

Examen Noviembre Desarrollo en Servidor

NORMAS DE EXAMEN : Son causas de suspenso: Hablar con un compañero durante el examen .Usar cualquier tipo de dispositivo, conexión , configuración o usuario no autorizado

Una **aplicación web sobre análisis y gestión de una flota de vehículos** trabaja con el siguiente modelo de datos:

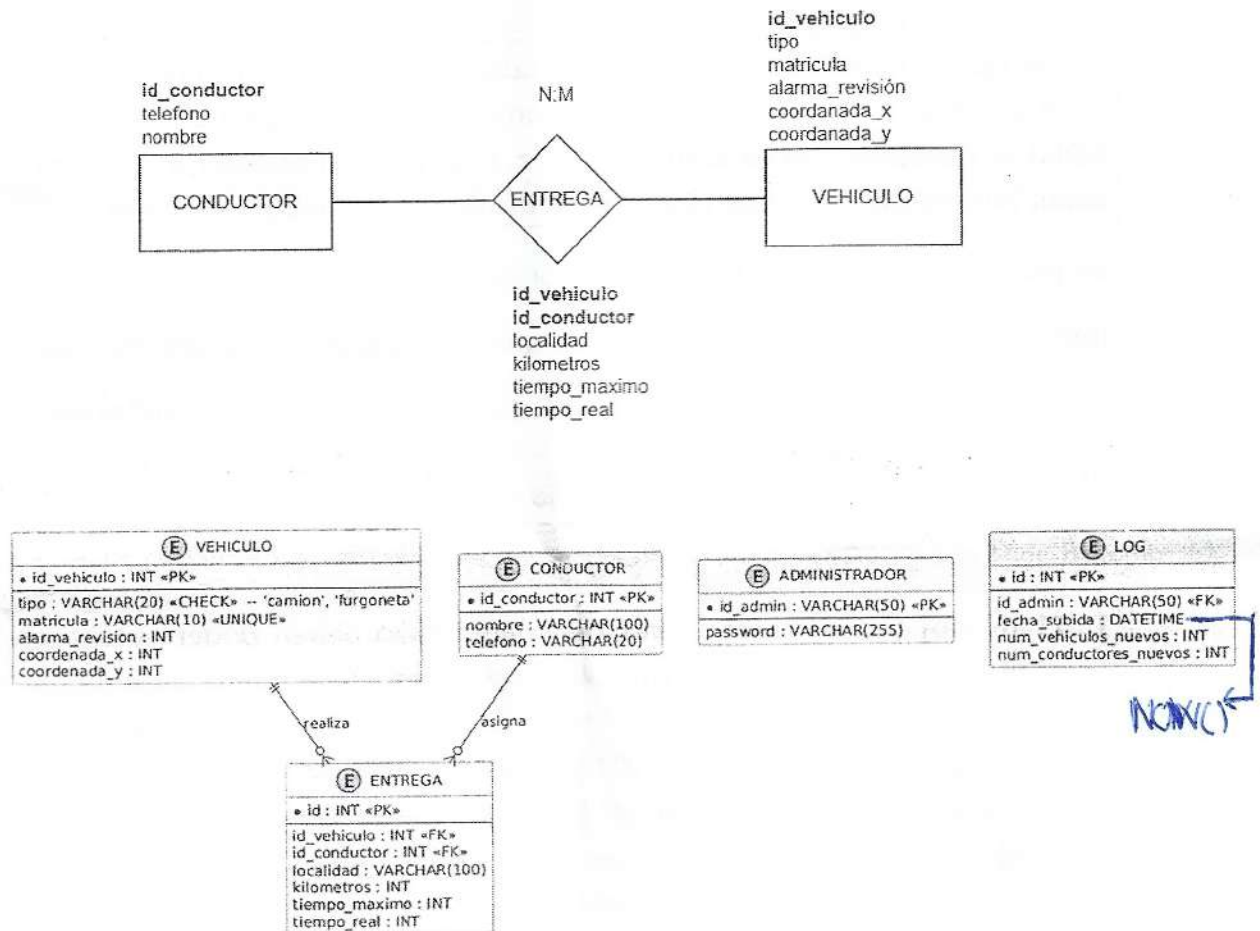


Figura 1 Modelo de datos ER de la FLOTA de vehículos

INTERFAZ GRAFICA DE USUARIO.

