


***A signed copy of this form must be submitted with every assignment.  
If the statement is missing your work may not be marked.  
For Level 7 assignments, please use the separate Level 7 Candidate  
Statement of Own Work form instead of this one.***

**Student Declaration**

I confirm the following details:

<b>Candidate Name:</b>	Ishan Bapardekar
<b>Candidate ID Number:</b>	213062
<b>Qualification:</b>	L4DC
<b>Unit:</b>	OOSAD
<b>Centre:</b>	FUT004
<b>Word Count:</b>	2159
<p>I have read and understood both NCC Education's <i>Academic Misconduct Policy</i> and the <i>Referencing and Bibliographies</i> document. To the best of my knowledge my work has been accurately referenced and all sources cited correctly.</p> <p>I confirm that I have not exceeded the stipulated word limit by more than 10%.</p> <p>I confirm that this is my own work and that I have not colluded or plagiarized any part of it.</p>	
<b>Candidate Signature:</b>	
<b>Date:</b>	12/05/2025

# Table of content

Task 1: Requirements Elicitation.....	03
Four stakeholders in the PVS system with justification.....	03
Two methods for requirements elicitation.....	04
Five additional pieces of information needed.....	04
Task 2: Requirements Specification.....	05
Difference between functional and nonfunctional requirements.....	05
Functional and nonfunctional Requirements.....	05
Task 3: Use Case Model.....	07
Purpose of a Use Case Model.....	07
Identifying Actors.....	08
use case diagram .....	09
Scenarios for "Make Appointment" Use Case.....	10
Use Case Documentation for "Make Appointment".....	10
Task 4: Sequence Diagram.....	11
Purpose and Benefits of a Sequence Diagram.....	11
Sequence diagram.....	12
Task 5: Class Diagram.....	13
Class Diagram for the system.....	13
Justification for Selected Classes.....	14
References.....	15

## Task 1: Requirements Elicitation

### a) Four stakeholders in the PVS system with justification:

Stakeholders	Justification
1. Branch Managers	<ul style="list-style-type: none"><li>They are the one to run each office location and oversee all operations. They need tools to monitor sales performance, manage staff schedules, and handle branch administration. They're key decision-makers who need higher-level system access.</li></ul>
2. Sales Representatives	<ul style="list-style-type: none"><li>The sales employee handles the day-to-day client interactions, property showings, and sales processes. They need quick access to property details, scheduling tools, and client information while potentially working from various locations.</li></ul>
3. Clients (Buyers and Sellers)	<ul style="list-style-type: none"><li>The customers using NYPS services have different needs - sellers want their properties marketed effectively while buyers need to find suitable properties and arrange viewings. Their experience determines NYPS's business success.</li></ul>
4. IT Department	<ul style="list-style-type: none"><li>The technical is team responsible for system maintenance, security updates, and user support. They need to ensure that the system runs smoothly, integrates with other business systems, and meets security requirements.</li></ul>

Table: 1.1 Stakeholders of PVS System

### b) Two methods for requirements elicitation:

Methods	Elicitation
Interview	<ul style="list-style-type: none"><li>Direct communication with key stakeholders (e.g., buyers, sellers, and managers) allows for in-depth discussions.</li><li>Helps uncover specific expectations, pain points, and workflow requirements.</li><li>Allows follow-up questions to clarify ambiguities in the problem statement.</li></ul> <p>Example: Conduct one-on-one interviews with buyers to understand their property search behaviour and appointment booking needs.</p>

<b>Use Case Workshops</b>	<ul style="list-style-type: none"> <li>Engages multiple stakeholders in brainstorming system features and interactions.</li> <li>Helps identify missing functionalities by walking through real-world scenarios.</li> <li>Encourages collaboration and ensures a shared understanding of requirements.</li> </ul> <p>Example Usage: A workshop with sellers and representatives to model the process of property listing, appointment scheduling, and confirmation.</p>
---------------------------	---

