**Funcionalidades mínimas:**

* Búsqueda por filtros (ciudad, alquiler/venta, valor)
* Publicación con imágenes, descripción, número de contacto.
* Ubicación en mapa.
* Gestión de múltiples propiedades para dueño/agente inmobiliario.
* Mensajería interna?

**Diferenciales de la competencia:**

* Destacar dueños de inmobiliarias.
* Habilitar espacio de reseñas de usuarios confirmados?
* Mejor interfaz de mensajería.

**Entidades, atributos, métodos y relaciones necesarios:  
  
Observaciones preliminares:   
  
1.** todo borrado/eliminado es lógico (es decir, un atributo booleano que se pasa a “falso” para que no aparezca en las búsquedas de los usuarios, sin borrar el registro en base de datos. Se prefiere así por razones legales y de integridad de datos en la base de datos.  
  
2. dada la complejidad del sistema de mensajería directa y el escaso valor agregado que aporta al producto (dado que es más común contactarse vía whapp o llamar), se prefiere iniciar con todas las otras funcionalidades y dejar esta para el último, sujeta a ajustes. Entretanto, se puede incluir el nro de teléfono del dueño/agente inmobiliario para hacer el contacto.  
  
3. asimismo, tampoco se incluye en este momento las entidades y relaciones necesarias para un foro de reseñas para cada propiedad.  
  
4. se prefiere la gestión de usuarios por roles (donde éstos aseguran tanto las pantallas a las que acceden como los comportamientos que pueden o no hacer dentro de la app) por sobre la herencia de clases y la especificación posterior de atributos y métodos singulares.  
\* Enum “Rol usuario”: Inquilino/Comprador, Dueño, Agente Inmobiliario, Administrador

\* Enum “Tipo propiedad”: casa, depto, PH…  
  
\* Usuario.  
 Atributos: id, nombre, email, contraseña, rol, fecha de registro, habilitado (bool)

Métodos: registrarse(), iniciarSesion(), actualizarPerfil(), eliminarCuenta(), cambiarContraseña()  
Relaciones: Tiene 0..\* publicaciones (si es dueño o agente). Envía y recibe mensajes. Puede ser deshabilitado por el admin.

\* Propiedad.

Atributos: id, título, descripción, tipo, operación (alquiler/venta), precio, dirección, ubicación, superficie, ambientes, fotos, estado (activa/inactiva), fecha de publicación/modificación de estado.  
Métodos: crearPublicación(), editarPublicación(), pausarPublicación(), eliminarPublicación(), verPublicación(), verTodasLasPublicaciones(), verTodasLasPublicacionesConFiltro().

Relaciones: Pertenece a un Usuario (dueño o agente), Permitir mensaje directo (que especifique ids de dueño/agente, de la propiedad y del inquilino/comprador). Puede ser editada/eliminada por el administrador.

\* Conversación:

Atributos: id, user\_1, user\_2, propiedad\_id, fecha\_inicio, habilitado (bool).

Métodos: iniciarConversacion(), obtenerConversacionEspecifica(), obtenerTodasLasConversaciones(), eliminarConversacion()

Relaciones: Conecta dos usuarios en relación a una propiedad específica. Puede ser leído solo por los participantes y administrador.

\* Mensaje:

Atributos: id, conversacion\_id, emisor\_id, receptor\_id, contenido, fecha, leido (bool)  
 Métodos: enviarMensaje(), borrarMensaje()

Relaciones: Muchos son parte de una sola conversación. Emisor y administrador puede “borrarlos” (se los oculta).