21/2/21 **EXP : 3**	**MINI PROJECT** **USE CASE MODEL**	**TEAM NO: 5**

Resource Used:

1) Equipment: ArgoUML

Theory:

Actors:

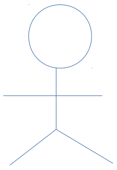
Actors represent anyone or anything that interact with the system. An actor may

- Only input information to a system
- Only retrieve information from a system
- Both input and retrieve information to and from a system

Typically, the actors are found in the problem statement, and also from conversation with the customers and domain experts. There are three types of actors:

1. users of the system,
2. external application systems, and
3. external devices that can independently interact with the system.

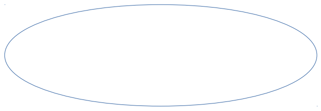
In UML, an actor is represented stickman symbol, as shown below:



Use cases:

Use cases eventually map to the menu option. Use cases represent the functionality provided by the system. Each individual functionality provided by a system is captured as a use case. A use case thus represents a dialog between an actor and the system. A collection of use cases for a system reflects all the defined ways in which a system can be used. A use case can be defined as a sequence of transactions performed by a system, that yields a measurable result of values for a particular actor.

In UML, a use case is represented as an oval, as shown below:

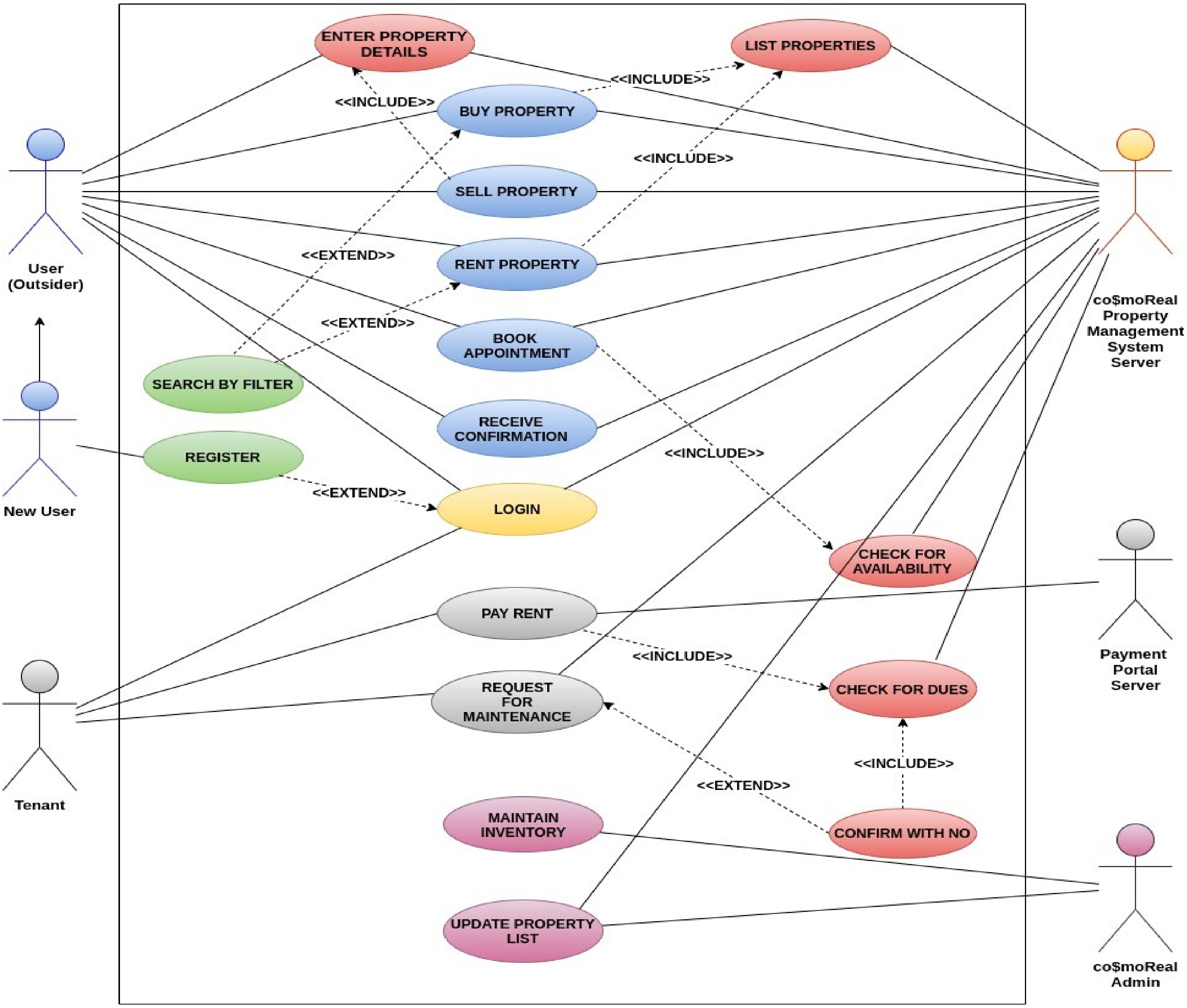


Use case Diagram:

A use case diagram is a interaction view of a some or all the actors, use cases and their interactions identified for a system. Each system typically has a main use case diagram, which is a picture of the system boundary (actors) and the major functionality provided by the system (use cases). Other use case diagrams may be created as needed. Some examples are:

- A diagram showing all the use cases for a selected actor.
- A diagram showing all the use cases being implemented in an interaction.
- A diagram showing all the use cases and all its relationship.

UML Diagram for Property Management System:



Actors:

1. General user
2. Tenant
3. co$moReal Property Management Server
4. co$moReal Admin
5. Payment Portal Server

Use Cases:

1. Login
2. Buy Property
3. Sell Property
4. Rent Property
5. Book Appointment
6. Receive Confirmation
7. Pay Rent
8. Request for Maintenance
9. Maintain Inventory
10. Update Property List

Fully Dressed Use Case Model:

1. **Login:** Registered users enter login credentials to get into the server.

- **Primary Actor:** General user/Tenant
- **Stakeholders/Interests:**
  - Users: can login to the server to avail the services provided
  - Server: Authenticates the users
- **Preconditions:**
  - Users should have access to the internet.
- **Post Conditions:**
  - Users are authenticated and have access to the server.
- **Basic Flow:**
  - Customers access the internet and enter the co$moReal Website.
  - They Login and get access to the server after getting authenticated.
- **Special Requirements:**
  - User authorization response within 30 seconds 90% of the time.
  - Language internationalization on the text displayed.
- **Technology and Data variations List:**
  - Login Credentials entered via a keyboard.
- **Frequency of Occurence:**
  - Almost Continuous
- **Open Issues:**
  - What Customization is required for users from different regions.

2. **Buy Property:** Displays a list of properties available to buy. Users can give inputs like the location of the property, price range, Type of property(apartments,villa,land), Number of BHK, Purpose(Commercial/ Residential) , Used or New.

- **Primary Actor:** General user
- **Stakeholders/Interests:**
  - Users: can view a list of co$moReal properties on sale. They can search for the properties by filtering the properties available based on different categories like price range,location etc. And book appointment with co$moReal to futher buy the property.
  - co$moReal: can make sales upon success.
  - Server: Displays list of properties(with or without filters)
- **Preconditions:**
  - Users should have access to the internet.
  - Users should be registered to view the properties.
  - Users should be authenticated.
- **Post Conditions:**
  - Users can look for a property of their interest and proceed with legal steps to buy it.
- **Basic Flow:**
  - Customers access the internet and enter the co$moReal Website.
  - They Login and get access to the server after getting authenticated.
  - They can view the list of properties on sale and select the property of their interest and proceed further to buy it.
- **Special Requirements:**
  - The Page shout auto reload every 5 minutes
  - Language internationalization on the text displayed.
- **Technology and Data variations List:**
  - Property filters to be entered via a keyboard.
- **Frequency of Occurence:**
  - Almost Continuous
- **Open Issues:**
  - What Customization is required for users from different regions.

3. **Sell Property:** Allows users to provide information about properties that they have to sell. If the owners are interested in the property,they will buy it and add it to their own list of properties.

- **Primary Actor:** General user
- **Stakeholders/Interests:**
  - Users: can sell a property of theirs to co$moreal.

    - ■ co$moReal: can potentially buy a property to further expand their list of properties.

- **Preconditions:**
    - ■ Users should have access to the internet.
    - ■ Users should be registered to view the properties.
    - ■ Users should be authenticated.
- **Post Conditions:**
    - ■ Users can enter details of a property they wish to sell to co$moReal and if co$moreal is interested they will reach out to the particular user.
- **Basic Flow:**
    - ■ Customers access the internet and enter the co$moReal Website.
    - ■ They Login and get access to the server after getting authenticated.
    - ■ They can enter details of the property they want to sell.
- **Special Requirements:**
    - ■ The Page shout auto reload every 5 minutes
    - ■ Language internationalization on the text displayed.
- **Technology and Data variations List:**
    - ■ Property details to be entered via a keyboard.
- **Frequency of Occurence:**
    - ■ Almost Continuous
- **Open Issues:**
    - ■ Where should the property be located at?