Table :1.2 methods of elicitation

### c) Five additional pieces of information needed:

<b>Additional information</b>	<b>Justification</b>
<b>1. Current system details</b>	<ul style="list-style-type: none"> <li>What methods are currently used to manage properties, appointments and client information? Understanding existing processes helps identify pain points and ensures the new system addresses actual problems rather than creating new ones.</li> </ul>
<b>2. Data retention policies</b>	<ul style="list-style-type: none"> <li>How long should client information be kept? Are there legal requirements for storing property transaction data? Clear understanding of data lifecycle requirements affects database design and storage planning.</li> </ul>
<b>3. After-hours access requirements</b>	<ul style="list-style-type: none"> <li>Do representatives need system access outside business hours when meeting clients? Does the system need 24/7 availability for client browsing? This impacts system availability requirements and mobile access design.</li> </ul>
<b>4. Marketing integration needs</b>	<ul style="list-style-type: none"> <li>How should the system connect with property advertising platforms? Does NYPS list properties on external websites, and if so, how should data flow between systems? This affects integration planning and API requirements.</li> </ul>
<b>5. Performance metrics</b>	<ul style="list-style-type: none"> <li>What key performance indicators does NYPS track? Understanding what metrics matter to the business helps design reporting features that deliver actionable insights for management.</li> </ul>

Table 1.3 additional information

## Task 2: Requirements Specification

### a) Difference between functional and nonfunctional requirements

#### 1) Functional requirements

Functional Requirements define what the system should do, describing the specific features and operations it must perform. These requirements ensure that the system provides the necessary services to users.  
Sommerville, I. (2011).

**Example from PVS:**

The system must allow buyers to book property viewings with sellers.

#### 2) Non-Functional Requirements:

Non-Functional Requirements specify how the system should perform rather than what it does. These cover aspects like performance, security, and usability.  
Sommerville, I. (2011).

**Example from PVS:**

The system should respond to booking requests within 2 seconds to ensure a seamless user experience.

### b) Functional Requirements

Functional Requirements	justification
User Registration & Authentication	Users (Buyers, Sellers, Managers, and Representatives) must be able to create accounts and log in securely.
Property Listing Management	Sellers must be able to add, edit, and remove property listings.
Search and Filter Properties	Buyers must be able to search for properties based on location, price, and property type.
Make an Appointment	Buyers must be able to schedule property viewings with sellers.
Confirm/Decline Appointments	Sellers must be able to approve or decline scheduled viewings.
View Appointment Schedule	Buyers and sellers must be able to view upcoming and past appointments.
Manage User Profiles	Users must be able to update their personal details and preferences.
Send Notifications	The system must notify buyers and sellers of appointment confirmations, cancellations, and reminders.
Generate Reports	Managers must be able to generate system usage and appointment reports.

<b>Secure Payment Handling (if applicable)</b>	If the system supports paid bookings, it must process payments securely.
<b>Appointment Cancellation &amp; Rescheduling</b>	Buyers must be able to modify appointments, ensuring flexibility.

Table:2.1 functional requirements

## b) Non-Functional Requirements

<b>Nonfunctional requirements</b>	<b>Justification</b>
<b>Performance</b>	The system must handle at least 100 concurrent users without performance degradation.
<b>Usability</b>	The system should have an intuitive interface for easy navigation.
<b>Security</b>	User data must be encrypted and protected against unauthorized access.
<b>Scalability</b>	The system should support future expansion, including more users and properties.
<b>Availability</b>	The system should maintain <b>99.9% uptime</b> to ensure accessibility.
<b>Compliance</b>	The system must comply with GDPR and data protection regulations.
<b>Device Compatibility</b>	The system must be accessible from mobile and desktop devices.
<b>Logging &amp; Auditing</b>	All critical actions must be logged for auditing purposes.
<b>Backup &amp; Recovery</b>	The system must perform <b>daily backups</b> to prevent data loss.
<b>Multi-Language Support</b>	The system should support multiple languages to cater to a wider audience.

Table: 2.2 nonfunctional requirements

## TASK 3: Use case

### A) Purpose of a Use Case Model

A use case model is a graphical model that explains how different users, or actors, utilize a system to achieve some goals. It acts as a link between system requirements and the design process and gives an overview at a high level of user expectations. The primary applications of use case model are:

**Requirement Analysis** – It is utilized to specify and define functional requirements by understanding how the users use the system.

**Communication Tool** – It serves as a shared point of reference for the stakeholders, including clients, developers, and testers, so that they will be aligned and clear.

**System Design and Development** – It provides the foundation on which the system architecture can be designed and its features implemented.

