

REAL ESTATE

App para el alquiler y venta de viviendas



1. Introducción

El proyecto, que se realizará individualmente, consiste en la implementación de una aplicación móvil para Android que sirva para gestionar el alquiler y venta de viviendas.

2. Arquitectura de la API

La API está implementada con el generador *rest generator* que hemos usado ampliamente en clase, y funciona de la misma forma.

Para conocer las diferentes peticiones que se pueden usar, los parámetros y los tipos de retorno, puedes ejecutar el comando

```
npm run docs
```

y automáticamente generará la documentación.

La versión funcional del api subida a heroku la podéis utilizar a través de la siguiente URL

<https://morning-chamber-30708.herokuapp.com>

La MASTERKEY que se está utilizando es hzasFUC4txHXVHHWvjR2R7UxQ6kzBlw

3. Funcionalidades

La aplicación debe implementar las siguientes funcionalidades:

3.1. Sin login

- **Mostrar un listado de las viviendas.** Como se comentó en clase, hay que tener en cuenta el ámbito que se le va a dar a la app, y en función del mismo, las viviendas que se mostrarán inicialmente. Se puede plantear el que la primera vez que el usuario inicie la app, aparezca una forma de seleccionar la categoría y la localización de las viviendas a buscar.
- Este listado debe implementar un **filtro** por:
 - Categoría (Alquiler, compra y obra nueva).
 - Ciudad
 - Código Postal
 - Provincia
 - Número de habitaciones
 - Tamaño (en metros cuadrados)

- Precio
- Las viviendas que se muestran en el listado filtrado deben poder visualizarse también a **través de un mapa**.
- Al pulsar sobre una vivienda, **mostrar la información completa de la vivienda**. Una vivienda puede tener más de una imagen, y se deben de poder visualizar todas ellas.

3.2. Para usuarios registrados

- Como es lógico, la posibilidad de realizar el **registro**, el **login** y el **logout**.
- Las *mismas funcionalidades que para un usuario sin registro*. Se añade la posibilidad de **marcar/desmarcar un inmueble como favorito**.
- **Añadir nuevos inmuebles**, a través de un formulario con los campos correspondientes. Un inmueble puede tener 0, 1 o N fotografías asociadas).
- **Eliminar las fotografías** de un inmueble. No se pueden editar las mismas.
- **Editar un inmueble** (todos los datos menos las fotografías, que solo se pueden añadir o eliminar).
- **Eliminar un inmueble** (el API se encarga de que, al eliminar un inmueble, se eliminen todas sus fotografías).

3.3. Se valorará

- La implementación de otras funcionalidades que se añadan a las descritas anteriormente.
- Un buen diseño y experiencia de usuario.
- La implementación a través de un código limpio, ordenado y documentado.
- La originalidad, tanto a nivel de diseño, navegabilidad y experiencia de usuario, como de código.

4. Entrega, calificación y corrección

La fecha de entrega del trabajo será **antes de las 14:30 del día 4 de Marzo**.

- Los profesores determinarán aquellas horas de clase que pueden dedicarse a realizar dicho trabajo.
- Se usarán los criterios de corrección establecidos en los módulos profesionales.
- Este trabajo producirá las siguientes calificaciones:
 - Una nota de examen para el módulo de Programación de Servicios y Procesos.
 - Una nota de examen para el módulo de Acceso a datos.

- Una nota de examen para el módulo de Programación Multimedia y Dispositivos Móviles.
- Una nota de examen para el módulo de Desarrollo de Interfaces.