

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



TRABAJO FIN DE GRADO

Aplicación de Gestión Inmobiliaria en la Nube

Dagmara Wójcik

Tutor: Simone Santini

Mayo 2014

Resumen

Título: Aplicación de Gestión Inmobiliaria en la Nube

El objetivo de este proyecto es la implementación de una aplicación de gestión inmobiliaria en la nube, entendiendo como tal una aplicación que presenta características propias de una aplicación web y de una aplicación de escritorio. En este proyecto, la aplicación es capaz de hacerse cargo de las tareas de gestión y administración de los inmuebles de un portal inmobiliario.

Para proceder al desarrollo del proyecto se estudian diferentes tecnologías que facilitan la implementación de una aplicación en nube. Se analizan las ventajas e inconvenientes de las distintas alternativas y el razonamiento con el que se ha llegado a elegir la implementación final del proyecto. Todos los estudios realizados han tenido por objetivo fijar una estrategia on-line para pequeñas empresas dedicadas al sector de la inmobiliaria.

Palabras Clave:

Aplicación en la Nube, Aplicación Web, Modelo-Vista-Controlador, Framework, Metodología, Mantenimiento, Gestión Inmobiliaria.

Abstract

Title: Real Estate Management Application in the Cloud

The objective of this project is the implementation of a cloud application for real estate management, with features both those of a web application and a desktop application. In this project, the application is able to handle real property portal administrative and management related tasks.

To proceed with the development of the project, several technologies are studied which ease the efforts on implementing a cloud application. Advantages and drawbacks are thoroughly analyzed along with the different approaches chosen for the final implementation of the project. The whole research has followed the objective to fix an online strategy for small real estate enterprises.

Keywords :

Cloud Application , Web Application , Model-View-Controller Framework , Methodology, Maintenance, Real Estate Management .

Índice de contenido

1. Introducción	16
2. Definición del proyecto.....	20
2.1. Contexto actual.....	20
2.2. Case Study: escenarios del cliente	22
2.3. Alcance del proyecto	26
2.4. Metodología.....	27
2.4.1. Control y Seguimiento	28
2.4.2. Diseño gráfico y funcional.....	29
2.4.3. Preparación de entornos.....	30
2.4.4. Desarrollo.....	31
2.4.5. Implantación	32
2.4.6. Carga de contenidos.....	33
2.4.7. Pruebas y Aceptación	33
2.4.8. Gestión del Cambio.....	34
2.5. Web App vs. Cloud App.....	34
2.6. Arquitectura Lógica	35
2.6.1. Capa de Presentación.....	37
2.6.2. Capa de Aplicaciones.....	37
2.6.3. Capa de Acceso a Datos.....	38
2.6.4. Capa de Datos.....	39
2.7. Cloud hosting vs. VPS.....	39
2.8. Blob Storage Account vs. File System.....	40
3. Requisitos del Proyecto.....	42

Aplicación de Gestión Inmobiliaria en la Nube

3.1.	Requisitos Funcionales.....	42
3.1.1.	EF1: Acceso a la página principal.....	42
3.1.2.	EF2: Acceso a Secciones de Inmuebles.....	44
3.1.3.	EF3: Acceso a la página Favoritos.....	47
3.1.4.	EF4: Acceso a la página de Registro.....	48
3.1.5.	EF5: Acceso a la página Login.....	49
3.1.6.	EF6: Acceso a la página de Administración.....	49
3.1.7.	EF7: Buscador.....	50
3.1.8.	EF8: Acceso a la página de Resultados de Búsquedas.....	51
3.1.9.	EF9: Acceso a la página de Detalle de un Inmueble.....	52
3.1.10.	EF10: Acceso a la página Creación de un Inmueble (Administración).....	53
3.1.11.	EF11: Acceso a la página Borrado de Inmuebles (Administración).....	54
3.1.12.	EF12: Acceso a la página Listado de Inmuebles (Administración).....	55
3.1.13.	EF13: Acceso a la página Edición un Inmueble.....	57
3.1.14.	EF14: Acceso a la página Creación de Estados de Inmuebles.....	58
3.1.15.	EF15: Acceso a la página Borrado de Estados de Inmuebles.....	59
3.1.16.	EF16: Acceso a la página Listado de Estados de un Inmueble.....	60
3.1.17.	EF17: Acceso a la página Edición de Estados de un Inmueble.....	61
3.1.18.	EF18: Acceso a la página Creación de Tipos de Inmueble.....	62
3.1.19.	EF19: Acceso a la página Borrado de Tipos de Inmueble.....	63
3.1.20.	EF20: Acceso a la página Listado de Tipos de Inmueble.....	64
3.1.21.	EF21: Acceso a la página Edición de Tipos de Inmueble.....	65
3.1.22.	EF22: Acceso a la página Creación de Personas.....	66
3.1.23.	EF23: Acceso a la página Borrado de Personas.....	67

Aplicación de Gestión Inmobiliaria en la Nube

3.1.24.	EF24: Acceso a la página Listado de Personas.....	68
3.1.25.	EF25: Acceso a la página Edición de Personas.....	69
3.1.26.	EF26: Acceso a la página Creación de Tipos de Persona	70
3.1.27.	EF27: Acceso a la página Borrado de Tipos de Persona.....	71
3.1.28.	EF28: Acceso a la página Listado de Tipos de Persona	72
3.1.29.	EF29: Acceso a la página Edición de Tipos de Persona	73
3.1.30.	EF30: Acceso a la página Creación de Interesados	74
3.1.31.	EF31: Acceso a la página Borrado de Interesados.....	75
3.1.32.	EF32: Acceso a la página Listado de Interesados.....	76
3.1.33.	EF33: Acceso a la página Edición de Interesados	77
3.1.34.	EF34: Acceso a la página Creación de Tipos de Contacto.....	78
3.1.35.	EF35: Acceso a la página Borrado de Tipos de Contacto	79
3.1.36.	EF36: Acceso a la página Listado de Tipos de Contacto	80
3.1.37.	EF37: Acceso a la página Edición de Tipos de Contacto.....	81
3.1.38.	EF38: Acceso a la página Creación de Poblaciones.....	82
3.1.39.	EF39: Acceso a la página Borrado de Poblaciones.....	83
3.1.40.	EF40: Acceso a la página Listado de Poblaciones.....	84
3.1.41.	EF41: Acceso a la página Edición de Poblaciones.....	85
3.1.42.	EF42: Acceso a la página Configuración de Ofertas	86
3.1.43.	EF43: Acceso a la página Edición de una Oferta	87
3.1.44.	EF44: Acceso a la página Configuración de Carrousel.....	88
3.1.45.	EF45: Acceso a la página Edición de un elemento del Carrousel	89
3.2.	Requisitos no Funcionales	89
3.2.1.	ENF1: Disponibilidad de la aplicación	89

Aplicación de Gestión Inmobiliaria en la Nube

3.2.2.	ENF2: Roles de Usuarios.....	90
3.2.3.	ENF3: Seguridad de la aplicación.....	90
3.2.4.	ENF4: Usabilidad de la aplicación.....	90
3.2.5.	ENF5: Compatibilidad con Navegadores.....	91
3.2.6.	ENF6: Diseño de la aplicación.....	91
3.2.7.	ENF7: Escalabilidad y extensibilidad de la aplicación.....	91
3.2.8.	ENF8: Sesiones de usuarios simultáneas.....	92
3.2.9.	ENF9: Mantenibilidad de la aplicación.....	92
3.2.10.	ENF10: Requisitos de Servidores.....	92
3.2.11.	ENF11: Tiempos de Respuesta.....	93
3.2.12.	ENF12: Bases de Datos.....	93
4.	Diseño del Software.....	94
4.1.	Estudio de alternativas tecnológicas (frameworks).....	94
4.2.	Implementación escogida.....	106
4.3.	Diseño técnico.....	107
4.3.1.	Base de datos.....	108
4.3.2.	Patrón.....	112
4.3.3.	Detalles de implementación.....	114
5.	Uso de la Aplicación.....	116
5.1.	Casos Reales.....	116
5.1.1.	Alta de inmuebles en el portal.....	116
5.1.2.	Portal corporativo.....	116
5.1.3.	Interacción con los visitantes.....	117
5.1.4.	Acceso a los detalles de un inmueble.....	118

Aplicación de Gestión Inmobiliaria en la Nube

5.2. Comparación con otras plataformas.....	119
6. Mantenimiento.....	120
6.1. Fases del servicio de mantenimiento.....	120
6.1.1. Definición.....	121
6.1.2. Transición.....	121
6.1.3. Servicio.....	121
7. Conclusiones.....	124
8. Líneas Futuras de Trabajo.....	126
Anexo 1: Manual de Usuario.....	128
Anexo 2: Clase BlobStorageService.....	142

Índice de figuras y tablas

Ilustración 1: Web actual de la empresa	22
Ilustración 2: Web actual de la empresa II	23
Ilustración 3: Web actual de la empresa III	23
Ilustración 4: Web actual de la empresa IV.....	24
Ilustración 5: Aplicación actual de la empresa.....	24
Ilustración 6: Scrum, metodología ágil de desarrollo software.....	27
Ilustración 7: Blob Storage Account	41
Ilustración 8: Popularidad de Frameworks PHP (fuente: www.sitepoint.com).....	95
Ilustración 9: Frameworks más populares de Java (fuente: zeroturnaround.com)	98
Ilustración 10: Mantenimiento propuesto.....	120
Ilustración 11: Fases del servicio de mantenimiento.....	120

Aplicación de Gestión Inmobiliaria en la Nube

Glosario de Términos

Framework: (marco de trabajo) define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

Metodología: conjunto de procedimientos utilizados para alcanzar los objetivos de un proyecto software.

Mantenimiento: fases en el ciclo de vida de desarrollo de sistemas (SDLC, sigla en inglés de system development life cycle), que se aplica al desarrollo de software.

Aplicación: tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajos

1. Introducción

El origen de este proyecto tiene lugar en una empresa dedicada al sector de la inmobiliaria que ve la necesidad de redefinir su estrategia de presencia on-line así como adecuar su herramienta de gestión a las necesidades del negocio. El presente proyecto constituye la primera entrega de un proyecto que responde a las necesidades reales de un modelo de negocio como el que presenta dicha empresa.

El principal objetivo del proyecto ha sido construir una aplicación que sustituya al software que la empresa tiene actualmente y que posea un módulo de administración con el fin de permitir la gestión del contenido del sitio web.

El software que actualmente maneja la empresa resulta a día de hoy obsoleto y presenta, entre otros, los siguientes inconvenientes:

- **No permite la gestión ubicua de contenido:** Para añadir ofertas a la web es necesario utilizar una herramienta de escritorio que funciona en una red de ordenadores, sólo disponible en la oficina de la empresa. Esto imposibilita el trabajo desde casa o cualquier otro lugar que no sea el propio puesto de trabajo.
- **El software está obsoleto:** La herramienta actual tiene alrededor de 10 años y la empresa encargada del mantenimiento de la misma sigue percibiendo el mismo o mayor importe de hace unos años sin ofrecer un mantenimiento evolutivo del software.
- **Alta complejidad de uso:** La herramienta posee muchas funcionalidades que los usuarios no utilizan en absoluto, aumentando el número de pasos para realizar una consulta sencilla en la aplicación.
- **Falta de funcionalidades:** Si bien es cierto que la actual herramienta proporciona muchas funcionalidades, no permite la más básica de todas: guardar una relación de clientes interesados en un inmueble. Esta información se tiene que registrar a mano actualmente.
- **Falta de interacción con los visitantes de la web:** El sitio actual de la inmobiliaria ha registrado cerca de 13 mil visitas mensuales y no brinda la

posibilidad de que los usuarios puedan tener su propia sesión en la web y guardar los inmuebles que les hayan parecido más interesantes para futuras visitas.

- **No se facilita la exportación de datos:** La empresa no dispone de acceso al medio de almacenamiento de los inmuebles. La herramienta tampoco dispone de una funcionalidad que lo facilite, salvo una impresión en papel, siendo imposible realizar una exportación digital de los datos.
- **Las operaciones de gestión sólo son realizables desde un PC:** El programa de gestión no se puede usar desde el teléfono móvil ni tableta. Ni siquiera desde otro ordenador que no esté en la misma red que la oficina.

Tras exponer los principales inconvenientes del software actual de la empresa, resultará más sencillo saber cuáles son los principales puntos que aborda el presente proyecto.

Estos son:

- **Permitir la gestión ubicua de contenido:** El software funcionará en ordenadores personales, tabletas y teléfonos móviles. La aplicación de escritorio desaparecerá para convertirse en una aplicación web.
- **Construir una página web con un diseño más atractivo:** Se construirán las bases de una página web más atractiva aunque no será objetivo de este proyecto presentar el sitio web definitivo ya que se quiere mantener el anonimato de la empresa. Como se mencionó anteriormente, este proyecto constituye una primera entrega.
- **Diseñar un software escalable:** Las ampliaciones a las que se someterá a la aplicación posteriormente buscarán una buena integración con Microsoft Office. Para ello se expondrán todas posibles implementaciones estudiadas para lograr este fin.
- **Agregar perfiles de usuarios web:** La aplicación presentará dos tipos de usuarios web:
 - **Administradores:** tendrán acceso a la aplicación web que permite la gestión del contenido web.
 - **Visitantes:** podrán navegar por la página web y guardar sus inmuebles favoritos para posteriores visitas.

Aplicación de Gestión Inmobiliaria en la Nube

- **Integrar el sitio web con redes sociales:** Se agrega a la página web la posibilidad de compartir inmuebles en Facebook y Twitter.

Los requisitos de la aplicación se han extraído a partir de reuniones y correos electrónicos con los gerentes de la empresa. Todos los referentes al presente Trabajo de Fin de Grado se expondrán a lo largo de este documento.

2. Definición del proyecto

El objetivo del proyecto ha sido el desarrollo de una aplicación *cloud-aware*, entendiendo como tal una aplicación capaz de funcionar tanto como aplicación web como aplicación en nube. Se entiende por aplicación web una aplicación accesible desde cualquier dispositivo con un navegador, mientras que una aplicación en nube presenta características propias tanto de una aplicación web como de una aplicación de escritorio. En este proyecto, la aplicación es capaz de hacerse cargo de las tareas de gestión y administración de los inmuebles.

A lo largo del documento se expondrán las diferentes tecnologías estudiadas para el desarrollo de la aplicación, las ventajas e inconvenientes de las distintas alternativas y el razonamiento con el que se ha llegado a elegir la implementación final del proyecto. Todos los estudios realizados han tenido por objetivo dar una solución a la nueva estrategia on-line de la empresa y sus necesidades de gestión, teniendo en cuenta el posible uso de la aplicación por más empresas que presenten un modelo de negocio similar.

2.1. Contexto actual

Durante los últimos tres años de actividad empresarial, la agencia inmobiliaria que ha inspirado el proyecto ha experimentado un importante aumento de visitas en la web, posicionándola en los primeros resultados de búsqueda en Google.

El incremento de posibles clientes ha llevado a la agencia a redefinir su estrategia de presencia on-line con el objetivo de adecuar su posicionamiento en Internet a las actuales necesidades del negocio.

Los principales objetivos que persigue la nueva estrategia web son los siguientes:

- ✓ Mejorar su posicionamiento interprovincial
- ✓ Conocer las necesidades y preferencias de los posibles clientes

Aplicación de Gestión Inmobiliaria en la Nube

- ✓ Reforzar su imagen comercial
- ✓ Dar a conocer todos los servicios de la agencia

Como respuesta a estos objetivos, se ha realizado un estudio de varias soluciones web que se encuentran ofertadas en Internet actualmente. Todas las ofertas ofrecían un software de caja, no ampliable, con problemáticas similares a la solución que la empresa posee actualmente: nula escalabilidad, excesiva funcionalidad, alta complejidad y nula adaptación del software al modelo de negocio de la empresa. Por tanto, el nuevo modelo de presencia on-line que persigue el presente proyecto se resume en el desarrollo de una aplicación basada en los siguientes puntos:

1. Portal corporativo de la empresa
2. Integración del portal con redes sociales
3. Interacción de la empresa con los visitantes web
4. Diseño web adaptable (Responsive Design)
5. Gestión ubicua del contenido del portal

A raíz de la decisión de la agencia de involucrarse y colaborar en la creación de un software que se adaptara a todas las necesidades del negocio, es deseable encajar la relación mantenida dentro de las **metodologías ágiles** de desarrollo software.

Las metodologías ágiles son un conjunto de métodos basados en el desarrollo iterativo e incremental del software. Los requisitos y las soluciones se definen a través de la estrecha colaboración de equipos multidisciplinares. En este caso, las disciplinas que intervienen directamente son la ingeniería del software junto con el sector inmobiliario. La creación de un software especializado en inmobiliarias ha requerido la manutención de un flujo constante de comunicación con el fin de extraer los diferentes requisitos funcionales y no funcionales que, además, deben adaptarse al marco legal español. Teniendo en mente que el portal desarrollado presenta ofertas de distintos inmuebles a los usuarios web, ha sido especialmente importante averiguar la información que obligatoriamente debe darse a conocer.

2.2. Case Study: escenarios del cliente

Portal web

Para definir el diseño del portal que se desarrolla en este proyecto, ha sido necesario un estudio previo del portal actual con el fin de identificar las carencias del mismo. El portal actual de la empresa presenta el siguiente aspecto:

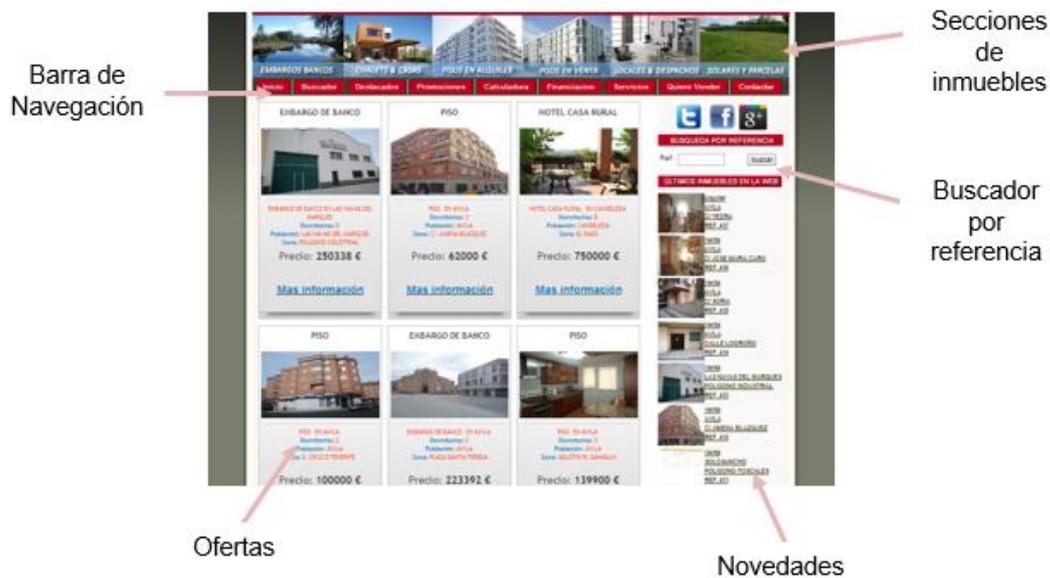


Ilustración 1: Web actual de la empresa

Comenzando por la página principal del sitio, se observan ciertas carencias:

- **Una búsqueda por referencia:** obliga a los visitantes de la web a apuntar a mano la referencia de los inmuebles que les han resultado interesantes. Esta situación puede desembocar en la pérdida de clientes ya que otros portales inmobiliarios incorporan sesiones de usuario que permiten marcar favoritos, sin obligarles a hacer un esfuerzo adicional en la navegación.
- **Diseño obsoleto:** Nada más entrar en la página se observa a simple vista que el aspecto no está cuidado. Esto puede indicar al visitante que la página no se actualiza muy a menudo o que el negocio no cuida su presencia online.

Aplicación de Gestión Inmobiliaria en la Nube

- **Menú de navegación inútil:** incorpora funcionalidades que no se utilizan. Los propios agentes inmobiliarios han señalado esta sección como inútil. Incorpora un enlace a la sección de búsqueda que debería estar en la página principal.
- **No presenta un buscador en primera página:** obliga al usuario a navegar por más páginas para llegar al buscador de inmuebles o irse a otro sitio.

Si se accede a alguna sección se pueden ver defectos como el siguiente:



Ilustración 2: Web actual de la empresa II

La información publicada en una sección de parcelas es la misma que en la de pisos, cuando los inmuebles poseen características distintas. ¿Acaso una parcela tiene baños o habitaciones? Este tipo de detalles dan mala imagen al negocio.

Si se navega a la sección de búsqueda, aparece un desplegable de operaciones que permite buscar: Alquileres, Ventas y Traspasos. Esto no es configurable y da a entender al usuario que la agencia opera con traspasos.

Busqueda de Propiedades

En nuestra cartera de inmuebles encontrará la mejor oferta de inmuebles.

Operación:

Precio desde: Hasta:

Provincia: Municipio:

Zona: Tipo:

Dormitorios desde: Hasta:

Busqueda por Referencia :

Ilustración 3: Web actual de la empresa III

Aplicación de Gestión Inmobiliaria en la Nube

Pero si buscamos algún inmueble para traspasos, nos encontramos con que no existe ninguno en todo el portal:

No se han encontrado inmuebles para su consulta.

Ilustración 4: Web actual de la empresa IV

¿Para qué se muestra la opción de traspasos entonces? ¿No sería más lógico mostrarla cuando hubiera inmuebles que ofertar?

Aplicación de escritorio

El software que utiliza la entidad para gestionar el contenido del portal web, entre otras funcionalidades, es una aplicación de escritorio que funciona en red. Es decir, fuera de la oficina la aplicación no es utilizable. Por ello, el principal inconveniente que presenta es que no se puede trabajar desde ningún otro sitio que no sea la oficina, limitando con ello la actividad empresarial.

A primera vista, el software puede parecer muy completo. Cantidad de botones, pestañas, menús... Lo cierto es que el 80% de lo que ofrece el software no se utiliza y a los usuarios les resulta muy difícil encontrar lo que quieren sin tener que indagar en la herramienta. No resulta intuitiva y requiere un tiempo de aprendizaje elevado para usarla correctamente.

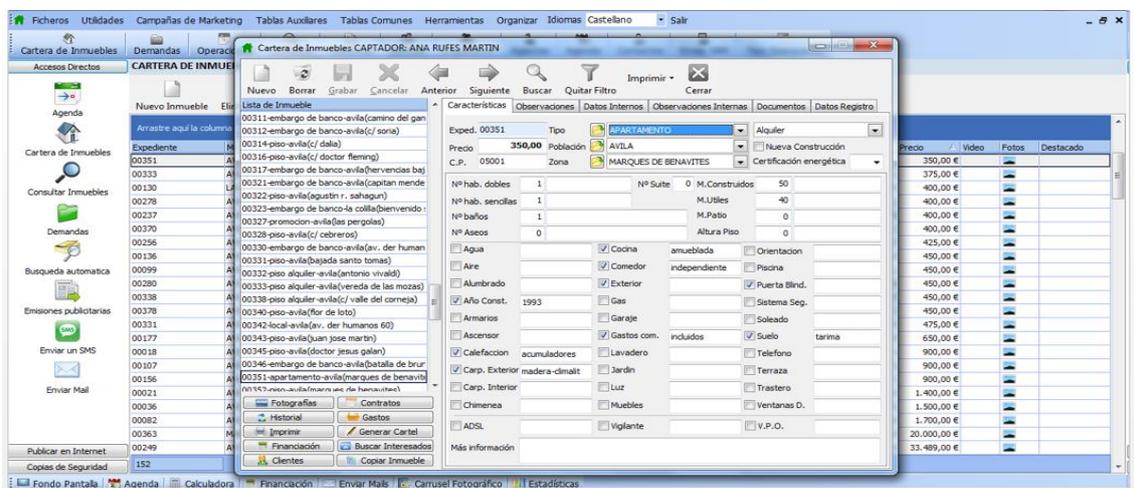


Ilustración 5: Aplicación actual de la empresa

Aplicación de Gestión Inmobiliaria en la Nube

Tras numerosas conversaciones con los gerentes de la empresa, se han recogido las carencias más importantes del software actual. Ya que posteriormente se definen los requisitos del nuevo proyecto es importante que las lagunas que actualmente existen se vean solventadas:

- ✓ **No se puede trabajar fuera de la oficina:** varios agentes trabajan para la misma empresa y no se puede gestionar el contenido desde otro sitio que no sea la red de la oficina. Esto obliga a tener más puestos de trabajo, pagar por un local más grande, comprar más licencias del producto y limitar la actividad empresarial. Como es evidente que una pequeña empresa puede no tener los medios para costear todo, se recurre a la segunda opción: perder oportunidades de negocio.
- ✓ **Aplicación accesible únicamente desde ordenadores personales:** ¿Y los smartphones y tabletas? ¿Es que no existe a día de hoy alguna tecnología que permita usar una misma aplicación desde cualquier dispositivo con conexión a Internet? Esta situación obliga a los agentes a tener que desplazarse hasta la oficina para registrar que han realizado una gestión con un inmueble (Ej: enseñar un inmueble) cuando esto se podría hacer in-situ.
- ✓ **Navegación incómoda por el portal desde móviles:** El diseño no se adapta a las pantallas más pequeñas, como es el caso de los teléfonos móviles. La navegación se hace pesada y, seguramente, invite a los visitantes web a abandonar la página.
- ✓ **La información de las secciones no se adapta a las especificaciones del inmueble:** Tal y como señalamos antes, se agrega demasiada información en blanco en muchos inmuebles.
- ✓ **¿Qué buscan los usuarios web?:** Esta es otra de las preguntas de los agentes. No es posible determinar qué inmuebles resultan más interesantes ni cuáles son las búsquedas más comunes de los internautas. Tampoco existe la posibilidad de enviar ofertas a los usuarios o informarles acerca de novedades en la página que cumplen sus criterios de búsqueda.
- ✓ **No se incorpora facturación ni contabilidad:** Es una labor costosa y normalmente se acaba recurriendo a Excel.

- ✓ **El programa no guarda una lista de interesados:** No se permite registrar clientes interesados en algún inmueble en particular. Es una funcionalidad muy básica que el software actual no incluye. Obliga a recoger esta información en papeles y cuadernos que se pierden o se olvidan, dificultando la labor del negocio.
- ✓ **No existen alertas sobre los vencimientos de contrato:** Cuando un contrato vence, sería conveniente generar alertas a modo recordatorio.
- ✓ **Escaso filtrado de datos:** Tampoco se puede realizar un filtrado de datos en la mayoría de los listados. Por ejemplo: si se quieren visualizar por separado los propietarios de las viviendas en alquiler y en venta, no es posible, aparecen todos juntos.
- ✓ **Ordenación no modificable:** los listados aparecen ordenados alfabéticamente, pero no se puede modificar esta visualización. Resulta incómodo pues normalmente los agentes prefieren visualizarlos por orden de inscripción.
- ✓ **Información redundante:** ¿Y si existe un mismo propietario para varios inmuebles? Resulta que la aplicación obliga a registrar los datos varias veces en lugar de seleccionar un propietario ya existente.