**Testing and Validation** – It facilitates the creation of test cases that reflect real user behavior, so that the system lives up to expectations.

Cockburn, A. (2001).

---

#### **Benefits of a Sequence Diagram in the Property Viewing System (PVS)**

Using a **Sequence Diagram** for the **Property Viewing System (PVS)** provides several advantages, ensuring a clear and efficient appointment scheduling process. Below are the key benefits:

##### **1. Clear Visualization of Appointment Scheduling Process**

- The diagram provides a step-by-step breakdown of how a buyer inquiry about properties, books an appointment, and provides feedback.
- It helps in identifying missing steps or redundant interactions, ensuring a smooth workflow.

##### **2. Improved System Design & Development**

- Helps developers understand how different components interact (e.g., Property Management System, Appointment Scheduler System).
- Ensures that all interactions are well-structured, reducing **errors and rework** during implementation.

##### **3. Better Communication Between Stakeholders**

- Helps developers, business analysts, and managers understand the system's requirements briefly.
- Reduces miscommunication by providing a visual representation of the appointment process.

##### **4. Enhanced User Experience for Clients & Employees**

- Ensures efficient appointment scheduling, minimizing delays for buyers and representatives.
- Helps in designing a user-friendly and responsive system that quickly provides property details and available time slots.

## 5. Facilitates Debugging and Testing

- The diagram acts as a blueprint for testers, helping them create test cases for different appointment scenarios.
- Makes it easier to trace errors in the system by following the interaction flow.

## 6. Supports Future System Expansion

- If NYPS expands its services (e.g., online property booking, virtual tours), the sequence diagram can be easily modified to include new features.
- Ensures scalability by clearly defining system boundaries and interactions.

## 7. Efficient Managerial Oversight

- The Manager oversees the appointment process and reviews feedback, helping in business decision-making.
- Ensures the company maintains high service quality based on buyer feedback and appointment success rates.

Booch, G., Rumbaugh, J., & Jacobson, I. (2005).

## b) Identifying Actors

Actors	Justification	Role in system
<b>Buyer</b>	The <b>Buyer</b> is a key user as they drive sales and interactions.	The primary user who search for products, makes purchases, books appointments, and tracks order progress.
<b>Seller</b>	The <b>Seller</b> is crucial as they provide the products or services.	A secondary user who manages products, responds to buyer inquiries, and keeps track of sale.
<b>Manager</b>	The <b>Manager</b> ensures smooth platform operations and user management.	Manages platform operations, manages user accounts, and resolves system issues.
<b>Representative</b>	The <b>Representative</b> plays a support role, assisting with queries and issues.	provides customer service, resolves disputes, and communicates with buyers and sellers.

