

UT1.-Repaso de POO e introducción al C#

Realizar la siguiente práctica aplicando los conceptos vistos en la **UT1.- Repaso de POO e introducción al C#**.

- Prestar atención en:
 - Denominación de clases, variables y métodos.
 - Declaración de variables.
 - Paso de parámetros a los métodos.
 - Comentarios.
 - Orden en el proyecto

Trabajamos en una empresa de desarrollo de software y nos piden crear un software para gestionar las interfaces en:

- Aplicaciones de Gestión, para el departamento de Administración.
- Aplicaciones de VR, para el departamento de I+D.
- Aplicaciones de 3D, para el departamento de Infoarquitectura.

Nos piden que cada uno de los puntos anteriores cumplan los siguientes requisitos:

- Para cada tipo de software, se debe de crear una clase. Las clases serán las siguientes:
 - ApGestion
 - ApVirtualReality
 - ApInfoarchitecture
- Estas tres clases, van a heredar de una clase padre denominada:
 - DesarrolloInterfaces

Las tres clases hijas ApGestion, ApVirtualReality y ApInfoarchitecture, solo pueden contener:

- Una variable **pública de tipo bool** (booleana), que se denominará: **activarGestion**, **activarVirtualReality** y **activarInfoarchitecture** respectivamente.
- Una **variable privada de tipo texto (string)**, que contendrá el mensaje a visualizar. Estas variables se denominarán **menGestion**, **menVirtualReality** y **menInfoarchitecture** respectivamente.

UT1.-Repaso de POO e introducción al C#

- Los mensajes que contendrán cada una de las variables anteriores, serán:
 - menGestion: **La herramienta de Gestión, está activa.**
 - menVirtualReality: **La herramienta de VR, está activa.**
 - menInfoarquitectura: **La herramienta de Infoarquitectura, está activa.**
- Todas ellas dispondrán de su método **Update()**.

La clase padre contendrá un **único método** que se denominará:

- VisualizarMensaje().

Este método debe de recibir como **parámetro de entrada** la **variable privada** de la clase hija activa para que se visualice en la ventana “Console” de Unity.

Cuando las variables de tipo booleano **se activen**, desde la ventana “Inspector”, se debe **visualizar el mensaje** en la ventana “Console”.



El material de apoyo con el que se cuenta es:

- **DI_UT1_Practica.unitypackage** del proyecto iniciado para que el alumno lo termine. Los paquetes de Unity, los estudiaremos más adelante. **Se trata de un proyecto comprimido que podemos cargar y seguir trabajando sobre él. Para ver como realizar esta operación ver el vídeo: UT1 CargarPaqueteUnity.**
- Anexo a este documento, el alumno dispone de una rúbrica con los puntos que serán evaluados en esta práctica.



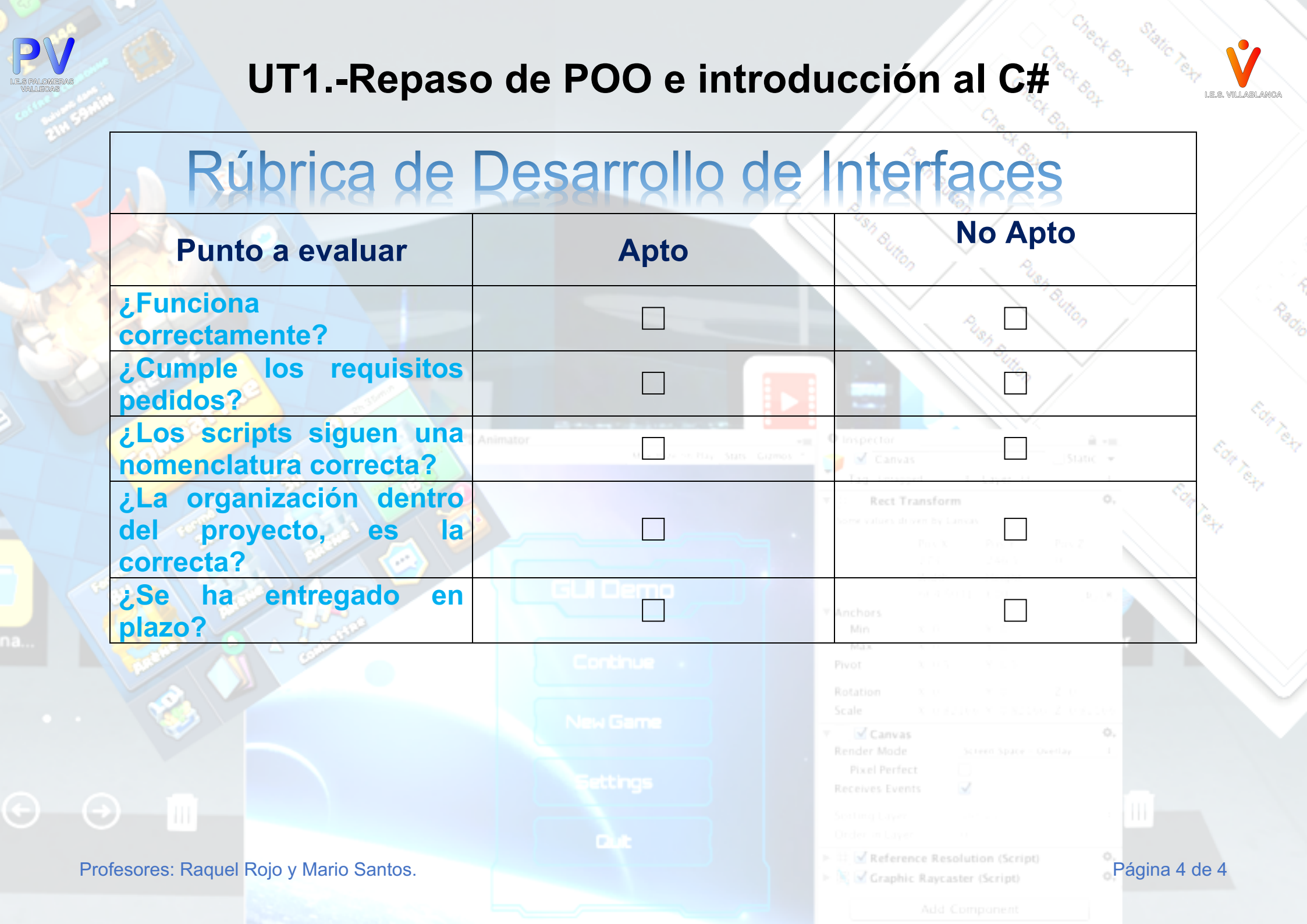
Ayuda

- Buscar en la documentación de Unity como funciona la orden **Debug**
- Buscar en la documentación de Unity como funciona el método **Update()**.



La solución a la práctica la tendréis disponible en
"UT1_Resolucion_Práctica.mp4"





Rúbrica de Desarrollo de Interfaces

Punto a evaluar	Apto	No Apto
¿Funciona correctamente?	<input type="checkbox"/>	<input type="checkbox"/>
¿Cumple los requisitos pedidos?	<input type="checkbox"/>	<input type="checkbox"/>
¿Los scripts siguen una nomenclatura correcta?	<input type="checkbox"/>	<input type="checkbox"/>
¿La organización dentro del proyecto, es la correcta?	<input type="checkbox"/>	<input type="checkbox"/>
¿Se ha entregado en plazo?	<input type="checkbox"/>	<input type="checkbox"/>