# User Stories

The earliest part of the requirements definition are user stories. Single declarative sentences which define an actor; the behaviour; and the purpose. <Actor> <Verbs> <For reason> is the general form and these are then fleshed out, first into a series of use case diagrams and sequence diagrams, and then into a class requirements document, class diagram and ERD to act as a master document for development.

1. *As a user, I want to be able to log in; log out; and update my account details, so that I can use the system*
2. *As a manager, I want to decide which realtors are responsible for each property, so that I can organise workload*
3. *As a realtor, I schedule showings for each property I am responsible for, so that I can solicit potential buyers*
4. *As a seller, I want to list my house so that I can sell it*
5. *As a buyer, I want to see local houses for sale so that I can buy it*

## User Story 1

*As a user, I want to be able to log in; log out; and update my account details, so that I can use the system*

A grid with a diagram

AI-generated content may be incorrect.

A diagram of a company

AI-generated content may be incorrect.

## User Story 2

*As a manager, I want to decide which realtors are responsible for each property, so that I can organise workload*

A diagram of a project

AI-generated content may be incorrect.

A diagram of a realtor

AI-generated content may be incorrect.

## User Story 3

*As a realtor, I schedule showings for each property I am responsible for, so that I can solicit potential buyers*

A diagram of a project

AI-generated content may be incorrect.

A diagram of a company

AI-generated content may be incorrect.

## User Story 4

*As a seller, I want to list my house so that I can sell it*

A diagram of a project

AI-generated content may be incorrect.

A diagram of a property for sale

AI-generated content may be incorrect.

## User Story 5

*As a buyer, I want to see local houses for sale so that I can buy it*

A diagram of a computer

AI-generated content may be incorrect.

A diagram of a property

AI-generated content may be incorrect.

# Requirements document

## Introduction

**Purpose:** This document details the functional requirements for the realty management system designed to facilitate realty viewings and sales.

**Scope:** The application will allow sellers to list property, potential buyers to view listed properties, buyers to buy properties, realtors to show property and facilitate the sale of properties and managers to assign properties to realtors.

## System overview

The system aims to provide a user-friendly interface for Hypothetical Realtors Inc to manage properties that they have been commissioned to sell or rent. Key features include property assignment, viewing organisation and sales commissioning.

## Business Requirements

Improve productivity by providing management an overview of property statuses

Notify all stakeholders of deadlines, viewings, and property updates

Allow data to discover bottlenecks in the sales process

## Functional Requirements

The system will allow users to log in and maintain correct user account details

The system will allow Buyers to view and register interest in local properties.

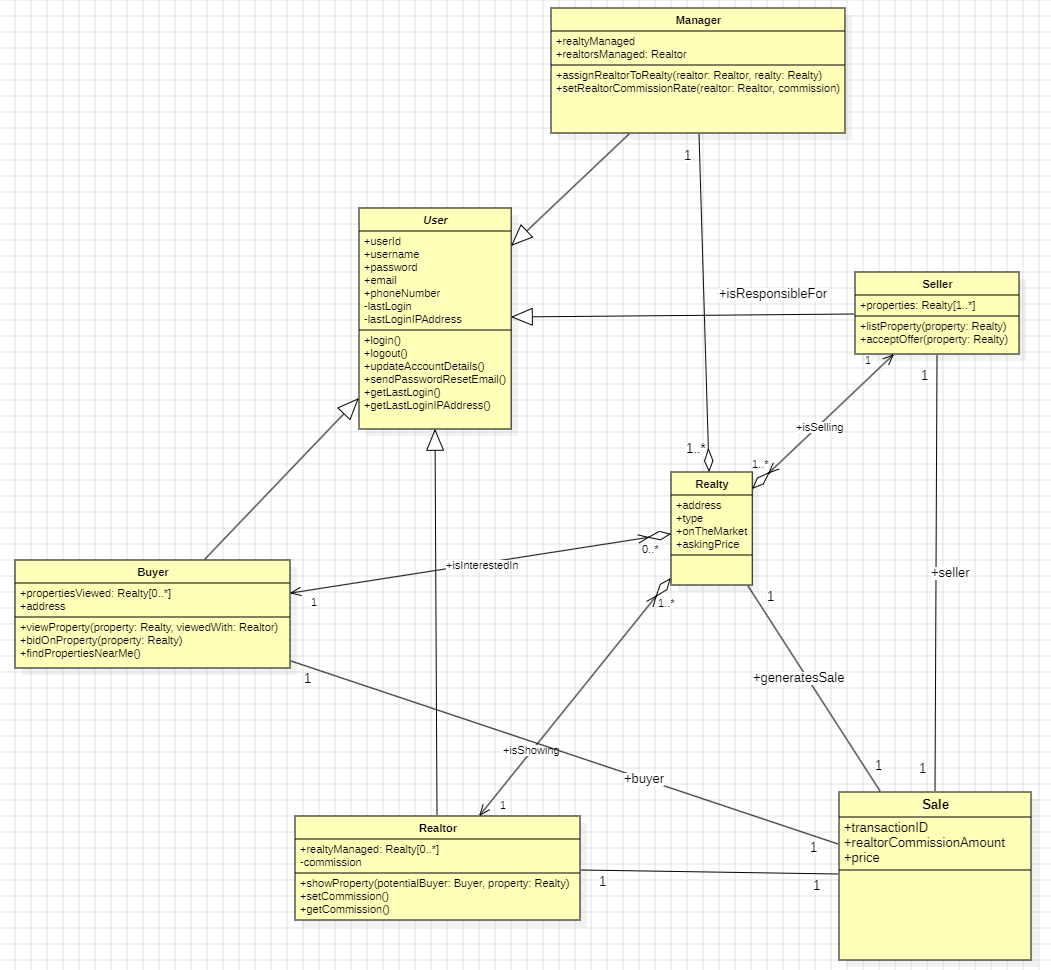
The system will allow Sellers to register properties they have for sale with Hypothetical Realtors inc.

