

## NORMAS PARA LA REALIZACIÓN DEL EJERCICIO PRÁCTICO

- 1) La realización del trabajo práctico es de carácter **OBLIGATORIO**, y constituye el mecanismo de evaluación de la asignatura.
- 2) Se debe elaborar una breve memoria que se entregará junto con todos los ficheros de código fuente y todos los ficheros necesarios para la generación y ejecución del ejecutable. Todos los archivos se empaquetarán en un único fichero que se entregará mediante la tarea habilitada al efecto en la página de la asignatura en la plataforma Studium (<http://studium.usal.es/>).
- 3) La memoria se entregará en formato electrónico (fichero PDF) como se indica en el punto anterior. **Obligatoriamente**, para la aceptación de la práctica para su defensa, deberá contener **como mínimo** los siguientes puntos:
  - Un breve manual de usuario.
  - Un manual del programador. Partiendo de un diagrama de objetos<sup>1</sup> se deben explicar las principales clases creadas, así como los métodos más significativos de las mismas. También se deben explicar las relaciones entre los objetos, por ejemplo, la utilización de delegaciones y eventos para comunicar los diferentes elementos visuales de la aplicación.
  - **Referencia de todas las fuentes utilizadas:** libros, código de ejemplos, URLs (las URL deberán ser concretas, **no se aceptan referencias genéricas** como <https://stackoverflow.com>, <https://www.youtube.com>, o <https://docs.microsoft.com>), ...

Se podrá añadir cualquier otro tipo de documentación que los alumnos consideren relevante. A modo de orientación, la memoria podrá tener entre 4 y 15 páginas.

- 4) Una vez entregados los archivos de la práctica, se contactará mediante correo electrónico con el profesor/a para concertar la fecha y hora de la defensa.
  - Ana de Luis Reboredo (Grupo A): [adeluis@usal.es](mailto:adeluis@usal.es)
  - Iván Álvarez Navia (Grupo B): [inavia@usal.es](mailto:inavia@usal.es)

En dicho correo el **alumno indicará claramente su identidad** y, por orden de preferencia, **varias opciones de fecha y hora de defensa**. Posteriormente, el profesor le citará, también mediante correo electrónico, para la defensa procurando atender, en la medida de lo posible, las preferencias del alumno.

**No se considerará entregada una práctica hasta que se haya recibido el correo del alumno indicando sus preferencias de defensa.** Por tanto, el envío de este correo está sometido a los mismos plazos que la entrega de la práctica.

- 5) **Dependiendo de la originalidad, complejidad y calidad de la práctica entregada, así como de las circunstancias sanitarias, el profesor podrá determinar una calificación sin necesidad de su defensa.** En este caso, se notificará la propuesta de calificación. Si el estudiante no estuviera de acuerdo con la misma, se realizará la defensa en fecha que acuerde con el profesor.
- 6) Cada alumno deberá defender individualmente su trabajo, salvo los casos mencionados en el punto 5. La defensa se realizará en la fecha y hora que previamente acordadas. **Durante la defensa del trabajo, los profesores podrán**

---

<sup>1</sup> Para la generación del diagrama de objetos se puede hacer uso de la opción Ver/Ver diagrama de clases, disponible en el menú emergente que Visual Studio muestra sobre los nombres de los proyectos en el explorador de soluciones.

**plantear cuestiones relativas al trabajo realizado, así como relativas a la implementación de nuevos supuestos prácticos que el estudiante deberá resolver en ese mismo acto de defensa.** A estos efectos, debe considerarse que el acto de defensa de la práctica es equivalente a la realización del examen de la asignatura.

- 7) Los plazos de entrega y defensa de la práctica para ambas convocatorias son los siguientes:

FECHAS FINALES DE DEFENSA Y ENTREGA		
	Entrega de práctica y solicitud de fecha de defensa	Defensa de Práctica
1ª Convocatoria	hasta las 14:00 del <b>12 de Diciembre</b>	hasta el <b>16 de Enero</b>
2ª Convocatoria	hasta las 23:59 del <b>10 de Febrero</b>	hasta el <b>24 de Febrero</b>

- 8) Las fechas que se indican en todos los plazos son fechas límite, es decir, el alumno siempre puede realizar la entrega y defensa en fechas anteriores. De hecho, se recomienda encarecidamente que las prácticas se entreguen y defiendan lo antes posible y de forma escalonada.
- 9) Obviamente, **si se detecta copia entre dos o más prácticas, todas ellas serán irremediabilmente calificadas con suspenso**, con independencia del papel desempeñado por los distintos implicados.

## ENUNCIADO GENERAL

Se desarrollará una aplicación en entorno WPF Microsoft Windows en C# que permita registrar información sobre los costes derivados del mantenimiento una flotilla de automóviles, así como de mostrar distintas gráficas comparativas. Se exigen, como costes mínimos a gestionar, los derivados del repostaje de combustible de los vehículos. El estudiante puede añadir otros costes como, por ejemplo, revisiones generales, cambios de aceite, cambio de ruedas, reparaciones, etc.