Table: 3.1 identifying actor



### c) Use case diagram for PSV

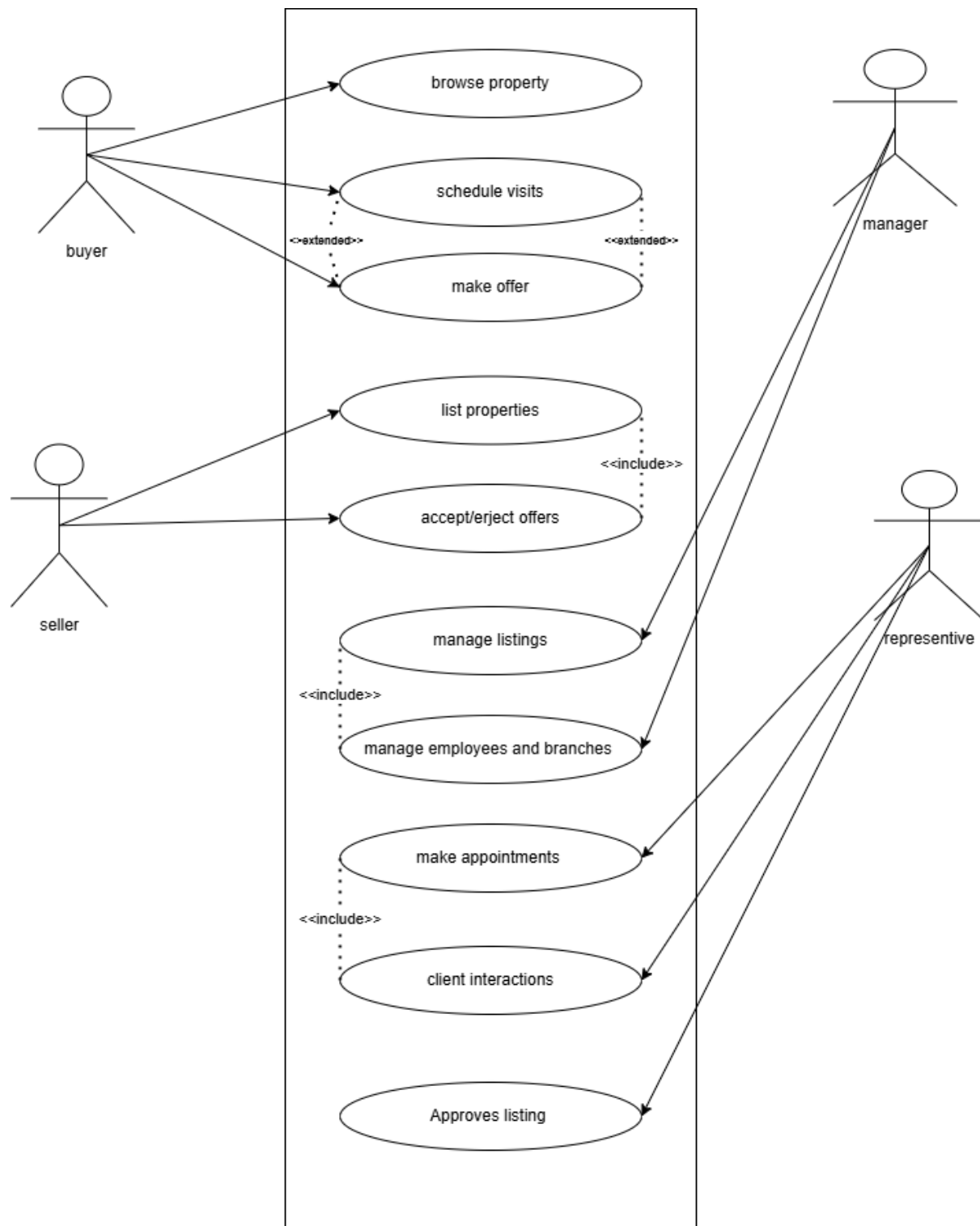


Fig 3.1 use case diagram for pvs

#### d) Scenarios for "Make Appointment" Use Case

Scenario 1: Buyer Schedules an Appointment with a Seller	Scenario 2: Buyer Cancels an Appointment
<ul style="list-style-type: none"> <li>Buyer selects a date and time for consultation.</li> </ul>	<ul style="list-style-type: none"> <li>Buyer accesses the scheduled appointments section.</li> </ul>
<ul style="list-style-type: none"> <li>System checks the seller's availability.</li> </ul>	<ul style="list-style-type: none"> <li>Buyer selects "Cancel Appointment."</li> </ul>
<ul style="list-style-type: none"> <li>If available, the appointment is confirmed.</li> </ul>	<ul style="list-style-type: none"> <li>The system updates the status and informs the seller.</li> </ul>
<ul style="list-style-type: none"> <li>The system notifies both the buyer and the seller.</li> </ul>	<ul style="list-style-type: none"> <li>The buyer and seller receive a cancellation notification.</li> </ul>
<ul style="list-style-type: none"> <li>Output: appointment is booked</li> </ul>	<ul style="list-style-type: none"> <li>Output: appointment is cancelled</li> </ul>

Table : 3.3 scenarios

#### e) Use Case Documentation for "Make Appointment"

<b>Use case name</b>	Make appointment
<b>Participating actors</b>	Buyer, representative, seller, manager
<b>Entry condition</b>	The buyer is interested in viewing a property and has logged into the system.
<b>Flow of event</b>	<ol style="list-style-type: none"> <li>1. Buyer searches for available properties.</li> <li>2. Buyer selects a property and requests an appointment.</li> <li>3. System notifies the representative of the appointment request.</li> <li>4. Representative reviews availability and confirms a time slot.</li> <li>5. System records the appointment details.</li> <li>6. System sends confirmation to the buyer and representative.</li> <li>7. Buyer attends the scheduled appointment.</li> <li>8. Representative records buyer's feedback after the appointment.</li> </ol>
<b>Exit condition</b>	Appointment is successfully scheduled and stored in the system.
<b>Special requirements</b>	<ul style="list-style-type: none"> <li>- The system must send automatic confirmation emails.</li> <li>- Only registered buyers can request appointments.</li> <li>- Representatives must have available slots before scheduling.</li> </ul>