A continuación se muestra la interfaz de usuario que deberás implementar

strlen(—)

↳ para comprobar la longitud de una cadena

Sistema de Flota de Vehículos de Transporte

Iniciar sesion Administrador

- 1.- Ver Indicadores de Explotación de Flota
- 2.- Mostrar Posiciones de Vehículos en Ruta
- 3.- Cargar nuevos datos de Flota (ADMIN)
- 4.- Ver vehiculos para revision (ADMIN)
- 5.- Ver entregas pendientes (ADMIN)
- 5.- Retrasar entregas (ADMIN)

Cerrar sesion

NOTA: Las opciones 1,2 pueden ser accedidas sin iniciar sesión como admin. Las otras opciones si es necesario iniciar sesión como administrador previamente. **SOLO es necesario iniciar sesión una vez. Si se intenta acceder a 3,4,5 sin están autenticado se debe redirigir al formulario de inicio de sesión**

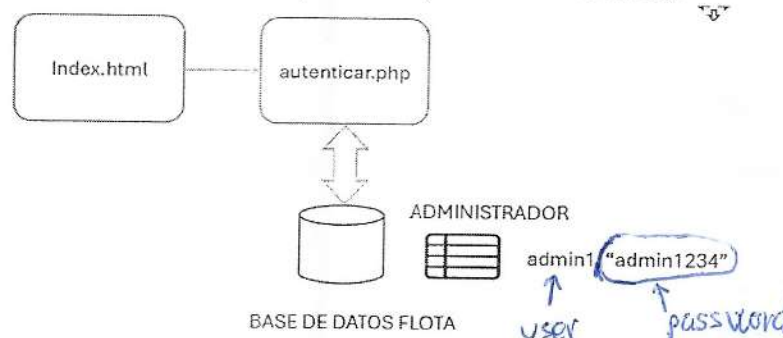
Se pide implementar la siguientes funciones:

IMPORTANTE Considera estos requisitos para la realización del examen

- Implementa una **clase VistaHTML** que centralice todas las salidas en el cliente mediante métodos
- Implementa una **clase Repositorio** para centralizar el acceso y operaciones con la base de datos

1. [1 punto] Los administradores de la aplicación deben poder autenticarse **para poder acceder a determinadas opciones**. Fíjate que la base de datos FLOTA tiene una tabla ADMINISTRADOR con los datos de los administradores existentes. **NOTA: Por motivos de seguridad solo es válido acceder desde la red del instituto, luego esto debe comprobarse en el controlador**

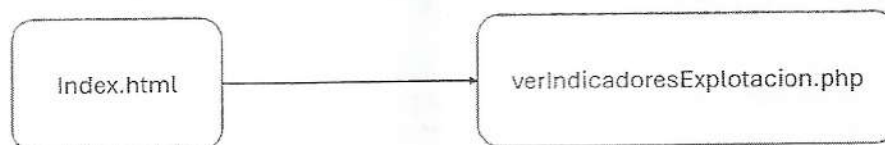
A partir de la autenticación, la sesión servirá para que el administrador pueda acceder a determinadas opciones (ver más adelante).



Módulo autenticar.php: Autentica a un usuario en la base de datos . Después de autenticar la aplicación devuelve al menú principal

NOTA : Ten en cuenta que la contraseña se guarda en la base de datos cifrada con HASH y que los datos de entrada deben ser validados para evitar la inyección, luego debes validar la entrada. Ver anexo

2. [1,5 puntos] Muestra los datos de explotación de la FLOTA.



