

Caso de Estudio 1

MVC - API - DesignPatterns - SOLID

Trabajo Grupal Fecha de Entrega 4 de Julio

Descripcion:

Utilizando la aplicacion creada en clase, la aplicacion provista por el profesor o una aplicacion nueva creada por el alumno debera crear un proyecto de MVC que se conecte a algun API de noticias e.g. [NY Times](#) o bien puede utilizar cualquiera o varios de este enlace [Public API News](#). Debera leer las noticias haciendo una peticion de al API (probablemente necesitara crear un api-key, consultar al profesor por ayuda, si la necesita). Las noticias leidas debera devolverlas en un objeto/modelo para que sea enviado hasta la UI (MVC), y debe mostrar las noticias en vistas en un formato amigable.

Consideraciones (requisitos) del proyecto

- Debe hacer el llamado como minimo a un API (puede dar uso del RestProvider visto en clase)
- Debe utilizar los ActionResult's vistos en clase para devolver el contenido a la UI en todos los endpoints
- Debe crear 1 Solucion y 3 proyectos/librerias. (Puede reutilizar el proyecto creado en clase)
 - El Namespace debe llevar el "name convention" siguiente: CEPAW.{nombre_del_proyecto}
 - Proyecto de Web (ASP Net Core - MVC)
 - Proyecto de Servicios o Business
 - Proyecto de Modelos
- Debe utilizar al maximo interfaces y evitar instanciaciones o declaraciones directas de clases de otros proyectos
- Debe utilizar Dependency Injection (DI)
- Debe implementar los Principios de SOLID Programming (debe indicar en el codigo mediante un comentario // SOLID: <> a donde se implemente el principio)
- El "Look & Feel / Design" de la pagina queda a discrecion del grupo
- El proyecto debe utilizar al menos 3 Design Patterns (Investigacion) y aplicarlos de manera logica en el codigo, o sea, lo que implemento debe tener sentido y trabajar en armonia con el sitio de manera fluida (debe indicar en el codigo mediante un comentario // DESIGN_PATTERN: <> a donde se implemente el patron)
- Debe crear documentacion para presentarlo en la clase
- Debe implementar la funcionalidad y codigo necesario (botones, links, codigo, clases, etc) para lograr que la aplicacion logre el objetivo

Calificacion

Item	Pts
1. Llamado al menos a 1 API	10
2. Uso de ActionResult's	10
3. Arquitectura del proyecto: CEPAW.Web, CEPAW.Services, CEPAW.Models	12

4. Utiliza Interfaces	10
5. Evita Instanciaciones de clases	10
6. Utiliza Patron de DI	10
7. Implementa SOLID Principles	10
8. Implementa al menos 3 Design Patterns	10
9. Creacion de la UI de MVC	8
10. Presentacion (con material)	10
Total:	100

Entregables

1. Crear un PDF con las especificaciones del proyecto:

Item	Detalle
1.Nombre del API utilizado	
2. Cantidad de ActionResults	
3. Nombre de Proyectos creados	
4. Cantidad de Interfaces	
5. Clases o lugares donde evito hacer una dependencia directa	
6. Utiliza Patron de DI	
7. Clases donde implementa SOLID	
8. Ubicacion donde utiliza los: Design Patterns	
9. Imagen con el "Landing Page"	
10. Presentacion (con material) PDF / PPT / OTRA	

2. Presentacion en modo PDF/PPT/Lienzo(diagrams.net / draw.io) etc
3. El proyecto comprimido en formato *.zip