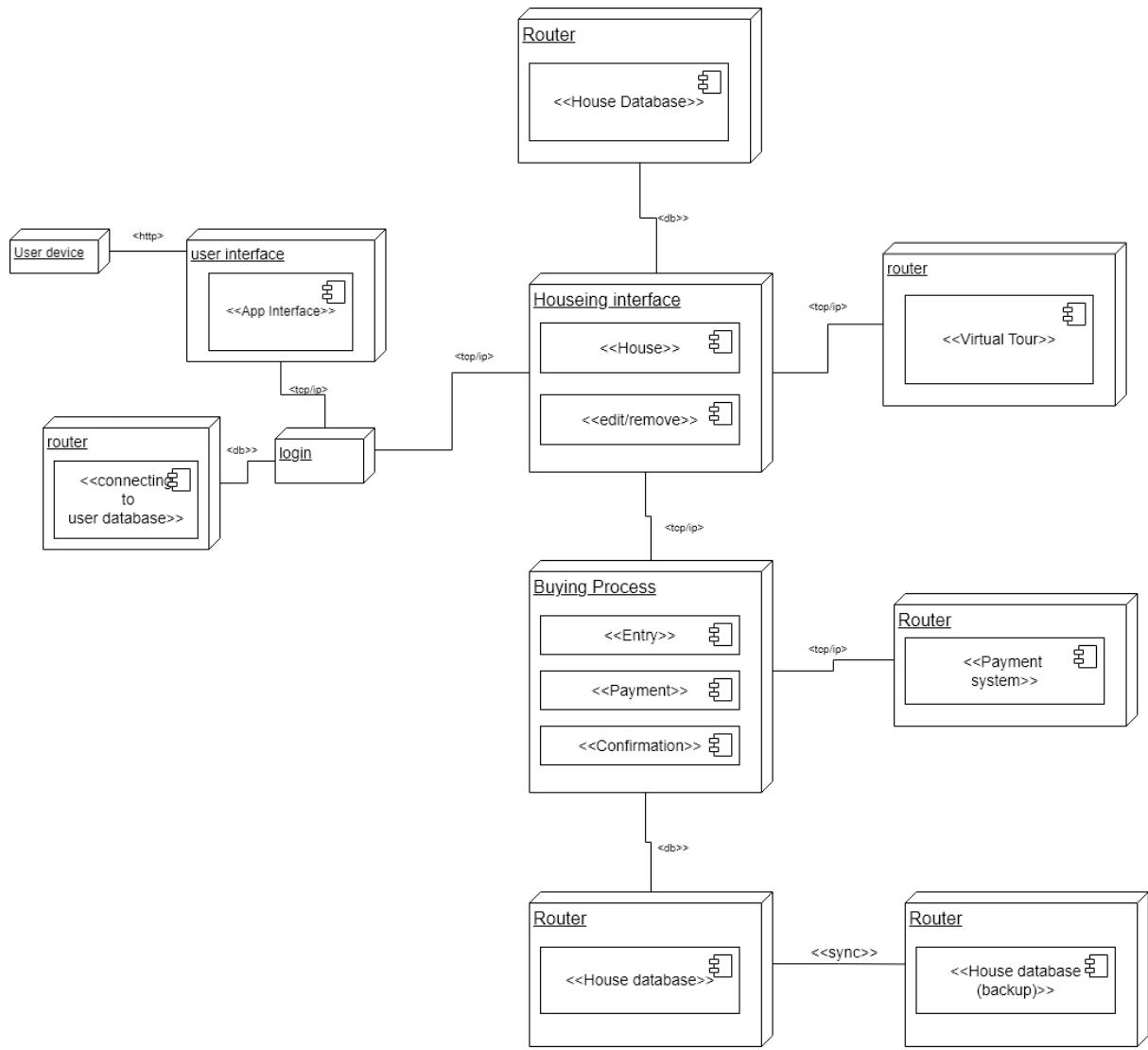
Architecture Style, Modelling and Evaluation

Architecture Style 1:

Component Diagram

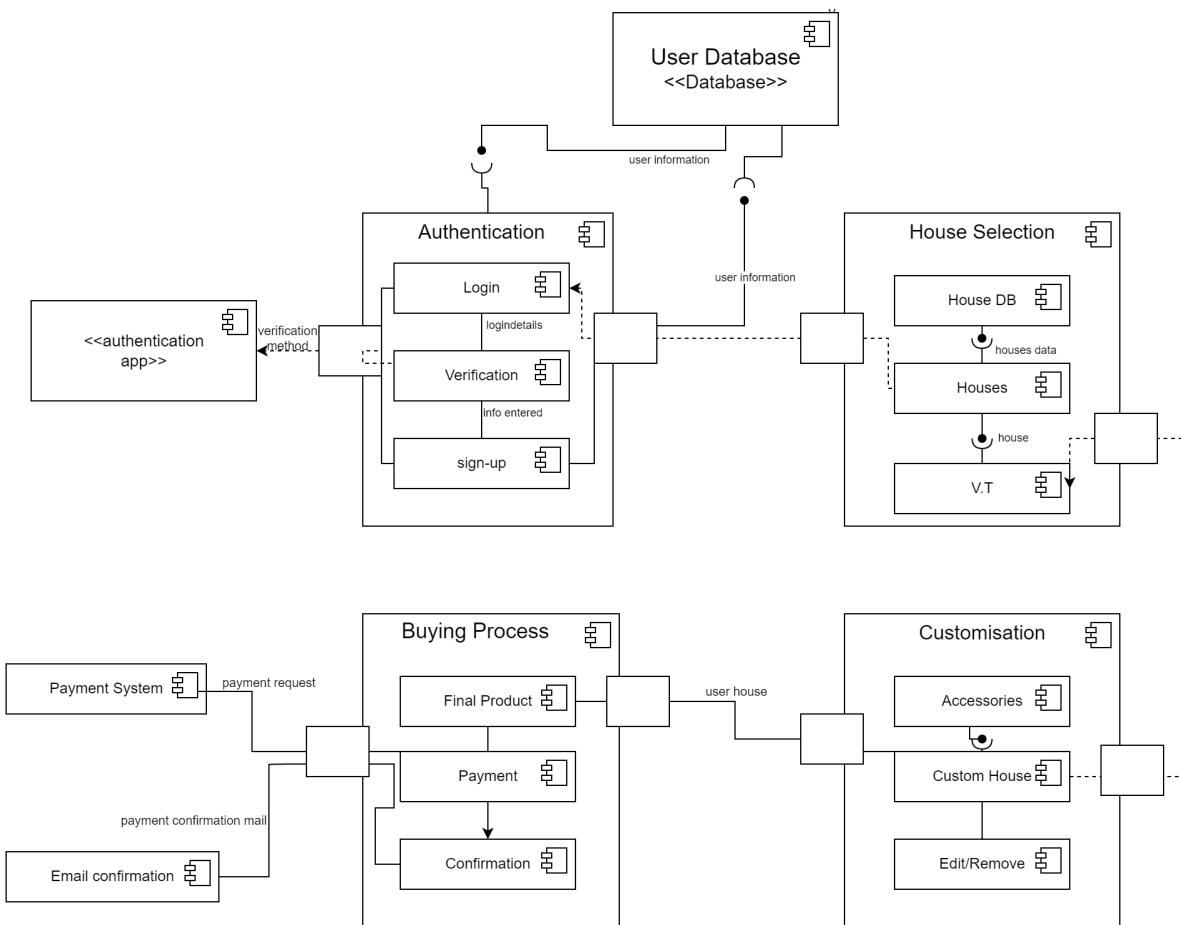


Deployment Diagram

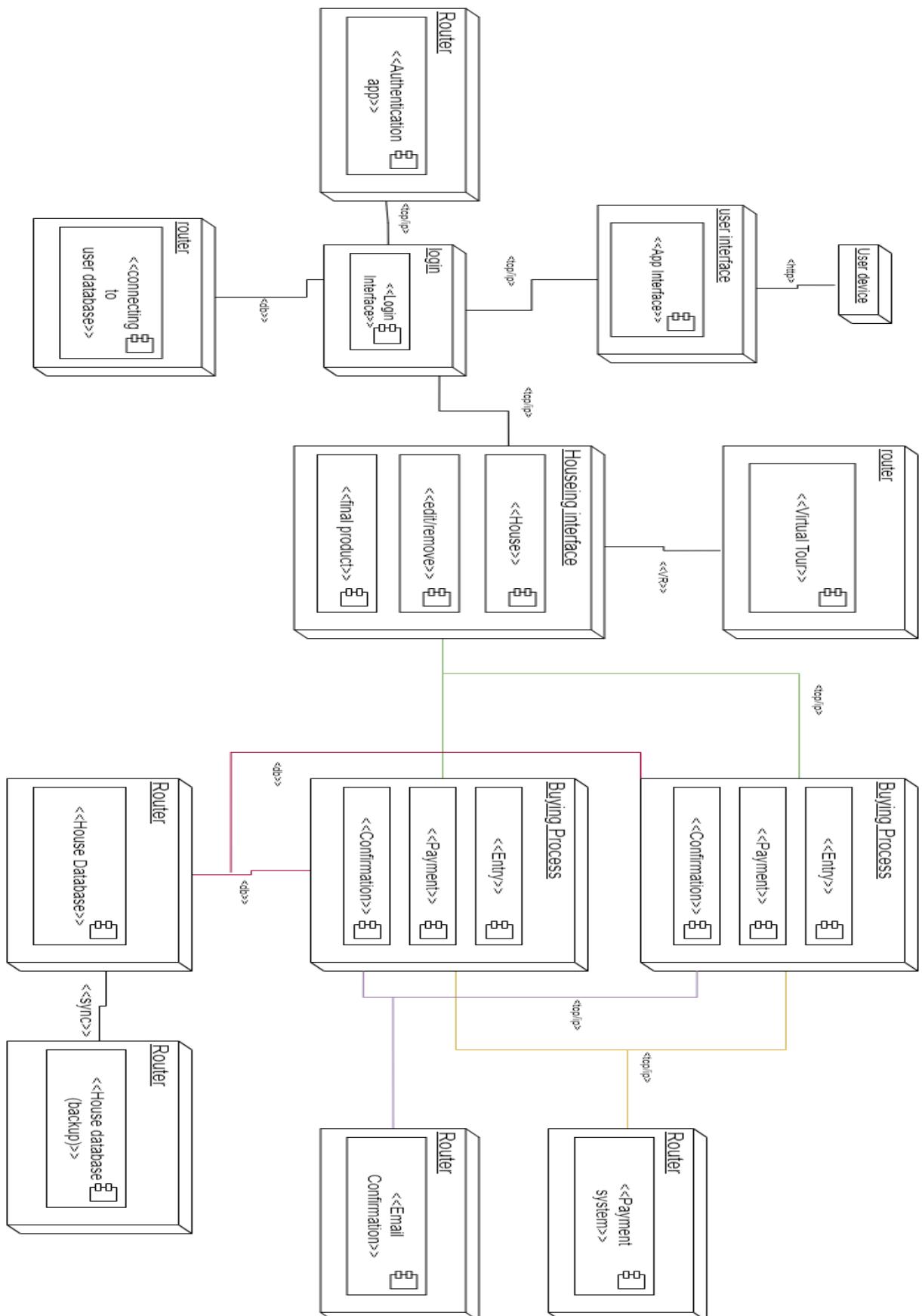


Architecture Style 2:

Component Diagram



Deployment Diagram



Architecture Comparison

When designing an architecture for MetaHouse, a number of factors need to be considered such as: security, efficiency, scalability, cost to setup and cost to maintain. All of these factors cannot be considered at the same time, therefore compromises and tradeoffs are made.

Both architectures are fairly similar with a simple approach and multiple subsystems and routers accessing data giving both good efficiency and security.

The first architecture style has a simpler approach being as concise and effective as possible. This allows for scalability and a high level of efficiency. Downsides to this simple architecture is that the security of the system is not as good as the second, and it doesn't deal with network traffic as well. However, it is more cost effective and easier to maintain which is ideal for a smaller business.