Módulo **verIndicadoresExplotación.php** :Deberá mostrar la siguiente información

Indicador	Descripción
Disponibilidad de la flota. [0,5 puntos]	<p>-Número de vehículos no disponibles (están en una entrega pendiente) y los vehículos que están disponibles.</p> <p>NOTA: Un vehículo está en entrega si existe en la tabla entregas y su tiempo de finalización es nulo <i>tiempo real</i></p>
Estado de las Entregas: .[0,4 puntos]	<p>-Entregas en proceso/pendientes (Tiempo_real es NULL)</p> <p>-Tiempos de Entrega: Promedio de tiempo_real de todas las entregas completadas promedio de tiempo máximo que tenían asignadas</p> <p>-Entregas en tiempo . A tiempo si en promedio tiempo real es inferior al promedio del tiempo_máximo</p>
Datos de Rendimiento y Coste. [0,6 puntos]	<p>-Km estimados para mantenimiento. Kms que le queda a cada vehículo para su revisión que es a los 100 KM <i>1000KM</i></p> <p>-Ciudad con más entregas . La ciudad que tiene el mayor número de entregas realizadas</p> <p>-Coste de cada conductor. Por km se le paga 2 euros por entregas realizadas (las pendientes no computan)</p>

Salida esperada:

Indicadores de Explotación de la Flota

Disponibilidad de Flota

Vehículos disponibles: 2, No disponibles: 2

Estado de las Entregas

Entregas en proceso: 2

Promedio tiempo real: 125.0000 horas, Promedio tiempo máximo: ~~125~~ 6667 horas

TIEMPO DE LLEGADA en promedio inferior al previsto

180

Datos derendimiento y coste

datos de mantenimiento:

vehiculo ABC1234 , recorridos : 300, quedan para mantenimiento :700

vehiculo DEF5678 , recorridos : 500, quedan para mantenimiento :500

vehiculo GHI9012 , recorridos : 1100, quedan para mantenimiento :-100

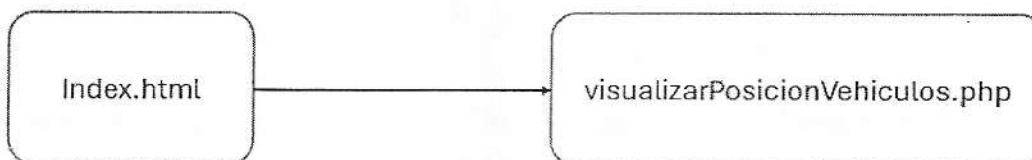
vehiculo JKL3456 , recorridos : 400, quedan para mantenimiento :600

Localidad con más entregas: Valencia

Coste de cada conductor

Nombre	Coste(euros)
Carlos Prez	600
Ana Garca	1000
Juan Lpez	2200
Mara Fernndez	800

3. [1 punto] Permite ver una matriz (10 x 10) la posición de cada vehículo de la FLOTA **que están en ruta (con entrega en curso)** usando sus coordenadas. **Cada coche se debe visualizar con su matrícula.**



Salida esperada (la matriz será de 10 x 10 simulando 100 m2)

Posición de Vehículos en Ruta

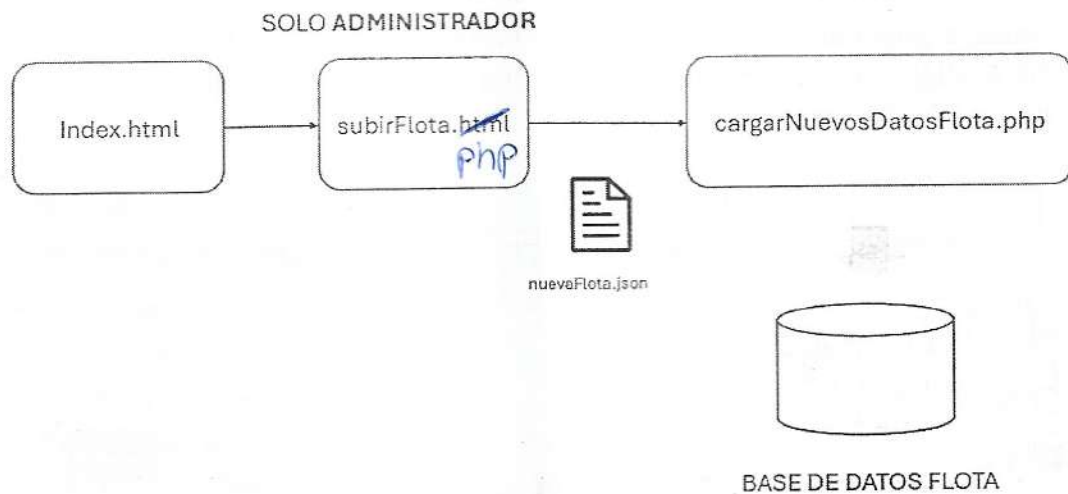
--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--

