

	Ciclo de grado superior: Desarrollo de aplicaciones multimedia			Curso: 2023-24
	Módulo: Entornos de Desarrollo (ED)			Calificación
	Grupo: 1º DAM	Fecha: 14/03/2024	Tiempo: 3h.	
	Evaluación: Segunda Evaluación			
Nombre y Apellidos:				

Enunciado

La empresa Ryanair nos ha contratado para realizar la aplicación su negocio. Ryanair se dedica a la venta de billetes de avión. En las reuniones mantenidas con ellos hemos obtenido la siguiente información:

- ☐ Para poder sacar a la venta un vuelo, antes hay que darlo de alta en el sistema. Un vuelo lleva asociada la siguiente información: código de vuelo, avión que realiza el vuelo, tripulantes que van en ese vuelo (piloto, azafatos, azafatas, etc.), pasajeros, fecha, hora de vuelo, precio, origen y destino.
- ☐ La empresa desea poder llevar un control de los empleados de su empresa. Para ello nos pide que la aplicación pueda añadir empleados y recuperar empleados por cargo. De un empleado se almacena el dni, nombre y apellidos, cargo (piloto o azafata).
- ☐ La empresa tiene diez aviones que usa para los vuelos. De los aviones se almacena: código de avión, nombre, modelo y asientos.
- ☐ Cuando un usuario compra un billete para un vuelo, esto es lo que se conoce como pasajero. Del pasajero se desea almacenar: dni, nombre y apellidos, localidad y provincia.

Ejercicio 1: Modelado y Arquitectura

Apartado A (5 puntos):

Modelar las entidades y crear los casos de usos **necesarios** para realizar el programa. Se pide:

- ☐ 1. Crear los empleados (tripulantes):
 - Dni: 700001, Nombre: Empleado1 Apellidos: EmpleadoApellido1 Cargo: Piloto
 - Dni: 700002, Nombre: Empleado2 Apellidos: EmpleadoApellido2 Cargo: Piloto
 - Dni: 700003, Nombre: Empleado3 Apellidos: EmpleadoApellido3 Cargo: Azafata
- ☐ 2. Crear los pasajeros:
 - Dni: 800001, Nombre: Pasajero1 Apellidos: PasajeroApellido1 Localidad: Ávila Provincia: Ávila
 - Dni: 800002, Nombre: Pasajero2 Apellidos: PasajeroApellido2 Localidad: Ávila Provincia: Ávila
- ☐ 3. Crear los aviones:
 - Código de Avión: 100001 Nombre: Pajarito Modelo Boing701 Asientos: 200
 - Código de Avión: 100002 Nombre: Nube Modelo Boing705 Asientos: 250
 - Código de Avión: 100003 Nombre: Rayo Modelo Boing710 Asientos: 100
- ☐ 4. Comprobar si un avión ya está asignado o no a un vuelo para no sobreasignar aviones.
- ☐ 5. Crear los siguientes vuelos:
 - Código: 1 Avión: 100001 (buscar este) Tripulantes: 700001, 700003 (buscar estos dos)
Pasajeros: 800001, 800002 (buscar estos dos) Fecha: 10/02/2024, Hora: 10:30h, Precio: 90e,
Origen: Madrid, Destino: Barcelona.
- ☐ 6. Obtener el siguiente vuelo: Código 1.

Las fuentes de datos que deben escogerse son:

- ☐ Empleados (Tripulantes): MemLocalDataSource
- ☐ Pasajeros: FileLocalDataSource
- ☐ Aviones: FileLocalDataSource
- ☐ Vuelos: FileLocalDataSource

Apartado B (5 puntos)

- ☐ En el apartado A se pide en el punto 1. Una funcionalidad que cree empleados. Los empleados se están almacenando en MemLocalDataSource. Haz que esta funcionalidad se pueda cambiar a FileLocalDataSource de una forma sencilla y óptima aplicando los principios SOLID (La O de Open-Closed).

Nota: Para aprobar la parte de modelado, arquitectura y patrones de diseño se debe tener todos los puntos del Apartado A correctamente. Esto implica hacer correctamente:

- ☐ Arquitectura separada por capas.
- ☐ Cada capa tendrá su responsabilidad.
- ☐ Uso del patrón Repository para la gestión de los datos.

Ejercicio 2: Test Unitarios (10 puntos)

Se pide realizar los siguientes test unitarios:

- ☐ 1. Test unitario del caso de uso que me permite validar si un avión ya está asignado a un vuelo o no. (5 puntos)
- ☐ 2. Test unitario del caso de uso que me permite obtener los vuelos (3 puntos).
- ☐ 3. ¿Qué son los test dobles? (2 puntos) (añade esta pregunta a la Pull Request).

Nota: Para aprobar esta parte se debe realizar correctamente el test 1. El test 2 y test 3 es para subir nota.

Ejercicio 3: Control de versiones (10 puntos)

En esta parte el alumno debe:

- ☐ 1. Debe realizar una de estas opciones:
 - Opción 1:
 - Crear un repositorio en modo privado con el nombre: exam-marzo-ryanair.
 - Clonar el repositorio <https://github.com/iesam-dam/ed-23-24-exam-marzo.git> cambiar la ruta al repositorio creado anteriormente.
 - Añadir como revisor al profesor (jmperezra).
 - Opción 2:
 - Hacer un fork del repositorio: <https://github.com/iesam-dam/ed-23-24-exam-marzo.git>
 - Añadir como revisor al profesor (jmperezra)
- ☐ 2. Por cada uno de los ejercicio se debe realizar una PullRequest usando la plantilla que viene ya en el repositorio del profesor. Es decir, dos PullRequest.
- ☐ 3. Bajar el código en un zip y subirlo al AulaVirtual junto con la url del repositorio.

Nota: Para obtener puntuación se deben realizar correctamente todos los puntos indicados.