2.3. Alcance del proyecto

El proyecto desarrollado contempla los siguientes puntos que se desarrollarán en detalle en este documento:

- **Plan de proyecto:** se especifica la metodología seguida para cumplir con los requisitos de las distintas entregas del proyecto, las especificaciones funcionales y no funcionales del software así como los servicios básicos de mantenimiento.
- **Diseño técnico de la solución:** se incluye el diseño de la arquitectura lógica y física de la solución a implementar. Se utiliza el framework ASP.NET MVC5.
- **Creación del portal** corporativo de la empresa.

- **Manual de usuario:** se anexa el documento con las instrucciones necesarias para dar de alta contenido en la aplicación.

2.4. Metodología

Es común que las empresas desarrolladoras de software definan una metodología desde los inicios del proyecto y se la presenten a los clientes en la propuesta de proyecto, de modo que ambas partes sepan cómo van a interactuar: la periodicidad de las reuniones, la frecuencia de entregas parciales, etc.

Para este proyecto se propone el uso de una metodología con un enfoque basado en metodologías ágiles que permitirá realizar entregas progresivas de la solución. Se plantea la ejecución del proyecto siguiendo el uso de metodologías ágiles, concretamente, se propone el uso de Scrum para cada fase del proyecto.

Scrum centra sus esfuerzos en desarrollar un producto que satisfaga las expectativas del cliente por lo que establece un plazo de tiempo máximo en el cual ir obteniendo un feedback sobre la evolución del mismo. Las iteraciones de scrum (sprints) tienen una duración fija de un máximo de 4 semanas.

La construcción del proyecto es similar a un puzzle o lego en el que Las piezas del proyecto se conciben de modo que cada una aporta un valor al cliente. Cada pieza se construye en un sprint (periodo corto que siempre tiene la misma duración), tras el que el cliente puede solicitar una entrega y proceder a su revisión.

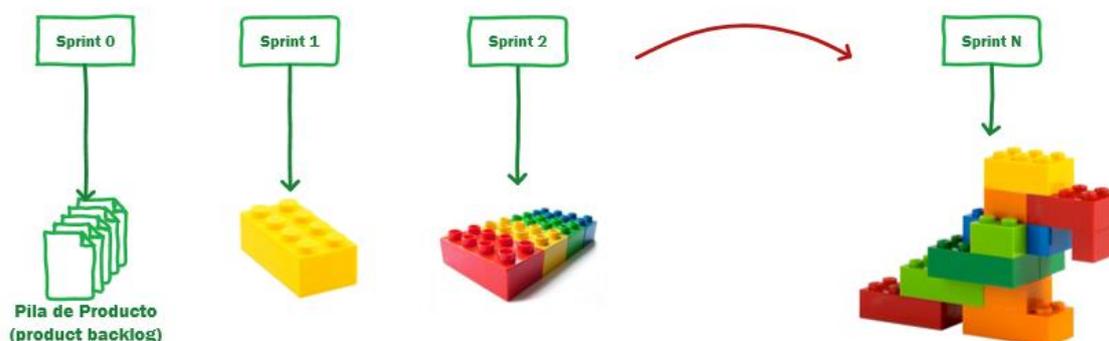
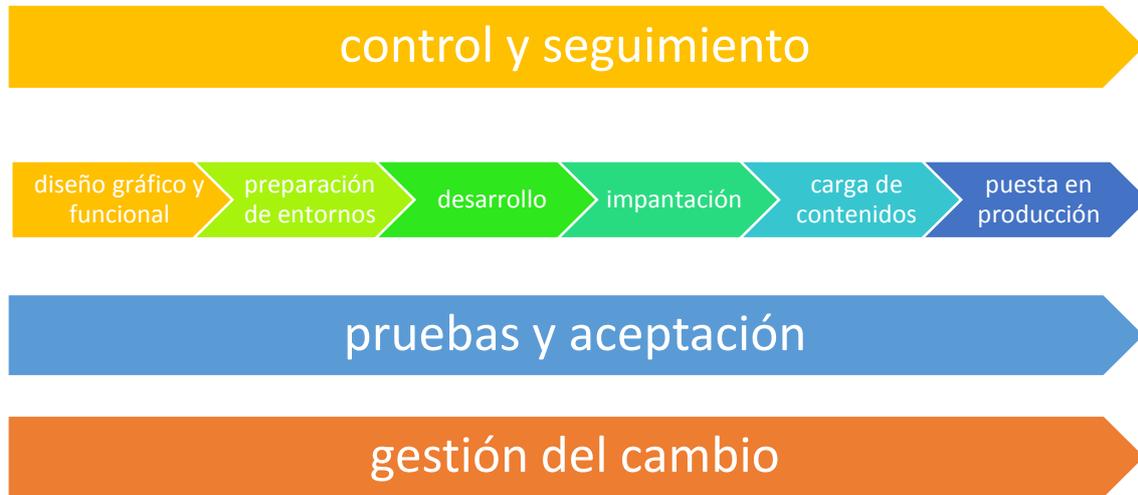


Ilustración 6: Scrum, metodología ágil de desarrollo software

A continuación se describen las actividades propias de cada fase del proyecto.



2.4.1. Control y Seguimiento

Durante todo el desarrollo del proyecto se realiza un seguimiento que comprende todas las tareas relacionadas con la organización, coordinación, seguimiento y control del proyecto desde su comienzo hasta el fin. Entre ellas:

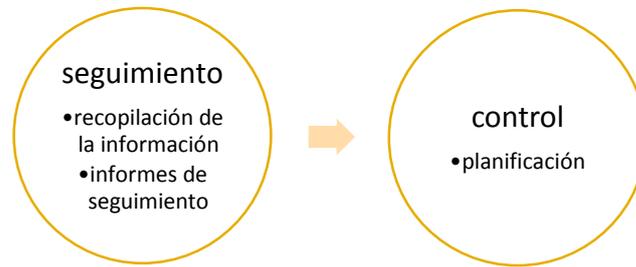
- Reuniones de kick-off
- Comités de seguimiento:
- Comités de dirección
- Gestión de riesgos
- Gestión del alcance

Después de cada reunión se genera un acta que recoja los acuerdos tomados entre el cliente y los desarrolladores, los puntos pendientes, las acciones a emprender, las responsabilidades de cada acción y la fecha de compromiso de finalización.

Las reuniones presenciales de seguimiento tienen una frecuencia entre 2-4 semanas para realizar un seguimiento del estado del proyecto. Esto no descarta mantener una comunicación vía telefónica o correos electrónicos.

Aplicación de Gestión Inmobiliaria en la Nube

En esta fase también se define el procedimiento de gestión del alcance y gestión de los riesgos. Con ello se pretende evitar el mal entendimiento de los requisitos del proyecto.



2.4.2. Diseño gráfico y funcional

La fase de diseño gráfico y funcional se desglosa en:

- ➔ Diseño técnico y arquitectura
- ➔ Maquetación

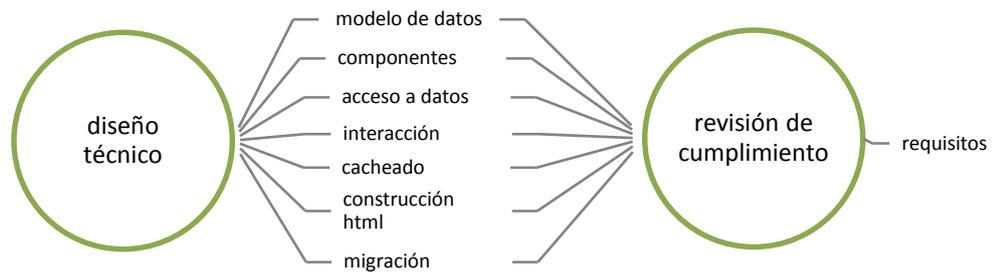
2.4.2.1. Diseño técnico y arquitectura

Una vez realizado el diseño funcional de la aplicación se procede a elaborar el diseño técnico de la solución incluyendo el diseño de los componentes, el diseño de las interfaces, integraciones y servicios web. Además, se incluye el diseño de la arquitectura.

Es especialmente importante no dejar puntos abiertos en el **diseño técnico** de la solución. Este debe incluir:

- Un mapa de sistemas
- El diseño lógico de la arquitectura
- El modelo de datos o tipos de contenido
- El diseño técnico de los componentes
- El diseño técnico de las integraciones
- El diseño de las interfaces (si las hubiere)

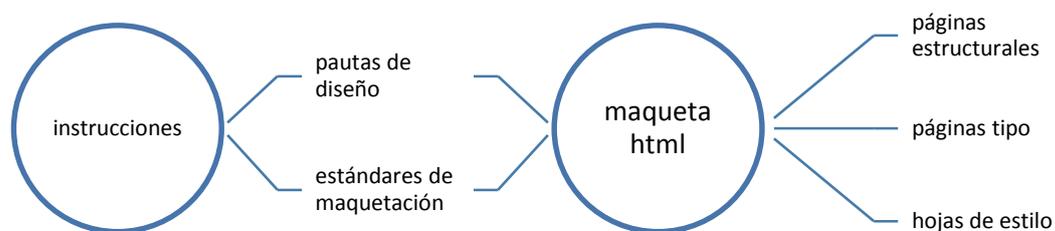
El **diseño de la arquitectura** debe incluir el dimensionamiento.



2.4.2.2. Maquetación

La maquetación diseñada será utilizada en la fase de construcción. Se adaptará a las necesidades y preferencias del cliente. Se hará especial hincapié en que los diseños sean responsivos y se adapten a distintos tipos de dispositivos y navegadores. Las especificaciones de la maquetación deben incluir:

- Maquetación de páginas estructurales.
- Maquetación de páginas tipo, formularios, etc.
- Hojas de estilo CSS.



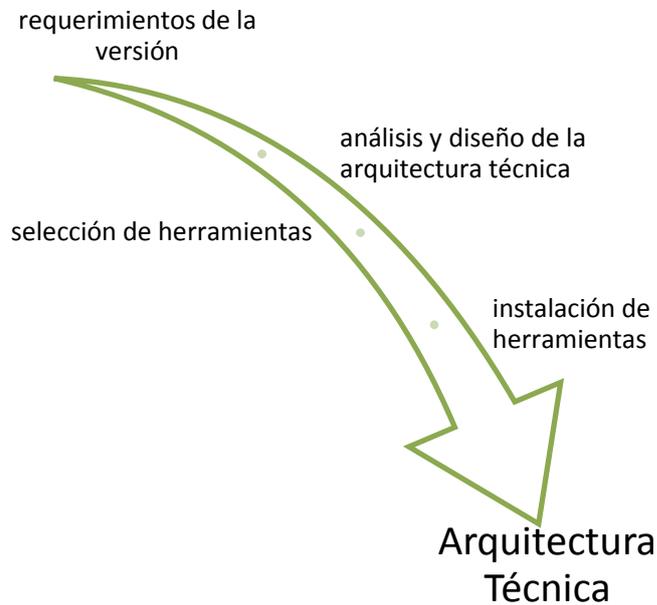
2.4.3. Preparación de entornos

En esta fase se realiza la instalación de los entornos de preproducción y producción. Se acordarán previamente los servicios de hosting a contratar así como los costes anuales que suponen. El cliente podrá decidir si desea un VPS hosting o Cloud Hosting.

Al cabo de esta fase se verá:

Aplicación de Gestión Inmobiliaria en la Nube

- El entorno de desarrollo instalado.
- El entorno de preproducción instalado.
- El entorno de producción instalado.



2.4.4. Desarrollo

En la fase de desarrollo se genera el código de todos los componentes de la aplicación, las plantillas, flujos de publicación, interfaces, servicios web, tipos de contenido, etc.

Se sugiere desarrollar en primer lugar las funcionalidades más complejas y críticas del proyecto. Para ello se debería disponer de un mapa de dependencias con el fin de identificarlas.

Seguir los estándares a la hora de desarrollar el código ayuda a una mejor legibilidad y mantenimiento del mismo.

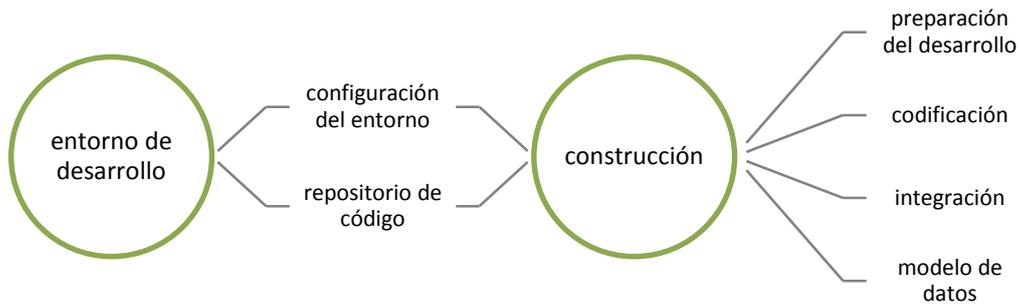
Se propone el uso de un gestor de versiones del código, como Team Foundation Server o GitHub.

Al término de la fase se verá el código fuente de:

- Componentes

Aplicación de Gestión Inmobiliaria en la Nube

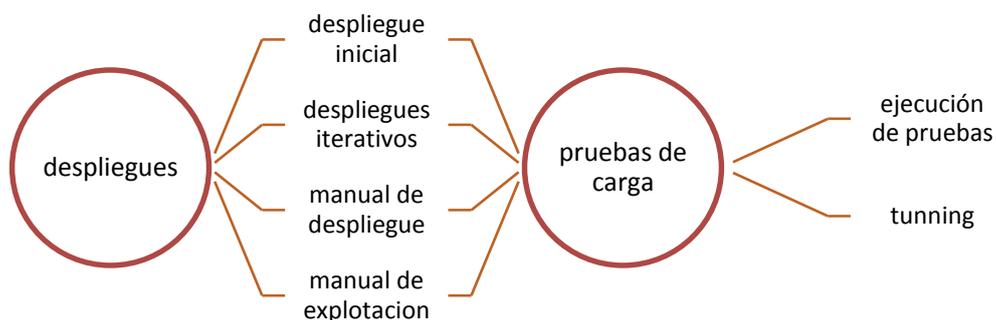
- Plantillas
- Tipos de contenido
- Servicios Web
- Integraciones
- Librerías
- Paquetes desplegables de la aplicación



2.4.5. Implantación

En esta fase se realiza el despliegue de la aplicación en el entorno de preproducción y producción de acuerdo al manual de instalación que se aporta. La realización de varios despliegues de prueba desde etapas tempranas del proyecto permite minimizar el riesgo y evitar el efecto túnel. Al cabo de esta fase se facilitarán:

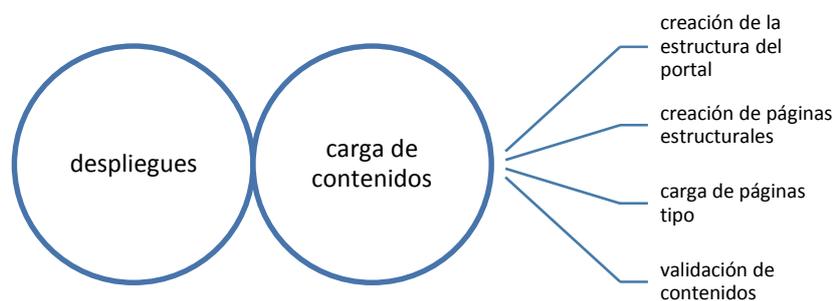
- Los manuales de instalación
- El manual de explotación.



2.4.6. Carga de contenidos

En esta fase se realiza la carga de contenidos a la base de datos y a la aplicación. Para ello es necesario disponer de los entornos de preproducción y producción correctamente configurados.

Se recomienda realizar cargas de contenidos incrementales y su mostración a los usuarios desde fases tempranas así como la optimización de las imágenes del portal para no penalizar su rendimiento.



2.4.7. Pruebas y Aceptación

En la fase de pruebas se realizará el diseño, ejecución y evaluación de las pruebas con el fin de asegurar el correcto funcionamiento del sistema global. Se contempla la realización de pruebas integradas, de carga y de aceptación de usuarios (UAT).



La fase de pruebas no debe ser infravalorada ya que siempre acaba habiendo más defectos de los inicialmente previstos. Se recomienda una prueba de concepto en las fases iniciales del proyecto a la vez que se realiza el diseño técnico. Al cabo de esta fase se generarán los siguientes entregables:

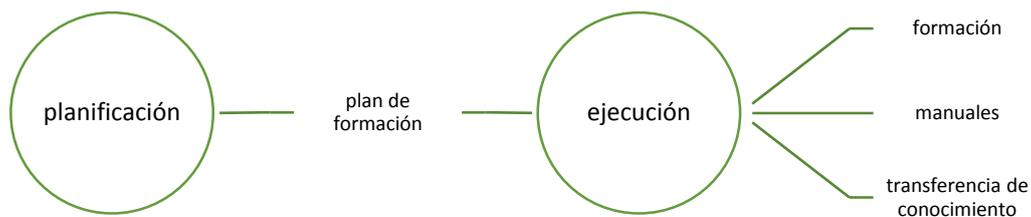
- Diseño del plan de pruebas
- Informe de ejecución de las pruebas

2.4.8. Gestión del Cambio

En esta fase se realizan todas las tareas relacionadas con la Gestión del cambio del proyecto a los colectivos afectados con la introducción de la aplicación así como la transferencia de conocimiento al equipo encargado de explotar la aplicación.

Es importante involucrar a los usuarios desde etapas tempranas del proyecto así como formar a los colectivos involucrados en él. Al término de esta fase se generarán:

- Manuales de usuario
- Plan de formación
- Transferencia de formación al equipo de explotación



2.5. Web App vs. Cloud App

Actualmente la línea que separa una aplicación en la nube (cloud app) de una aplicación web (web app) es muy fina. Existen similitudes entre ambas aunque si se indaga en la lógica de ambas se puede comprobar que existen diferencias notables.

Muchos consideran que una aplicación de nube es en realidad una aplicación web evolucionada. Se utiliza de la misma forma para acceder a servicios online como en las aplicaciones web, pero no siempre depende de un navegador para trabajar. Una aplicación en la nube se suele caracterizar por:

- Los datos se encuentran almacenados en una estructura nube.
- Los datos pueden estar localmente almacenados en caché para el modo sin conexión.
- La aplicación soporta diferentes necesidades de los usuarios, como backup de datos.

Aplicación de Gestión Inmobiliaria en la Nube

- Se puede usar desde un navegador web u otras aplicaciones como móviles o de escritorio.
- Se puede usar para acceder a un amplio rango de servicios como almacenamiento u otras aplicaciones.

Entre los ejemplos más comunes de aplicaciones en nube cabe destacar iCloud, Google Drive, Ubuntu One, OneDrive y Dropbox. Otros ejemplos relacionados con el correo electrónico como Google, Yahoo y Hotmail pueden parecer menos obvios pero en realidad dependen de una tecnología en la nube y se encuentran disponibles offline si los usuarios han seleccionado configurarlos como tal. Las web apps, sin embargo, están diseñadas exclusivamente para usarse a través de un navegador web.

A efectos prácticos, cuando se desarrolla una web app no se tiene por qué desechar la opción de convertirla posteriormente a una cloud app, es lo que se conoce como una aplicación **cloud-aware**. ¿Por qué puede ser conveniente desarrollar una aplicación cloud-aware si esta se va a transformar en cloud app igualmente? La respuesta podría estar en los costes que cada una de ellas conlleva.

A menudo se tiende a confundir las aplicaciones web con aplicaciones en nube por el simple hecho de que son accesibles mediante un navegador. Incluso muchos negocios empaquetan aplicaciones web y las venden como "software en la nube" por ser una palabra comercialmente atractiva cuando en realidad están ofreciendo una suite de funcionalidades mucho más pobre que la que ofrece una aplicación en la nube.

2.6. Arquitectura Lógica

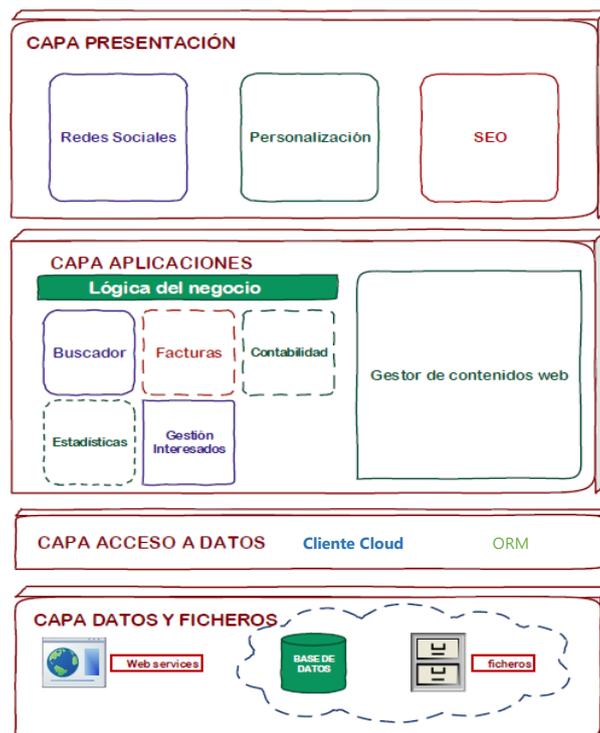
Es muy común confundir el término **patrón de diseño** con el término **arquitectura**. Puede ocurrir que por el hecho de hacer una separación de componentes, como ocurre en MVC, se crea que se está trabajando con una arquitectura n-layer y esto no es así.

Una arquitectura lógica debería ser diseñada sin tener en cuenta la tecnología con la que se va a implementar cada funcionalidad. Una vez diseñada esta, se procedería a elegir qué alternativa tecnológica se adapta mejor a los resultados que se pretenden conseguir. No realizar un diseño lógico previo puede desembocar en un diseño físico poco refinado y desorganizado.

Aplicación de Gestión Inmobiliaria en la Nube

El diseño lógico se utiliza en las especificaciones funcionales que dan lugar al diseño físico de la solución. Es conveniente plantear varios modelos lógicos hasta escoger uno.

Concretamente, en este proyecto se pensó inicialmente en una arquitectura de tres capas: presentación, aplicaciones y datos. El problema es que la aplicación a desarrollar no siempre realizaba un mismo acceso a los datos, por lo que se decidió incorporar una capa más a la arquitectura lógica: la capa de acceso a datos.



En la imagen superior queda representada la organización lógica de la aplicación. Esta presenta cuatro capas:

- ➔ **Capa de presentación:** (o capa de usuario) es la capa que ve el usuario que hace uso de la aplicación. Se encarga de presentar la información e interactuar con el usuario a través de una interfaz gráfica. Se comunica exclusivamente con la capa de aplicaciones.
- ➔ **Capa de aplicaciones:** (o capa de negocio) en ella residen los programas que se ejecutan. Se encarga de recibir las peticiones del usuario y comunicar una respuesta. Se comunica con la capa de presentación y con la capa de acceso a datos.

- **Capa de acceso a datos:** contiene la lógica principal de acceso y persistencia de datos dentro de la aplicación. Soporta el almacenamiento de datos y la concurrencia de múltiples usuarios accediendo a la información. Se comunica con la capa de aplicaciones y con la capa de datos.
- **Capa de datos:** es la capa en la que residen los datos utilizados por la aplicación. Se accede a ella desde la capa de acceso a datos.

2.6.1. Capa de Presentación

En la capa de presentación se definen 3 funciones principales: Redes Sociales, Personalización y SEO.

Se incluyen controles que facilitan la compartición del contenido de la web en las **redes sociales**, Twitter y Facebook principalmente. Con ello se pretende invitar a los usuarios de las redes sociales a entrar a la página y dar a conocer el sitio.

La **personalización** comprende desde la selección de inmuebles favoritos por los visitantes hasta la gestión de las ofertas de la página por parte de los administradores del sitio.

Por último, dada la importancia que tiene para un negocio pequeño aparecer en los primeros resultados de búsqueda de los diferentes buscadores, se incluye el **posicionamiento en buscadores** (SEO, Search Engine Optimization). Para mejorar la notoriedad de la web en los medios online es necesario cuidar el contenido de la web y su accesibilidad.

2.6.2. Capa de Aplicaciones

En la imagen se puede observar que existen funcionalidades representadas con líneas discontinuas. Estas se refieren a funcionalidades que no se van a implementar en este proyecto pero que van a intervenir a la hora escoger la tecnología con la que este se va a desarrollar.

El **buscador** permite filtrar los inmuebles ofertados en base a unos criterios (tipo de inmueble, habitaciones, precio, ubicación, etc.) Es vital tener un buscador eficiente que

disponga de los filtros adecuados para que los usuarios web puedan encontrar fácilmente lo que buscan.

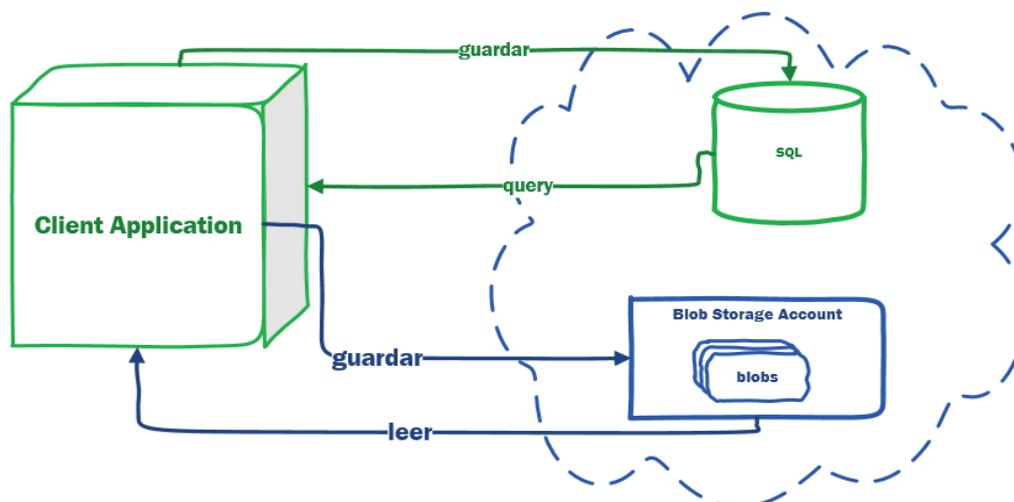
La gestión de **interesados** permite a los agentes inmobiliarios registrar los clientes que pueden ser candidatos a comprar o alquilar un inmueble. Esta funcionalidad pretende suplir la actual carencia del software de la inmobiliaria, que obliga a apuntar estas relaciones a mano.

