



1 Airline Rest

La empresa Airline Rest nos ha contratado para crear un servicio web rest para la administración de sus vuelos. En concreto quiere administrar lo siguiente:

- Consulta de vuelos.
 - Se pedirá como entrada un origen y un destino.
 - Siendo el endpoint un method GET: /flights?ori=ORI&des=DES
 - Se obtendrá un JSON como respuesta indicando los vuelos que hay
 - Si no hay vuelos se mostrara un HTTP Status NOT FOUND
- Consulta un vuelo. Se pedirá un código de vuelo y una fecha:
 - Se pedirá el id del vuelo
 - Siendo el endpoint un method GET /flights/ID_VUELO?date=FECHA
 - Se obtendrá un JSON como respuesta indicando el vuelo
 - Si no existe el vuelo se mostrará un HTTP Status NOT FOUND
- Asociar un pasajero a un vuelo.
 - Endpoint con method POST: /flights/ID_VUELO/passenger.
 - Los datos del pasajero irán en un JSON en el cuerpo de la petición.
 - Se devolverá un HTTP Status CREATED si ha sido creado.
 - Si no existe el vuelo se devolverá un NOT_FOUND
 - Si hay algún tipo de error en la validación del pasajero se devolverá un HTTP Status BAD REQUEST
- Consulta si un pasajero va en un vuelo.
 - Endpoint con method GET: /flights/ID_VUELO/passenger/NIF.
 - Se mostrará un JSON si existe.
 - Si no existe el vuelo o el pasajero se mostrara un HTTP Status NOT FOUND
- Actualizar un pasajero en un vuelo
 - Endpoint con method PUT: /flights/ID_VUELO/passenger/NIF.
 - Los datos del pasajero irán en un JSON en el cuerpo de la petición.
 - Se devolverá un HTTP Status OK si ha sido actualizado.
 - Si el vuelo o el pasajero no existe se devolverá NOT FOUND
 - Si hay algún tipo de error en la validación del pasajero se devolverá un HTTP Status BAD REQUEST
- Eliminar un pasajero en un vuelo
 - Endpoint con method DELETE: /flights/ID_VUELO/passenger/NIF.
 - Se devolverá un HTTP Status OK si ha sido eliminado.
 - Si el vuelo o el pasajero no existe se devolverá NOT FOUND
- Saber todos los pasajeros que van en un vuelo
 - Endpoint con method GET: /flights/ID_VUELO/passengers.
 - Se mostrará un JSON con los pasajeros.
 - Si no existe el vuelo o el pasajero se mostrara un HTTP Status NOT FOUND

Se pide crear un servicio web usando Springboot y un cliente que permita ejecutar todas y cada una de las peticiones que se piden en el servicio.

Se facilita dos proyectos maven (uno para el servicio y otro para el cliente) con la estructura y dependencias creadas.



1.1 Criterios de evaluación

CRITERIOS DE EVALUACIÓN ASOCIADOS AL RA Y SU PONDERACIÓN EN EL RA
RA4.Desarrolla aplicaciones que ofrecen servicios en red, utilizando librerías de clases y aplicando criterios de eficiencia y disponibilidad.

Criterios		Instrumentos	%
d)	Se han desarrollado y probado servicios de comunicación en red.	Práctica	10
e)	Se han utilizado clientes de comunicaciones para verificar el funcionamiento de los servicios.	Práctica	8
f)	Se han incorporado mecanismos para posibilitar la comunicación simultánea de varios clientes con el servicio.	Práctica	4
g)	Se ha verificado la disponibilidad del servicio.	Práctica	5
h)	Se han depurado y documentado las aplicaciones desarrolladas.	Práctica	5
			32%

1.2 Entrega

Se deberá subir el proyecto servicio, el cliente y un pdf con una memoria individual con evidencias que demuestren paso a paso, como se han resuelto algunos problemas que hayáis encontrado, en caso de no haber encontrado ningún error, cómo habéis depurado alguna ejecución.

El nombre del fichero comprimido debe contener vuestro nombre y apellidos sin tildes, eñes ni ningún otro carácter especial, con el siguiente formato:

apellidos_nombre_practicaUT4.zip