In the second architecture an emphasis was put on security with multiple routers to connect to different databases and it also has multiple buying process subsystems each with a router. This direction leads to great security as even when one of the databases or routers is breached, the data in the other subsystems are still intact. Another upside to having these separate systems is network traffic is spread out over the different routers so the user shouldn't notice a drop in speed during peak use hours. A downside to all these different routers is setup cost and cost to maintain the system as it is using more routers than the other architecture style. It has benefitted from being able to upgrade or add more routers to a system at a cheaper cost as each subsystem isn't affected by each other so the system in the worst condition can be focussed on, however the downside of the scalability is, to fully scale up the entire system, a large cost is needed.

In conclusion, the second architecture is preferred as the focus on security is sufficient enough to allow for the slight difference in cost. Due to the additional advantages in network traffic and efficiency, this will be the architecture chosen to implement the system.

Testing

Introduction

The system will allow the user to search for properties within a given address, as well as matching any attributes they require. The user can choose to view virtual tours of the house, which will allow them to be taken around a 3D model of the property, which they can then choose to customise with any furniture or wall colours that they wish. The user can choose to purchase a property if they wish, which will utilise an external payment system. All details of the house will be stored on a database server, with some other external systems being used for extra functions.

The objective for testing is to determine if the system meets the functional and non-functional requirements specified for it, as well as ensuring that the system meets all quality specifications. Any bugs and errors that affect the critical functionality of the system should be identified and fixed before the system is released, as well as the system having a simple user experience and being quick to navigate. The response time of the system should be short enough so that the overall experience remains smooth. The goal is to create a system that can be used for its intended purpose with minimal issues and be performant at the same time, these factors will be evaluated during the testing process.

Assumptions

For these tests, the following assumptions will be made:

- For testing, a test database will be created and connected to, with sample entries.
- The login and sign-up service can be run in a test environment, with sample logins created for testing.
- There are no restrictions on the format of passwords for accounts.
- The payment system can be run in a test environment with “dummy” accounts.

Test Items

The systems to be tested include the login system, the house selection system, some aspects of the buying system, virtual tour system, house detail system and the house customisation system. These will all be tested for functionality and ease of use. These should be tested using multiple headsets, however not all devices can be tested, so only the most popular will be used for this test.

Features to be Tested

Features that will be tested include:

- As a user, logging into the system.
- As a user, signing up for the system.
- As a user, searching for a property.

- As a user, selecting a property.
- As a user, purchase a house and receive an invoice.
- As a user, starting a virtual tour.
- As a user, navigating a virtual tour.
- As a user, adding an estate-agent to the tour.
- As a user, view the details of the property.
- As a user, view the last sale price of the property.
- As a user, enter the customisation mode.
- As a user, change the colour of walls.
- As a user, add furniture to the house.
- As a user, export the 3D model.
- As a user, simply navigate the system.

Features not to be Tested

Features that will not be tested include the full functionality of the payment subsystem, as dummy accounts will be created which have no real money, however can be used to simulate them for some functions of the payment system. This will allow some testing to be performed on the payment subsystem, without any real funds having to be transferred (which would not be possible with tools available) whilst still testing the critical sections of the buying system. Dummy accounts will have an account number and a balance (the minimum required for testing), with no other features of real bank accounts. The map subsystem cannot be tested due to reliance on external data, which cannot be easily altered and then tested in a controlled testing environment.

Strategy

Testing for this system will be performed with both black-box and white-box methods, with the white-box methods being performed by programmers throughout the implementation of the system, with the goal of ensuring functionality of small sub-sections. This will be done with the use of unit testing methods, as well as some integration tests. The tests specified in the “Pass/Fail Criteria” section will be performed in a black-box method, with most of the tests utilising a class partitioning method for defining the test data. This will be performed with system testing methods. These tests will be performed multiple times to ensure functionality, and each of the tests will be split between a group of testers by the testing team.

Tests should be marked with Pass/Fail, depending whether the test matches the expected outcome. If a test is marked as Fail, then a request should be created and assigned to a developer. The developer will then fix the issue, and this will then be retested. If the new test fails again then the process is repeated until it passes.

Quality factors that will be measured during the testing process include the usability of the system, measured in a test case as well as being monitored throughout the testing process for feedback. Efficiency will also be tested, as well as correctness of the system and interoperability with 3rd party systems.

Pass/Fail Criteria

The system should have no critical defects to the functionality of the system, in order to ensure the system works as expected. Test cases will be run as described below.