The system will allow managers to assign properties to realtors.

The system will allow realtors to arrange viewings of properties to potential buyers

The system will implement a Model View Controller architecture to allow for a responsive, device agnostic, front end but a consistent back end API that models the functionality.

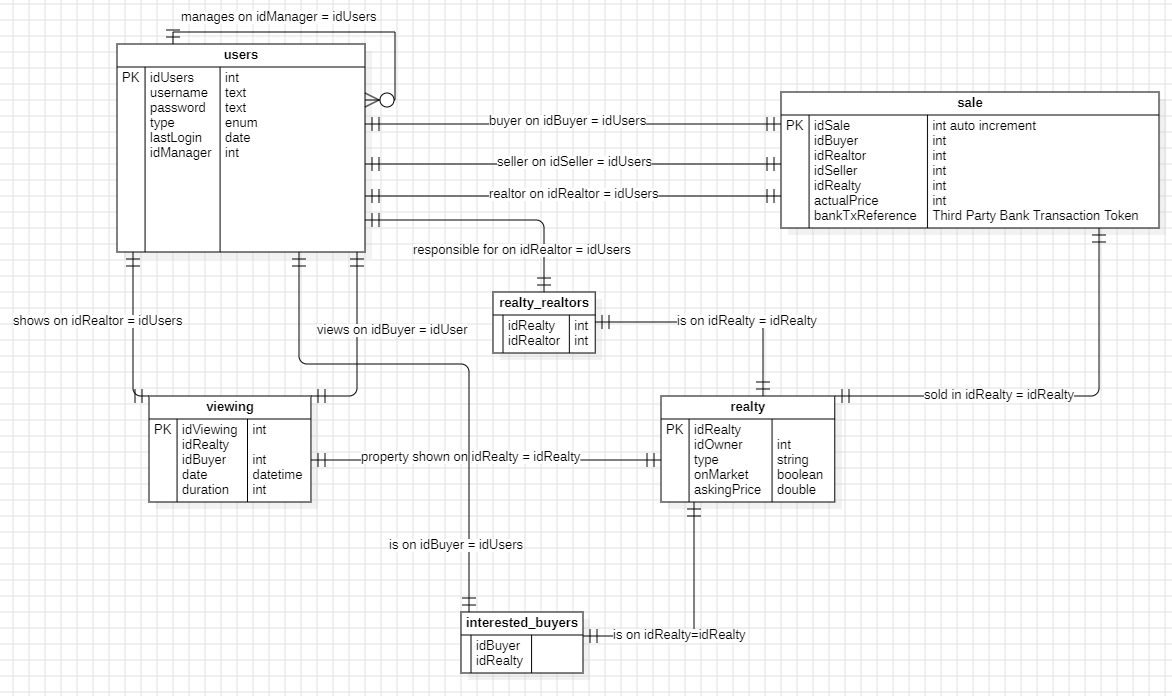
# Class diagram



The class diagram derived from the functional requirements will allow the software developers to independently design aspects of the realty application independently of each other.

Each separate user type inherits the basic user functionality from the abstract User class and is responsible only for the behaviours associated with their individual implementation concerns, ie the Buyer class is only concerned with buyer specific business concerns, etc.

# Entity Relationship Diagram



The initial database schema required to design a Minimum Viable Product. This is a conservative estimate of the data structure that arises from the class diagram. Extra tables and relations may arise over the course of development and this diagram would accordingly require updating.

# System tests

## Login

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Inputs** | | **Results** | |  |
| **Correct Username** | **Correct Password** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| Yes | Yes | Login Succeeded |  |  |
| Yes | No | Login Failed |  |  |
| No | Yes | Login Failed |  |  |
| No | No | Login Failed |  |  |

## Account update

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Inputs** | | **Results** | |  |
| **Account details changed** | **Password** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| Yes | Correct | Account details updated |  |  |
| Yes | Incorrect | Account password incorrect message |  |  |
| Yes | Not provided | Account password required message |  |  |
| No | - | No response required |  |  |