

Practica 1 - Taller de Coches en GO

Sistemas Distribuidos - GIT - URJC

Enunciado

Desarrolle un sistema en **GO** para gestionar un taller mecánico de coches que cuenta con las siguientes características:

Requisitos:

1. El taller dispone de 2 plazas para reparación de coches por cada mecánico.
2. Para cada cliente se debe almacenar:
 - ID del cliente
 - Nombre
 - Teléfono
 - Email
 - Vehículo asociado
3. Para cada vehículo se debe registrar:
 - Matrícula
 - Marca
 - Modelo
 - Fecha de entrada
 - Fecha estimada de salida
 - Incidencia detectada
4. Para cada incidencia/reparación se debe registrar:
 - ID de la incidencia
 - Mecánicos asignados, pueden ser varios
 - Tipo de incidencia (mecánica, eléctrica, carrocería)
 - Prioridad (baja, media, alta)
 - Descripción de la incidencia
 - Estado (abierta, en proceso, cerrada)
5. Para cada mecánico se debe almacenar:
 - ID del mecánico
 - Nombre

- Especialidad (mecánica, eléctrica, carrocería)
- Años de experiencia
- Si está activo o de baja

6. El sistema debe permitir realizar (al menos) las siguientes operaciones:

- Crear/Visualizar/Modificar/Eliminar clientes
- Crear/Visualizar/Modificar/Eliminar vehículos
- Crear/Visualizar/Modificar/Eliminar incidencias
- Crear/Visualizar/Modificar/Eliminar mecánicos
- Asignar vehículos a plazas del taller (respetando la limitación de plazas)
- Visualizar estado actual del taller (plazas ocupadas/libres)
- Dar de alta/baja a un mecánico
- Cambiar el estado de una incidencia
- Listar todas las incidencias de un vehículo
- Listar todos los vehículos de un cliente
- Listar todos los mecánicos disponibles (no asignados a ninguna incidencia)
- Listar todas las incidencias asignadas a un mecánico
- Listar todos los clientes con vehículos en el taller
- Listar todas las incidencias en el taller y su respectivo estado
- Menú para seleccionar las diferentes opciones, puede haber submenús si lo considera necesario

Consideraciones

- Solo se debe usar las estructuras de código vistas hasta el momento (14/10/2025)
- Se debe realizar en un único fichero `.go`
- No es necesario implementar persistencia de datos (base de datos o ficheros)
- Incluya validaciones para evitar asignar elementos de forma errónea (por ejemplo, no asignar más vehículos que plazas disponibles)

Entregables

Un documento en formato **PDF** que incluya:

1. Explicación del diseño del sistema (estructuras de datos, funciones principales, funcionamiento de la aplicación, etc.) haciendo uso explícito de diagramas de UML. (40%)
2. Código fuente del sistema en **GO**, debe estar en un repositorio de github y debe incluir un enlace al mismo en el documento. (20%)

3. Link al vídeo explicativo del funcionamiento del sistema, disponible en cualquier sitio de alojamiento de vídeos no necesariamente de forma pública (Youtube, Vimeo, etc.) siempre debe estar disponible para consulta del profesor, el vídeo debe visualizar la ejecución de todas las operaciones solicitadas en el **requisito 6**, **no serán tenidos en cuenta los vídeos grabados con una cámara externa** y su duración máxima es de **5 minutos**. (40%)

Indicaciones adicionales

Guarda el archivo con un nombre como `Practica_1_Nombre_SS00_dist.pdf` . Donde `Nombre` ha de ser tu nombre de usuario.

Luego dicho documento se entrega en la actividad en aula virtual.