Shown below is data to be created for use in the test cases:

Data	Description
Account with attributes: <ul style="list-style-type: none">• Email: "test@email.com"• Password: "testPassword1"	This is a user account that will be used throughout the testing.
House entry in database with attributes: <ul style="list-style-type: none">• Address: "AB12 CDE"• Bedrooms: 2• Bathrooms: 1• CCTV installed: Yes• Previous selling price: £200,000• Current sale price: £250,000• 3D model of the house should be created.	This property will be used for most valid use cases in the system.
House entry in database with attributes: <ul style="list-style-type: none">• Address: "AB12 CDE"• Bedrooms: 3	This house will also be used in test cases.
House entry in database with attributes: <ul style="list-style-type: none">• Address: "AB12 CDE"• Bedrooms: 1	This house will also be used in test cases.
House entry in database with attributes: <ul style="list-style-type: none">• Address: "XY99 ZZZ"• Current sale price: £400,000	This house will be used for some invalid cases.
Payment account with attributes: <ul style="list-style-type: none">• Account number: 1111 2222 3333 4444• Funds: £300,000	This account will be used for test cases for payment.
Payment account with attributes: <ul style="list-style-type: none">• Account number: 5555 6666 7777 8888• Funds: £300,000	This account will be used for test cases for payment.