El **gestor de contenidos** web permite a los administradores del sitio seleccionar los inmuebles que se van a publicar en la web así como configurar las ofertas y el carrousel de la página principal.

2.6.3. Capa de Acceso a Datos

Tal y como se especificaba en el apartado anterior (2.4. Web App vs. Cloud App), este proyecto está ideado para poder usarse tanto como aplicación web y como aplicación en la nube. Concretamente, el acceso a los datos es lo que diferencia ambas en cuanto a implementación se refiere.

Generalmente, una aplicación en nube almacena los datos a través de una cuenta de almacenamiento en nube (Blob Storage Account). Para acceder a esta cuenta e interactuar con ella se necesita un cliente (se incluye con líneas discontinuas). El cliente se encarga de leer y almacenar contenido en la cuenta o en la base de datos si esta se encuentra también en nube.



Aplicación de Gestión Inmobiliaria en la Nube

Pero puede ser que la aplicación no utilice los servicios de almacenamiento en nube recurriendo a un VPS (Virtual Private Server). De este modo se eliminaría el cliente que accede a la cuenta de almacenamiento en nube y se utilizaría un guardado tradicional de archivos en un sistema de ficheros.

En esta capa también se incluye el ORM (Object-Relational mapping), el conjunto de herramientas y apis para el acceso a datos a través de un modelo basado en entidades, abstrayendo al programador de la base de datos.

Dado que en este apartado sólo se define una arquitectura lógica, no será aquí donde se especifique qué tecnologías se utilizan para cada caso, sino en el apartado de Diseño del Software.

2.6.4. Capa de Datos

En la capa de datos se incluyen los posibles servicios web de los que pueda hacer uso la aplicación. En esta entrega del proyecto no se hace uso de ninguno aunque no se descarta tener que disponer de alguno en ampliaciones futuras de la aplicación.

Tal y como se explicaba anteriormente, es posible que la base de datos resida en nube o no, según cómo se elija hospedar la aplicación y se utilice un sistema de archivos tradicional o una cuenta de almacenamiento en nube. En cualquier caso, estos sistemas de almacenamiento residen en la capa de datos.

En el siguiente punto se explican las diferencias existentes entre el uso de nube o VPS para hospedar la aplicación.

2.7. Cloud hosting vs. VPS

Muchos usuarios piensan que VPS (Virtual Private Server) y el cloud hosting es lo mismo cuando esto no es así. Las diferencias son verdaderamente notables aunque a primera vista no pueda parecerlo. Sin embargo, en las grandes compañías de negocios el uso de una u otra opción supone grandes cambios.

Aplicación de Gestión Inmobiliaria en la Nube

Un VPS es físicamente un único servidor que puede ser compartido por un número limitado de usuarios que poseen una cuenta para acceder al mismo. La principal diferencia en la instalación



de ambos recursos es que en VPS los usuarios tienen un mayor control sobre su sección del servidor. La virtualización es lo que divide a un usuario de otro, aunque ello no evita el hecho de que un mal uso del servidor por otro usuario no pueda afectarle.



En el caso de los servidores en nube, los recursos son distribuidos en varios servidores físicos. La redundancia es lo que permite utilizar la red de servidores sin ningún tipo de problema pues si un servidor falla, los datos se encuentran replicados en otra máquina. El usuario que contrata el servicio ve el recurso como un

servidor pero en realidad hay múltiples máquinas detrás.

En cuanto al coste, los servidores en nube ofrecen un servicio mucho más avanzado y por ello son también más caros. Normalmente se paga por el uso real de los recursos, mientras que en los VPS se paga una cuota fija. Esto también quiere decir que las aplicaciones alojadas en nube son más escalables pues no se oponen a las solicitudes de más recursos. En el caso de los VPS esto no es así, la escalabilidad de la aplicación está limitada a los recursos de la máquina que la aloja.

2.8. Blob Storage Account vs. File System

Desde el punto de vista de la implementación, ¿es lo mismo orientar una aplicación para su uso en nube que para su uso en un VPS? Es posible usar una aplicación web en un hosting en nube aunque seguramente no se esté aprovechando todo su potencial. Es decir, se puede trabajar con un sistema de ficheros que almacene diferentes documentos

en los directorios del proyecto pero es más correcto usar cuentas de almacenamiento en nube.

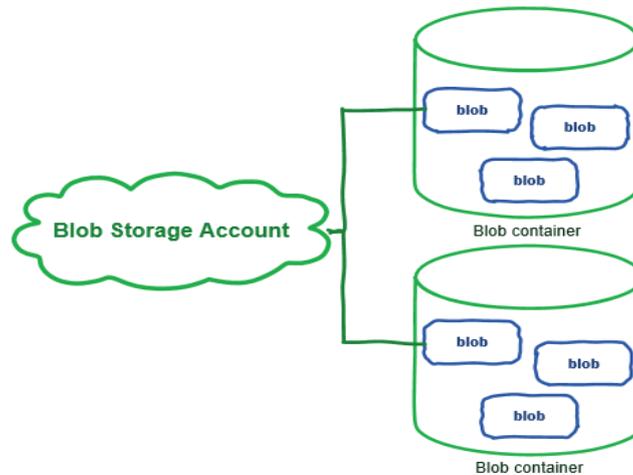


Ilustración 7: Blob Storage Account

Cuando se desea guardar un archivo en la aplicación no se trabaja con carpetas o directorios. Se trabaja con cuentas de almacenamiento en nube (Blob Storage). Estas cuentas nos permiten guardar un documento sin necesidad de especificar un directorio concreto. Se maneja unidades más abstractas, cuya dirección de almacenamiento exacta se desconoce, lo que devuelve la cuenta al guardar el contenido es una URL. De este modo, cuando se guarda un archivo, se encapsula este en un "Blob" (gota) que se guarda en un contenedor. Ese contenedor no se sabe dónde está ya que está en la nube, replicado múltiples veces en varias máquinas. ¿Qué aporta esto? Seguridad: los recursos van a estar siempre disponibles ya que el fallo de un servidor no provocará que el archivo se encuentre inaccesible, estará replicado geográficamente.

¿Se debería trabajar siempre en nube? Depende del modelo de negocio. Hay que tener en cuenta los costes que ello supone y si realmente va a aportar ventajas palpables a los usuarios. Para empresas grandes, seguramente sea una apuesta más segura, mientras que para las empresas pequeñas y medianas pueda resultar un servicio cuyos costes por uso y mantenimiento estén muy por encima de sus posibilidades. El problema es que muchos clientes piden servicios en nube refiriéndose a una aplicación web, pues no acaban de discernir ambos conceptos.

3. Requisitos del Proyecto

3.1. Requisitos Funcionales

Los requisitos funcionales definen las funcionalidades a cumplir del software a desarrollar. Estos se detallan describiendo los comportamientos del sistema para los diferentes casos de uso.

Ya que intervienen varios tipos de usuarios en la aplicación, se especifica en cada requisito el actor que puede realizar la acción:

- ➔ **Genérico**: cualquier usuario visitante del sitio que puede no disponer de cuenta de usuario.
- ➔ **Visitante**: usuario del sitio sin privilegios de administración.
- ➔ **Administrador**: usuario que cuenta con todos los privilegios y puede gestionar el contenido de la página.

3.1.1. EF1: Acceso a la página principal

Código de Especificación Funcional:	EF1
Nombre	Acceso a la página principal sin inicio de sesión previo
Versión:	1.0
Actores:	Genérico
Descripción:	El usuario accede desde un navegador a la dirección web de la página Home.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web.
Postcondiciones:	El sistema redirige a la página principal del portal inmobiliario.
Flujo	<ol style="list-style-type: none">1) El Actor accede a la página principal del portal.2) El Sistema permite el acceso a la aplicación.3) La página principal sin una sesión iniciada se presenta con la siguiente estructura:

	<ul style="list-style-type: none">• Un menú de navegación con diferentes secciones de inmuebles y acceso a login y registro en la aplicación.• Un carrousel con imágenes.• Un buscador de inmuebles.• Una sección de ofertas de inmuebles. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al rellenar los filtros de búsqueda y pulsar sobre el botón de búsqueda se redirige a una página de resultados con los criterios especificados.</p> <p>Al pulsar sobre Iniciar Sesión se redirige a la página login.</p> <p>Al pulsar sobre Registrarse se redirige a la página de registro.</p> <p>Al pulsar sobre una oferta se redirige al detalle del inmueble ofertado.</p>
--	--

Alternativa 1: Acceso a la página principal por un usuario Administrador

Actores	Administrador
Flujo	<ol style="list-style-type: none">1) El Actor introduce su identificador y contraseña en la página login.2) El Sistema comprueba si ha habido autenticación y permite el acceso a la aplicación.3) La página principal con una sesión iniciada se presenta con la siguiente estructura:<ul style="list-style-type: none">• Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación.• Acceso a favoritos.• Acceso a administración.• Un carrousel con imágenes.• Un buscador de inmuebles.• Una sección de ofertas de inmuebles. <p>Al pulsar sobre favoritos se accede al listado de inmuebles que el usuario haya marcado en sesiones anteriores.</p>

	Al pulsar sobre administración se redirige a la página de administración.
	El resto de acciones son análogas al actor genérico.
Requisitos	El usuario debe estar registrado en la aplicación como Administrador
Alternativa 2: Acceso a la página principal por un usuario Visitante	
Actores	Visitante
Flujo	<ol style="list-style-type: none"> 1) El Actor introduce su identificador y contraseña en la página login. 2) El Sistema comprueba si ha habido autenticación y permite el acceso a la aplicación. 3) La página principal con una sesión iniciada se presenta con la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Acceso a favoritos. • Un carrousel con imágenes. • Un buscador de inmuebles. • Una sección de ofertas de inmuebles. <p>Al pulsar sobre favoritos se accede al listado de inmuebles que el usuario haya marcado en sesiones anteriores.</p> <p>El resto de acciones son análogas al actor genérico.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Visitante.

3.1.2. EF2: Acceso a Secciones de Inmuebles

Código de Especificación Funcional:	EF2
Nombre	Acceso a las secciones de inmuebles
Versión:	1.0
Actores:	Genérico
Descripción:	El usuario accede desde un navegador a la dirección web de una sección de inmuebles.
Prioridad:	Alta

Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página principal.
Postcondiciones:	El sistema redirige a la página de sección de inmuebles correspondiente.
Flujo	<ol style="list-style-type: none"> 1) El Actor accede a una sección de inmuebles del portal. 2) El Sistema permite el acceso a la aplicación. 3) Una sección de inmuebles sin una sesión iniciada se presenta con la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a login y registro en la aplicación. • Un listado de inmuebles con información básica. • Un buscador de inmuebles. • Botones de acceso a detalle y añadir a favoritos. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al rellenar los filtros de búsqueda y pulsar sobre el botón de búsqueda se redirige a una página de resultados con los criterios especificados.</p> <p>Al pulsar sobre Iniciar Sesión se redirige a la página login.</p> <p>Al pulsar sobre Registrarse se redirige a la página de registro.</p> <p>Al pulsar sobre un inmueble se redirige a la página de detalle del mismo.</p> <p>Al pulsar sobre el botón favorito se redirige a la página login.</p>
Alternativa 1: Acceso a la página principal por un usuario Administrador	
Actores	Administrador
Flujo	<ol style="list-style-type: none"> 1) El Actor introduce su identificador y contraseña en la página login. 2) El Sistema comprueba si ha habido autenticación y permite el acceso a la aplicación. 3) La página principal con una sesión iniciada se presenta con la siguiente estructura:

	<ul style="list-style-type: none">• Un menú de navegación con diferentes secciones de inmuebles y acceso cerrar sesión en la aplicación.• Acceso a favoritos.• Acceso a administración.• Un listado de inmuebles con información básica.• Un buscador de inmuebles.• Botones de acceso a detalle y añadir a favoritos. <p>Al pulsar sobre el botón favorito de un inmueble se agrega el inmueble al listado de favoritos del usuario.</p> <p>Al pulsar sobre favoritos se accede al listado de inmuebles que el usuario haya marcado en sesiones anteriores.</p> <p>Al pulsar sobre administración se redirige a la página de administración.</p> <p>Las demás acciones son análogas al actor genérico.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador
Alternativa 2: Acceso a la página principal por un usuario Visitante	
Actores	Visitante
Flujo	<ol style="list-style-type: none">1) El Actor introduce su identificador y contraseña en la página de login.2) El Sistema comprueba si ha habido autenticación y permite el acceso a la aplicación.3) La página principal con una sesión iniciada se presenta con la siguiente estructura:<ul style="list-style-type: none">• Un menú de navegación con diferentes secciones de inmuebles y acceso cerrar sesión en la aplicación.• Acceso a favoritos.• Un listado de inmuebles con información básica.• Un buscador de inmuebles.• Botones de acceso a detalle y añadir a favoritos. <p>Al pulsar sobre el botón favorito de un inmueble se agrega el inmueble al listado de favoritos del usuario.</p>

	Al pulsar sobre favoritos se accede al listado de inmuebles que el usuario haya marcado en sesiones anteriores. Las demás acciones son análogas al actor genérico.
Requisitos	El usuario debe estar registrado en la aplicación como Visitante.

3.1.3. EF3: Acceso a la página Favoritos

Código de Especificación Funcional:	EF3
Nombre	Acceso a la página favoritos
Versión:	1.0
Actores:	Visitante y Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de una página favoritos
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página principal. Estar autenticado en el portal.
Postcondiciones:	El sistema redirige a la página favoritos.
Flujo	<p>4) El Actor se autentica en el sistema 5) El Sistema comprueba las correctas credenciales y permisos del usuario. 6) El sistema redirige al usuario a la página favoritos. 7) La página favoritos presenta la siguiente estructura:</p> <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Un listado de inmuebles con información básica, previamente marcados por el usuario. • Un buscador de inmuebles. • Enlace a sección de administración si el usuario autenticado es administrador. • Botones de acceso a detalle y desagregar favoritos. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p>

	<p>Al rellenar los filtros de búsqueda y pulsar sobre el botón de búsqueda se redirige a una página de resultados con los criterios especificados.</p> <p>Al pulsar sobre un inmueble se redirige a la página de detalle del mismo.</p> <p>Al pulsar sobre el botón desagregar favorito se elimina el inmueble del listado de favoritos.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Visitante o Administrador.

3.1.4. EF4: Acceso a la página de Registro

Código de Especificación Funcional:	EF4
Nombre	Acceso a la página de registro
Versión:	1.0
Actores:	Genérico
Descripción:	El usuario accede desde un navegador a la dirección web de la página de registro de la aplicación.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página principal.
Postcondiciones:	El sistema redirige a la página de registro.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea registrarse: <ol style="list-style-type: none"> a. Insertando un nombre y contraseña nuevos b. Mediante una cuenta Facebook. c. Mediante una cuenta Google. 2) El sistema comprueba si el usuario ya existe. Si así fuera muestra un mensaje de error. 3) El Sistema registra las credenciales y asigna permisos de Visitante al usuario. 4) El sistema redirige al usuario a la página principal.
Requisitos	Ninguno.

3.1.5. EF5: Acceso a la página Login

Código de Especificación Funcional:	EF5
Nombre	Acceso a la página login
Versión:	1.0
Actores:	Genérico
Descripción:	El usuario accede desde un navegador a la dirección web de la página login de la aplicación.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página principal.
Postcondiciones:	El sistema redirige a la página login.
Flujo	<ol style="list-style-type: none">1) El Actor indica las credenciales con las que desea iniciar sesión.2) El Sistema comprueba las correctas credenciales y permisos del usuario.3) El sistema redirige al usuario a la página principal.
Requisitos	El usuario debe estar registrado en la aplicación como Visitante o Administrador.

3.1.6. EF6: Acceso a la página de Administración

Código de Especificación Funcional:	EF6
Nombre	Acceso a la página de Administración
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página de Administración de la aplicación.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página principal. Autenticarse en el sistema como usuario Administrador.
Postcondiciones:	El sistema redirige a la página de Administración.

Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El usuario navega a la página de Administración. 4) El sistema comprueba si el usuario posee permisos de Administrador. 5) El sistema redirige al usuario a la página de Administración. 6) La página de Administración presenta la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Acceso a la gestión de inmuebles. • Acceso a la gestión de tipos de inmueble. • Acceso a la gestión de tipos de estados del inmueble. • Acceso a la gestión de personas. • Acceso a la gestión de interesados. • Acceso a la gestión de tipos de contacto. • Acceso a la gestión de tipos de persona. • Acceso a la configuración de ofertas de la página home. • Acceso a la configuración del carrousel de la página home. • Acceso a la gestión de poblaciones.
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.7. EF7: Buscador

Código de Especificación Funcional:	EF7
Nombre	Buscador del portal
Versión:	1.0
Actores:	Genérico
Descripción:	El usuario selecciona uno o varios filtros disponibles del buscador.

Aplicación de Gestión Inmobiliaria en la Nube

Prioridad:	Alta
Precondiciones:	Encontrarse en la página principal, sección de inmuebles, favoritos o resultados.
Postcondiciones:	El sistema realiza una búsqueda con los criterios especificados en los filtros del buscador.
Flujo	<ol style="list-style-type: none">1) El Actor selecciona los criterios de búsqueda mediante los filtros disponibles en el buscador:<ol style="list-style-type: none">a. Operación (Alquilar, Vender...)b. Ubicación (Madrid, Barcelona...)c. Precio máximod. Número de habitacionese. Superficief. Tipo (Piso, Chalet, Finca...)2) El Sistema realiza sobre los inmuebles publicados en el portal.3) El sistema redirige al usuario a la página de resultados.
Requisitos	Ninguno.

3.1.8. EF8: Acceso a la página de Resultados de Búsquedas

Código de Especificación Funcional:	EF8
Nombre	Acceso a la página de resultados de búsquedas
Versión:	1.0
Actores:	Genérico
Descripción:	El usuario especifica los filtros de búsqueda en el buscador y pulsa el botón buscar.
Prioridad:	Alta
Precondiciones:	Haber especificado filtros de búsqueda en el buscador.
Postcondiciones:	El sistema redirige a la página de resultados de búsqueda.
Flujo	<ol style="list-style-type: none">1) El Actor rellena el formulario de búsqueda y pulsa el botón buscar.

	<ol style="list-style-type: none"> 2) El Sistema realiza una búsqueda de los inmuebles publicados que coinciden con los criterios de la búsqueda. 3) El sistema redirige al usuario a la página de resultados de búsqueda. 4) La página de resultados de búsqueda presenta la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a iniciar/cerrar sesión en la aplicación. • Un listado de inmuebles filtrados según los criterios especificados por el usuario. (Puede aparecer vacío si no hay inmuebles con dichos criterios) • Un buscador de inmuebles.
Requisitos	Ninguno.

3.1.9. EF9: Acceso a la página de Detalle de un Inmueble

Código de Especificación Funcional:	EF9
Nombre	Acceso a la página detalle de un inmueble
Versión:	1.0
Actores:	Genérico
Descripción:	El usuario accede desde un navegador a la dirección web de la página detalles de un inmueble.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página principal, resultados de búsqueda o secciones de inmuebles.
Postcondiciones:	El sistema redirige a la página detalle de un inmueble.
Flujo	<ol style="list-style-type: none"> 1) El actor selecciona ver el detalle de un inmueble. 2) El Sistema comprueba si el inmueble existe y recoge sus datos. 3) El sistema redirige al usuario a la página favoritos.

	<p>4) La página detalle presenta la siguiente estructura:</p> <ul style="list-style-type: none"> • Una sección con la siguiente información del inmueble: <ul style="list-style-type: none"> ○ Dirección ○ Población ○ Número de habitaciones ○ Aseos ○ Precio ○ Superficie ○ Certificación energética ○ Detalles ○ Fotos • Un botón para agregar el inmueble al listado de favoritos. • Un menú de navegación con las secciones de inmuebles y inicio/cierre de sesión. • Sección favoritos (si el usuario ha iniciado sesión previamente) • Sección administración (si el usuario autenticado es administrador) <p>Al pulsar sobre cualquier foto se abre un visualizador de fotos que permite ver las imágenes ampliadas.</p>
Requisitos	Ninguno.

3.1.10. EF10: Acceso a la página Creación de un Inmueble (Administración)

Código de Especificación Funcional:	EF10
Nombre	Acceso a la página creación de un inmueble
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página creación de inmuebles
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página principal.

	Estar autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	El sistema redirige a la página creación de inmuebles.
Flujo	<ol style="list-style-type: none"> 1) El sistema comprueba si el usuario posee privilegios de administrador. 2) El usuario accede a la sección de inmuebles. 3) El sistema redirige al usuario a la página creación de inmuebles. 4) La página creación de inmuebles presenta la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Un formulario para introducir los datos del inmueble. • Un formulario para introducir los datos del propietario. • Un formulario para introducir los datos de tipo y estado del inmueble. • Un formulario para agregar los documentos relacionados con el inmueble desde la computadora. 5) Al pulsar sobre el botón crear el sistema comprueba la validez de los datos. 6) Si los datos son correctos el sistema registra el inmueble, el propietario y los documentos en la base de datos. 7) Si el registro se ha producido correctamente el sistema redirige al usuario al índice de inmuebles.
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.11. EF11: Acceso a la página Borrado de Inmuebles (Administración)

Código de Especificación Funcional:	EF11
Nombre	Acceso a la página borrado de un inmueble

Aplicación de Gestión Inmobiliaria en la Nube

Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página borrado de inmuebles
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Estar autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	El sistema redirige a la página de borrado de inmuebles.
Flujo	<ol style="list-style-type: none">1) El sistema comprueba si el usuario posee privilegios de administrador.2) El usuario accede a la sección de inmuebles.3) El sistema redirige al usuario a la página listado de inmuebles.4) El usuario selecciona eliminar un inmueble.5) La página borrado de inmuebles presenta la siguiente estructura:<ul style="list-style-type: none">• Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación.• Un formulario con la información del inmueble y un botón de eliminar.8) Al pulsar sobre el botón borrar el sistema elimina el inmueble.9) Se redirige al usuario a la página listado de inmuebles.
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.12. EF12: Acceso a la página Listado de Inmuebles (Administración)

Código de Especificación Funcional:	EF12
Nombre	Acceso a la página listado de inmuebles

Aplicación de Gestión Inmobiliaria en la Nube

Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página listado de inmuebles.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Encontrarse autenticado en el sistema como usuario administrador.
Postcondiciones:	El sistema redirige a la página listado de inmuebles.
Flujo	<ol style="list-style-type: none">1) El Actor indica las credenciales con las que desea registrarse.2) El Sistema comprueba las correctas credenciales y permisos del usuario.3) El sistema comprueba si el usuario posee permisos de administrador.4) El usuario accede al panel de administración.5) El usuario selecciona la sección de inmuebles y accede al listado.6) El sistema redirige al usuario a la página listado de inmuebles.7) La página listado de inmuebles presenta la siguiente estructura:<ul style="list-style-type: none">• Menú de navegación con secciones de inmuebles, cierre de sesión, acceso a favoritos y administración.• Un listado de inmuebles con información básica• Un buscador de inmuebles.• Botones de acceso a detalle y agregado a favoritos. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al rellenar los filtros de búsqueda y pulsar sobre el botón de búsqueda se redirige a una página de resultados con los criterios especificados.</p> <p>Al pulsar sobre un inmueble se redirige a la página de detalle del mismo.</p> <p>Al pulsar sobre el botón desagregar favorito se elimina el inmueble del listado de favoritos.</p>

Requisitos	El usuario debe estar registrado en la aplicación como Administrador.
------------	---

3.1.13. EF13: Acceso a la página Edición un Inmueble

Código de Especificación Funcional:	EF13
Nombre	Acceso a la página edición de un inmueble
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página de edición de un inmueble.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Estar autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	El sistema redirige a la página de edición de inmuebles.
Flujo	<ol style="list-style-type: none"> 1) El sistema comprueba si el usuario posee privilegios de administrador. 2) El usuario accede a la sección de inmuebles y selecciona editar uno existente. 3) El sistema redirige al usuario a la página edición del inmuebles. 4) La página creación de inmuebles presenta la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Un formulario para editar los datos del inmueble. 10) Al pulsar sobre el botón Guardar el sistema comprueba la validez de los datos. 11) Si los datos son correctos el sistema registra las modificaciones del inmueble, el propietario y los documentos en la base de datos. 12) Si la modificación se ha producido correctamente el sistema redirige al usuario al índice de inmuebles.

Requisitos	El usuario debe estar registrado en la aplicación como Administrador.
------------	---

3.1.14. EF14: Acceso a la página Creación de Estados de Inmuebles

Código de Especificación Funcional:	EF14
Nombre	Acceso a la página alta de estados del inmueble.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página alta de estados del inmueble.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Encontrarse autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	El sistema redirige a la página alta de estados del inmueble.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El sistema comprueba que el usuario tenga permisos de administrador. 4) El sistema redirige al usuario a la página alta de estados. 5) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos • Un formulario para crear un estado nuevo. <p>Al pulsar sobre crear se validan los datos introducidos en el formulario y se procede a la creación del estado.</p> <p>Una vez creado el estado se redirige a la página listado de estados.</p>

Requisitos	El usuario debe estar registrado en la aplicación como Administrador.
------------	---

3.1.15. EF15: Acceso a la página Borrado de Estados de Inmuebles.

Código de Especificación Funcional:	EF15
Nombre	Acceso a la página borrado de estados del inmueble.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página borrado de estados del inmueble.
Prioridad:	Alta
Precondiciones:	<p>Introducir en el navegador web la dirección de la web o acceder desde la página listado de estados.</p> <p>Encontrarse autenticado en el sistema con una cuenta de administrador.</p>
Postcondiciones:	El sistema redirige a la página borrado de estados del inmueble.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El sistema comprueba que el usuario tenga permisos de administrador. 4) El sistema redirige al usuario a la página borrado de estados. 5) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos • Un formulario para borrar el estado descrito. <p>Al pulsar sobre borrar elimina el estado del sistema.</p> <p>Una vez borrado el estado se redirige a la página listado de estados.</p>

Requisitos	El usuario debe estar registrado en la aplicación como Administrador.
------------	---

3.1.16. EF16: Acceso a la página Listado de Estados de un Inmueble

Código de Especificación Funcional:	EF16
Nombre	Acceso a la página estados del inmueble.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página gestión de estados del inmueble.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Encontrarse autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	El sistema redirige a la página gestión de estados del inmueble.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El sistema comprueba que el usuario tenga permisos de administrador. 4) El sistema redirige al usuario a la página gestión de estados. 5) La página gestión de estados presenta la siguiente estructura: <ul style="list-style-type: none"> • Un listado de estados. • Enlaces para editar / borrar / crear nuevos estados. <p>Al pulsar sobre crear se redirige a la página creación de estados del inmueble.</p>

	Al pulsar sobre borrar se redirige a la página eliminar el estado.
	Al pulsar sobre editar se redirige a la página edición del estado.
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.17. EF17: Acceso a la página Edición de Estados de un Inmueble

Código de Especificación Funcional:	EF17
Nombre	Acceso a la página edición de estados del inmueble.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página edición de estados del inmueble.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Encontrarse autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	El sistema redirige a la página edición de estados del inmueble.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El sistema comprueba que el usuario tenga permisos de administrador. 4) El sistema redirige al usuario a la página edición de estados. 5) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos • Un formulario para editar un estado de inmueble.