Nota: Solo se deben mostrar lo vehículos cuyas coordenadas tengan valor distinto de 0

→ crear archivo PHP

3.1 [0.5 puntos] Haz que pulsando sobre la matricula la aplicación lleve a una pantalla de visualización de todos los datos del vehículo :
matricula, tipo, alarma , coordenada X,Y

4. [1,25 punto] El administrador del sistema necesita **cargar nuevos vehículos y conductores regularmente** mediante un archivo JSON desde la parte cliente. El esquema de módulos necesarios se muestra a continuación:



- **subirFlota.html:** Es una interfaz HTML para subir archivo un archivo JSON
- **cargarNuevosDatosFlota.php:** Módulo que procesa el archivo JSON y los guarda los nuevos datos en la base de datos.

El archivo JSON para subir nuevos datos de flota tiene la siguiente estructura:

```

{
  "vehiculos": [
    { "id": 60, "matricula": "3284ABCD", "tipo": "Camion"},
    { "id": 61, "matricula": "3779DEFD", "tipo": "Furgoneta"},
    { "id": 62, "matricula": "A47788KK", "tipo": "Coche"}
  ],
  "conductores": [
    { "id": 62, "nombre": "Juan Pérez", "telefono": "600123456"},
    { "id": 63, "nombre": "Ana Gómez", "telefono": "600654321"} ]
}
  
```

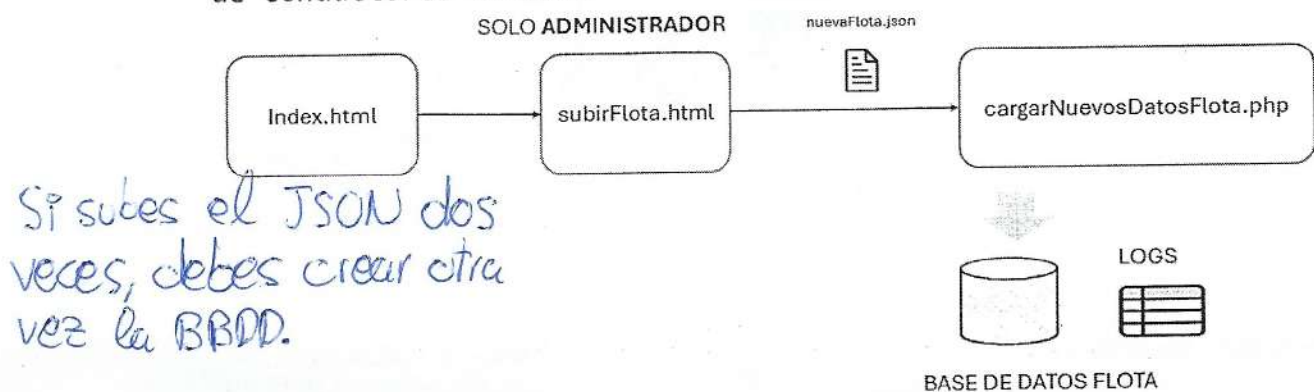
NOTA IMPORTANTE: Los datos que faltan en el archivo JSON **deben ser inicializados a valores por defecto en el controlador de proceso** y son :
Coordenadas X,Y a valor 0 y alarma a valor 0.