Table: 3.4 Use case documentation for "Make Appointment"

## Task 4: Sequence Diagram

### A) Purpose and Benefits of a Sequence Diagram

A Sequence Diagram is a UML tool used to represent the interaction between different objects in a system over time. It focuses on the order of messages exchanged between actors and system components, showing how a process unfolds step by step.

#### Benefits in the Development of the Property Viewing System (PVS)

1. Clear Process Visualization – It illustrates how buyers, representatives, and the system interact during the property search, appointment booking, and transaction completion.
2. Identifying System Components – Helps developers understand objects like database, system, representatives are involved in the workflow.
3. Error Prevention & Refinement – By visually mapping out interactions, developers can detect missing steps or potential issues before implementation.
4. Enhanced Communication – Provides a shared understanding between developers, designers, and stakeholders, ensuring the system meets business needs.

#### Example:

In the "Making an Appointment" scenario, the sequence diagram ensures that:

- Buyers can browse properties before requesting an appointment.
- Representatives confirm and schedule appointments in the system.
- The database securely stores all transactions.
- The negotiation and finalization process follows a structured flow

Booch, G., Rumbaugh, J., & Jacobson, I. (2005).

## B) Sequence diagram

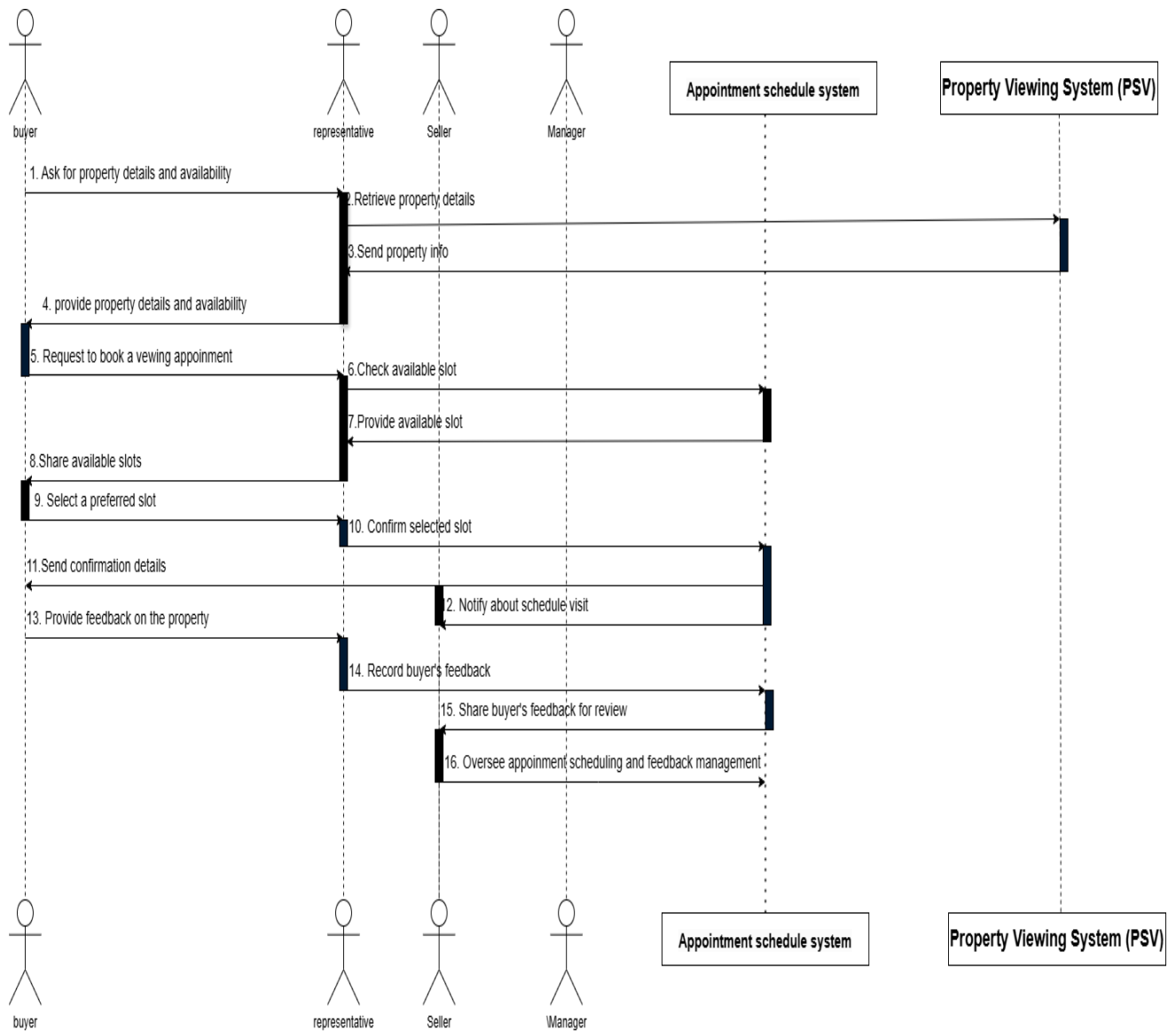


Fig: 4.1 sequence diagram

## Task 5: Class Diagram

### a) Class Diagram for the system.

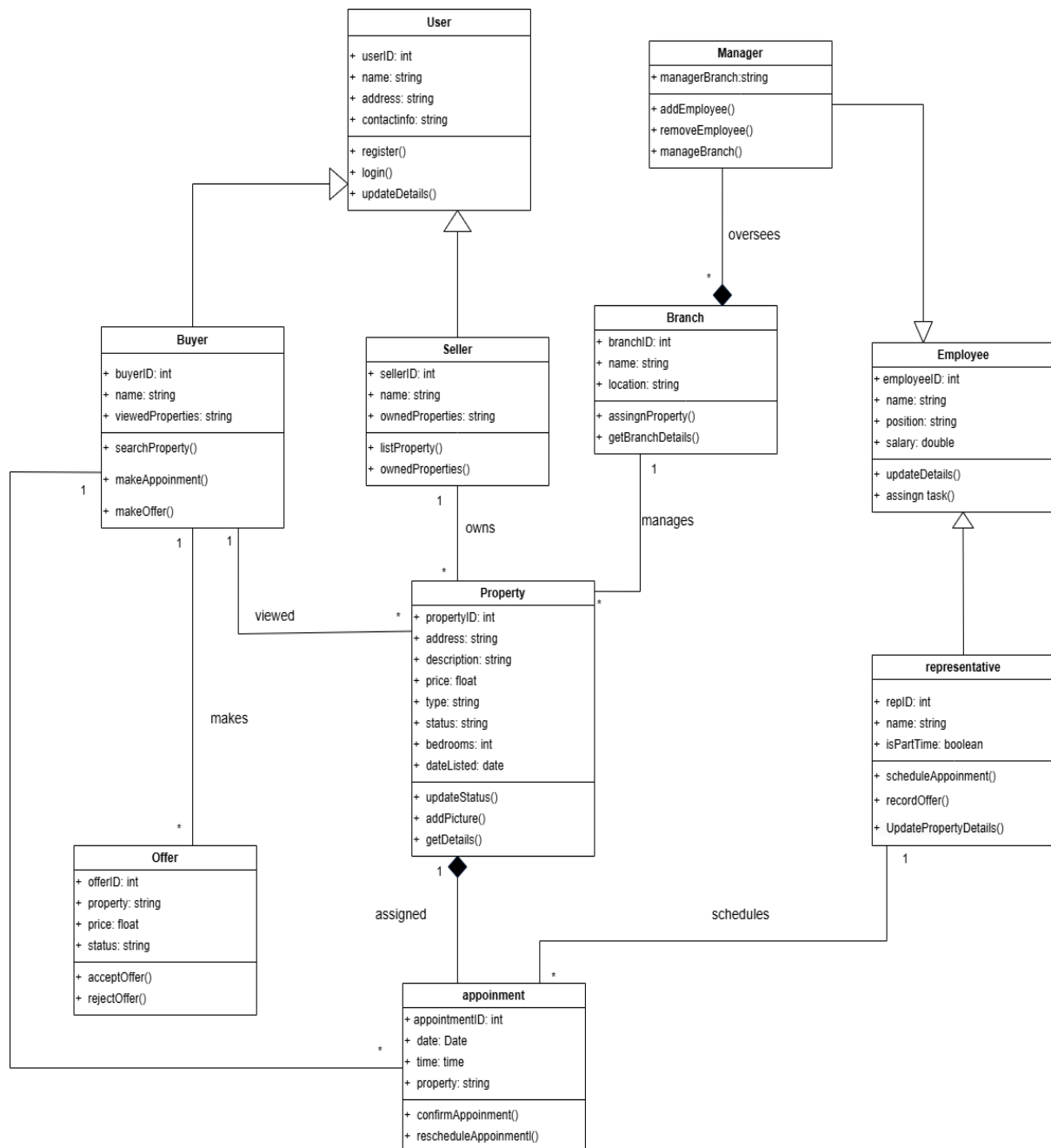


Fig 5.1 class diagram for system