	<p>Al pulsar sobre modificar se validan los datos introducidos en el formulario y se procede a la modificación del estado.</p> <p>Una vez modificado el estado se redirige a la página listado de estados.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.18. EF18: Acceso a la página Creación de Tipos de Inmueble

Código de Especificación Funcional:	EF18
Nombre	Acceso a la página alta de tipos de inmueble.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página alta de tipos de inmueble.
Prioridad:	Alta
Precondiciones:	<p>Introducir en el navegador web la dirección de la web o acceder desde la página de administración.</p> <p>Encontrarse autenticado en el sistema con una cuenta de administrador.</p>
Postcondiciones:	El sistema redirige a la página alta de tipos de inmueble.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El sistema comprueba que el usuario tenga permisos de administrador. 4) El sistema redirige al usuario a la página alta de tipos de inmueble. 5) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos

	<ul style="list-style-type: none"> Un formulario para crear un tipo nuevo (Piso, Chalet, Adosado, Estudio, Dúplex...) <p>Al pulsar sobre crear se validan los datos introducidos en el formulario y se procede a la creación del tipo.</p> <p>Una vez creado el estado se redirige a la página listado de tipos de inmueble.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.19. EF19: Acceso a la página Borrado de Tipos de Inmueble

Código de Especificación Funcional:	EF19
Nombre	Acceso a la página borrado de tipos de inmueble.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página borrado tipos de inmueble.
Prioridad:	Alta
Precondiciones:	<p>Introducir en el navegador web la dirección de la web o acceder desde la página de administración.</p> <p>Encontrarse autenticado en el sistema con una cuenta de administrador.</p>
Postcondiciones:	El sistema redirige a la página borrado de tipos de inmueble.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El sistema comprueba que el usuario tenga permisos de administrador. 4) El sistema redirige al usuario a la página borrado de tipos de inmueble. 5) La página presenta la siguiente estructura:

	<ul style="list-style-type: none"> • Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos • Un formulario con los datos del tipo de inmueble. <p>Al pulsar sobre borrar se elimina el tipo de inmueble del sistema.</p> <p>Una vez eliminado el tipo de inmueble se redirige a la página listado de tipos de inmueble.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.20. EF20: Acceso a la página Listado de Tipos de Inmueble

Código de Especificación Funcional:	EF20
Nombre	Acceso a la página tipos de inmueble
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página tipos de inmueble.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página registro.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El sistema redirige al usuario a la página favoritos. 4) La página presenta la siguiente estructura:

	<ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Un listado de inmuebles con información básica, previamente marcados por el usuario. • Un buscador de inmuebles. • Enlace a sección de administración si el usuario autenticado es administrador. • Botones de acceso a detalle y desagregar favoritos. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al rellenar los filtros de búsqueda y pulsar sobre el botón de búsqueda se redirige a una página resultados con los criterios especificados.</p> <p>Al pulsar sobre un inmueble se redirige a la página detalle del mismo.</p> <p>Al pulsar sobre el botón desagregar favorito se elimina el inmueble del listado de favoritos.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.21. EF21: Acceso a la página Edición de Tipos de Inmueble

Código de Especificación Funcional:	EF21
Nombre	Acceso a la página edición de tipos de inmueble.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página edición de tipos de inmueble.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.

	Encontrarse autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	El sistema redirige a la página edición de tipos de inmueble.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El sistema comprueba que el usuario tenga permisos de administrador. 4) El sistema redirige al usuario a la página tipos de inmueble y el usuario selecciona editar uno de los existentes. 5) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos • Un formulario para editar el tipo de inmueble. <p>Al pulsar sobre modificar se validan los datos introducidos en el formulario y se procede a la modificación del tipo.</p> <p>Una vez modificado el estado se redirige a la página listado de tipos de inmueble.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.22. EF22: Acceso a la página Creación de Personas

Código de Especificación Funcional:	EF22
Nombre	Acceso a la página alta de personas.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página alta de personas.
Prioridad:	Alta

Precondiciones:	<p>Introducir en el navegador web la dirección de la web o acceder desde la página de administración.</p> <p>Encontrarse autenticado en el sistema con una cuenta de administrador.</p>
Postcondiciones:	El sistema redirige a la página alta de personas.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El sistema comprueba que el usuario tenga permisos de administrador. 4) El sistema redirige al usuario a la página alta de personas. 5) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos <p>Un formulario para crear una nueva persona.</p> <p>Al pulsar sobre crear se validan los datos introducidos en el formulario y se procede a la creación de la persona.</p> <p>Una vez creada la persona se redirige a la página listado de personas.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.23. EF23: Acceso a la página Borrado de Personas

Código de Especificación Funcional:	EF23
Nombre	Acceso a la página borrado de personas.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página borrado de personas.
Prioridad:	Alta

Precondiciones:	<p>Introducir en el navegador web la dirección de la web o acceder desde la página de administración.</p> <p>Encontrarse autenticado en el sistema con una cuenta de administrador.</p>
Postcondiciones:	El sistema redirige a la página borrado de personas.
Flujo	<ol style="list-style-type: none"> 1) El Actor indica las credenciales con las que desea autenticarse. 2) El Sistema comprueba las correctas credenciales y permisos del usuario. 3) El sistema comprueba que el usuario tenga permisos de administrador. 4) El sistema redirige al usuario a la página personas y el usuario selecciona borrar una existente. 5) La página borrado presenta la siguiente estructura: <ul style="list-style-type: none"> • Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos • Un formulario con los datos de la persona que se va a eliminar. <p>Al pulsar sobre eliminar se elimina la persona seleccionada del sistema</p> <p>Una vez eliminada la persona se redirige a la página listado de personas.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.24. EF24: Acceso a la página Listado de Personas

Código de Especificación Funcional:	EF24
Nombre	Acceso a la página listado de Personas
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página listado de personas.
Prioridad:	Alta

Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página listado de personas.
Flujo	<ol style="list-style-type: none"> 1) El Sistema comprueba las correctas credenciales y permisos del usuario. 2) El sistema redirige al usuario a la página listado de personas. 3) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Un listado de personas con información básica. • Botones de acceso a detalle/creación y borrado de personas. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre crear se redirige a la página creación de personas.</p> <p>Al pulsar sobre el botón borrar se redirige a la página borrado de personas.</p> <p>Al pulsar sobre el botón editar se redirige a la página edición de personas.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.25. EF25: Acceso a la página Edición de Personas

Código de Especificación Funcional:	EF25
Nombre	Acceso a la página edición de Personas
Versión:	1.0
Actores:	Genérico
Descripción:	El usuario accede desde un navegador a la dirección web de la página listado de personas.
Prioridad:	Alta

Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página principal.
Postcondiciones:	El sistema redirige a la página registro.
Flujo	<ol style="list-style-type: none">1) El Actor indica las credenciales con las que desea registrarse.2) El Sistema comprueba las correctas credenciales y permisos del usuario.3) El sistema redirige al usuario a la página favoritos.4) La página favoritos presenta la siguiente estructura:<ul style="list-style-type: none">• Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación.• Un listado de inmuebles con información básica, previamente marcados por el usuario.• Un buscador de inmuebles.• Enlace a sección de administración si el usuario autenticado es administrador.• Botones de acceso a detalle y desagregar favoritos. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al rellenar los filtros de búsqueda y pulsar sobre el botón de búsqueda se redirige a una página de resultados con los criterios especificados.</p> <p>Al pulsar sobre un inmueble se redirige a la página de detalle del mismo.</p> <p>Al pulsar sobre el botón desagregar favorito se elimina el inmueble del listado de favoritos.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.26. EF26: Acceso a la página Creación de Tipos de Persona

Código de Especificación Funcional:

EF26

Aplicación de Gestión Inmobiliaria en la Nube

Nombre	Acceso a la página de alta de tipos de persona.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página de alta de tipos de persona.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Encontrarse autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	El sistema redirige a la página alta tipos de persona.
Flujo	<ol style="list-style-type: none">1) El Sistema comprueba las correctas credenciales y permisos del usuario.2) El sistema comprueba que el usuario tenga permisos de administrador.3) El sistema redirige al usuario a la página alta de tipos de persona.4) La página presenta la siguiente estructura:<ul style="list-style-type: none">• Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos.• Formulario para crear un tipo de persona nuevo. <p>Al pulsar sobre crear se validan los datos introducidos en el formulario y se procede a la creación del tipo de persona.</p> <p>Una vez creado el tipo de persona se redirige a la página listado de tipos de persona.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.27. EF27: Acceso a la página Borrado de Tipos de Persona

Código de Especificación Funcional:

EF27

Aplicación de Gestión Inmobiliaria en la Nube

Nombre	Acceso a la página borrado de tipos de persona.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página borrado de tipos de persona.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Encontrarse autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	1) El sistema comprueba la autenticación del usuario y los permisos sobre el sitio. 2) El sistema redirige a la página borrado de tipos de persona.
Flujo	3) La página presenta la siguiente estructura: <ul style="list-style-type: none">• Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos.• Formulario para borrar el tipo de persona. 4) Al pulsar sobre borrar se elimina el tipo de persona del sistema. 5) Una vez borrado el tipo de persona se redirige a la página listado de tipos de persona.
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.28. EF28: Acceso a la página Listado de Tipos de Persona

Código de Especificación Funcional:	EF28
Nombre	Acceso a la página listado tipos de persona.
Versión:	1.0

Aplicación de Gestión Inmobiliaria en la Nube

Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página listado de tipos de persona.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página listado de tipos de persona.
Flujo	<ol style="list-style-type: none">1) El Sistema comprueba las correctas credenciales y permisos del usuario.2) El sistema redirige al usuario a la página listado de tipos de personas.3) La página presenta la siguiente estructura:<ul style="list-style-type: none">• Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación.• Listado de poblaciones agregadas.• Acceso a borrado/edición/creación de poblaciones. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre edición se redirige a la página edición del tipo de persona.</p> <p>Al pulsar sobre borrado se redirige a la página borrado del tipo de persona.</p> <p>Al pulsar sobre crear se redirige a la página creación de un tipo de persona.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.29. EF29: Acceso a la página Edición de Tipos de Persona

Código de Especificación Funcional:

EF29

Aplicación de Gestión Inmobiliaria en la Nube

Nombre	Acceso a la página edición de tipos de persona.
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página edición de tipos de persona.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Encontrarse autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	
Flujo	<ol style="list-style-type: none">1) El Sistema comprueba las correctas credenciales y permisos del usuario.2) El sistema comprueba que el usuario tenga permisos de administrador.3) El sistema redirige al usuario a la página edición de tipos de persona.4) La página presenta la siguiente estructura:<ul style="list-style-type: none">• Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos.• Formulario para editar un tipo de persona. <p>Al pulsar sobre modificar se validan los datos introducidos en el formulario y se procede a la modificación del tipo de persona en el sistema.</p> <p>Una vez modificado el tipo de persona se redirige a la página listado de tipos de persona.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.30. EF30: Acceso a la página Creación de Interesados

Código de Especificación Funcional:	EF30
Nombre	Acceso a la página creación de interesados.
Versión:	1.0

Aplicación de Gestión Inmobiliaria en la Nube

Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página creación de interesados.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Encontrarse autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	
Flujo	<ol style="list-style-type: none">1) El Sistema comprueba las correctas credenciales y permisos del usuario.2) El sistema comprueba que el usuario tenga permisos de administrador.3) El sistema redirige al usuario a la página alta de interesados.4) La página presenta la siguiente estructura:<ul style="list-style-type: none">• Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos.• Formulario para relacionar una persona con un inmueble. <p>Al pulsar sobre crear se validan los datos introducidos en el formulario y se procede a la creación de la relación de interesados.</p> <p>Una vez creada la relación se redirige a la página listado de interesados.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.31. EF31: Acceso a la página Borrado de Interesados

Código de Especificación Funcional:	EF31
Nombre	Acceso a la página borrado de interesados.
Versión:	1.0

Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página borrado de interesados.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Encontrarse autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	
Flujo	<ol style="list-style-type: none"> 1) El Sistema comprueba las correctas credenciales y permisos del usuario. 2) El sistema comprueba que el usuario tenga permisos de administrador. 3) El sistema redirige al usuario a la página borrado de interesados. 4) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos. • Formulario para eliminar la relación de una persona con un inmueble. <p>Al pulsar sobre borrar se procede a la eliminación de la relación de interesados.</p> <p>Una vez eliminada la relación se redirige a la página listado de interesados.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.32. EF32: Acceso a la página Listado de Interesados

Código de Especificación Funcional:	EF32
Nombre	Acceso a la página interesados
Versión:	1.0
Actores:	Administrador

Aplicación de Gestión Inmobiliaria en la Nube

Descripción:	El usuario accede desde un navegador a la dirección web de la página registro de la aplicación.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página interesados.
Flujo	<ol style="list-style-type: none">1) El Sistema comprueba las correctas credenciales y permisos del usuario.2) El sistema redirige al usuario a la página interesados.3) La página interesados presenta la siguiente estructura:<ul style="list-style-type: none">• Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación.• Un listado de inmuebles relacionados con clientes.• Botones de acceso a creación, edición y borrado de relaciones. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre crear se redirige a la página creación de interesados.</p> <p>Al pulsar sobre edición se accede a la página edición de una relación de interesados.</p> <p>Al pulsar sobre borrar una relación se accede a la página borrado de interesados.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.33. EF33: Acceso a la página Edición de Interesados

Código de Especificación Funcional:	EF33
Nombre	Acceso a la página edición de interesados.
Versión:	1.0
Actores:	Administrador

Aplicación de Gestión Inmobiliaria en la Nube

Descripción:	El usuario accede desde un navegador a la dirección web de la página edición de interesados.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración. Encontrarse autenticado en el sistema con una cuenta de administrador.
Postcondiciones:	
Flujo	<ol style="list-style-type: none">1) El Sistema comprueba las correctas credenciales y permisos del usuario.2) El sistema comprueba que el usuario tenga permisos de administrador.3) El sistema redirige al usuario a la página edición de interesados.4) La página presenta la siguiente estructura:<ul style="list-style-type: none">• Menú de navegación con secciones de inmuebles, cerrar sesión, acceso a administración y favoritos.• Formulario para editar la relación de un cliente con un inmueble. <p>Al pulsar sobre modificar se validan los datos introducidos en el formulario y se procede a la edición de la relación de interesados.</p> <p>Una vez modificada la relación se redirige a la página listado de interesados.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.34. EF34: Acceso a la página Creación de Tipos de Contacto

Código de Especificación Funcional:	EF34
Nombre	Acceso a la página alta de tipos de contacto
Versión:	1.0
Actores:	Administrador

Descripción:	El usuario accede desde un navegador a la dirección web de la página alta de tipos de contacto.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página tipos de contacto.
Flujo	<p>1) El Sistema comprueba las correctas credenciales y permisos del usuario.</p> <p>2) El sistema redirige al usuario a la página alta de tipos de contacto.</p> <p>3) La página presenta la siguiente estructura:</p> <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Un formulario para la creación de un nuevo tipo de contacto. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre crear el sistema añade el nuevo tipo de contacto.</p> <p>Una vez creado se redirige a la página listado de tipos de contacto.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.35. EF35: Acceso a la página Borrado de Tipos de Contacto

Código de Especificación Funcional:	EF35
Nombre	Acceso a la página borrado de tipos de contacto
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página alta de tipos de contacto.
Prioridad:	Alta

Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página tipos de contacto.
Flujo	<ol style="list-style-type: none"> 1) El Sistema comprueba las correctas credenciales y permisos del usuario. 2) El sistema redirige al usuario a la página borrado de tipos de contacto. 3) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Un formulario para el borrado de un tipo de contacto. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre borrar el sistema elimina el tipo de contacto.</p> <p>Una vez borrado se redirige a la página listado de tipos de contacto.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.36. EF36: Acceso a la página Listado de Tipos de Contacto

Código de Especificación Funcional:	EF36
Nombre	Acceso a la página listado de tipos de contacto
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página tipos de contacto.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.

Postcondiciones:	El sistema redirige a la página tipos de contacto.
Flujo	<p>4) El Sistema comprueba las correctas credenciales y permisos del usuario.</p> <p>5) El sistema redirige al usuario a la página listado de tipos de contacto.</p> <p>6) La página presenta la siguiente estructura:</p> <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Un listado de tipos de contacto (ej: redes sociales, carteles, amigos...), previamente añadidos por el usuario. • Botones de acceso a detalle, creación y borrado.. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre crear se redirige a la página creación de tipos de contacto.</p> <p>Al pulsar sobre edición se accede a la página edición de tipos de contacto.</p> <p>Al pulsar sobre borrar una relación se accede a la página borrado de tipos de contacto.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.37. EF37: Acceso a la página Edición de Tipos de Contacto

Código de Especificación Funcional:	EF37
Nombre	Acceso a la página edición de tipos de contacto
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página edición de tipos de contacto.
Prioridad:	Alta

Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página edición de tipos de contacto.
Flujo	<ol style="list-style-type: none"> 1) El Sistema comprueba las correctas credenciales y permisos del usuario. 2) El sistema redirige al usuario a la página edición de tipos de contacto. 3) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Un formulario para la edición de un nuevo tipo de contacto. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre modificar el sistema guarda los cambios sobre el tipo de contacto.</p> <p>Una vez modificado se redirige a la página listado de tipos de contacto.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.38. EF38: Acceso a la página Creación de Poblaciones

Código de Especificación Funcional:	EF38
Nombre	Acceso a la página alta de Poblaciones
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página creación de poblaciones.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración
Postcondiciones:	El sistema redirige a la página alta de poblaciones.

Flujo	<ol style="list-style-type: none"> 1) El Sistema comprueba las correctas credenciales y permisos del usuario. 2) El sistema redirige al usuario a la página poblaciones. 3) La página alta de poblaciones presenta la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Formulario para dar de alta una nueva población. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre crear el sistema añade la nueva población.</p> <p>Una vez creada la población se redirige a la página listado de poblaciones.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.39. EF39: Acceso a la página Borrado de Poblaciones

Código de Especificación Funcional:	EF39
Nombre	Acceso a la página borrado de Poblaciones
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página borrado de poblaciones.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración
Postcondiciones:	El sistema redirige a la página alta de poblaciones.
Flujo	<ol style="list-style-type: none"> 1) El Sistema comprueba las correctas credenciales y permisos del usuario. 2) El sistema redirige al usuario a la página poblaciones.

	<p>3) La página borrado de poblaciones presenta la siguiente estructura:</p> <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Formulario para borrar una población. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre borrar el sistema elimina la población.</p> <p>Una vez borrada la población se redirige a la página listado de poblaciones.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.40. EF40: Acceso a la página Listado de Poblaciones

Código de Especificación Funcional:	EF40
Nombre	Acceso a la página listado de Poblaciones
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página listado de Poblaciones.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración
Postcondiciones:	El sistema redirige a la página listado de poblaciones.
Flujo	<ol style="list-style-type: none"> 1) El Sistema comprueba las correctas credenciales y permisos del usuario. 2) El sistema redirige al usuario a la página poblaciones. 3) La página poblaciones presenta la siguiente estructura:

	<ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Un listado de poblaciones dadas de alta. • Botones de acceso a edición/borrado y creación de poblaciones. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre edición se redirige a la página edición de la población.</p> <p>Al pulsar sobre borrado se redirige a la página borrado de la población.</p> <p>Al pulsar sobre crear se redirige a la página creación de una población nueva.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.41. EF41: Acceso a la página Edición de Poblaciones

Código de Especificación Funcional:	EF41
Nombre	Acceso a la página edición de Poblaciones
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página edición de poblaciones.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración
Postcondiciones:	El sistema redirige a la página edición de poblaciones.
Flujo	<ol style="list-style-type: none"> 4) El Sistema comprueba las correctas credenciales y permisos del usuario. 5) El sistema redirige al usuario a la página edición de poblaciones. 6) La página edición de poblaciones presenta la siguiente estructura:

	<ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. • Formulario para editar una población. <p>Al pulsar sobre una sección se redirige a una página que lista inmuebles del tipo seleccionado.</p> <p>Al pulsar sobre modificar el sistema guarda los cambios de la población.</p> <p>Una vez modificada la población se redirige a la página listado de poblaciones.</p>
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.42. EF42: Acceso a la página Configuración de Ofertas

Código de Especificación Funcional:	E42
Nombre	Acceso a la página configuración de ofertas
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página configuración de ofertas.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página configuración de ofertas.
Flujo	<ol style="list-style-type: none"> 1) El Sistema comprueba las correctas credenciales y permisos del usuario. 2) El sistema redirige al usuario a la página configuración de ofertas. 3) La página presenta la siguiente estructura:

	<ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. Acceso a administración y a favoritos. • Un listado de las ofertas actuales de la página principal con enlaces a la página edición de oferta.
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.43. EF43: Acceso a la página Edición de una Oferta

Código de Especificación Funcional:	E43
Nombre	Acceso a la página edición de ofertas
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página edición de ofertas.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página edición de ofertas.
Flujo	<ol style="list-style-type: none"> 1) El Sistema comprueba las correctas credenciales y permisos del usuario. 2) El sistema redirige al usuario a la página edición de ofertas. 3) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. Acceso a administración y a favoritos. • Un listado de las ofertas actuales de la página principal con enlaces a la página edición de oferta.

Requisitos	El usuario debe estar registrado en la aplicación como Administrador.
------------	---

3.1.44. EF44: Acceso a la página Configuración de Carrousel

Código de Especificación Funcional:	EF44
Nombre	Acceso a la página configuración de carrousel
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página configuración de carrousel.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página configuración de carrousel.
Flujo	<ol style="list-style-type: none"> 4) El Sistema comprueba las correctas credenciales y permisos del usuario. 5) El sistema redirige al usuario a la página configuración de carrousel. 6) La página presenta la siguiente estructura: <ul style="list-style-type: none"> • Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. Acceso a administración y a favoritos. • Un listado de las fotos actuales del carrousel con enlaces a la página edición de un elemento del carrousel.
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.1.45. EF45: Acceso a la página Edición de un elemento del Carrousel

Código de Especificación Funcional:	EF45
Nombre	Acceso a la página configuración de carrousel
Versión:	1.0
Actores:	Administrador
Descripción:	El usuario accede desde un navegador a la dirección web de la página configuración de carrousel.
Prioridad:	Alta
Precondiciones:	Introducir en el navegador web la dirección de la web o acceder desde la página de administración.
Postcondiciones:	El sistema redirige a la página configuración de carrousel.
Flujo	<ol style="list-style-type: none">7) El Sistema comprueba las correctas credenciales y permisos del usuario.8) El sistema redirige al usuario a la página configuración de carrousel.9) La página presenta la siguiente estructura:<ul style="list-style-type: none">• Un menú de navegación con diferentes secciones de inmuebles y acceso a cerrar sesión en la aplicación. Acceso a administración y a favoritos.• Un listado de las fotos actuales del carrousel con enlaces a la página edición de un elemento del carrousel.
Requisitos	El usuario debe estar registrado en la aplicación como Administrador.

3.2. Requisitos no Funcionales

3.2.1. ENF1: Disponibilidad de la aplicación

Código de Especificación Funcional:	ENF1
-------------------------------------	------

Aplicación de Gestión Inmobiliaria en la Nube

Nombre	Disponibilidad de la aplicación
Versión:	1.0
Descripción:	La aplicación deberá estar disponible para ser usada las 24h del día durante los 7 días de la semana.
Prioridad:	Alta

3.2.2. ENF2: Roles de Usuarios

Código de Especificación Funcional:	ENF2
Nombre	Roles de usuarios
Versión:	1.0
Descripción:	La aplicación deberá dar acceso a tipos de usuario administradores, usuarios visitantes y usuarios sin cuenta.
Prioridad:	Alta

3.2.3. ENF3: Seguridad de la aplicación

Código de Especificación Funcional:	ENF3
Nombre	Seguridad de la aplicación
Versión:	1.0
Descripción:	La aplicación deberá incluir medidas de protección para prevenir inyecciones SQL y cross site scripting.
Prioridad:	Alta

3.2.4. ENF4: Usabilidad de la aplicación

Código de Especificación Funcional:	ENF4
Nombre	Usabilidad de la aplicación
Versión:	1.0
Descripción:	La aplicación incluirá elementos que faciliten su usabilidad como menús, botones y enlaces. La

	familiarización con las páginas no debería llevar más de 3 minutos.
Prioridad:	Alta

3.2.5. ENF5: Compatibilidad con Navegadores

Código de Especificación Funcional:	ENF5
Nombre	Compatibilidad con Navegadores
Versión:	1.0
Descripción:	La aplicación deberá ser compatible con las últimas versiones de los navegadores: Google Chrome, Internet Explorer, Mozilla Firefox, Opera y Safari.
Prioridad:	Alta

3.2.6. ENF6: Diseño de la aplicación

Código de Especificación Funcional:	ENF6
Nombre	Diseño de la aplicación
Versión:	1.0
Descripción:	La aplicación deberá presentar un diseño responsivo en la mayoría de las páginas.
Prioridad:	Alta

3.2.7. ENF7: Escalabilidad y extensibilidad de la aplicación

Código de Especificación Funcional:	ENF7
Nombre	Escalabilidad y extensibilidad de la aplicación
Versión:	1.0
Descripción:	La aplicación deberá estar diseñada para poder integrarse con Microsoft Office en futuras ampliaciones.

Prioridad:	Alta
------------	------

3.2.8. ENF8: Sesiones de usuarios simultáneas

Código de Especificación Funcional:	ENF8
Nombre	Sesiones de usuarios simultáneas
Versión:	1.0
Descripción:	La aplicación deberá estar preparada para soportar el acceso a datos de múltiples usuarios de forma simultánea.
Prioridad:	Alta

3.2.9. ENF9: Mantenibilidad de la aplicación

Código de Especificación Funcional:	ENF9
Nombre	Mantenibilidad de la aplicación
Versión:	1.0
Descripción:	La aplicación deberá prestarse a una fácil mantenibilidad a través de: <ul style="list-style-type: none">• Una correcta documentación• Comentarios de código• Manual del programador.• Uso de patrones de diseño.• Uso de software para control de versiones.
Prioridad:	Alta

3.2.10. ENF10: Requisitos de Servidores

Código de Especificación Funcional:	ENF2
Nombre	Requisitos de servidores.
Versión:	1.0

Descripción:	La aplicación deberá poder alojarse en servidores basados en Windows Server. No es necesaria su compatibilidad con Linux o cualquier otra plataforma.
Prioridad:	Alta

3.2.11. ENF11: Tiempos de Respuesta

Código de Especificación Funcional:	ENF11
Nombre	Tiempos de Respuesta
Versión:	1.0
Descripción:	La aplicación deberá ofrecer un tiempo de respuesta máximo de 10 segundos.
Prioridad:	Alta

3.2.12. ENF12: Bases de Datos

Código de Especificación Funcional:	ENF12
Nombre	Bases de Datos
Versión:	1.0
Descripción:	La aplicación deberá funcionar sobre una base de datos SQL Server.
Prioridad:	Alta

4. Diseño del Software

El diseño del software constituye el núcleo técnico de la ingeniería y se aplica una vez se han analizado y especificado los requisitos. A lo largo del siguiente apartado se establece una plataforma para la construcción de la aplicación tras el análisis de las alternativas más utilizadas actualmente.

