

**TEMA:**

CHECK IN

**CASO DE ESTUDIO:**

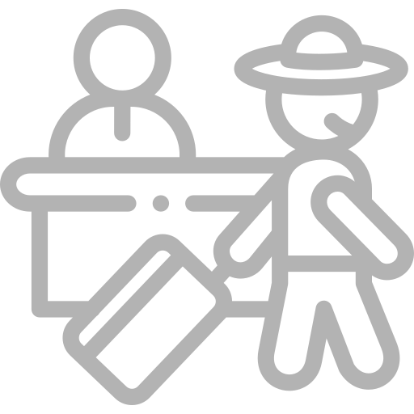
AEROLINEA

SANTA CRUZ - BOLIVIA

**AUTOR:**

JONATHAN SIMONS AÑEZ

**PROYECTO**



1. **Check – Caso de Estudio “Aerolinea”**

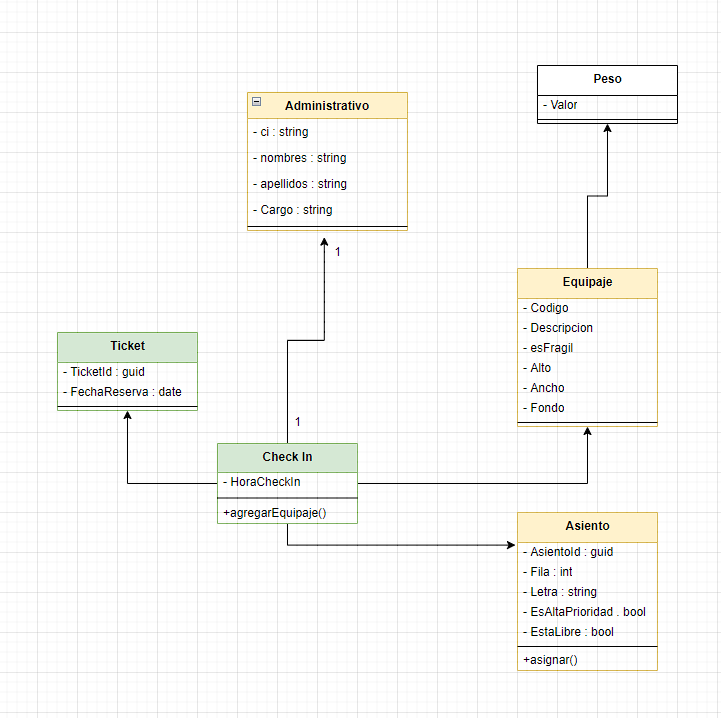
“Check In

Cuando un pasajero se acerca a la recepción en el día del vuelo, se realiza el proceso de Check In en el cual se etiqueta el equipaje y se le asigna un asiento en base a la hora de llegada y la condición del pasajero, si necesita asistencia se le asigna un asiento cerca a la puerta de ingreso.”

1. **Diseño**



1. **Diagrama**



1. **Repositorio del proyecto**

<https://github.com/jonathansimonsa/airline-ms-checkin.git>

1. **Contenerizacion en Docker**

Link de descarga de Docker:

<https://www.docker.com/products/docker-desktop/>

Documentacion de Docker:

<https://docs.docker.com/docker-hub/>

Repositorio de Docker:

<https://hub.docker.com/>

**Paso 1.**

Desde Visual Studio publicar archivos del proyecto:

(Terminal)

dotnet publish -c Release

El comando ejecutado genera los archivos de publicación en su directorio (en este caso):

D:\GitHub\airline-ms-checkin\CheckIn.WebApi\bin\Release\net5.0\publish

Se puede probar la ejecución de la aplicación con el comando:

dotnet .\MyApiDocker.dll

**Paso 2.**

En esta ubicación crearemos nuestro archivo Dockerfile para la creación de la imagen de docker:



**Paso 3.**

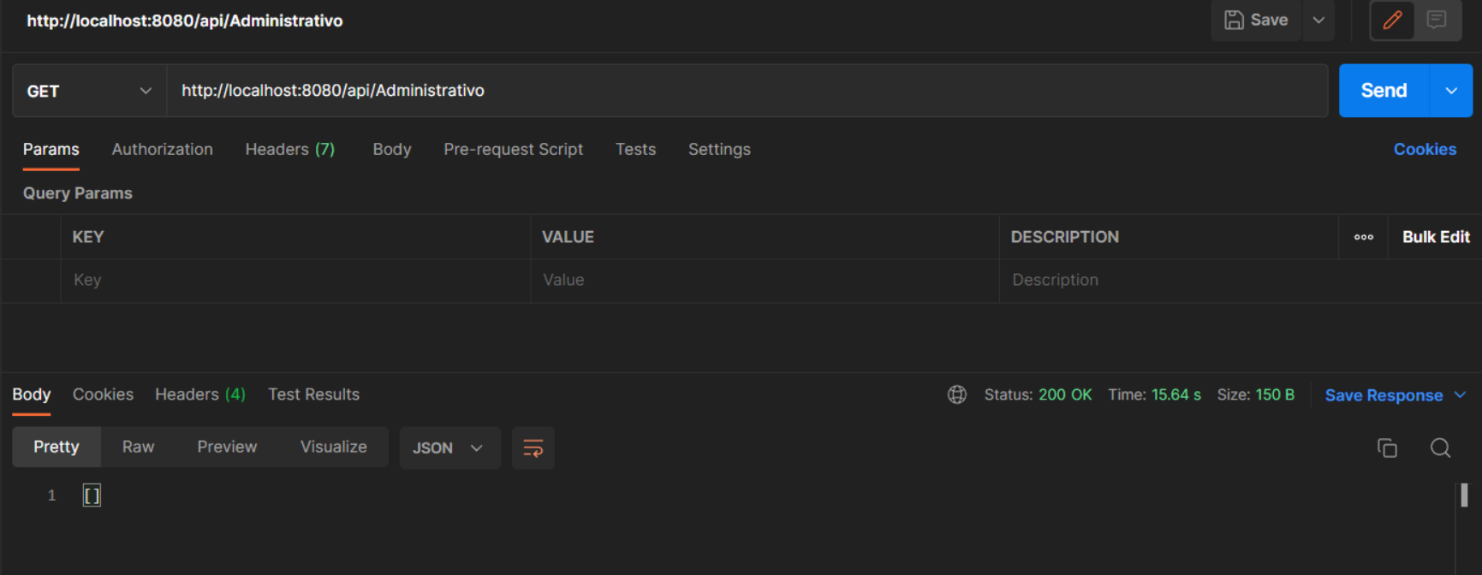
Una vez creado el archivo, abrimos la consola de PowerShell y ejecutamos el comando:

docker build -t ms-checkin:1.0 .

**Paso 4.**

después de ejecutar dicho comando podremos ejecutar la imagen en un contenedor con:

docker container run -d -p 8080:80 ms-checkin:1.0

en este caso como se trata de una aplicación WebApi, comprobaremos la funcionalidad con Postman:

Respuesta con Status 200 OK

Hay respuesta positiva del servicio

**Paso 5.**

Detenemos el contenedor:

Docker container stop [id container]

**Paso 6.**

Añadir un Tag a la imagen creada:

docker image tag ms-checkin:1.0 jsimons182/ms-checkin:1.0

**Paso 7.**

Publicamos nuestra imagen al hub de Docker:

docker image push jsimons182/ms-checkin:1.0

una vez subidos los archivos de la imagen nos mostrara el id de nuestra imagen subida al hub.