## b) Justification for Selected Classes

Actor	Role
<b>User</b>	<ul style="list-style-type: none"> <li>Represents a general user of the system.</li> <li>Allows role-based access control.</li> <li>Supports inheritance for different user types (e.g., Manager, Buyer, Seller, Employee).</li> </ul>
<b>Manager</b>	<ul style="list-style-type: none"> <li>Has administrative privileges.</li> <li>Can manage employees, property details, and appointments.</li> <li>Plays a key role in overseeing the system's operations.</li> </ul>
<b>Buyer</b>	<ul style="list-style-type: none"> <li>Represents customers interested in purchasing properties.</li> <li>Can browse available properties and request appointments.</li> <li>Can make offers on properties and provide feedback.</li> </ul>
<b>Seller</b>	<ul style="list-style-type: none"> <li>Represents property owners using NYPS services.</li> <li>Can list properties for sale.</li> <li>Can review offers received on listed properties.</li> </ul>
<b>Employee</b>	<ul style="list-style-type: none"> <li>Represents staff members working at NYPS.</li> <li>Includes representatives who handle buyer inquiries and schedule appointments.</li> <li>Has attributes such as employment type (full-time/part-time).</li> </ul>
<b>Representative</b>	<ul style="list-style-type: none"> <li>Specializes in assisting buyers and sellers.</li> <li>Responsible for booking viewings and managing property inquiries.</li> <li>Facilitates communication between buyers and sellers.</li> </ul>
<b>Branch</b>	<ul style="list-style-type: none"> <li>Represents different NYPS locations.</li> <li>Each branch manages multiple properties and staff members.</li> <li>Ensures that property sales are handled at specific branches.</li> </ul>