Los datos que no sean válidos deben ser descartados por el controlador: No son válidos los vehículos con una matrícula que no sean 8 caracteres o que el tipo no sea camión o furgoneta

5. [1,25 puntos]. Todo lo que hace el administrador debe quedar registrado en la tabla LOG. Guarda los siguientes datos en la base de datos cada vez que un administrador suba un nuevo archivo de flota.

Datos a guardar

ID_ADMIN, fecha de subida, Numero de vehículos nuevos, Número de conductores nuevos.



- 5.1 [0.25 puntos] Además, debes guardar el archivo subido en la carpeta uploads del SERVIDOR con el siguiente formato día_mes_nuevaflota.json. date('d') y date('m') devuelven el día y la fecha
Ejemplo 1_12_nuevaflota.json

6. [1 punto] Otra de las opciones que **SOLO puede hacer el administrador autenticado** es ver qué vehículos tienen que pasar la revisión. Muestra qué vehículos tienen que pasar la próxima revisión. Los criterios para pasar la revisión son :

- ~~Que el vehículo tenga una alarma pendiente o~~
- Que el vehículo haya hecho más de 1000 kilómetros

Salida esperada

Pasar la revisión vehículo :GHI9012

7. [1,25 puntos] **Consulta de entregas pendientes.** Desarrolla la funcionalidad de poder ver las entregas completadas y las pendientes (o en

curso). La búsqueda debe tener la posibilidad de filtrar por conductor o vehículo. **Valida la entrada para prevenir inyección de código. Utiliza COOKIES para recordar el último valor introducido y que lo autocomplete. Si se meten los dos valores la consulta la hará por Conductor**

Ejemplo de interacción

Consulta de Entregas Pendientes

Conductor (ID): o Vehículo (ID):

Salida esperada :

ID	Matrícula	Conductor	Localidad	Kilómetros	Tiempo Máximo
6	JKL3456	Mara Fernández	Sevilla	400	240 min

[Volver a búsqueda](#)

Pulsando la opción Volver a búsqueda

Consulta de Entregas Pendientes

Conductor (ID): o Vehículo (ID):

COOKIE recordando el valor introducido

Otro ejemplo (consulta por el conductor)

Consulta de Entregas Pendientes

Conductor (ID): o Vehículo (ID):

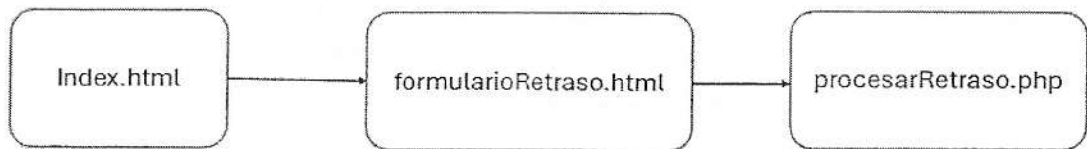
Salida esperada :

ID	Matrícula	Conductor	Localidad	Kilómetros	Tiempo Máximo
2	DEF5678	Ana García	Barcelona	500	300 min

[Volver a búsqueda](#)

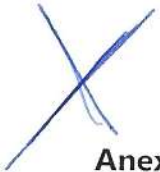
8. [1 punto] Implementa una nueva funcionalidad que permita incrementar en un 20% ($\text{tiempo_maximo} * 1.2$) los tiempos máximos de todas la entregas que están en curso para una localidad determinada. El administrador introduce la localidad por la interfaz que sabe que van a subir retraso.

SOLO ADMINISTRADOR



Actualizar entregas

localidad:



Anexo

password_verify()

```
<?php
// See the password_hash() example to see where this came from.
$hash = '$2y$10$.vGA1O9wmRjrwAVXD98HNOgsNpDczlqm3Jq7KnEd1rVAGv3Fykk1a';

if (password_verify('rasmuslerdorf', $hash)) {
    echo 'Password is valid!';
} else {
    echo 'Invalid password.';
}
?>
```