

Caso de Uso: Sugerencias Automáticas para el Agente

1. Información general

Nombre del caso de uso

Sugerencias automáticas para el agente inmobiliario.

Versión

1.0

Autor

Claudia Maritza Velásquez R.

Fecha

Octubre 2025

Propuesta Funcional

Nueva funcionalidad basada en inteligencia contextual, desarrollada como una simulación simplificada del *Caso de Uso Propuesto* a partir del análisis de la información pública disponible en la web oficial de Inmoweb sobre vuestro [CRM Inmobiliario](#).

Esta mejora ilustra cómo podría integrarse un sistema de sugerencias automáticas para el agente inmobiliario, capaz de identificar coincidencias entre las características de un inmueble y las preferencias registradas de los clientes.

Objetivo

El propósito de esta simulación es mostrar una pequeña muestra práctica de mi experiencia en mantenimiento evolutivo y ampliación funcional de sistemas existentes, aplicándolo en este caso al contexto de un CRM inmobiliario similar al de Inmoweb.

Debido a que los proyectos en los que he participado pertenecen a entornos empresariales con repositorios privados y cláusulas de confidencialidad, no es posible compartir su código fuente.

Por ello, he preparado esta implementación simplificada que reproduce la lógica de un caso de uso realista —con un enfoque en arquitectura, patrones de diseño (MVC y Strategy) y claridad del flujo funcional—, manteniendo el alcance en un nivel demostrativo.

No se pretende replicar las capas de seguridad, autenticación OAuth ni las pruebas unitarias que ya podrían formar parte del CRM real de Inmoweb.

El objetivo es demostrar capacidad técnica, análisis funcional y criterio arquitectónico en un entorno controlado, respetando las buenas prácticas y las posibles funcionalidades descritas en la documentación pública del producto.

Entregables y entorno de despliegue

- **Repositorio Git:**

Incluye el código fuente completo de la simulación desarrollada, junto con la documentación funcional y técnica del caso de uso “Sugerencias automáticas para el agente”. El repositorio se organiza siguiendo buenas prácticas de control de versiones y convenciones de nomenclatura, facilitando su revisión y mantenimiento.

- https://github.com/cmvelasquezr/crm_inmoweb/tree/main

- **Instalación en entorno privado:**

El proyecto se encuentra instalado en un **servidor privado**, con acceso restringido a las personas que la empresa designe. Este entorno permite la evaluación práctica de la funcionalidad propuesta sin comprometer información sensible ni exponer recursos públicos.

- <http://135.236.152.41/index.php>

- **Despliegue en entorno Linux:**

El sistema está preparado para su ejecución en un **servidor Linux** bajo **Apache y MySQL**.

2. Descripción general

El objetivo de este caso de uso es **optimizar la gestión comercial del agente inmobiliario**, proporcionándole **recomendaciones automáticas de clientes potenciales** interesados en un inmueble determinado.

El sistema calcula un **índice de afinidad** entre las características del inmueble y las preferencias registradas de los clientes en el CRM.

Esto permite que el agente identifique rápidamente a qué clientes podría ofrecer un inmueble concreto, reduciendo el tiempo de búsqueda manual y aumentando la efectividad de cierre.

3. Actores involucrados

Actor	Rol
Agente inmobiliario	Usuario principal. Consulta la ficha de un inmueble y recibe sugerencias de clientes potenciales.
Sistema CRM Inmoweb	Procesa los datos de inmuebles y clientes, calcula el índice de afinidad y muestra las sugerencias.
Administrador (opcional)	Define los parámetros del algoritmo de afinidad (peso de ubicación, presupuesto, número de habitaciones, etc.).

4. Flujo principal del caso de uso

Escenario: Desde la ficha de inmueble

1. El **agente** accede a la ficha de un inmueble dentro del CRM.
2. El **sistema** identifica automáticamente el inmueble activo.
3. El **módulo de afinidad** busca en la base de datos los clientes registrados.
4. Se calcula un **índice de afinidad (0–100%)** para cada cliente según:
 - Coincidencia de **zona o barrio**.
 - Ajuste del **rango de precio**.
 - Coincidencia de **número de habitaciones**.
 - Preferencias adicionales (terraza, garaje, ascensor, etc.).
5. El sistema ordena los resultados de mayor a menor afinidad.
6. El **agente visualiza** una lista con los clientes sugeridos, mostrando:
 - Nombre del cliente.