La aplicación debe permitir recoger para cada vehículo como mínimo su matrícula, y marca, así como la información relacionada con cada uno de los repostajes de combustible, que debe incluir como mínimo la fecha del repostaje, el número de km del cuentakilómetros del coche en el momento de repostar, el número de litros de combustible adquirido y el coste de estos.

### Requisitos indispensables de la aplicación

1. La aplicación mostrará obligatoriamente al menos dos ventanas:
  - La ventana principal, que obligatoriamente incluirá un área donde se representarán los gráficos (Fig 1)

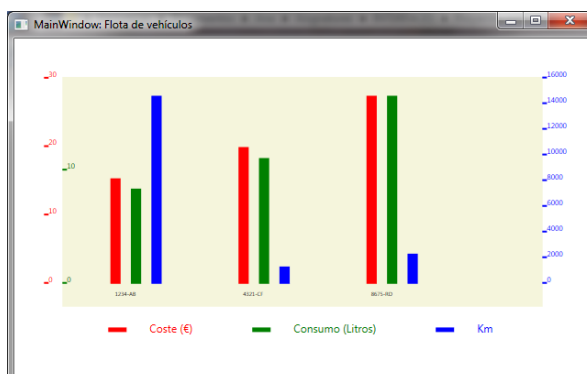
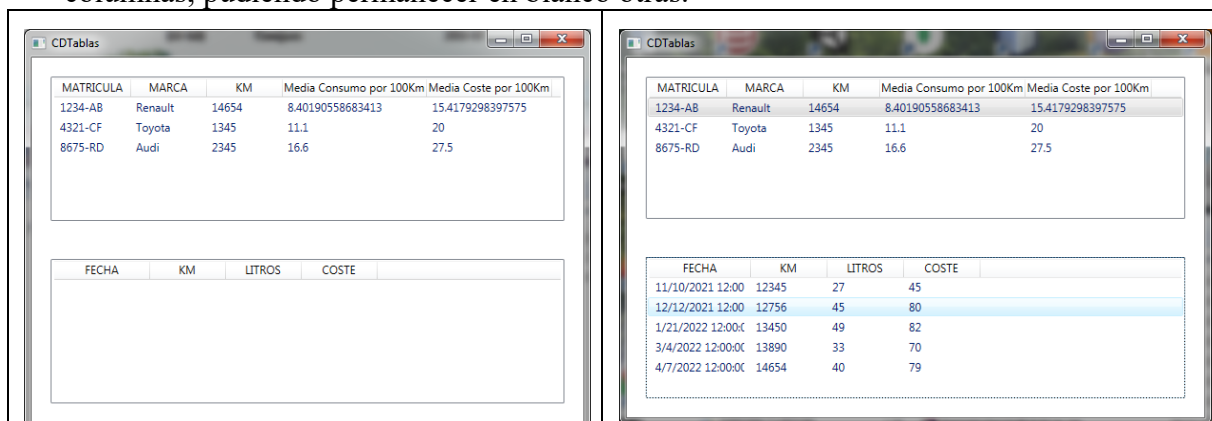


Figura 1

- Una ventana secundaria (o cuadro de diálogo en modo NO MODAL) en donde se mostrarán dos tablas: la primera contendrá el listado de vehículos de la flotta, incluyendo el número actual de km de cada uno, así como su consumo medio y su coste medio por cada 100 km; la segunda aparecerá inicialmente vacía y, cada vez que se seleccione un coche en la primera tabla, la segunda mostrará el detalle de cada uno de los repostajes realizados por el vehículo y de los otros conceptos de gastos en caso de haberse incluido conceptos adicionales. (Figuras 2 y 3). Cada ítem de esta tabla tendrá información, al menos, en tres columnas: la fecha, el kilometraje y el coste. En el caso de los repostajes, se mostrará en otra columna la cantidad de litros adquiridos. Si se han añadido otros costes de mantenimiento, como los indicados en el enunciado general, se añadirán las columnas correspondientes que visualicen los datos de estos conceptos. Esto implicará que podrá haber ítems que tendrán datos en unas columnas, pudiendo permanecer en blanco otras.



Figuras 2 y 3

2. Se deberán representar como mínimo dos tipos de gráficos en la ventana principal:

- Un gráfico con los datos cada uno de los coches. Se representarán al menos tres columnas por cada vehículo: una para representar los kilómetros actuales del vehículo; otra para la media de consumo del vehículo por cada 100 kilómetros; otra para la media de costes del vehículo por cada 100 kilómetros. Este es el gráfico que se mostrará por defecto en la ventana principal (Figura 1). Se podrá completar la información de este gráfico con cualquier otro dato del vehículo que se considere de interés.
- Cuando en la primera tabla de la ventana secundaria haya un coche seleccionado, se mostrará en la ventana principal el segundo tipo de gráfico que

datallará los datos correspondientes a todos los costes asociados al vehículo seleccionado (Figura 4). Nuevamente, como mínimo se mostrarán tres gráficas lineales, correspondientes a los valores de los kilómetros recorridos por el vehículo desde el repostaje anterior, el número de litros repostados y el coste de estos. Si se han añadido otros costes de mantenimiento, se podrán añadir otras gráficas de cualquier tipo que representen cada uno. Cuando se esté mostrando este tipo de gráfico, deberá existir un mecanismo que permita volver a representar el gráfico con los datos globales.

