

Apellido y nombre: _____ Legajo: _____

EXAMEN FINAL

14/12/2021

ENUNCIADO

Un taller que se dedica a realizar servicios de reparación y mantenimiento de automóviles te encomienda un pequeño prototipo de app web con persistencia a través de una base de datos no relacional.

La información que interesa mantener registrada es la siguiente:

- **Clientes:** Tendrán un ID único, además del DNI, nombre, apellido, mail y teléfono móvil.
- **Automóviles:** Tendrán un ID único, además de la patente, la marca, el modelo y el año de fabricación y a qué dueño pertenece.
- **Servicios:** Tendrán un ID único, la fecha de realización, el costo y la información del automóvil al que se le realizó, junto a la cantidad de kms que tenía hasta el momento.

Se desea una aplicación web de una sola página (SPA) capaz de manejar las altas, bajas, modificaciones y consultas de cada una de las entidades involucradas. Además, debe ser posible visualizar un listado con todos los automóviles de más de 3 años de antigüedad que posean UN solo servicio realizado.

FORMATO DE ENTREGA

Archivo **.zip** o **.rar** nombrado de la siguiente manera (reemplazando lo que está en verde):

INSPT-ProgIII-3601-final-2021-service-automotor-TUAPPELLIDO-TUNOMBRE

El **.zip** o **.rar** debe contener:

- El código fuente del proyecto de lado servidor junto a los archivos necesarios para regenerar localmente la base de datos (salvo que se haya usado un servicio de base de datos en la nube).
- El código fuente del proyecto del lado cliente junto a los recursos utilizados.

En ningún caso incluir archivos y carpetas que puedan regenerarse mediante scripts (por ejemplo, la carpeta **node_modules**).

No cumplir con alguna indicación del formato de entrega disminuye la nota del examen.

CRITERIOS DE EVALUACIÓN

Para considerar aprobado el examen, se deben aplicar correctamente los siguientes conceptos vistos durante la materia:

- Inicialización de un proyecto de backend y el manejo automatizado de las dependencias (por ejemplo, Node y NPM).
- Conceptos del lado del servidor (enrutamiento, request-response, errores, autenticación, seguridad, REST, etc).
- Interacción y modelado utilizando base de datos no relacional (por ejemplo, MongoDB).

- Diseño y funcionalidad de una aplicación del lado cliente.
- Uso de librerías/frameworks para desarrollar la interfaz de usuario (por ejemplo, React).
- Lenguaje JavaScript bajo el estándar ECMAScript 6 y posteriores.
- Algoritmos regidos bajo el paradigma funcional, usando funciones de orden superior por sobre bucles, además de callbacks y funciones puras.
- Modularización reutilizable y mantenible en carpetas nombradas siguiendo la convención.
- Autonomía y autodeterminación para poder relacionar conceptos y descubrir nuevos mediante la lectura de la documentación de las tecnologías introducidas u otras necesarias.
- Buenas prácticas de programación:
 - Uso de constantes en lugar de números mágicos.
 - Nombres de variables, métodos y clases descriptivos, que sigan la convención del lenguaje.
 - Uso adecuado de los ciclos. No alterar la naturaleza de los mismos utilizando instrucciones de control como **break** y **continue** (puede desacreditar el examen).
 - Código ordenado e indentado correctamente.

% Correcto	0 a 20	25 a 45	50 a 55	60	65 a 70	75	80	85 a 90	95	100
Nota	1	2	¿4?	4	5	6	7	8	9	10