4.1. Estudio de alternativas tecnológicas (frameworks)

Para la realización del proyecto se han estudiado 3 alternativas diferentes:

- PHP
- Java
- .Net

A la hora de determinar cuál de “los grandes” se debería utilizar para un proyecto como este, sería ideal conocer las ventajas y desventajas de cada uno teniendo en cuenta las características de la aplicación web.

4.1.1.1. PHP

Se trata de un lenguaje de programación **interpretado** en el servidor. Creado en 1994, en la actualidad es ampliamente utilizado en entornos de desarrollo web por su facilidad de uso, su perfecta integración con HTML y su versatilidad de uso en cualquier plataforma. Se utiliza en más de 20 millones de sitios web en todo el mundo y cuenta con numerosos frameworks que facilitan el desarrollo.

Cada vez que una página php se ejecuta, se revisa que la sintaxis sea correcta y se construye un árbol sintáctico. Este árbol sintáctico es ejecutado por lo que se conoce como Zend Engine. Además, existen aceleradores del rendimiento como APC, Xcache y eAccelerator que cachean lo que se construye en el árbol sintáctico, aunque la interpretación se realice siempre.

Aplicación de Gestión Inmobiliaria en la Nube

Entre los más populares encontramos:

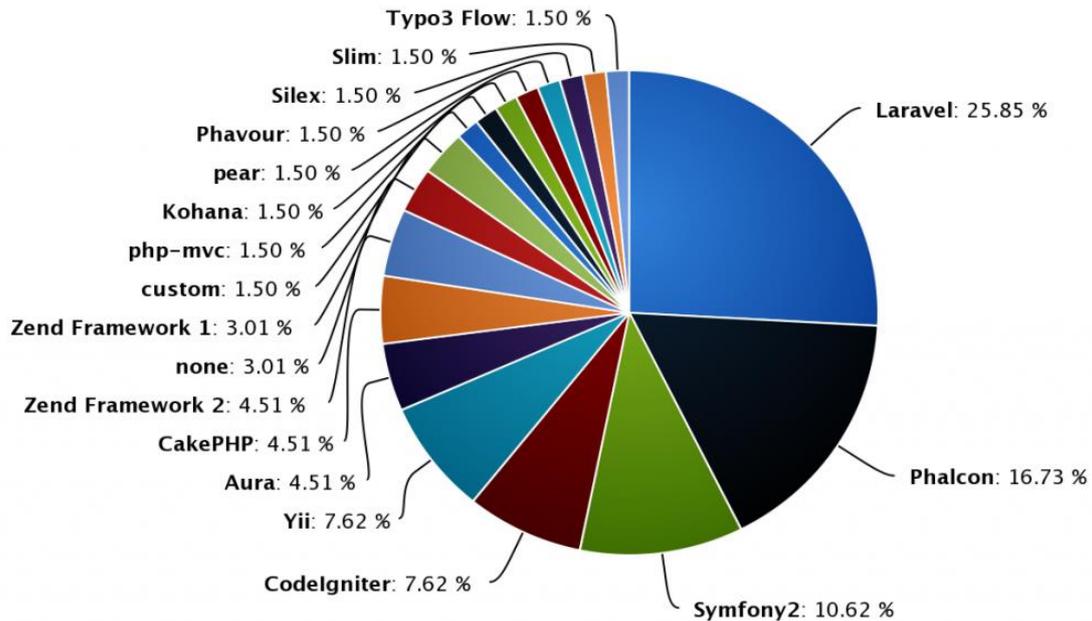
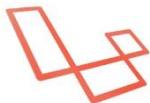


Ilustración 8: Popularidad de Frameworks PHP (fuente: www.sitepoint.com)



laravel

Laravel, el framework de desarrollo web MVC más popular de PHP, se caracteriza por los siguientes puntos:

- **Sintaxis expresiva muy clara** y un set completo de funcionalidades que ahorran tiempo de trabajo.
- Prima la convención sobre la configuración.
- **Dicta normas rígidas sobre codificación** pero siguiéndolas el framework muchas veces sabe qué tiene que hacer sin necesidad de escribir código.
- **Punto de encuentro entre minimalismo y funcionalidad** (filosofía zen), es decir, la programación es algo más que escribir líneas de código.
- Resultado de la evolución de frameworks de desarrollo anteriores de PHP.
- **Ofrece una modularidad del código** gracias a los drivers y paquetes.

- **El empaquetado permite reutilizar código** y compartirlo con toda la comunidad Laravel. Ello facilita de modo abrumador el desarrollo web.
- **El sistema de migración** invita a diseñar y modificar bases de datos de un modo independiente a la plataforma.
- **Integra Fluent Query Builder**, un sistema de consultas a la base de datos llamado que solventa las diferencias de consultas entre las diferentes bases de datos, lo que permite ejecutar consultas mucho más robustas.
- **La implementación active record Eloquent** interactúa con la base de datos de un modo totalmente orientado a objetos. Siguiendo los estándares más modernos es posible crear, recuperar, actualizar y borrar datos sin necesidad de escribir una sola línea de código SQL. Provee un poderoso sistema de manejo de relaciones y es capaz de gestionar la paginación automáticamente.
- **Incluye Artisan**, una herramienta de interface de línea de comandos totalmente configurable. Con ella es posible interactuar con la aplicación para ejecutar acciones como migraciones, pruebas unitarias y pruebas programadas.
- **El sistema de manejo de rutas** permite controlar completamente las direcciones URL del sitio usando el helper HTML integrado. Permite crear enlaces desde dentro del sitio que se actualizarán automáticamente.
- **El motor de plantillas Blade** hace las vistas mucho más limpias y sencillas incluye importantes mejoras en la presentación de la aplicación.



Phalcon sigue un diseño MVC al igual que Laravel, sin embargo, es considerado el framework más rápido de PHP. A continuación se detallan las peculiaridades del framework:

- **Implementado como una extensión de C**, ofrece un alto rendimiento y un menor consumo de recursos por cada petición.

- **Componentes débilmente acoplados** dejan libertad para usar la totalidad o sólo las partes que resulten necesarias. Con ello se facilita y aumenta la escalabilidad de proyectos PHP sin aumentar la complejidad.
- **Utiliza un ORM escrito en C** que logra una alta eficiencia en las operaciones con la base de datos.
- Accede directamente a las estructuras internas de PHP optimizando cada ejecución.
- **El paquete que contiene el framework es ligero** como para no necesitar librerías de terceros pues contiene el ORM, un motor de plantillas, PHQL, etc.
- **Se instala como una extensión** y el marco tiene muy poca sobrecarga, sin embargo, esto hace que sea difícil de instalar. Muchos desarrolladores lo consideran una ventaja pues se aseguran de estar en una comunidad interesada en el framework, eliminando a los aficionados.
- **Aprovecha características de otros frameworks** populares tales como la gestión de eventos, un ORM de fácil uso y los espacios de nombres nativos.



Otro de los frameworks más destacados para la programación web en PHP es Code Igniter. Al igual que los frameworks PHP anteriores, implementa el modelo de tres capas comúnmente conocido como MVC, con el fin de organizar mejor el código, facilitar la creación y el mantenimiento. Los rasgos más significativos de CodeIgniter son:

- **Versatilidad** para trabajar con la mayoría de los entornos o servidores, incluso en sistemas de alojamiento compartido, donde sólo hay un acceso por FTP para enviar los archivos al servidor y donde no se dispone de acceso a su configuración.
- **Facilidad de instalación** a la hora de subir la solución al servidor

Aplicación de Gestión Inmobiliaria en la Nube

- **Mayor flexibilidad** en comparación con otros frameworks. Define una manera de trabajar específica aunque no es obligatorio seguirla.
- **Suaviza la curva de aprendizaje** gracias a sus módulos opcionales.
- **Posee un núcleo ligero**, lo que permite que el servidor no se sobrecargue con grandes porciones de código y la mayoría de los módulos o clases que ofrece se pueden cargar sólo cuando se van a utilizar realmente.
- **Proporciona documentación tutorizada** fácil de seguir y de asimilar.

4.1.1.2. Java

Java es un lenguaje de propósito general utilizado tanto para aplicaciones de escritorio, web y móviles. Actualmente existen muchos IDEs y frameworks que facilitan el desarrollo. A continuación se analizan los más comunes teniendo en cuenta la velocidad de prototipado de la aplicación, complejidad del framework, facilidad de uso y documentación sobre el mismo.

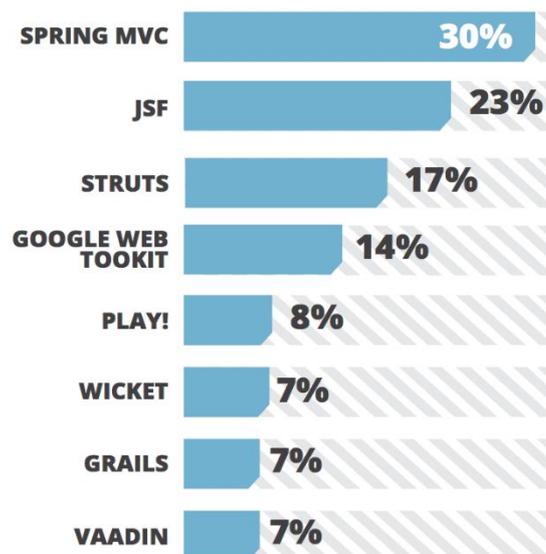


Ilustración 9: Frameworks más populares de Java (fuente: zereturnaround.com)



Spring es un framework MVC basado en HTTP y servlets que se centra en la programación y configuración de aplicaciones empresariales basadas en Java. Favorece la implementación de la lógica de negocio, la llamada "fontanería" (plumbing) del software. Comprende los siguientes módulos:

- **Un contenedor IoC (inversión de control)** que permite la configuración de los componentes de aplicación y la administración del ciclo de vida de los objetos Java, se lleva a cabo principalmente a través de la inyección de dependencias.
- **Acceso a datos:** se trabaja con RDBMS en la plataforma java, usando Java Database Connectivity y herramientas de Mapeo objeto relacional con bases de datos NoSQL.
- **Gestión de transacciones:** unifica distintas APIs de gestión y coordina las transacciones para los objetos Java.
- **El acceso remoto** permite la importación y exportación de objetos Java a través de redes que soportan RMI, CORBA y otros protocolos basados en HTTP incluyendo servicios web (SOAP).
- **Convención sobre configuración:** el módulo Spring Roo ofrece una solución rápida para el desarrollo de aplicaciones basadas en Spring Framework, privilegiando la simplicidad sin perder flexibilidad.
- **Procesamiento por lotes:** framework orientado a aplicaciones con alta carga de procesamiento de datos (gestión de transacciones, estadísticas, gestión de recursos, etc.).
- **Autenticación y Autorización:** procesos de seguridad configurables que soportan un rango de estándares, protocolos, herramientas y prácticas a través del subproyecto Spring Security (antiguamente Acegi).

- **Administración Remota:** Configuración de visibilidad y gestión de objetos Java para la configuración local o remota vía JMX.
- **Mensajes:** Registro configurable de objetos receptores de mensajes, para el consumo transparente desde la a través de JMS, una mejora del envío de mensajes sobre las API JMS estándar.
- **Testing:** Soporte de clases para desarrollo de unidades de prueba e integración.



JavaServer Faces (JSF) es un framework MVC que trata de simplificar la construcción de interfaces de usuario a través de componentes reutilizables y controladores de eventos. En cuanto a desarrollo, aporta los siguientes beneficios:

- **Emplea componentes UI reutilizables** que reducen el esfuerzo empleado para crear y mantener aplicaciones que corren sobre Java.
- **Los wizard para JSF** disponibles son su mejor baza en cuanto a productividad se refiere. Muchos IDEs como Netbeans se encargan de generar la mayor parte del código repetitivo y demás configuraciones.
- **Archetype, un proyecto de Maven**, aporta también numerosas plantillas disponibles para facilitar la creación de una aplicación base.
- **Utiliza páginas JSP para generar las vistas** incluyendo una biblioteca propia de etiquetas para crear los elementos de los formularios HTML (helpers).
- **Las managed beans facilitan la manipulación** de elementos de los formularios asociando cada vista a un conjunto de objetos java manejados por el controlador.
- **Es el único que forma parte del estándar J2EE** para crear la capa de presentación.
- **Es extensible** por lo que se pueden desarrollar nuevos componentes a medida así como modificar el comportamiento del framework mediante APIs que controlan su funcionamiento.

Struts

Apache Struts es un framework libre de código abierto orientado también al desarrollo de aplicaciones web. Implementa un modelo MVC para separar el código de la capa de presentación del código de la capa de negocio, facilitando la asignación de roles y la especialización de funciones.

- **Está basado en tecnologías estándares** como Java Servlets, JavaBeans o XML, hereda el ambiente de estos estándares y puede operar con cualquier tecnología basada en JSP o Servlet.
- **Implementa varios patrones de diseño de la arquitectura J2EE** tales como:
 - Service to worker
 - Dispatcher View
 - Front Controller
 - View Helper
 - Synchronize (Déjà vu) Token
- **Posee un plugin de convenciones** que fuerza la convención sobre la configuración.
- **Provee anotaciones para configurar el mapeo de direcciones URL** que acelera en muchos casos las aplicaciones.
- **Potencia la reutilización de código** y soporta múltiples interfaces de usuario (HTML, SHTML, WML, aplicaciones de escritorio, etc.).



Play, escrito en Scala y Java, es elegido por muchos principiantes por su simplicidad. El framework se originó por muchos desarrolladores de Java que envidiaban el rápido prototipado de Ruby on Rails.

Soporta código escrito en Scala.

- ✓ **Posee componentes binarios utilizados para scaffolding**, similares a los rails de Ruby on Rails
- ✓ **La documentación de Play** para poder iniciarse en el uso del framework es un completo tutorial. Ayuda a los desarrolladores primerizos a ser más productivos.
- ✓ **La capa de datos tiene un rol central.**
- ✓ **Utiliza el framework Hibernate**, una implementación de JPA para realizar el mapeo de los objetos de Java.



Grails es un framework desarrollado sobre el lenguaje de programación Groovy, basado en la plataforma Java. Se destacan sus siguientes características:

- **Proporciona un entorno de desarrollo estandarizado** y oculta gran parte de los detalles de configuración al programador.
- **Grails se caracteriza por su simplicidad.** Los desarrolladores que buscan implementar una aplicación CRUD (Create, Read, Update and Delete, operaciones básicas en bases de datos propias de la capa de persistencia) pequeña-mediana se decantan por este framework.
- **La instalación es rápida e incorpora scaffolding** (generación de código) que ahorra mucho tiempo.
- **Se rige por el principio de convención sobre la configuración** que ayuda a olvidarse de las molestias que esta ocasiona.
- **Incorpora un mecanismo de recarga de código** que actualiza las clases cuando ha habido una modificación en la base de datos, aunque este tiene muchas limitaciones. En general, sólo recarga clases Groovy.

4.1.1.3. .NET

.NET es un framework de Microsoft que corre principalmente sobre Windows Server. Hace especial hincapié en el rápido desarrollo de aplicaciones que permitan integrar la totalidad de sus productos: desde el sistema operativo hasta las herramientas de mercado como la suite de Office.

Existen otros frameworks como OpenRasta, Monorail, CSLA y DotNetNuke que se basan en la plataforma .NET. Sin embargo, no son opciones lo suficientemente atractivas como para competir con ASP.NET MVC u otras tecnologías de Microsoft orientadas a contenidos web como SharePoint.



SharePoint es constituido por un conjunto de herramientas de Microsoft que integran numerosas funcionalidades y modelos que propician el desarrollo de portales web.

Su potente gestión de contenidos ha extendido su uso en las empresas siendo capaz de integrarse con otras tecnologías como SAP. Ofrece las siguientes ventajas:

- **Permite construir soluciones de forma ágil, flexible y escalable.**
- **Incluye capacidades para la gestión documental** como la vista previa de los documentos y un sistema "drag & drop" que permite compartir y guardar los documentos cómodamente.
- **Incluye un potente buscador** que permite hacer todo tipo de refinamientos sobre los contenidos del sitio.
- **Cuenta con sitios de colaboración** donde los usuarios pueden compartir discusiones, foros y conversaciones.
- **Incorpora Skydrive Pro**, un lugar para guardar documentos y compartirlos con otros usuarios a través de diferentes dispositivos y desde cualquier lugar.
- **Compatibilidad** entre distintos dispositivos móviles.
- **SharePoint Apps** permite instalar aplicaciones mediante.

- **Proporciona herramientas de colaboración** para crear redes sociales corporativas
- **Preparado para nube** a partir de la última versión SharePoint 2013.

ASP.NET Web Forms

ASP.NET es una parte del framework para aplicaciones web ASP.NET. Es uno de los cuatro modelos de programación para crear aplicaciones web ASP.NET (ASP.NET MVC, ASP.NET Web Pages y ASP.NET Single Page). El marco basado en Web Forms ofrece las siguientes ventajas:

- **Beneficia el desarrollo de aplicaciones web de línea de negocio** gracias un modelo de eventos que conserva el estado sobre HTTP.
- **Proporciona numerosos eventos** que son compatibles con cientos de controles de servidor.
- **Utiliza un patrón Page Controller** que agrega funcionalidad a las páginas individuales.
- **Funciona bien para pequeños equipos de desarrolladores Web** que quieren aprovecharse de la gran cantidad de componentes disponibles para desarrollo rápido de aplicaciones.
- **Es menos compleja para el desarrollo de aplicaciones** porque los componentes están estrechamente integrados y generalmente requieren menos código que el modelo MVC.



El marco de ASP.NET MVC proporciona una alternativa al modelo de formularios Web Forms de ASP.NET para crear aplicaciones web. Su última versión ASP.NET MVC5 incorpora las siguientes propiedades:

- **Facilita la administración de la complejidad**, al dividir una aplicación en el modelo, la vista y el controlador.
- **One ASP.NET** permite la personalización y configuración de la autenticación a través de un wizard.
- **ASP.NET Identity** facilita la autenticación y gestión de identidad incluyendo la membership API.
- **El diseño con Bootstrap** provee un look and feel responsivo y fácil de personalizar.
- **Incluye filtros de autenticación** que permiten especificar la lógica de autenticación por acción, por controlador o global para todos los controladores.
- **Permite la sobreescritura de filtros** para especificar qué filtros se aplican a un determinado método.
- **Integra Attribute routing** mediante el que se pueden especificar rutas mediante anotaciones en acciones y controladores.
- **Se pueden cambiar las opciones de autenticación** por defecto para muchas de las plantillas disponibles. Por ejemplo, cuando se crea un proyecto con ASP.NET Web Forms se puede seleccionar cualquiera de las siguientes opciones:
 - No autenticación
 - Cuentas de usuario individuales (ASP.NET membership o social provider log in)
 - Cuentas Organizativas (Directorio Activo en una aplicación de internet)
 - Autenticación de Windows (Directorio Activo en una aplicación de internet)
- **ASP.NET Scaffolding** es la generación de código para aplicaciones web .NET. Facilita agregar código repetitivo mediante el que el proyecto interactúa con el modelo de datos.

4.2. Implementación escogida

Teniendo en cuenta las ampliaciones posteriores a las que se someterá el proyecto, cabe destacar que la **integración con la suite Microsoft Office** juega un papel fundamental en la elección del framework. En este aspecto .NET ofrece un abanico más amplio de posibilidades ya que es posible utilizar múltiples librerías e incluso servicios web.

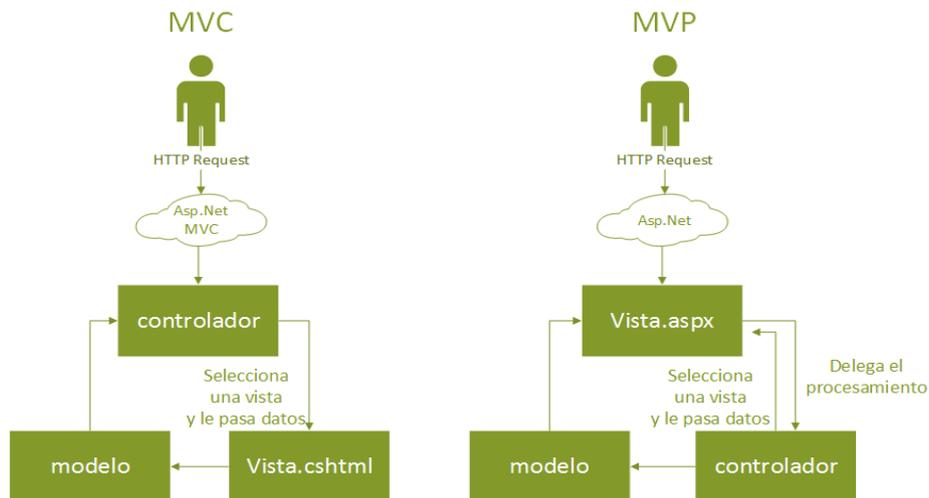
Las librerías que permiten integrar Office con las aplicaciones normalmente necesitan abrir una instancia del programa para manipular los datos. Esto quiere decir que precisan tener Microsoft Office instalado en el servidor en el que se alojan, lo cual puede ser un problema a la hora de contratar un proveedor. Este problema se soluciona a través del uso de servicios web. (Ej. Excel Services para SharePoint).

SharePoint es el producto más completo de Microsoft en cuanto a integración con Office se refiere. La desventaja que presenta es que es caro y está más orientado a intranets. También se desvía del propósito del proyecto que es crear una aplicación sencilla de usar al incluir una gran cantidad de funcionalidades a las que no se va a dar uso. Si el objetivo del proyecto fuera desarrollar una aplicación para un modelo de negocio propio de empresas más grandes (multinacionales), definitivamente sería la mejor opción.

Desechando la opción de SharePoint, aún queda por discutir si sería posible usar **WebForms o MVC**. Lo cierto es que ambas alternativas son perfectas: nos permiten usar todas las librerías disponibles para Office y desarrollar aplicaciones web de forma rápida. Sin embargo, una ventaja de MVC frente a los WebForms es que es más ligero. Los Web Forms incluyen muchos controles de usuarios y recibidores de eventos que, aunque sean más pesados, ahorran tiempo y dinero. Son una alternativa muy utilizada en el mundo empresarial aunque no por ello la más fácil de mantener. ¿Quiere esto decir que una aplicación con WebForms no puede llegar a ser mantenible? Sí que puede serlo, de hecho se puede implementar un modelo MVP (modelo-vista-presentación).

La principal diferencia entre MVC y MVP reside en cómo se gestionan las vistas. En MVC el controlador elige una vista y se la manda al cliente, mientras que en MVP esta elección no es tan clara, la vista se gestiona mediante un presentador. ¿Qué ventaja ofrece MVP?

El controlador tiene acceso constante a los objetos de las vistas, siendo capaz de editar sus propiedades. MVC, en cambio, requiere que todos los datos sean enviados en una petición.



Dado que otro de los propósitos del proyecto es una fácil mantenibilidad, MVC cumple mejor este requisito que los Web Forms, por lo que definitivamente es la solución elegida.

La versión más reciente que implementa el patrón modelo-vista-controlador es ASP.Net **MVC5, lanzada en Octubre de 2013** y permite construir tanto aplicaciones web como aplicaciones en la nube. El uso de Visual Studio 2013 facilita especialmente el desarrollo en nube ya que integra funcionalidades que despliegan la aplicación directamente en **Windows Azure**, la plataforma en nube de Microsoft.

4.3. Diseño técnico

El propósito de esta sección es describir los componentes técnicos de la aplicación partiendo de los requisitos funcionales iniciales del apartado 3.

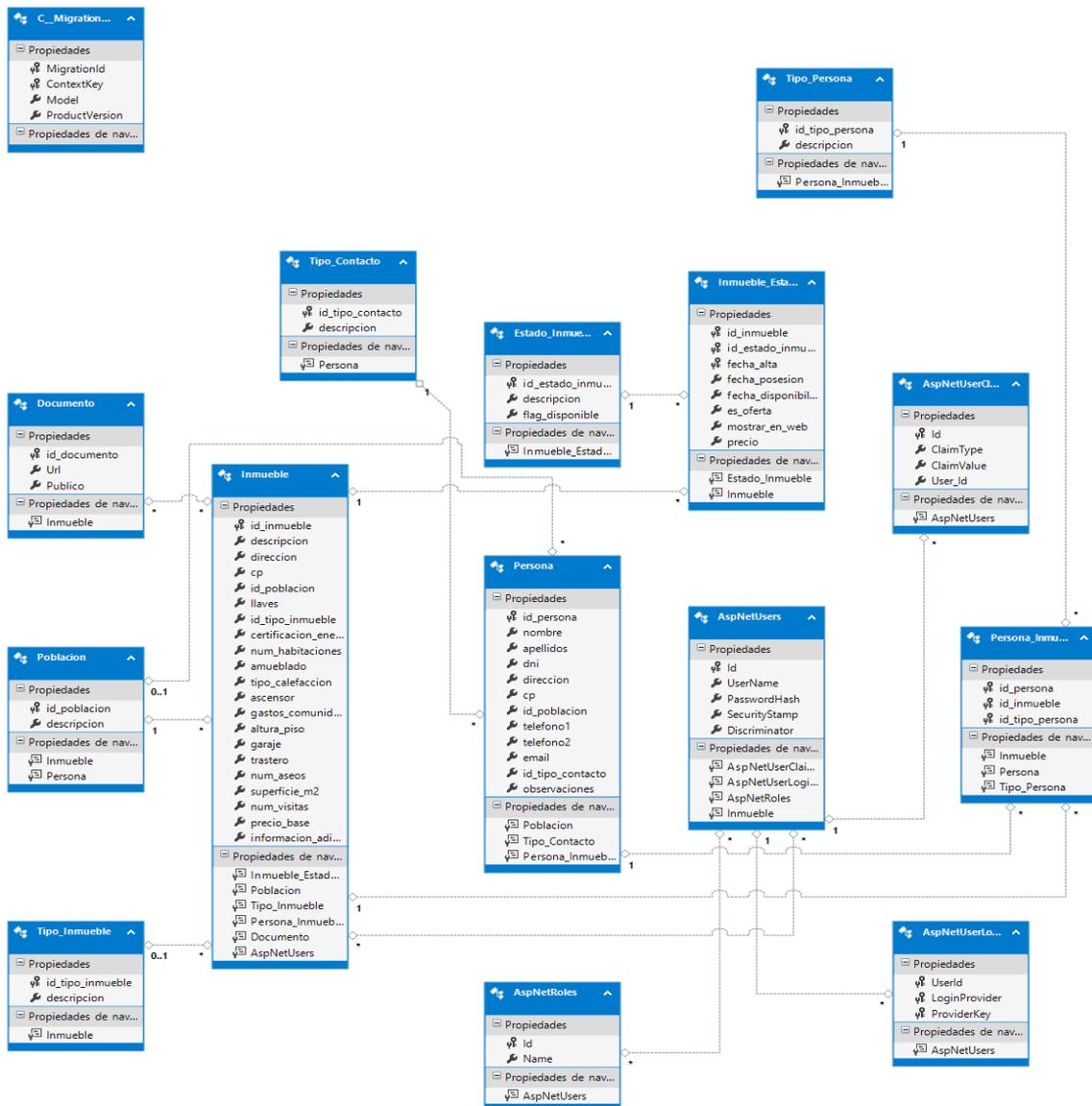
Se abordará el diseño de la base de datos especificando la utilidad de cada tabla en la aplicación así como el diseño de los componentes de la solución.