Test Cases

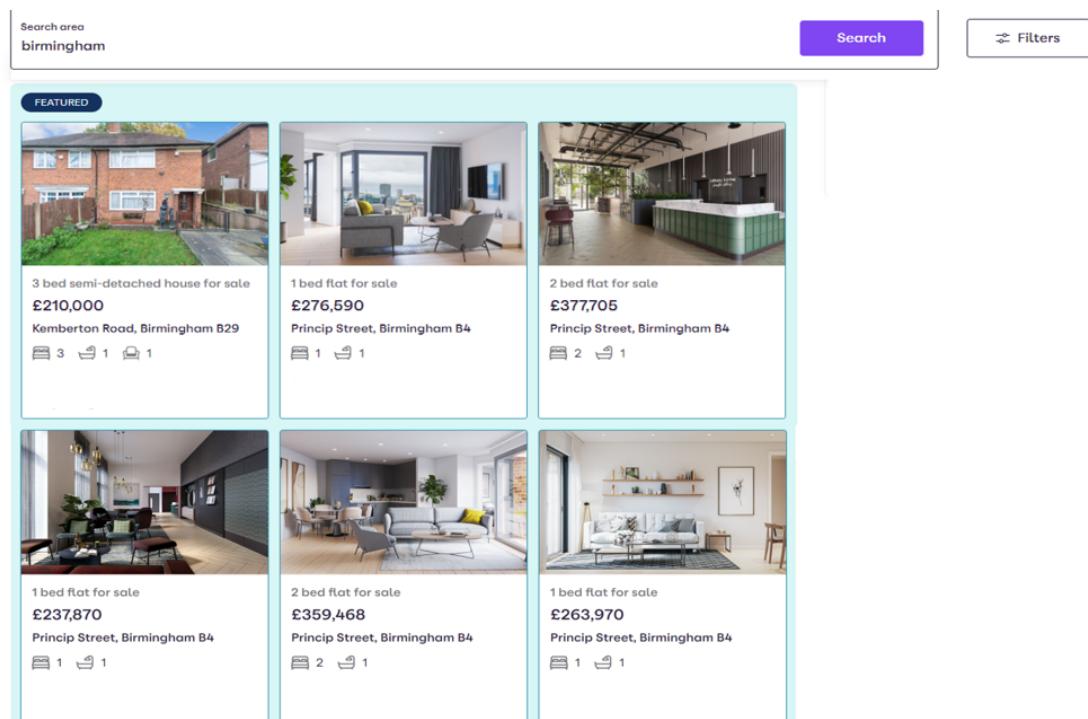
Test ID	Req ID	Description	Test steps	Test data	Expected result	Actual result	Pass / Fail	Comments
T1.1	REQ1.2	System login: valid email and password.	1. Open system. 2. Select "Login" option. 3. Enter email. 4. Enter password. 5. Click "Submit".	Email: "test@email.com" Password: "testPassword1"	Login accepted and user can enter the system.			
T1.2	REQ1.2	System login: valid email, invalid password.	1. Open system. 2. Select "Login" option. 3. Enter email. 4. Enter password. 5. Click "Submit".	Email: "test@email.com" Password: "Password"	Login rejected and the user is told incorrect password, to try again or signup for an account.			
T1.3	REQ1.2	System login: invalid account.	1. Open system. 2. Select "Login" option. 3. Enter email. 4. Enter password. 5. Click "Submit".	Email: "fake@email.com" Password: "	Login is rejected and the user is asked to try again or sign up.			
T2.1	REQ1.3	System sign-up: valid attempt.	1. Open system. 2. Select "Sign Up". 3. Enter email. 4. Enter password. 5. Select "Submit".	Email: "test2@email.com" Password: "newPassword2"	The user should be shown a confirmation message and then be logged into the system with the new account.			
T2.2	REQ1.3	System sign-up: invalid attempt.	1. Open system. 2. Select "Sign Up". 3. Enter email. 4. Enter password. 5. Select "Submit".	Email: "testinput" Password: "testPassword"	A message shown stating the email is of an incorrect form and the user should try again.			
T3.1	REQ2.2	House searching/selection: valid address.	1. Go to house search page. 2. Enter address. 3. Click "Search". 4. Select the first house given.	Address: "AB12 CDE"	The system should show 3 houses with the address AB12 CDE.			
T3.2	REQ2.2	House searching/selection: empty address.	1. Go to house search page. 2. Enter address. 3. Click "Search". 4. Select the first house given.	Address: "JK56 LMN"	The system should tell the user that there are no properties at this location and prompt them to attempt the search again.			
T3.3	REQ2.2	House searching/selection: invalid address.	1. Go to house search page. 2. Enter address. 3. Click search.	Address: "is this an address?"	Search is rejected and the user is told to try again.			
T4.1	REQ3.4	Purchasing a house: valid account, valid funds.	1. Go to house search page. 2. Input address. 3. Select "Search". 4. Select the first property from list. 5. Select "Purchase". 6. Input account number. 7. Confirm purchase.	Address: "AB12 CDE" Account Number: 1111 2222 3333 4444	The user should be greeted with a screen showing the details of the property, as well as receiving an invoice sent to their email. The balance of the account should now be £100,000.			
T4.2	REQ3.4	Purchasing a house: valid account, invalid funds.	1. Go to house search page. 2. Input address. 3. Select "Search". 4. Select the first property from list. 5. Select "Purchase". 6. Input account number. 7. Confirm purchase.	Address: "XY99 ZZ2" Account Number: 5555 6666 7777 8888	The transaction should be rejected due to invalid funds and the user should be told so. Funds of the account should not be altered (remained at £300,000).			