- Porcentaje de afinidad.
- Motivos de coincidencia (“Busca en Chamberí”, “Presupuesto dentro del rango”, etc.).

7. El agente podría (Caso de uso no implementado):

- Abrir la ficha del cliente.
- Enviar un correo directo desde el CRM.
- Marcarlo como “contactado”.

5. Flujos alternativos

Alternativa	Descripción
A1: No existen coincidencias	El sistema muestra el mensaje “No hay clientes compatibles actualmente para este inmueble”.
A2: Filtros personalizados	El agente puede ajustar manualmente los parámetros de búsqueda (por ejemplo, ampliar el rango de precios o añadir ubicación cercana).

6. Reglas de negocio

1. El cálculo de afinidad debe realizarse en menos de 1 segundo por inmueble.
 2. Los resultados se actualizan automáticamente si cambian las características del inmueble o las preferencias del cliente.
-

7. Consideraciones técnicas

- **Patrón de diseño empleado:**
 - **MVC (Model-View-Controller):** para separar la lógica de negocio, la presentación y el control de flujo.





- **Strategy:** para encapsular el algoritmo de cálculo de afinidad, permitiendo cambiar la lógica sin alterar el resto del sistema.

Algoritmo de afinidad (simplificado):

```
$score = 0;  
if ($client->matchesZone($property->zone)) $score += 40;  
if ($client->priceInRange($property->price)) $score += 30;  
if ($client->matchesRooms($property->rooms)) $score += 20;  
if ($client->matchesExtras($property)) $score += 10;
```

- Resultado → **Afinidad total = \$score / 100 × 100%**
- **Tecnologías:**
 - PHP 8.2 (NTS)
 - HTML5 + CSS3
 - XAMPP / Apache
 - MySQL (o simulación en memoria)

8. Beneficios esperados

Beneficio	Descripción
 Eficiencia comercial	Reduce el tiempo del agente en la búsqueda manual de clientes.
 Mayor tasa de conversión	Las coincidencias de afinidad aumentan la probabilidad de cierre.
 Automatización inteligente	Usa datos existentes sin requerir intervención manual.
 Visibilidad estratégica	Facilita reportes sobre qué tipo de inmuebles tienen más demanda.

9. Posibles mejoras futuras

1. Integrar un **motor de aprendizaje automático (Machine Learning)** que aprenda de los cierres anteriores.
 2. Permitir que el agente **ajuste el peso** de los factores de afinidad.
 3. Mostrar sugerencias inversas: “clientes con afinidad alta que aún no tienen inmuebles disponibles”.
 4. Integración con **notificaciones automáticas** (correo o WhatsApp).
-

9. Estructura del proyecto

```
crm-inmobiliario/
|
|— index.php # Punto de entrada principal
|
|— src/
|
|— assets/
|   |— css/
|   |   |— style.css #Estilos para la vista HTML de la ficha de
inmueble
|   |   |— js/
|   |       |— script.js #Javascript para la vista HTML de la ficha de
inmueble
|— Config/
|   |— Database.php # Configuración BBDD
|
|   |— Controllers/
|   |   |— PropertyController.php # Controlador principal (coordina modelo
y vista)
|   |
|   |— Models/
|   |   |— Property.php # Modelo de inmueble
|   |   |— Client.php # Modelo de cliente
|   |   |— AffinityCalculator.php # Calculadora de afinidad (usa Strategy)
|   |
|   |— Repositories/
|   |   |— PropertyRepository.php # Interfaz de repositorio
|   |   |— ClientRepository.php # Interfaz del cliente
```

```
| | └─ PropertyRepository.php # Interfaz de la propiedad
| | └─ MySQLPropertyRepository.php # PDO de la propiedad
| | └─ MySQLClientRepository.php # PDO del cliente
| |
| └─ Strategies/
| | └─ AffinityStrategy.php # Interfaz del patrón Strategy
| | └─ BasicAffinityStrategy.php # Estrategia concreta de afinidad
básica
|
└─ views/
    └─ property_view.php # Vista HTML de la ficha de inmueble
```