4.3.1. Base de datos

Para el proyecto se ha elegido una base de datos SQL Server aunque la aplicación se puede adaptar fácilmente a cualquier otra (mySQL, Oracle, Access...). Los motivos por los que se ha decidido implementar el proyecto sobre SQL Server son principalmente:

- ➔ La documentación disponible acerca de cómo integrar MVC5 con SQL Server.
- ➔ El tamaño de los registros.
- ➔ Sintonía con la tecnología de Microsoft utilizada

Para el modelo de negocio al que se enfoque este proyecto se propone el siguiente modelo de base de datos:



4.3.1.1. Tablas AspNet

Asp.Net propone una serie de tablas con el fin de facilitar la creación de usuarios con distintos roles en la aplicación.

MVC 5, concretamente, está preparado para integrarse con APIs de Google, Facebook, Twitter y Microsoft y permitir a los usuarios autenticarse sin necesidad de crear una cuenta específica en la aplicación. Sus credenciales quedan cifradas en las tablas de modo que el dueño de la base de datos no puede visualizar las contraseñas.

Los roles en ASP se refieren a los tipos de usuarios o grupos de usuarios que disfrutan de la aplicación (Administradores, Visitantes, Contribuidores, ...) Por defecto, la tabla que los alberga (AspNetRoles) se encuentra vacía, de modo que el desarrollador puede implementar los roles que quiera y con los permisos que desee.

Esto resulta especialmente útil en este proyecto, ya que existe la necesidad de ocultar el perfil de Administración a los usuarios visitantes. Para los agentes inmobiliarios, sin embargo, sí que interesa dejar esta sección visible. En este punto, definimos dos roles en la aplicación:

- Administrador: usuarios administradores que tendrán permisos para dar de alta contenido en la aplicación.
- Visitante: usuarios que no tendrán permisos para acceder a la gestión de los contenidos. Podrán ver la web y marcar favoritos.

4.3.1.2. Tabla Favoritos

Para permitir a los usuarios guardar sus favoritos para futuras visitas se ha creado una tabla "Favoritos". Se trata de una tabla que guarda la relación entre los usuarios y los

Nombre	inmuebles favoritos. Dado que la relacion es n:m no queda representada en el diagrama superior, sino que se visualiza la relación
UserId	entre las tablas AspNetUsers e Inmueble.
id_inmueble	

4.3.1.3. Tabla Inmueble

La tabla inmueble alberga las especificaciones de los inmuebles que se vayan a dar de alta en la aplicación. Se usa la misma tabla para todos los tipos de inmueble (pisos, casas, fincas, despachos...).

4.3.1.4. Tabla Población

Ya que se permite que los administradores del sitio añadan poblaciones nuevas en las que oferten inmuebles, se ha creado la tabla Población.

Las poblaciones aparecerán en los desplegados de las búsquedas de inmuebles y se utilizan también en las direcciones de las personas.

4.3.1.5. Tabla Personas

La tabla personas está ideada para que se puedan dar de alta en ella tanto clientes como propietarios, inicialmente. Se entiende que una persona puede ser propietaria y cliente a la vez y que de usar dos tablas distintas, sería necesario dar de alta a una misma persona dos veces. El modo en el que se diferencia un propietario de un cliente, por tanto, es a través del TipoPersona.

4.3.1.6. Tabla Tipo_Persona

El modelo de la base de datos es muy flexible para poder agregar cualquier tipo de relación entre el inmueble y la persona, no necesariamente propietarios y clientes de forma exclusiva. Con este método se pretende tener que evitar crear tablas adicionales a medida que aparezcan relaciones nuevas entre los inmuebles y las personas.

4.3.1.7. Tabla Tipo_Inmueble

Con los inmuebles se sigue el mismo método. No hay tablas para cada tipo de inmueble (piso, chalet, dúplex). Existe una tabla Inmueble y otra tabla "Tipo_Inmueble" que almacena los distintos tipos de inmuebles que oferta la inmobiliaria.

Los tipos de inmueble también aparecen en los desplegados del buscador del portal, facilitando con ello la interacción con los visitantes de la web.

4.3.1.8. *Tabla Estado_Inmueble*

La forma en la que se indica si un inmueble está disponible o no para una determinada operación es a través de la relación de la tabla Inmueble con la tabla Estado Inmueble.

La tabla Estado Inmueble está pensada para optimizar las consultas a la base de datos relacionadas con la disponibilidad del inmueble. Para ello se utiliza una descripción del estado (en alquiler, alquilado, en venta, vendido) y un flag de disponibilidad. Si se desea crear un estado que indique la ocupación de un inmueble, el flag_disponible deberá ser falso. Por el contrario, si el inmueble se encuentra disponible, el flag deberá ser verdadero.

4.3.1.9. *Tabla Tipo_Contacto*

Las empresas están interesadas en conocer cuál de sus medios de publicidad es más efectivo. La tabla tipo contacto sirve para dar de alta nuevos medios de contacto con los clientes (Amigos, Redes Sociales, Carteles, Web, Buscadores...).

4.3.1.10. *Tabla Inmueble_Estado_Inmueble*

Esta tabla es el resultado de la relación entre un Inmueble y uno o varios Estado_Inmueble, agregando además los campos relativos a las fechas de alta y baja de cada estado.

Está orientada a guardar un histórico de los inmuebles así como servir de precursor para la posterior creación de alertas de vencimiento de contrato en la aplicación.

4.3.1.11. *Tabla Persona_Inmueble*

En esta tabla se relacionan las Personas con los Inmuebles, pudiendo seleccionar el tipo de relación que guardan a través del tipo de persona.

4.3.1.12. *Tabla Documento*

La tabla documento guarda la dirección Url de los documentos asociados a un inmueble (fotos, pdf, video...)

4.3.2. Patrón

Según se mencionó anteriormente, para el desarrollo de la aplicación se ha elegido el framework ASP.Net MVC5 que permite construir aplicaciones en nube siguiendo el patrón Modelo-Vista-Controlador. La elección de este patrón de diseño tiene que ver especialmente con el mantenimiento de la aplicación. Dado que esta se someterá a fuertes ampliaciones, resulta esencial esforzarse en mantener un buen diseño, limpio y escalable. En el contexto de ASP.NET MVC:

- El modelo contiene toda la lógica de negocio y el acceso a datos.
- Las vistas albergan el código que se envía al navegador, es decir el código HTML junto con el código de servidor asociado que realice tareas de presentación, no de lógica de negocio.
- Los controladores reciben las peticiones del navegador y se encargan de elegir la vista y los datos de vuelta.

Entre las ventajas de ASP.NET MVC destaca también la facilidad con la que se generan URL semánticas, que se indexan mejor en los buscadores y son una práctica SEO habitual.

A nivel técnico, ASP.NET MVC es especialmente amigable a la hora de depurar y realizar pruebas unitarias. Ello propicia una correcta implementación del patrón MVC y la generación de código reutilizable.

A continuación, se detallan las características del framework en cada una de las capas de la arquitectura que facilitan la implementación del proyecto.

4.3.2.1. Modelo

El modelo se encarga de representar la lógica del negocio. MVC5 incorpora el denominado Entity Framework, encargado de modelar las entidades, las relaciones y trabajar con los motores de datos que se usan para almacenar y recuperar los mismos.

La forma en la que Entity Framework trabaja con los datos permite abstraerse completamente del modelo físico que presentan las bases de datos. Una de las ventajas

que aporta es que la creación de bases de datos y modelo es totalmente bidireccional, es decir:

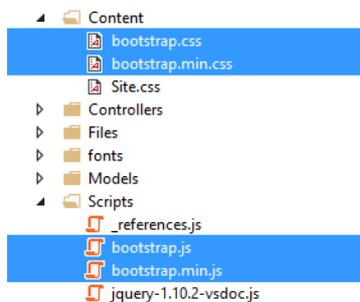
- ➔ Se puede crear el modelo a partir de las tablas de las bases de datos.
- ➔ Se pueden crear las tablas de las bases de datos a partir del modelo (Code First)

Ya que las clases están escritas en C#, un lenguaje orientado a objetos, las relaciones entre las tablas se representan a través de colecciones de objetos que facilitan las consultas en los controladores.

4.3.2.2. Vista

La vista transforma el modelo en una página web con la que los usuarios pueden interactuar. MVC5 se sirve de Razor y Twitter Bootstrap para agilizar el desarrollo de páginas con un look and feel atractivo y actual.

Razor es un motor de vistas (View Engine) cuyo principal objetivo es facilitar la programación de vistas simplificando la sintaxis. Además, fomenta la reutilización de código gracias a directivas que permiten combinar secciones de diferentes vistas.



En Visual Studio 2013, Twitter Bootstrap es añadido como el framework de interfaz de usuario por defecto de la aplicación. Se trata de una colección gratuita de plantillas HTML y CSS para el diseño de formularios, menús de navegación, botones, tablas, etc. Provee a la aplicación de un look and feel atractivo a la vez que agiliza la

maquetación a los programadores sin demasiadas nociones de diseño.

4.3.2.3. Controlador

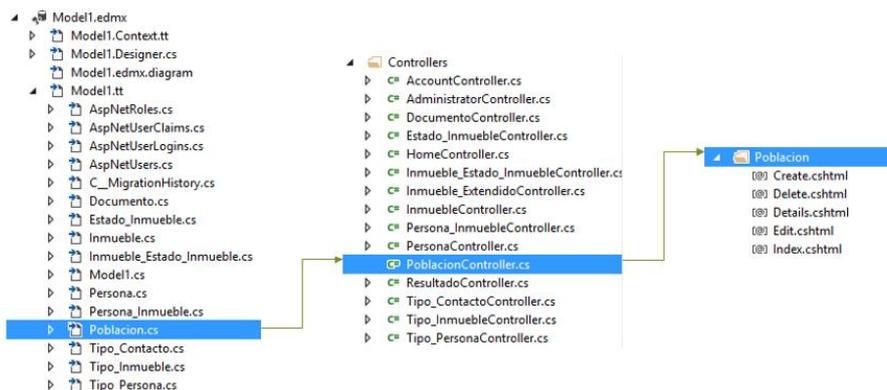
Los controladores actúan de intermediarios entre el usuario, el modelo y las vistas. Se encargan de recoger las peticiones del usuario, interactuar con el modelo y decidir la vista que debe mostrar los datos.

Aplicación de Gestión Inmobiliaria en la Nube

A la hora de interactuar con el modelo, la realización de consultas se realiza con LINQ, no en código SQL. **LINQ** (Language-Integrated Query) realiza consultas sobre las colecciones de objetos programadas tanto en C# como Visual Basic. Su uso previene las posibles incompatibilidades entre las consultas a diferentes bases de datos (SQL Server, MySQL, Oracle...) ya que trabaja con datos cargados en memoria. Como consecuencia, el código resulta aún más reutilizable ya que no depende de los cambios en la base de datos. El acceso a la base de datos se realiza únicamente cuando los datos no se encuentran almacenados en memoria.

Scaffolding

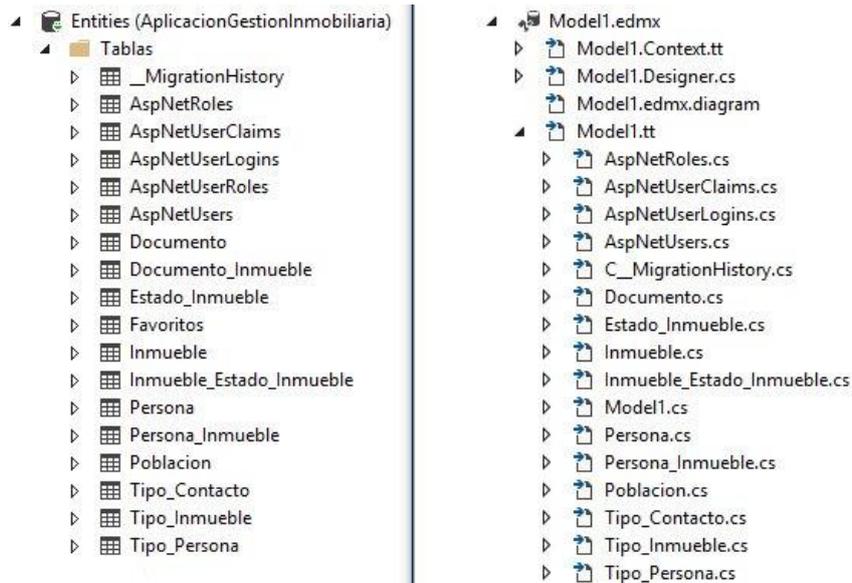
Gracias al Scaffolding es posible generar el código de las operaciones CRUD (Create, Read, Update, Delete) de los controladores y las vistas a partir del modelo de la aplicación. Por ejemplo, para el modelo correspondiente a las Poblaciones:



4.3.3. Detalles de implementación

Con la ayuda del Entity Framework de .NET se ha generado la siguiente correspondencia entre las tablas de la base de datos y el modelo de la aplicación:

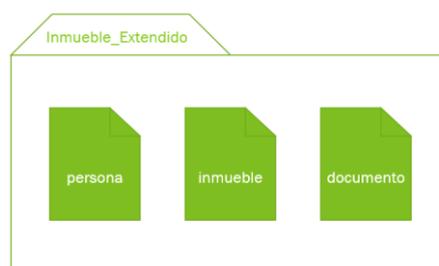
Aplicación de Gestión Inmobiliaria en la Nube



A partir del modelo se han generado las vistas y los controladores con la ayuda del Scaffolding anteriormente descrito. Con ello, la aplicación ya cuenta con las páginas necesarias para manipular los datos en la aplicación.

Persistir varios modelos a la vez

Sin embargo, la problemática surge a la hora de crear vistas que manejen varios modelos a la vez. Para ello se ha hecho uso de los “**modelos contenedores**”. Conceptualmente, un modelo contenedor es un modelo constituido por varios modelos. En el proyecto se utilizan varios modelos contenedores, como por ejemplo Inmueble_Extendido:



Para poder crear los objetos en la base de datos junto con sus relaciones, se crea el controlador correspondiente y las vistas con él asociadas. En el método “Crear” del controlador, la forma de recoger todos los datos sería tan simple como recibir por POST el modelo contenedor.

```
public ActionResult Create(Inmueble_Extendido inmueble_extendido) { ... }
```

5. Uso de la Aplicación

A continuación se analiza cómo resuelve la solución implementada diferentes escenarios del cliente.

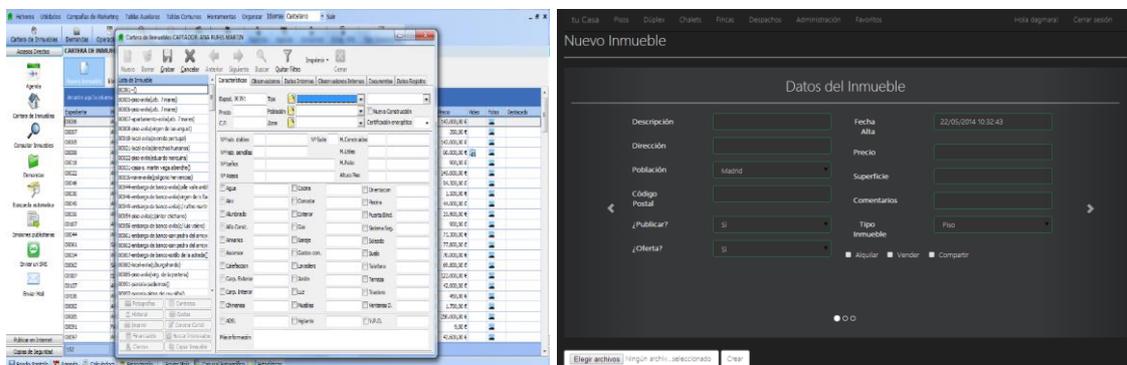
Se encuentra anexo al proyecto el manual de usuario que explica cómo dar de alta contenido en el portal web.

5.1. Casos Reales

Se presentan cuatro escenarios reales comparando la nueva aplicación web con el anterior sistema de la empresa inmobiliaria.

5.1.1. Alta de inmuebles en el portal

Las imágenes mostradas a continuación reflejan las diferencias entre la solución actual de la empresa y el proyecto nuevo:



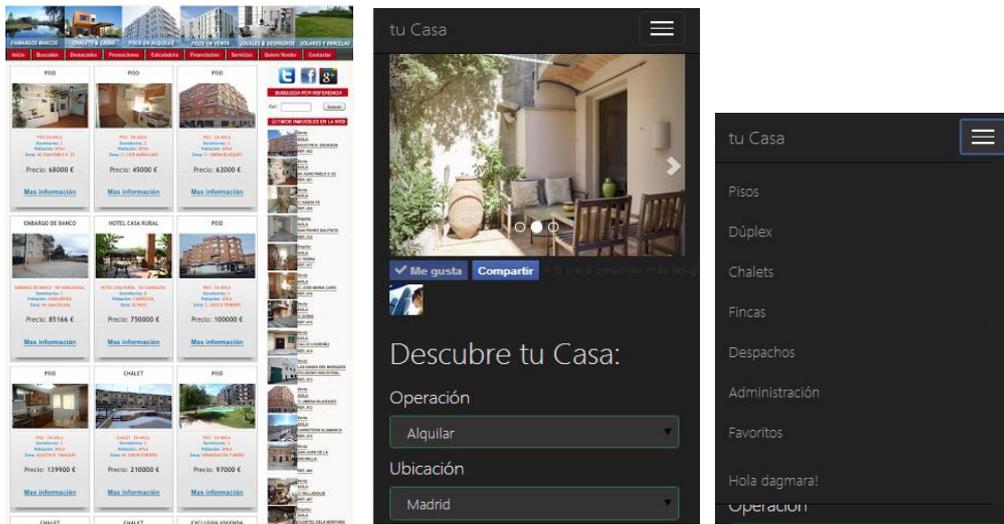
A simple vista se puede observar que el proyecto presenta una interfaz mucho más usable e intuitiva. Sin embargo, las ventajas no sólo se reducen a la presentación sino también al acceso desde otros dispositivos.

5.1.2. Portal corporativo

El acceso al portal corporativo desde otros dispositivos con una pantalla más pequeña se ve penalizado en la aplicación anterior, pues no es posible visualizar un texto y enlaces

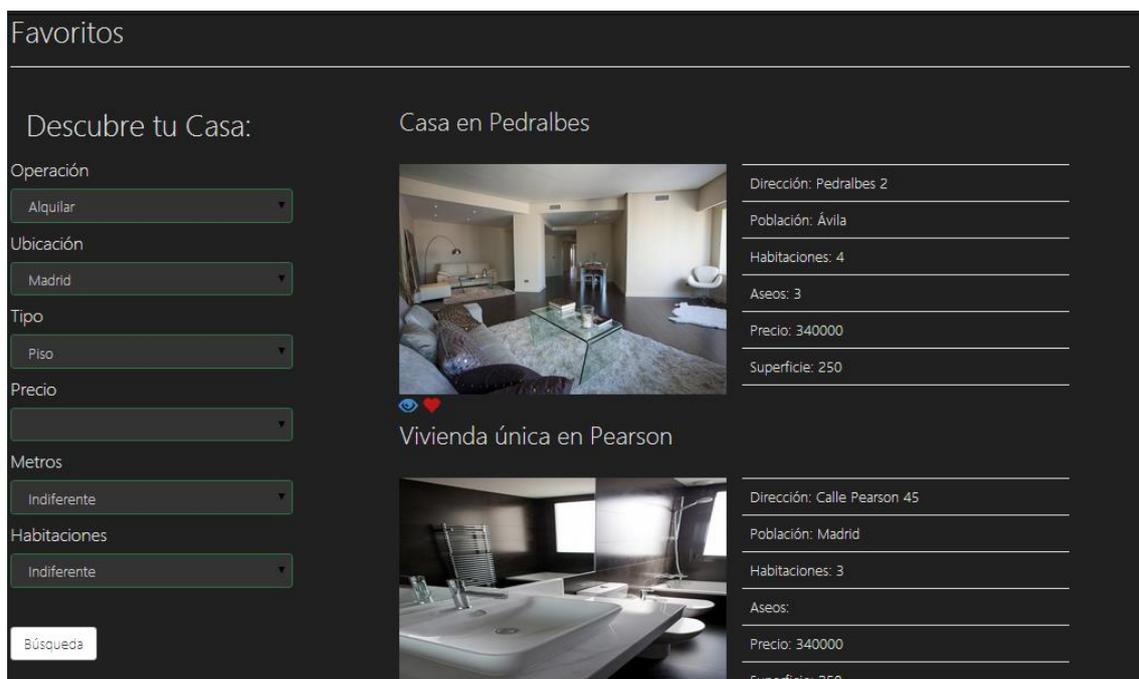
Aplicación de Gestión Inmobiliaria en la Nube

tan pequeños. En cambio, en la nueva aplicación el diseño del portal es responsivo y se adapta a todo tipo de pantallas:



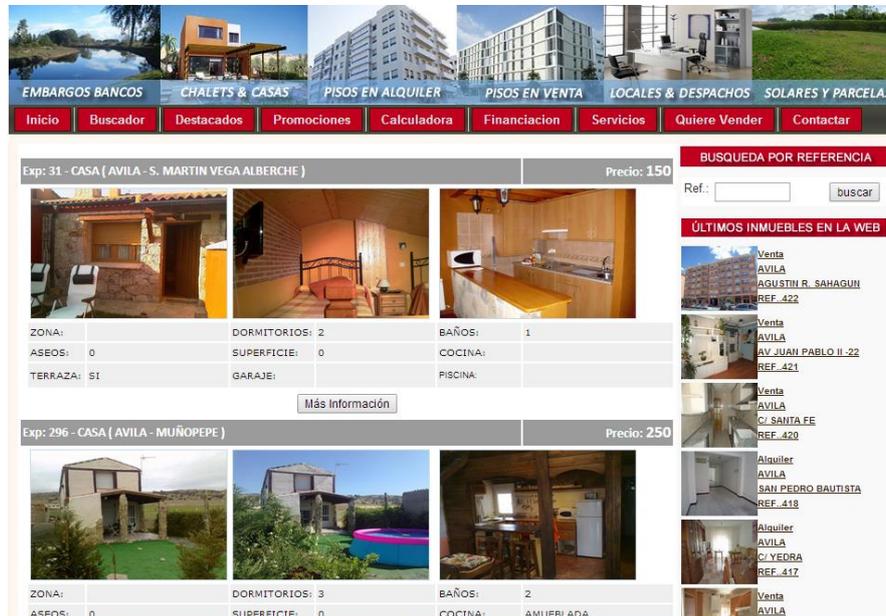
5.1.3. Interacción con los visitantes

Se facilita la interacción con los visitantes de la web añadiendo la posibilidad de que estos guarden los inmuebles que les hayan resultado más interesantes para posteriores visitas. Además, desde todas las páginas de listados de inmuebles, se muestra un buscador completo:



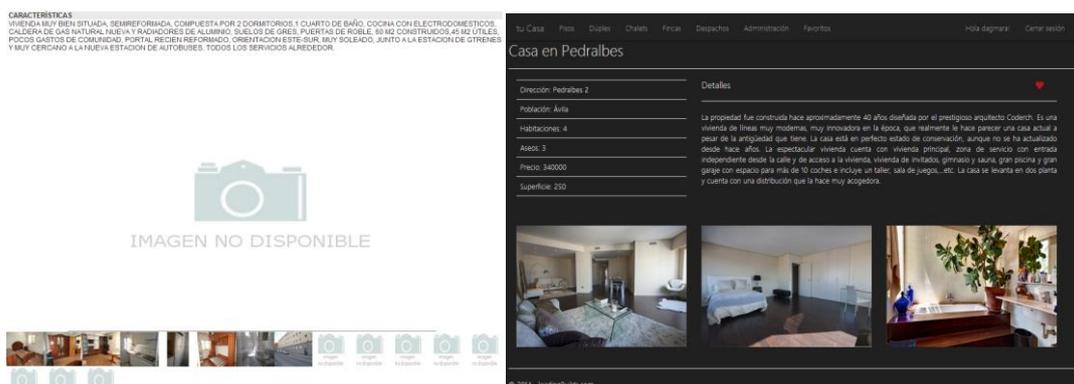
Aplicación de Gestión Inmobiliaria en la Nube

En el portal anterior la búsqueda sólo se podía realizar por referencia obligando a los visitantes interesados en un inmueble a apuntar manualmente el identificador del inmueble, tarea que resulta tediosa e invita al visitante a abandonar el portal.

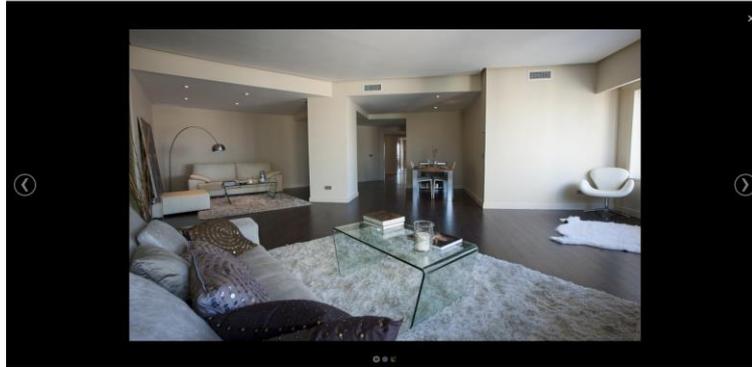


5.1.4. Acceso a los detalles de un inmueble

La imagen de la empresa tampoco se ve favorecida a la hora de presentar la información de los inmuebles. La aplicación no permite una correcta visualización de las fotos de los inmuebles, entorpeciendo la promoción de los mismos.