T4.3	REQ3.4	Purchasing a house: invalid account.	1. Go to house search page. 2. Input address. 3. Select "Search". 4. Select the first property from list. 5. Select "Purchase". 6. Input account number. 7. Confirm purchase.	Address: "AB12 CDE" Account Number: 0000 0000 0000 0000	The transaction should be rejected due to non-existent account, and the user should be prompted as such to enter an existing account number.			
T5.1	REQ4.1	Start virtual tour: 3D model present.	1. Go to house search page. 2. Enter address. 3. Select "Search". 4. Select first property from list. 5. Select "Virtual Tour". 6. Connect VR headset. 7. Select "Enter Tour".	Address: "AB12 CDE"	User should be taken to the virtual tour view, with a 3D model of the house AB12 CDE.			
T5.2	REQ4.2	Start virtual tour: 3D model absent.	1. Go to house search page. 2. Enter address. 3. Select "Search". 4. Select first property from list. 5. Select "Virtual Tour". 6. Connect VR headset. 7. Select "Enter Tour".	Address: "XY99 ZZ2"	The user should be told that there is no 3D model for this property and be returned to the previous page.			
T6.1	REQ5.1	View house details: valid attempt.	1. Go to house search page. 2. Input address. 3. Select "Search". 4. Select the first property from the list. 5. Select "View Properties".	Address: "AB12 CDE"	The user should be able to see the house has 2 bedrooms, 1 bathroom, CCTV installed and a previous price of £200,000.			
T6.2	REQ5.1	View house details: invalid attempt.	1. Go to house search page. 2. Input address. 3. Select "Search". 4. Select the first property from the list. 5. Select "View Properties".	Address: "XY99 ZZ2"	The user should see a message that the house at the address doesn't have any attributes linked to it, and should be returned to the previous menu.			
T7.1	REQ6.1	House customisation: valid attempt.	1. Go to house search page. 2. Input address. 3. Select "Search". 4. Select the first property from the list. 5. Select "Customise".	Address: "AB12 CDE"	The user should be brought into an AR view of the property, where they can select actions to customise the house. Options to customise elements of the house should be visible and selectable.			
T7.2	REQ6.1	House customisation: invalid attempt.	1. Go to house search page. 2. Input address. 3. Select "Search". 4. Select the first property from the list. 5. Select "Customise".	Address: "XY99 ZZ2"	The user should be shown an error that the required 3D model is not present, and then be returned to the previous menu.			
T8.1	NFREQ2.1	Virtual tour load time	1. Go to house search page. 2. Enter address. 3. Select "Search". 4. Select first property from list. 5. Select "Virtual Tour". 6. Connect VR headset. 7. Select "Enter Tour".	Address: "AB12 CDE"	The user should be loaded into the virtual tour within 10 seconds of selecting the "Enter Tour" button.			
Tb.1	NFREQ3.1	User navigation	1. Open system. 2. Navigate to the details page of a property with address given.	Address: "AB12 CDE"	The user should be able to navigate to the details page of the property without checking the help document more than once.			
Tc.1	NFREQ2.5	Server performance	Have 10+ users perform the following actions: 1. Go to house search page. 2. Input address. 3. Select "Search". 4. Select the first property from the list. 5. Select "View Properties".	Address: "AB12 CDE"	The system should be able to sustain multiple users accessing the same data at once, so each client should have loading times of <3 seconds.			