3. Como ya se ha indicado en los puntos anteriores, se podrá añadir cualquier otro tipo de información y representación gráfica que se considere de interés para gestionar y controlar los costes de la flota de vehículos, como, por ejemplo, revisiones, reparaciones, cambio de ruedas, ... Aunque se ha indicado que estos costes adicionales se pueden incluir, tanto en las tablas como en las gráficas definidas como mínimo obligatorio, si el estudiante lo considera oportuno, puede añadir tablas y gráficas adicionales y separadas de las obligatorias. Por ejemplo, puede mantener una tabla separada para las revisiones obligatorias del vehículo, con los apartados de información que se consideren oportunos, y una gráfica, también separada de la obligatoria, para la representación de estos costes. Cuando el usuario seleccione un vehículo en concreto en la primera tabla, se mostrarán dos tablas: la obligatoria con los datos de repostaje y una segunda con los costes de las revisiones (fecha, coste y otros apartados de detalle que se consideren). De la misma manera, en la gráfica se puede representar todos los costes mediante gráficas lineales en una única gráfica, o se pueden detallar los costes añadidos en otro tipo de gráficas.

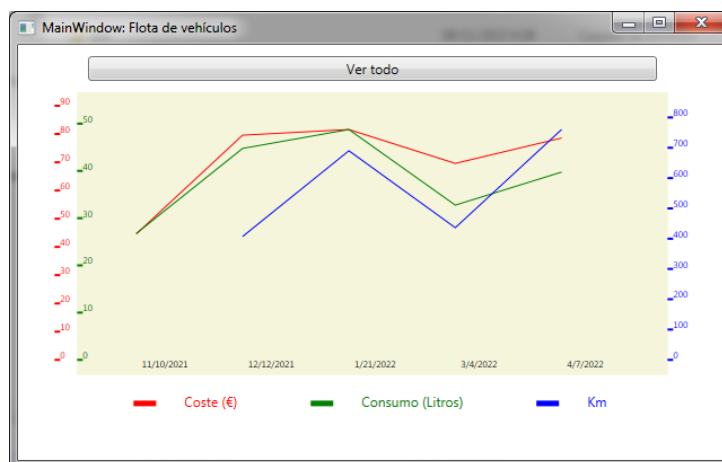


Figura 4

4. El código para la representación de los gráficos de los dos tipos obligatorios descritos en el punto anterior estará escrito sin hacer uso de bibliotecas externas de generación de gráficos.

5. La aplicación deberá ofrecer una interfaz correcta para la introducción de todos los datos de vehículos y repostajes.

6. La aplicación asegurará una correcta gestión de las aperturas y cierres de las ventanas, garantizando que, si es posible cerrar una ventana secundaria sin que la aplicación finalice, existirá una opción para reabrirla sin pérdida de los datos previamente introducidos.

Estos son los requisitos mínimos que debe cumplir la aplicación desarrollada, y constituyen el mínimo para conseguir la calificación de aprobado en la misma. Por

supuesto, el alumno puede incorporar cuantas mejoras desee, lo que repercutirá positivamente en la calificación obtenida. Se sugiere que dichas mejoras estén relacionadas con la utilización de aspectos de interfaz no estudiados en clase.

## VALORACIÓN DE LAS PRÁCTICAS

- El enunciado propuesto es bastante concreto, en cuanto a las ventanas a mostrar, pero se deja abierto el detalle, de manera que cada alumno pueda completar la especificación según su propio criterio.
- Se valorará tanto la calidad de la aplicación desarrollada como el grado de conocimiento del alumno sobre la misma.
- Serán requisitos mínimos para aprobar:
  - la correcta compilación y generación del ejecutable, así como su correcto funcionamiento
  - el cumplimiento estricto de todos los requisitos obligatorios
  - la entrega de la memoria con todos los apartados descritos en el punto 3 de las normas, así como de todo el código fuente y, en su caso, el ejecutable
- Se valorarán positivamente (según el caso), es decir, servirán para subir nota:
  - la incorporación de especificaciones a la aplicación que impliquen un mayor grado de dificultad en lo que se refiere a su interfaz
  - la utilización de varios tipos de controles, incluyendo los no utilizados en clase
  - la extensión de requisitos propuestos, en general, el uso de cualquier característica que haya implicado la investigación personal del alumno
- Serán motivo de **suspenso**:
  - el **incumplimiento** de las especificaciones del enunciado, en especial en lo referente a la estructura de la aplicación y de las ventanas mínimas que se deben presentar
  - la detección de **errores graves de concepto** o de mala utilización de los procedimientos establecidos
  - la comprobación de **desconocimiento** del funcionamiento o implementación de la aplicación
  - la comprobación de **plagio** completo, o bien parcial en un porcentaje importante, de otra práctica entregada en esta, u otra convocatoria