La presentación se ve notablemente mejorada incluyendo una galería de fotos que permite una mejor visualización. También se invita al usuario a añadir el inmueble a sus favoritos.



5.2. Comparación con otras plataformas

A continuación se realiza una comparativa del trabajo de fin de grado frente a otras plataformas de gestión inmobiliaria.

Sistema	Ventajas	Desventajas
<i>Trabajo Fin de Grado</i>	<ul style="list-style-type: none"> • Eficiente y escalable, nueva tecnología, duradera, con la interfaz que el usuario desee, en nube, barata a largo plazo • Permite exportar la base de datos. • Acceso desde todos los puntos, ya que está en nube. • Propiedad del código y de la BD. • Acceso por PCs y dispositivos móviles. 	<p>Más cara a corto plazo pues hay que hacerse cargo de los costes de desarrollo.</p>
<i>NetFincas</i>	<ul style="list-style-type: none"> • Paquete ya comprado, evitaría volver a pagar. • Permite un control de los datos pues su uso está limitado a los ordenadores en red de la empresa. 	<ul style="list-style-type: none"> • Ineficiente, interfaz poco atractiva, es necesaria una mejora, mantenimiento demasiado caro con respecto a calidad/precio • No escalable ni extensible. • No permite la gestión ubicua de contenido. • No accesible desde dispositivos que no dispongan de la licencia del producto. • No permite exportación de la base de datos.
<i>GestionInmobiliaria en Cloud</i>	<ul style="list-style-type: none"> • Accesible desde navegadores. Se vende como software en nube aunque no se han encontrado características propias de una cloud app. • Barata. 	<ul style="list-style-type: none"> • Incluyen funcionalidades que no van a utilizarse, no se puede cambiar lo que traen. Problemáticos a largo plazo puesto que no se pueden mejorar • No permite exportación de la base de datos.
<i>InmoFLEX</i>	<ul style="list-style-type: none"> • Herramienta de gestión inmobiliaria. No incorpora portal. • Accesible y diseño responsivo. • Acceso mediante navegador web facilita su uso desde móviles y tabletas. 	<ul style="list-style-type: none"> • Caro. La versión básica parte de 100€ y sólo incluye la gestión de un máximo de 100 inmuebles. • Al no incorporar portal inmobiliario no se adapta al modelo de negocio de la empresa.

6. Mantenimiento

El mantenimiento diseñado para el modelo de negocio al que se enfoca este proyecto está integrado por los siguientes servicios:

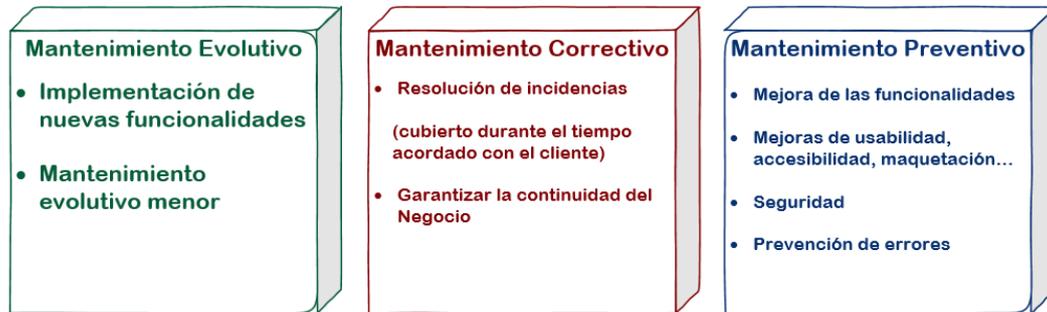


Ilustración 10: Mantenimiento propuesto

La metodología para el mantenimiento de servicios y aplicaciones tiene por objetivo principal optimizar los procesos de mantenimiento ofreciendo el apoyo de un servicio completo de gestión.

6.1. Fases del servicio de mantenimiento

Las fases que se definen en la prestación del servicio de mantenimiento son las siguientes:



Ilustración 11: Fases del servicio de mantenimiento

6.1.1. Definición

En esta fase se especifican los distintos procedimientos necesarios para la viabilidad y desarrollo del servicio.

- Procedimientos operativos
- Procedimientos de control y seguimiento
- Modelo de relación

Se detallan los indicadores de nivel de servicio así como su proceso de medición.

- Estudio de priorización de necesidades
- Análisis de indicadores
- Definición del proceso de medición

De igual modo, se preparan las infraestructuras necesarias para el desarrollo de las prácticas especificadas.

6.1.2. Transición

En la fase de transición se persigue arrancar el servicio de mantenimiento de modo que se valide los diferentes procedimientos e infraestructuras asegurando un correcto marco de colaboración.

6.1.3. Servicio

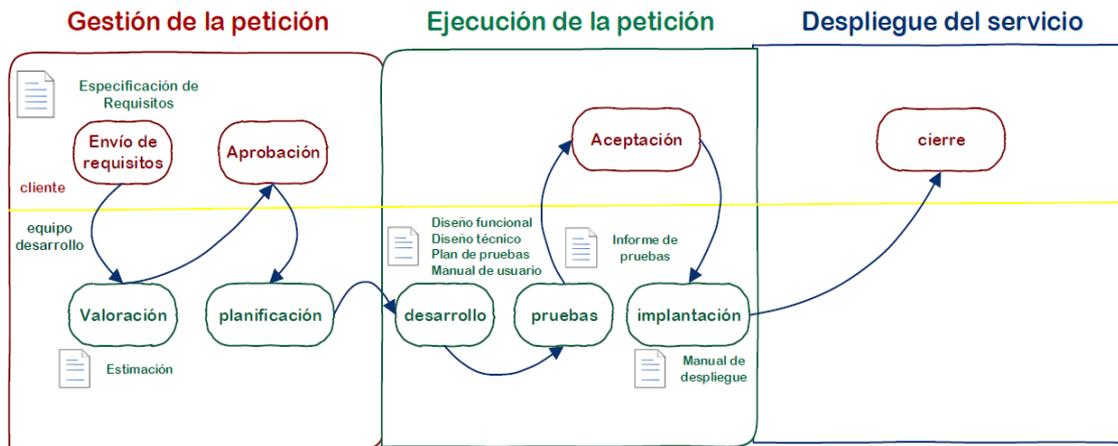
La fase de prestación del servicio cuenta con los mecanismos del modelo de relación probados y en funcionamiento, permitiendo la realización del seguimiento y adecuación constante del mismo.

Se lleva a cabo la monitorización del sistema con el fin de detectar puntos de mejora que beneficien el rendimiento y la usabilidad de forma que el equipo encargado de la prestación del servicio pueda dedicar más tiempo a la evolución y perfección del sistema. De este modo se pretende obtener un ciclo de vida más duradero para el software y una continua mejora de la plataforma.

Aplicación de Gestión Inmobiliaria en la Nube

Adicionalmente, se inicia un proceso de aseguramiento de la calidad orientado a mantener la calidad del producto, sus servicios y procedimientos.

Flujo de actividades en la petición de un servicio evolutivo



7. Conclusiones

El principal objetivo durante todo el proyecto ha sido el desarrollo de las bases de una buena aplicación de gestión inmobiliaria en la nube.

Dado que el proyecto surgió a raíz de una inmobiliaria real, ha sido necesario establecer una comunicación constante con la empresa para extraer los requisitos del proyecto, teniendo presente su modelo de negocio. Si bien es cierto que se podría haber realizado cualquier otro trabajo de fin de grado, se ha preferido proponer uno que tuviera un uso real posteriormente.

Ser consciente que de este proyecto va a depender la actividad empresarial de una inmobiliaria que ya cuenta con un elevado número de visitas mensuales ha planteado varias preguntas:

- ¿Qué tipo de personas van a utilizar esta aplicación?
- ¿Cuántos conocimientos de informática tienen?
- ¿Quién se va a encargar después de mantener la aplicación?
- ¿Qué les ha llevado a pedir una aplicación nueva?

La empresa solicitó el desarrollo de una aplicación en nube pues era lo que otras compañías le ofertaban por precios realmente bajos. Sin embargo, la empresa optó por colaborar en el desarrollo de este proyecto en lugar de contratar una oferta.

Con la aplicación en nube en mente y su posterior integración con Microsoft Office, se llevó a cabo una prueba de concepto que consumió una importante cantidad de tiempo pues no se tenían conocimientos previos del framework finalmente elegido: ASP.Net MVC5. Tras la prueba de concepto se comprendió que lo que esta empresa en concreto necesitaba no era una aplicación en nube, sino una aplicación web, ya que el coste por el alojamiento de la aplicación en servidores en nube todavía no le resulta asequible.

A pesar de ello, el título del TFG no se cambió y se trató de buscar cómo implementar una solución que fácilmente se pudiera adaptar a ambos modelos seguramente el coste de los servidores en nube descienda pronto y entonces sea factible desplegar la solución

Aplicación de Gestión Inmobiliaria en la Nube

en nube. El resultado ha sido una aplicación que puede ser utilizada tanto como cloud app como web app.

El resultado final de la primera entrega del proyecto, correspondiente a este Trabajo de Fin de Grado, está cumpliendo con todas las expectativas e incluso más de las inicialmente previstas, fruto de una buena elección del framework. Los problemas surgidos a lo largo del proyecto han contribuido a que el aprendizaje haya sido mucho más profundo y provechoso tanto a nivel personal como de cara al mundo laboral, donde la demanda de analistas .NET es elevada.

Tras el tiempo y esfuerzo invertido en el desarrollo del proyecto cabe destacar:

- ➔ Que se ha aprendido un nuevo lenguaje de programación y el uso de un framework cuya última versión es muy reciente.
- ➔ Que se ha propuesto el seguimiento de una metodología de software y unas pautas de mantenimiento que contribuyen al éxito del proyecto.
- ➔ Que se han estudiado diferentes alternativas tecnológicas y se ha procedido a elegir la que mejor se adaptaba a los requisitos del cliente.
- ➔ Que se ha aprendido a establecer diferencias entre aplicaciones en nube y aplicaciones web.
- ➔ Que se han estudiado las ventajas que ofrecen las bases de datos más utilizadas actualmente y se ha elegido la que presentaba una mejor sintonía con el framework utilizado.
- ➔ Que se ha priorizado el desarrollo de un software mantenible frente a un software con más funcionalidades de caja.

8. Líneas Futuras de Trabajo

Este Trabajo de Fin de Grado ha consistido en el desarrollo del núcleo principal de una aplicación de gestión inmobiliaria en la nube.

Ya que desde el inicio del proyecto se te tuvo en mente el desarrollo de una aplicación a mayor escala que integrara todas las funcionalidades propias de una aplicación en nube, se eligieron los requisitos que debería cumplir la primera entrega, cuyo resultado es el presente proyecto.

Las líneas futuras de trabajo se fundamentan en:

- La personalización del look and feel de la aplicación con los logos e imágenes corporativas de la empresa.
- Proveer de un diseño responsivo a todas las páginas del portal para mejorar la experiencia de usuario en móviles y tabletas.
- Proveer al proyecto de una solución de escritorio que permita la sincronización offline de los contenidos.
- Incorporar índices a la base de datos para mejorar el rendimiento de las consultas.
- Integrar la aplicación en nube con la suite Microsoft Office y desarrollar los módulos encargados de la facturación y contabilidad.
- Permitir el uso de la aplicación de forma conjunta con el correo electrónico.
- Desarrollar un sistema de guardado de búsquedas de los visitantes del portal para la generación de avisos en caso de que sean incorporados al portal inmuebles que los cumplan.
- Crear un sistema de alertas que informe acerca de los vencimientos de contrato de los inmuebles a los administradores de la aplicación.
- Implementar un sistema de eventos que facilite el registro de actividades de los empleados del negocio.
- Añadir mapas de Google a los inmuebles para facilitar la localización.
- Incorporar pasarelas a portales como idealista.com y fotocasa.com.

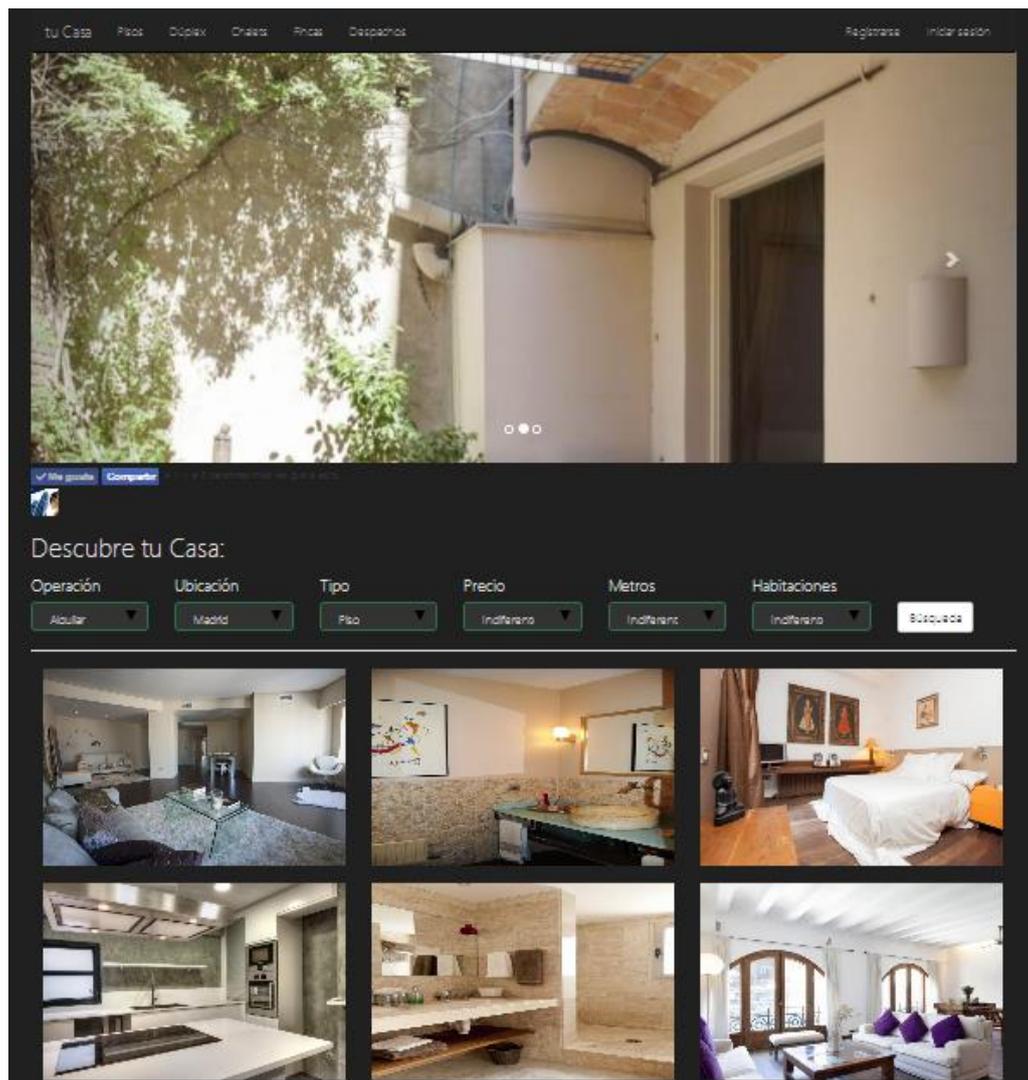
Anexo 1: Manual de Usuario

Página principal

Desde la página principal de la aplicación es posible acceder a las diferentes secciones de la aplicación. Desde la página es posible:

- ➔ Navegar por las secciones de inmuebles del menú de navegación superior.
- ➔ Acceder al detalle de los inmuebles ofertados.
- ➔ Realizar búsquedas.
- ➔ Compartir la página en redes sociales.
- ➔ Iniciar una sesión de usuario.
- ➔ Registrarse con una nueva cuenta de usuario.

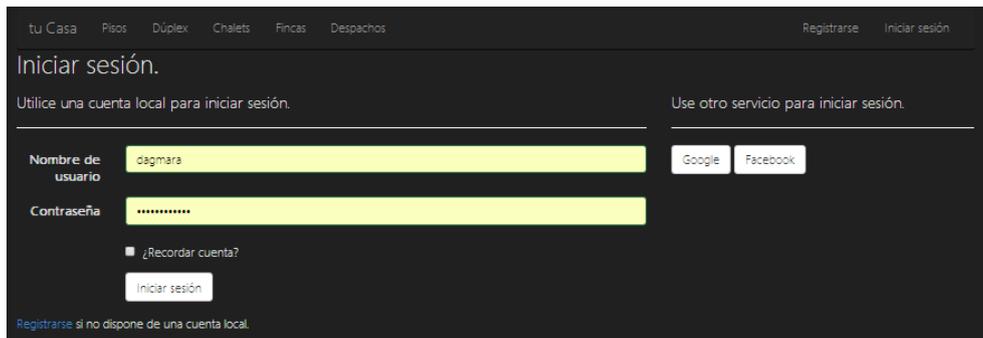
Para visualizar la sección de Administración será necesario iniciar sesión con un usuario que posea privilegios de administrador.



Aplicación de Gestión Inmobiliaria en la Nube

Iniciar sesión

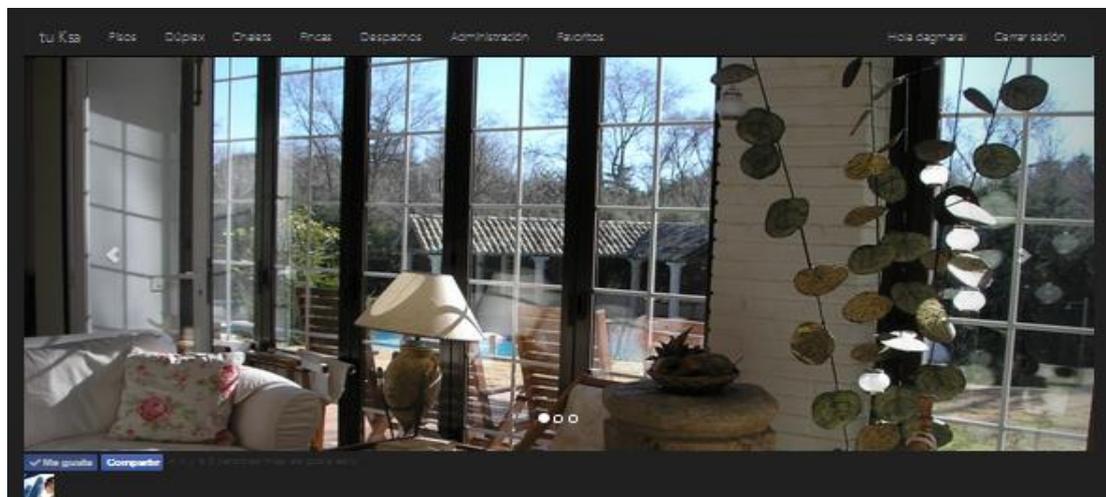
Para iniciar sesión en la aplicación será necesario pulsar sobre el enlace **“Iniciar sesión”** del menú superior de la página. Tras ello el usuario será redirigido a la página login desde la que podrá escribir sus credenciales o bien acceder a través de una cuenta Google o Facebook:



Si el usuario no tuviera una cuenta registrada o no deseara autenticarse mediante alguno de los servicios externos, deberá registrarse en la aplicación:



Una vez iniciada la sesión, aparecerá el nombre del usuario en el menú de la página junto con la sección de favoritos y administración si el usuario fuere administrador:



Aplicación de Gestión Inmobiliaria en la Nube

Realizar una búsqueda

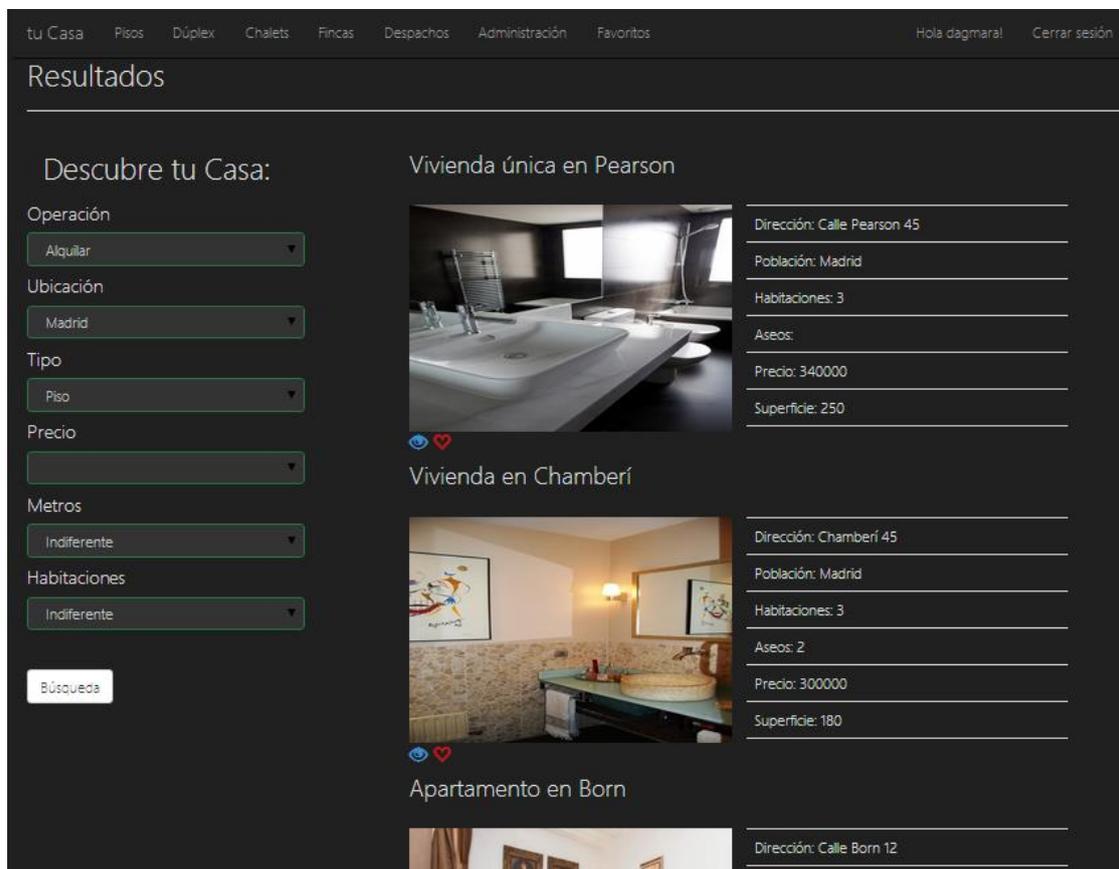
Para realizar una búsqueda de inmuebles bastará con rellenar la sección de búsqueda desde cualquiera de las páginas en las que sea visible, rellenando uno o varios filtros y presionando el botón **Búsqueda**.



Descubre tu Casa:

Operación	Ubicación	Tipo	Precio	Metros	Habitaciones	Búsqueda
Alquilar	Madrid	Piso	Indiferente	Indiferente	Indiferente	

Tras ello aparecerá la página de resultados:



tu Casa Pisos Dúplex Chalets Fincas Despachos Administración Favoritos Hola dagmaral Cerrar sesión

Resultados

Descubre tu Casa:

Operación: Alquilar

Ubicación: Madrid

Tipo: Piso

Precio: Indiferente

Metros: Indiferente

Habitaciones: Indiferente

Búsqueda

Vivienda única en Pearson

Dirección: Calle Pearson 45

Población: Madrid

Habitaciones: 3

Aseos:

Precio: 340000

Superficie: 250

Vivienda en Chamberí

Dirección: Chamberí 45

Población: Madrid

Habitaciones: 3

Aseos: 2

Precio: 300000

Superficie: 180

Apartamento en Born

Dirección: Calle Born 12

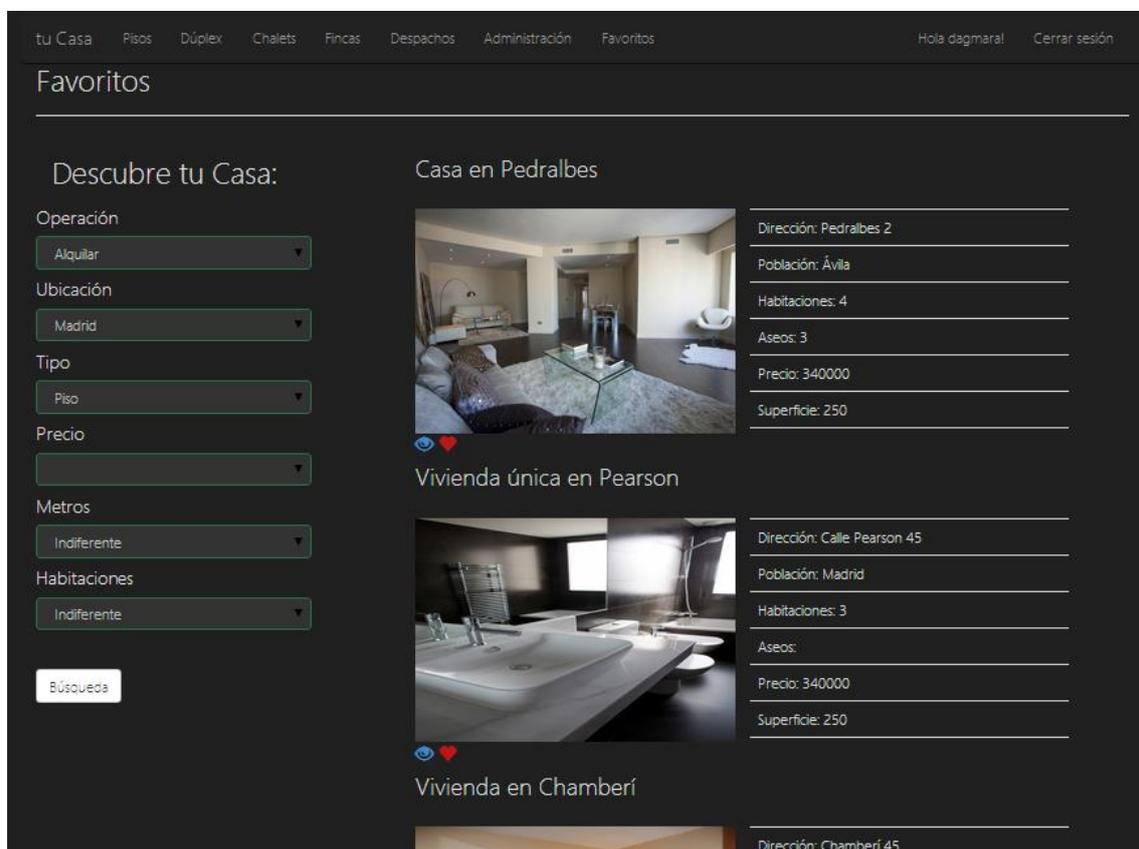
Agregar un inmueble a favoritos

Para agregar un inmueble a la sección de favoritos basta con pulsar sobre la imagen del corazón vacío. El corazón se ve lleno si el inmueble está actualmente añadido a favoritos. Para proceder al guardado el usuario deberá iniciar sesión en el sitio si no lo ha hecho previamente.