Exit Criteria

No functional tests should fail, with all other tests having a 95% pass rate. There should also be a very minimal amount of usability concerns brought up with the system.

Usability and Prototyping

We created 6 screens to be used in the video for our system. These were based off of a current real estate company's website:



Screen 1



Select an Option :

[Virtual tour](#)

[Customise house](#)

[View properties](#)

[Buy now](#)

HouseID: 311

£580,000

5 Bedrooms Villa in Birmingham

B29 , Selly Oak, Birmingham, England

[Log out](#)

Screen 2



Screen 3



Screen 4

A screenshot of a real estate listing for HouseID: 17. The listing includes details: 4 bedrooms, 3 bathrooms, and a price of 580,000£. It also includes links to "View securities", "View facilities", and "View related". The main image shows a modern living room with a grey sofa, a large abstract painting, and a dining area with a wooden table and chairs. A "Back" button is located in the bottom right corner of the image.

Screen 5

Amount to pay : 580,000£

Back

Enter card details

Card number

Accepted credit and debit card types



Expiry date

For example, 10/20

Month Year

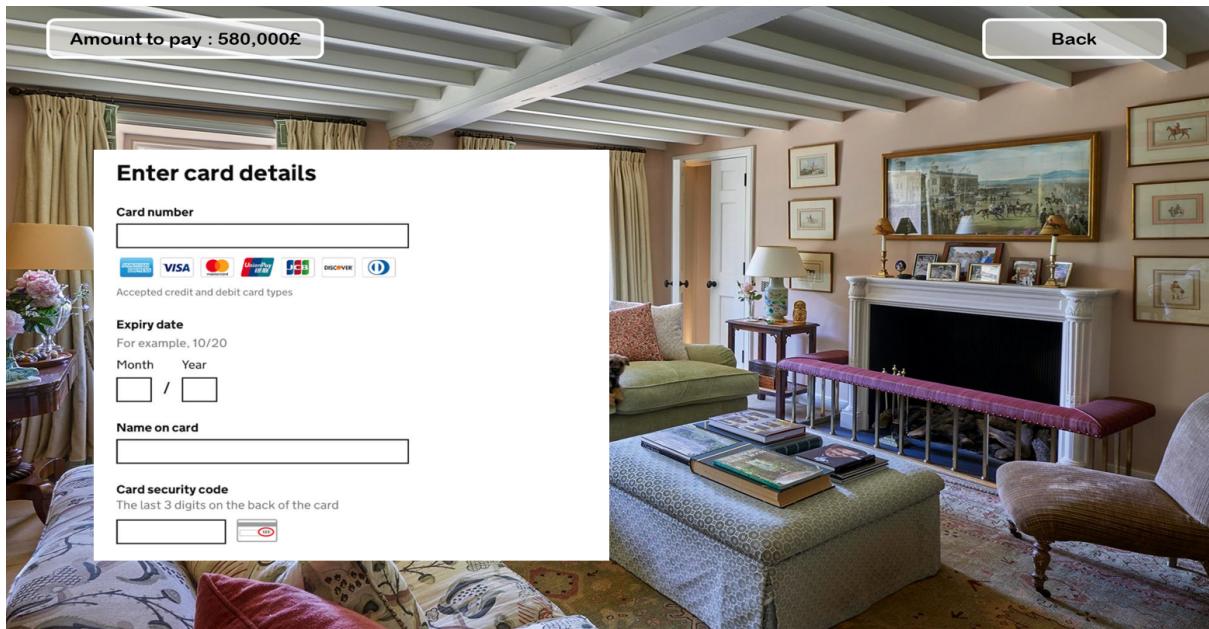
/

Name on card

Card security code

The last 3 digits on the back of the card





Screen 6

Ethics & Professional Practice

Our system has carefully abided to the ACM Code Of Ethics to provide an ethical & responsible design

ACM1: General Ethical Principles

ACM1.1:

- Our system helps contribute to society and minimises negative effects, by giving house viewing opportunities to those who may have not had the opportunity

ACM1.2:

- Our system minimises the risk of harm
- Health & Safety warnings on the use of VR headsets are prompted to such users

ACM1.3:

- Full transparency of our system & how user data is used is given to users

ACM1.4:

- Our system takes actions in not to discriminate against ethnicity, disability, age, gender identity race, religion or belief & sexual orientation ect.

ACM1.5:

- Efforts have been made to respect the work required to make new ideas e.g., our payment infrastructure is handled externally

ACM1.6:

- Only the minimum amount of personal information is needed to use our system
- Transparency on how user data is being used is given to all users

ACM1.7:

- Confidential user information is kept secure and can only be accessed by the user
- When our system needs to access personal information (e.g., bank details) the user needs to consent

ACM2: Professional Responsibilities

ACM2.1:

- The work for our system is respected
- Producing high quality software for the metaverse is promoted

ACM2.2:

- Independent study is promoted to continue maintaining a high standard of competence

ACM2.3:

- Rules outside of the ACM are also respected (regional & international laws etc.)
- Any rules or laws broken in regard to our system can face repercussions for their actions

ACM2.4:

- Our system will continually be provided with professional reviews to promote further innovation

ACM2.5:

- Any potential risks in regard to VR and our system will be carefully evaluated at each software lifecycle stage

ACM2.6:

- To maintain high quality software, only individuals qualified in VR systems can work on our system

ACM2.7:

- The aid which VR brings and its implications are made aware to our users

ACM2.8:

- Individuals who no longer user our system can request their data to be deleted

ACM2.9:

- The security of our system is kept at one of the highest priority to keep our users faith in us

ACM3: Professional Leadership Principles

ACM3.1:

- Our users are always our central concern for our system

ACM3.2:

- Social responsibility is made clear to members of our system

ACM3.3:

- The quality of life when enhancing our system will be constantly improved

ACM3.4:

- Compliance with ACM Codes will be recognised and therefore rewarded

ACM3.5:

- Opportunities to learn & grow is promoted to ensure our members grow professionally

ACM3.6:

- Retiring systems will be made clear to our users
- Our system will clearly communicate to our users when outdated systems will no longer be supported

ACM3.7:

- There is continual monitoring on how users use our system so we can keep up with any ethical or legal obligations

ACM4: Compliance With Code

ACM4.1:

- Our compliance with ACM will be promoted to others (investors/users ect.)

ACM4.2:

- If such ACM Code principle is broken, it will be reported