4. **Rent Property:** Display a list of properties available to rent. Users can give inputs like the location of the property, price range, Type of property(apartments,villa,land), Number of BHK, Purpose(Commercial/ Residential) , Used or New.

- **Primary Actor:** General user
- **Stakeholders/Interests:**
    - ■ Users: can view a list of co$moReal properties for rent. They can search for the properties by filtering the properties available based on different categories like price range,location etc.
    - ■ co$moReal: gets monthly payments from the potential tenants upon success.
    - ■ Server: Displays list of properties(with or without filters)
- **Preconditions:**
    - ■ Users should have access to the internet.
    - ■ Users should be registered to view the properties.
    - ■ Users should be authenticated.
- **Post Conditions:**

    - Users can look for a property of their interest and proceed with legal steps to rent it.
- **Basic Flow:**
    - Customers access the internet and enter the co$moReal Website.
    - They Login and get access to the server after getting authenticated.
    - They can view the list of properties on sale and select the property of their interest and proceed further to rent it.
- **Special Requirements:**
    - The Page shout auto reload every 5 minutes
    - Language internationalization on the text displayed.
- **Technology and Data variations List:**
    - Property filters to be entered via a keyboard.
- **Frequency of Occurence:**
    - Almost Continuous
- **Open Issues:**
    - What Customization is required for users from different regions.

5. **Book Appointment:** User can book an appointment with the co$moReal Manager from the list of dates available.

- **Primary Actor:** General user
- **Stakeholders/Interests:**
    - Users: can book an appointment with co$moReal Manager to discuss and buy/sell/rent a property.
    - co$moReal: can provide great customer services upon successful booking of an appointment.
    - Server: Displays a list of available dates.
- **Preconditions:**
    - Users should have access to the internet.
    - Users should be authenticated.
- **Post Conditions:**
    - Users can choose a date from the list of available dates.
- **Basic Flow:**
    - Users access the internet and enter the co$moReal Website.
    - They Login and get access to the server after getting authenticated.
    - They can view the list of available dates and select a date to book an appointment.
- **Special Requirements:**
    - The Page shout auto reload every 5 minutes
    - Language internationalization on the text displayed.
- **Technology and Data variations List:**
    - Date to be chosen from the dropdown with the help of a mouse pointer

- **Frequency of Occurence:**
    - ■ Almost Continuous
- **Open Issues:**
    - ■ What if more than one user book an appointment at the same time?

6. **Receive Confirmation:** User receives confirmation on successful booking of an appointment.

- **Primary Actor:** General user

- **Stakeholders/Interests:**
    - ■ Users: receive a confirmation upon successful booking of an appointment with co$moReal Manager to discuss and buy/sell/rent a property.
    - ■ co$moReal: can provide great customer services upon successful booking of an appointment.
    - ■ Server: provides confirmation.
- **Preconditions:**
    - ■ Users should have access to the internet.
    - ■ Users should be authenticated.
- **Post Conditions:**
    - ■ Users receive confirmation of their appointment booked.
- **Basic Flow:**
    - ■ Users access the internet and enter the co$moReal Website.
    - ■ They Login and get access to the server after getting authenticated.
    - ■ They can view the list of available dates and select a date to book an appointment.
    - ■ They receive confirmation of their appointment.
- **Special Requirements:**
    - ■ The Page shout auto reload every 5 minutes
    - ■ Language internationalization on the text displayed.
- **Technology and Data variations List:**
    - ■ A digital copy of the appointment booked.
- **Frequency of Occurence:**
    - ■ Almost Continuous
- **Open Issues:**
    - ■ What if more than one user book an appointment at the same time?
    - ■ What if the booking fails?

7. **Pay Rent:** Allows the user to submit a new payment. It will have inputs for the payment amount, the date on which to pay, the user’s name and the user’s bank account information.