Aplicación de Gestión Inmobiliaria en la Nube



Tras seleccionar el inmueble favorito, este se encontrará visible en la sección de favoritos del usuario:



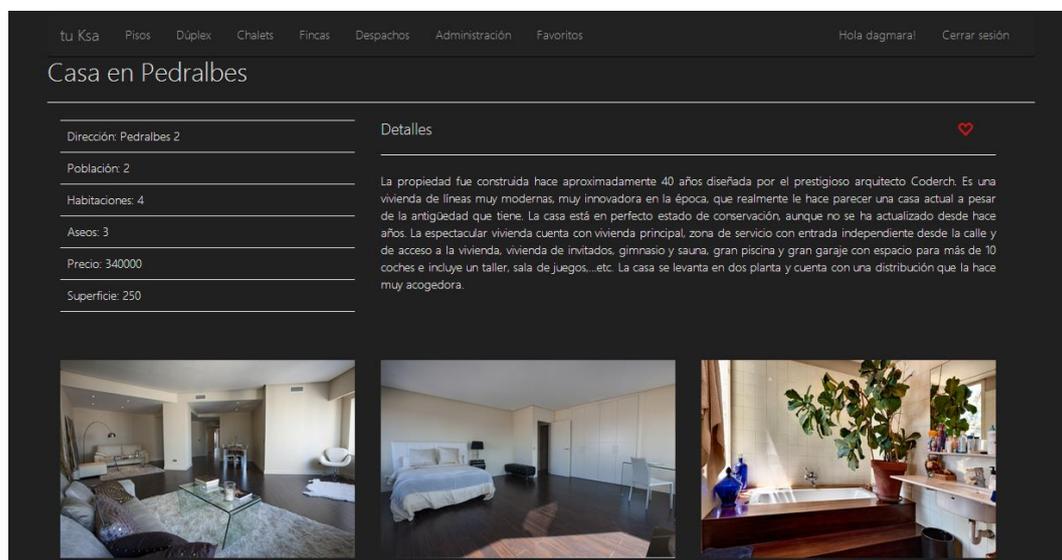
Ver el detalle de un inmueble

Es posible acceder a la información detallada de un inmueble pulsando sobre el icono en forma de ojo o sobre la imagen de cualquier inmueble visible en el portal:

Aplicación de Gestión Inmobiliaria en la Nube



Tras ello se accede a la página en la que aparece la información detallada. Si el inmueble dispone de imágenes asociadas es posible visualizarlas en grande haciendo clic sobre cualquiera de ellas.



Cuando se abre la galería de imágenes se puede navegar por ellas pulsando sobre las flechas derecha e izquierda. Para salir de la galería se pulsa el aspa superior:



Aplicación de Gestión Inmobiliaria en la Nube

Acceso al perfil de administración

Para acceder a la administración del sitio es necesario disponer de privilegios de administrador. Se puede acceder desde cualquier página pulsando sobre el enlace "Administración" de la barra de navegación superior:

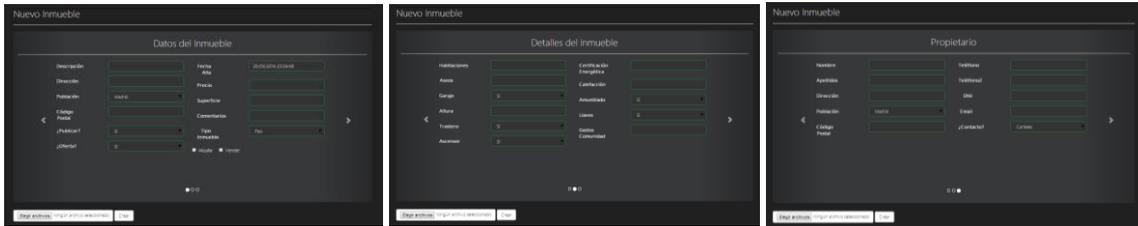


Alta de un Inmueble

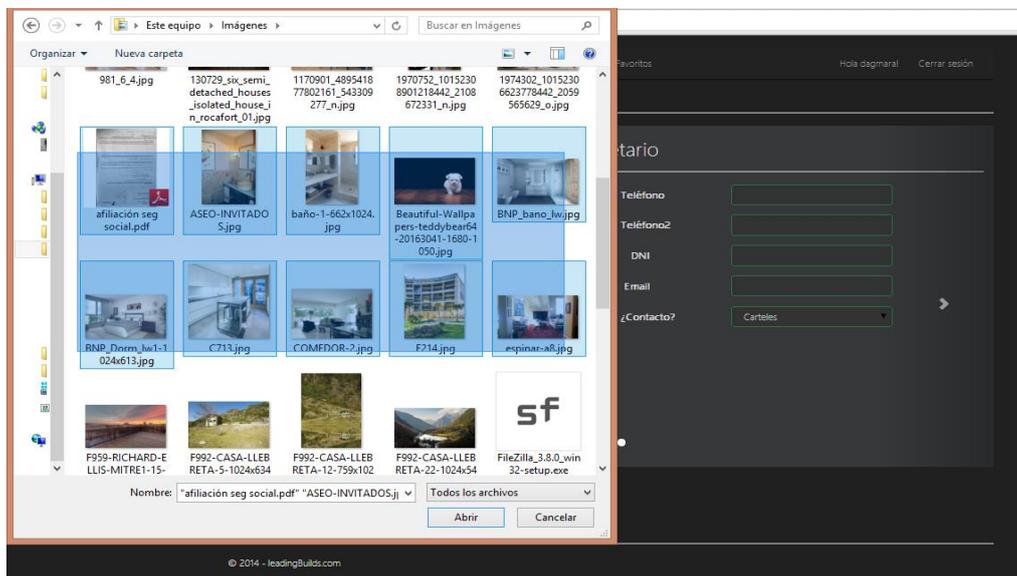
Para dar de alta un inmueble se accederá a la sección de inmuebles desde el panel de administración y se pulsará la opción "**Crear Nuevo**". Tras ello se podrá visualizar la siguiente página:

Aplicación de Gestión Inmobiliaria en la Nube

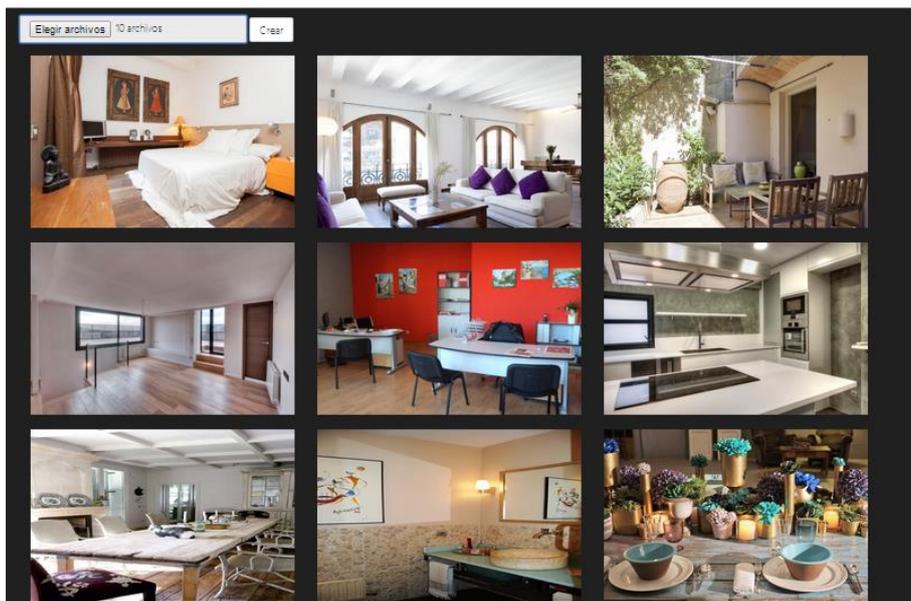
El formulario cuenta con 3 secciones referentes a los datos del inmueble, detalles y propietario:



Para agregar fotos y documentos relativos al inmueble se deberá hacer clic sobre el botón **"Elegir archivos"**:



Tras pulsar **"Abrir"** se verán agregados a la página:



Aplicación de Gestión Inmobiliaria en la Nube

En la sección de Datos del Inmueble del formulario, es importante tener en consideración lo siguiente:

- El campo "Publicar" indica si un inmueble es visible en los listados de inmuebles del portal inmobiliario.
- El campo "Oferta" indica si el inmueble se desea publicar como oferta en la página principal del portal.
- Las operaciones "Alquilar" y "Vender" son variables. Aparecen tantas operaciones para seleccionar como estados de inmueble dados de alta (con flag_disponible igual a Verdadero, ver "Alta de un Estado de Inmueble").

The screenshot shows a web form titled "Nuevo Inmueble" with a sub-section "Datos del Inmueble". The form contains the following fields and values:

Descripción	Casa Independiente en La Finca	Fecha Alta	20/05/2014 23:26:48
Dirección	La Finca 45	Precio	2000000
Población	Madrid	Superficie	500
Código Postal	28750	Comentarios	Ubicada en una lujosa urbanización, c...
¿Publicar?	Sí	Tipo Inmueble	Chalet
¿Oferta?	No		

Below the form, there are radio buttons for "Alquilar" (unchecked) and "Vender" (checked). At the bottom, there is a section for "Elegir archivos" (10 archivos) and a "Crear" button. Three images are displayed: a bedroom with a white bed, a living room with a white sofa and purple cushions, and an outdoor patio area with a table and chairs.

La sección de Detalles del Inmueble contiene la información relativa a las habitaciones, aseos, certificación energética...

Los datos de esta sección son opcionales pues se entiende que un inmueble como un solar no tiene por qué tener habitaciones. En general, la información obligatoria se encuentra en la primera sección. Con ello se pretende agilizar el dado de alta.

The screenshot shows a mobile application interface for adding a new property. The title is 'Nuevo Inmueble' and the section is 'Detalles del Inmueble'. The form contains the following fields:

Habitaciones	4	Certificación Energética	G
Aseos	3	Calefacción	Eléctrica
Garaje	Sí	Amueblado	Sí
Altura	2	Llaves	Sí
Trastero	Sí	Gastos Comunidad	100
Ascensor	Sí		

Navigation arrows are visible on the left and right sides, and a progress indicator at the bottom shows three dots, with the second dot filled, indicating the current step.

La información del propietario del inmueble se rellena en la tercera sección del formulario. Es obligatorio indicar al menos el nombre, el apellido y el contacto, que se refiere al medio por el que la persona ha conocido la inmobiliaria.

The screenshot shows the 'Propietario' section of the 'Nuevo Inmueble' form. The fields are filled with the following information:

Nombre	Álvaro	Teléfono	91822388
Apellidos	Bermúdez	Teléfono2	60654433
Dirección	Calle Los Sauces 29	DNI	67644778M
Población	Madrid	Email	alvaro.bermudez@mail.com
Código Postal	28764	¿Contacto?	Web

Navigation arrows are visible on the left and right sides, and a progress indicator at the bottom shows three dots, with the third dot filled, indicating the current step.

Dar de alta un nuevo tipo de inmueble

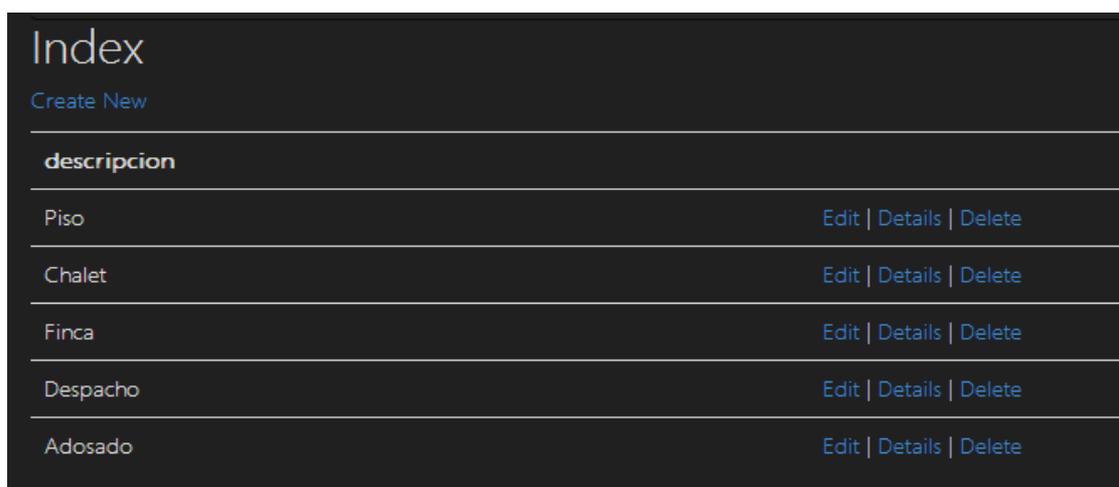
Los tipos de inmueble se refieren a las categorías a las que puede pertenecer un inmueble (Pisos, Chalets, Fincas, etc).

Aplicación de Gestión Inmobiliaria en la Nube

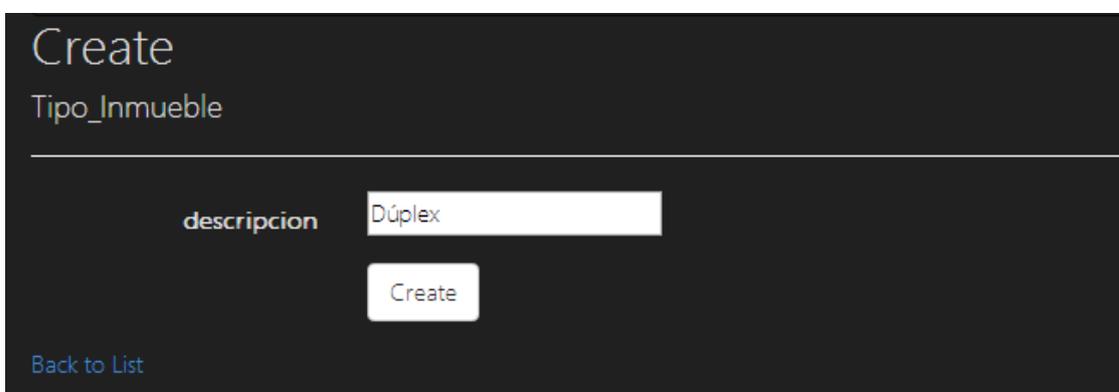
Para dar de alta un nuevo tipo de inmueble se accederá desde el menú de administración a la opción "**Categorías**":



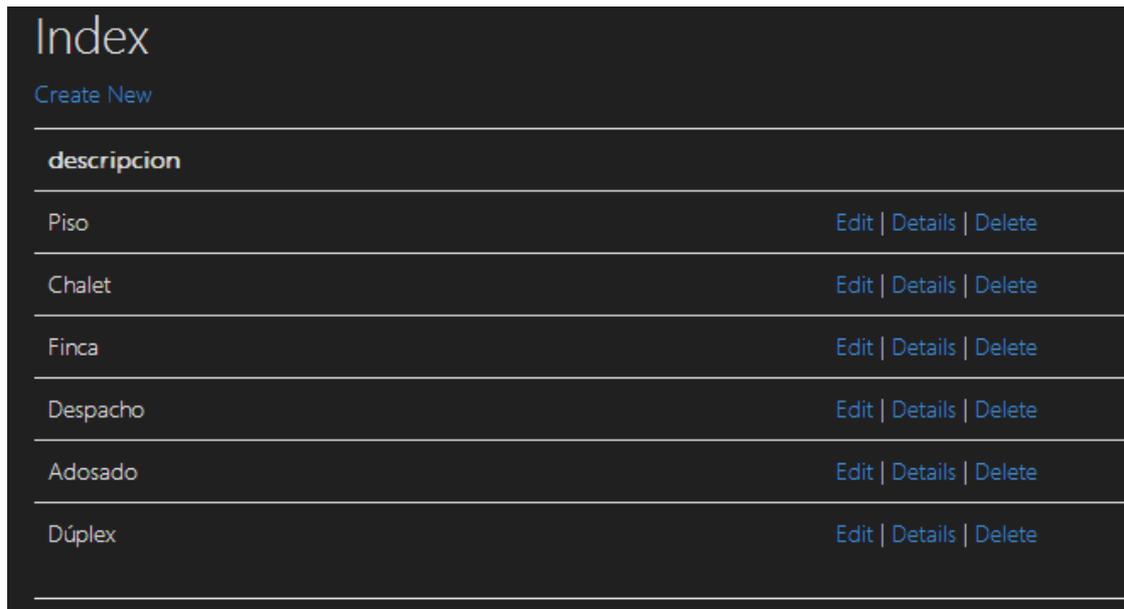
Tras ello, se pulsa sobre la opción "**Create New**":



Y se rellena el formulario con el nombre del nuevo tipo de inmueble:



Tras pulsar el botón "**Crear**" se verá el nuevo tipo incluido en el listado:



The screenshot shows a dark-themed web interface. At the top left, the word 'Index' is displayed in a large, light-colored font. Below it, there is a link labeled 'Create New' in a smaller, light blue font. The main content is a table with a dark background and light text. The table has a header row with the word 'descripcion' in a bold, light-colored font. Below the header, there are seven rows, each representing a property type. Each row contains the property type name on the left and a set of three links ('Edit | Details | Delete') on the right, all in a light blue font. The property types listed are: Piso, Chalet, Finca, Despacho, Adosado, and Dúplex.

descripcion	
Piso	Edit Details Delete
Chalet	Edit Details Delete
Finca	Edit Details Delete
Despacho	Edit Details Delete
Adosado	Edit Details Delete
Dúplex	Edit Details Delete

Dar de alta un nuevo estado de inmueble

Los estados de los inmuebles sirven para indicar la operación para la que está disponible un inmueble y si este se encuentra o no ocupado.

Ej: Un inmueble está disponible para venta, por tanto al crear el inmueble se selecciona la opción "Vender". El flag_disponible del estado vender es verdadero, lo que indica que el inmueble está disponible para ser vendido.

Si el inmueble se llega a vender, se modificaria su estado asignándole el estado "Vendido", cuyo flag_disponible igual a falso indicaría que el inmueble ya está ocupado.

Es posible asignar varios estados a un inmueble pero hay que tener en cuenta que si alguno de los estados indica que el inmueble está ocupado, entonces el inmueble no estará disponible para ninguna operación más.

Seleccionando la opción "**Create New**" se accede a la página que permite crear un estado nuevo:

Aplicación de Gestión Inmobiliaria en la Nube

descripcion	flag_disponible	
Alquilado	Falso	Edit Details Delete
Vendido	Falso	Edit Details Delete
Alquilar	Verdadero	Edit Details Delete
Vender	Verdadero	Edit Details Delete

Pulsar el botón **"Create"**:

Estado_Inmueble

descripcion:

flag_disponible:

[Back to List](#)

Una vez dado de alta el estado nuevo, será visible en el listado de estados:

descripcion	flag_disponible	
Alquilado	Falso	Edit Details Delete
Vendido	Falso	Edit Details Delete
Alquilar	Verdadero	Edit Details Delete
Vender	Verdadero	Edit Details Delete
Compartir	Verdadero	Edit Details Delete

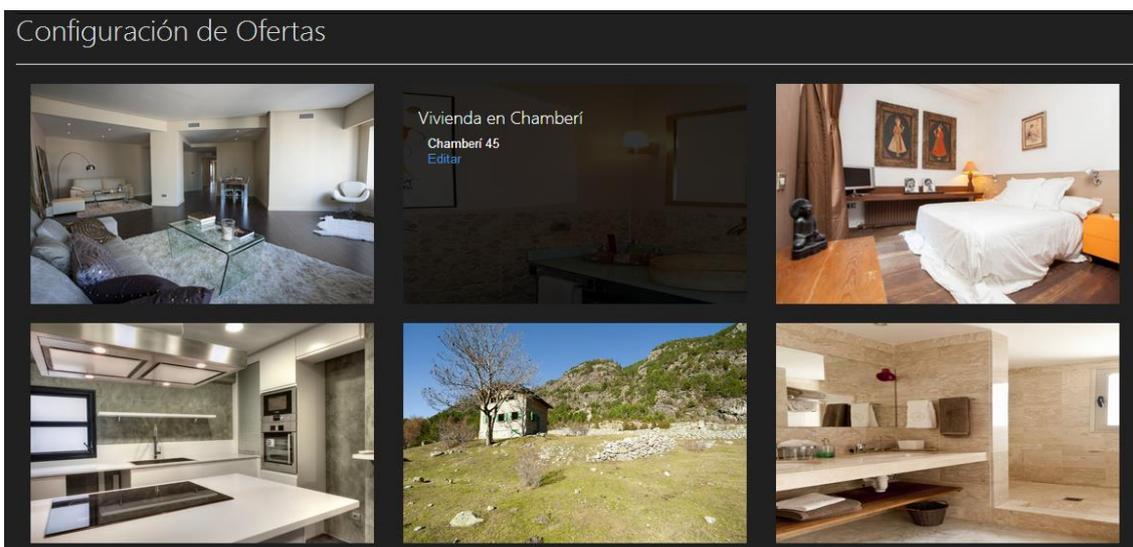
Configurar las ofertas de la página principal

La sección de ofertas de inmuebles visible en la página principal del portal se puede configurar accediendo a la sección **"Ofertas"** del panel de administración:

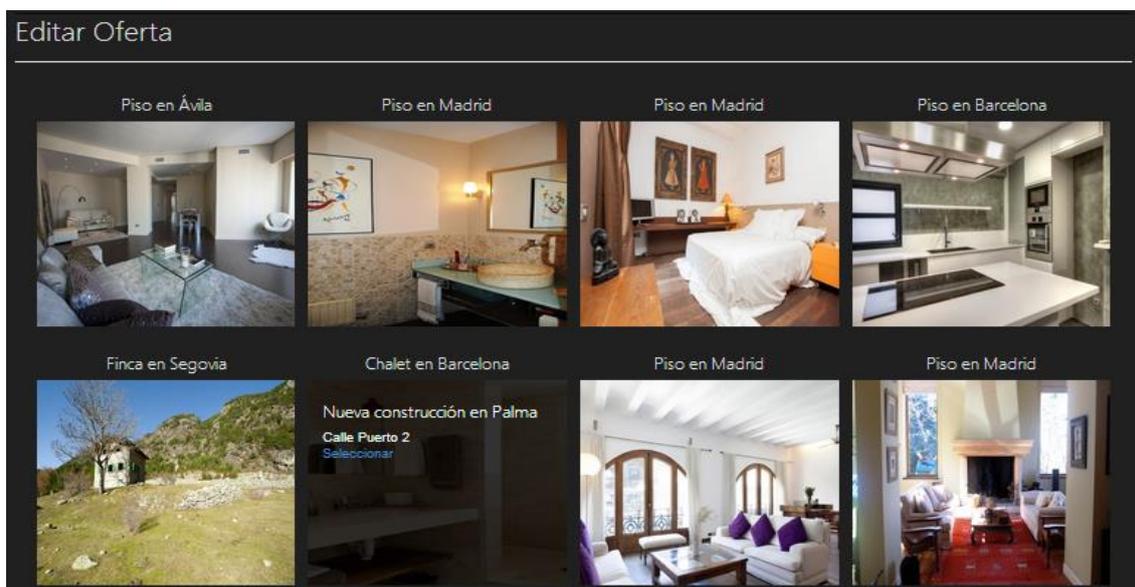
Aplicación de Gestión Inmobiliaria en la Nube



Tras ello, aparecen visibles las ofertas actuales de la web. Para cambiar una oferta se deberá pulsar la opción **"Editar"** que aparece sobre la imagen de la misma:



Tras ello, aparecen los inmuebles publicados en la web. Para seleccionar la nueva oferta basta con pulsar **"Seleccionar"**:



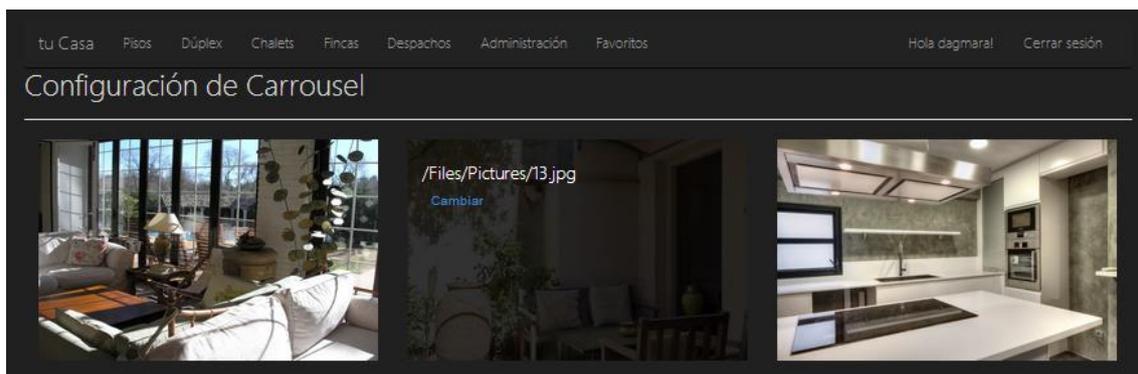
Aplicación de Gestión Inmobiliaria en la Nube

Configurar el carrusel de la página principal

El carrusel de fotos de la página principal del portal se puede configurar accediendo a la sección "**Carrusel**" del panel de administración:



Tras ello, aparecen visibles las fotos actuales del carrusel. Para cambiar una foto se deberá pulsar la opción "**Cambiar**" que aparece sobre la imagen de la misma:



Tras ello, aparecen las imágenes publicados en la web. Para seleccionar la nueva foto del carrusel basta con pulsar "**Seleccionar**":

Anexo 2: Clase BlobStorageService

```
using Microsoft.WindowsAzure.ServiceRuntime;
using Microsoft.WindowsAzure.Storage;
using Microsoft.WindowsAzure.Storage.Blob;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace AplicacionGestionInmobiliaria.Controllers
{
    public class BlobStorageService
    {
        public CloudBlobContainer GetCloudBlobContainer()
        {
            CloudStorageAccount storageAccount = CloudStorageAccount.Parse(RoleEnvironment.GetConfigurationSettingValue("ImageStorageAccountConn"));
            CloudBlobClient blobClient = storageAccount.CreateCloudBlobClient();
            CloudBlobContainer blobContainer = blobClient.GetContainerReference("myimages");
            if (blobContainer.CreateIfNotExists())
            {
                blobContainer.SetPermissions(new BlobContainerPermissions { PublicAccess = BlobContainerPublicAccessType.Blob });
            }
            return blobContainer;
        }
    }
}
```