<b>Property</b>	<ul style="list-style-type: none"> <li>Central to the system, representing the real estate for sale.</li> <li>Contains details such as address, type, price, and status.</li> <li>Linked to a seller and a branch.</li> </ul>
<b>Offer</b>	<ul style="list-style-type: none"> <li>Tracks purchase offers made by buyers.</li> <li>Contains details such as offer price, status (accepted/rejected), and buyer information.</li> <li>Helps in managing property negotiations.</li> </ul>
<b>Appointment</b>	<ul style="list-style-type: none"> <li>Represents scheduled property viewings.</li> <li>Involves a buyer, a representative, and a property.</li> <li>Helps in tracking viewing history and managing appointments.</li> </ul>

Table: 5.1 justification for classes

## Reference:

1. Sommerville, I. (2011). *Software engineering* (9th ed.). Addison-Wesley.
2. Cockburn, A. (2001). *Writing effective use cases*. Addison-Wesley.
3. Booch, G., Rumbaugh, J., & Jacobson, I. (2005). *The unified modelling language user guide* (2nd ed.). Addison-Wesley.
4. Booch, G., Rumbaugh, J., & Jacobson, I. (2005). *The unified modelling language user guide* (2nd ed.). Addison-Wesley.

## Candidate checklist

Please use the following checklist to ensure that your work is ready for submission.

Have you read the NCC Education document *Academic Misconduct Policy* and ensured that you have acknowledged all the sources that you have used in your work? ☐

Have you completed the *Statement and Confirmation of Own Work* form and attached it to your assignment? **You must do this.** ☐

Have you ensured that your work does not contain viruses and can be run directly? ☐