- **Primary Actor:** Tenant

- **Stakeholders/Interests:**
  - ■ Tenants: can pay their rent.
  - ■ Payment Portal: will connect the server with the tenant's bank.
  - ■ co$moReal: Their bank account will be credited with the rent amount.
  - ■ Server: Calculates the amount to pay.
- **Preconditions:**
  - ■ Tenants should have access to the internet.
  - ■ Tenants should be authenticated.
- **Post Conditions:**
  - ■ Tenants pay their rent.
  - ■ co$moReal receives the rent amount.
- **Basic Flow:**
  - ■ Tenants access the internet and enter the co$moReal Website.
  - ■ They Login and get access to the server after getting authenticated.
  - ■ They pay their rent with debit card.
  - ■ They are redirected to a payment portal where they enter their card details.
  - ■ On success, co$moReal receives the payment.
- **Special Requirements:**
  - ■ The Page shouldn't be refreshed while in the payment portal.
  - ■ Language internationalization on the text displayed.
- **Technology and Data variations List:**
  - ■ Card details entered via keyboard
  - ■ CVV number entered and OTP authorization
- **Frequency of Occurence:**
  - ■ Almost Continuous.
- **Open Issues:**
  - ■ What if the amount is debited but nor received by co$moReal?

8. **Maintenance Request:** Allows the user to submit a new maintenance request. This form will allow them to describe the issue and indicate whether maintenance personnel have permission to enter the complex.

- **Primary Actor:** Tenant

- **Stakeholders/Interests:**
  - ■ Tenants: can request for maintenance.
  - ■ Server: will check for any rent dues. If none,then maintenance request is accepted.
  - ■ co$moReal: will receive maintenance charge, if applied.
- **Preconditions:**
  - ■ Tenants should have access to the internet.

    - Tenants should be authenticated.
- **Post Conditions:**
    - Tenant’s maintenance request is accepted.
- **Basic Flow:**
    - Tenants access the internet and enter the co$moReal Website.
    - They Login and get access to the server after getting authenticated.
    - They request for maintenance.
    - Server checks for rent dues.
    - On success, tenant’s request is accepted.
- **Special Requirements:**
    - The Page should auto reload every 5 minutes.
    - Language internationalization on the text displayed.
- **Technology and Data variations List:**
    - Maintenance Issue filled out via keyboard.
- **Frequency of Occurence:**
    - Almost Continuous.
- **Open Issues:**
    - What if there are rent dues?

9. **Maintain Inventory:** co$moReal Admin maintains property and customer data.

- **Primary Actor:** Admin

- **Stakeholders/Interests:**
    - Admin: Maintains all the users,tenants’ and property details.
    - co$moReal: can maintain all their buying/selling/renting of properties.
- **Preconditions:**
    - Admin should be registered and authenticated.
- **Post Conditions:**
    - Admin collects and stores all the data which might prove useful later.
- **Basic Flow:**
    - Admin enters the server with internet access.
    - Admin redirects all the data on the site to a stroage device on a 6 month basis
- **Special Requirements:**
    - The Page should be refreshed every 5 minutes.
    - Language internationalization on the text displayed.
- **Technology and Data variations List:**
    - Storage devices to store all the data on the website.
- **Frequency of Occurence:**
    - Almost Continuous.
- **Open Issues:**

- What if some of the data get lost?

10. **Update Property List:** The Admin updates the list of properties for sale and rent.

- **Primary Actor:** Admin
- **Stakeholders/Interests:**
  - Admin: Maintains all the property details.
  - co$moReal: can maintain all their buying/selling/renting of properties.
- **Preconditions:**
  - Admin should be registered and authenticated.
- **Post Conditions:**
  - Admin updates the list of properties on sale or for rent.
- **Basic Flow:**
  - Admin enters the server with internet access.
  - Admin updates all the properties on sale and for rent.
- **Special Requirements:**
  - The Page should be refreshed every 5 minutes.
  - The updation should occur then and there.
  - Language internationalization on the text displayed.
- **Technology and Data variations List:**
  - Admin gets data from a storage device/datacenter and updates the list of properties.
- **Frequency of Occurence:**
  - Almost Continuous.
- **Open Issues:**
  - What if some of the data get lost?

Appendix:

1. **Stereotype:** Stereotypes defines a new model element in terms of another model element. It is represented by <<stereotypes>>

2. **System boundary boxes:** We can draw a rectangle around the use cases, called the System boundary box, to indicate the scope of your system.

3. **Abstract use case:** Use case, which is inherited, by some use case is called as abstract use case.

4. **Concrete use case:** Use case, which is directly inherited by actor, is called as concrete use case.

Conclusion:

Use cases define the behavior provided by the system. They are a central theme for the entire development process. They play a role in a various stages of the development process as follows:

- During requirements determination, they specify what the system should do from the user’s point of view.
- During analysis and design, the use cases are realized in a design model .Use case realizations describe the interaction between various objects in the design model.
- During implementation, the design model becomes the implementation specification.
- During testing, the use cases are the basis for identifying test cases. The system is verified by performing each use case.
- As part of project management, they provide a basis for planning the iterations.