

 	EXAMEN (OPCIÓN A) REC. SEGUNDA EVALUACIÓN (2020/21)	DWESE – 2.º DAW	
		12-mar-2021	Pág. 1 de 2

Apellidos y nombre: _____

Escribir en **Yii 2 framework** una aplicación web para la reserva de billetes de avión.

Los usuarios de la aplicación podrán seleccionar un vuelo de los disponibles (y que aún dispongan de plazas libres) para hacer su reserva.

Cuando un usuario entre en la aplicación con su nombre y contraseña, podrá consultar las reservas que ya tenga o crear reservas nuevas.

Se pide:

1. (1 punto) Crear la base de datos con las siguientes tablas:

a) usuarios (id, nombre, password)

El nombre identifica al usuario y debe ser único.

La contraseña debe estar cifrada mediante el algoritmo de cifrado estándar de Yii 2.

b) aeropuertos (id, codigo, denominacion)

El código es una cadena única de tres caracteres alfabéticos que identifica al aeropuerto.

c) vuelos (id, origen_id, destino_id, salida, llegada, plazas, precio)

El origen y el destino son los aeropuertos de origen y destino, y no deben coincidir.

La salida y la llegada son la fecha y hora de salida y la fecha y hora de llegada.

Las plazas son el número de plazas totales que tiene el vuelo (un entero no negativo).

El precio es el precio de cada reserva en Euros.

d) reservas (id, usuario_id, vuelo_id, asiento)

El asiento es el número de asiento ocupado en la reserva (un entero mayor que cero).

No puede haber dos reservas con el mismo número de asiento en el mismo vuelo.

Igualmente, un usuario no puede tener más de una reserva para el mismo vuelo.

2. (1,5 puntos) Crear los modelos correspondientes y todas las relaciones adecuadas entre ellos, implementando las restricciones indicadas en el punto anterior.

3. (2 puntos) En el modelo Reservas, se debe impedir que se puedan hacer reservas de un vuelo si éste no tiene plazas libres. *Indicación:* el número del asiento de la reserva no puede ser mayor que el número de plazas totales que tiene el vuelo.

4. (2 puntos) Cuando se entre en la aplicación, la página principal será `vuelos/index`, donde aparecerá la lista de vuelos que aún no han salido. Al lado de cada vuelo, aparecerá un botón «Reservar» si el usuario aún no tiene ninguna reserva en ese vuelo (o no está logueado) o «Anular» si ya la tiene. Indicar en la lista de vuelos el número de plazas libres que quedan en cada vuelo.

5. (2 puntos) Al pulsar «Reservar» se deberá ir a `vuelos/reservar/id`, siendo *id* el ID del vuelo. Allí se mostrarán los datos generales del vuelo. El usuario podrá elegir un número de asiento si está libre, el cual tiene que estar comprendido entre 1 y el número de plazas totales que tenga el vuelo. Para ello, podrá elegirlo de una lista desplegable donde sólo aparecerán los asientos que queden libres en ese vuelo. Al pulsar el botón «Confirmar reserva» se creará la reserva correspondiente y se volverá a `vuelos/index`. Sólo se podrá entrar en `vuelos/reservar` si el usuario está logueado (en caso contrario, se le redirigirá primero a `site/login` para que se loguee previamente).
6. (1,5 puntos) Al pulsar «Anular» se deberá anular la reserva correspondiente entrando en `vuelos/anular/id`, siendo *id* el ID de la reserva. Allí se visualizará el detalle de la reserva con todos sus datos y un botón «Confirmar anulación» que, al pulsarlo, anulará la reserva y volverá a `vuelos/index`. Sólo se podrá entrar en `vuelos/anular` si el usuario está logueado (en caso contrario, se le redirigirá primero a `site/login` para que se loguee previamente).

La evaluación de cada ejercicio se llevará a cabo siguiendo la siguiente rúbrica:

% del total	Evaluación
0 %	No entregado, apenas empezado o manifiestamente incorrecto.
25 %	Se esboza un mínimo planteamiento pero no se acerca a una solución funcional.
50 %	Hay una solución incompleta que no cumple todos los requisitos.
75 %	La solución es completa pero ineficiente, poco elegante o mal documentada.
100 %	Solución completa, eficiente, elegante y bien documentada.

Se tendrá en cuenta, especialmente:

- Aplicación coherente del patrón MVC.
- Uso de comentarios adecuados en el código.
- Documentación del mismo en formato PHPDoc.
- Aplicación de los estándares de codificación usados por el framework.
- Elección adecuada de las rutas de acceso a la funcionalidad de la aplicación.
- Formato apropiado de los datos.
- Uso de mecanismos de comunicación asíncrona (AJAX).