



Para garantizar que cualquier modificación en la propiedad `TipoLicencia` de la clase `Chofer` sea validada correctamente después de la inicialización del objeto, es esencial implementar mecanismos que aseguren la integridad de los datos. A continuación, se presentan dos enfoques recomendados para lograr esta validación:

1. Implementación de la interfaz `INotifyPropertyChanged`

Este enfoque permite que la clase `Chofer` notifique a los suscriptores cada vez que una propiedad cambie su valor, facilitando la validación en tiempo real y la actualización de otras partes del sistema que dependan de esta propiedad.

Implementación:

```
using System;
using System.ComponentModel;

public class Chofer : INotifyPropertyChanged
{
    private string tipoLicencia;
    public event PropertyChangedEventHandler PropertyChanged;

    public string TipoLicencia
    {
        get => tipoLicencia;
        set
        {
            if (tipoLicencia != value)
            {
                if (ValidarTipoLicencia(value))
                {
                    tipoLicencia = value;
                    OnPropertyChanged(nameof(TipoLicencia));
                }
                else
                {
                    throw new ArgumentException("Tipo de licencia no válido.");
                }
            }
        }
    }

    protected void OnPropertyChanged(string propertyName)
    {
        PropertyChanged?.Invoke(this, new PropertyChangedEventArgs(propertyName));
    }

    private bool ValidarTipoLicencia(string nuevaLicencia)
    {
        // Lógica de validación según los requisitos
        return true; // Ejemplo simplificado
    }
}
```



Ventajas:

- Proporciona un mecanismo estándar para notificar cambios en las propiedades.
- Facilita la integración con interfaces de usuario que requieren actualización en tiempo real.

Consideraciones:

- Requiere que las partes interesadas se suscriban al evento `PropertyChanged` para responder a los cambios. [Iteramos](#)
- Es importante manejar adecuadamente las excepciones para evitar interrupciones en la aplicación.

CHAT CON: CHATGPT



Enfoque 1: Validación en el Setter de la Propiedad

Este enfoque utiliza un método de validación dentro del setter de la propiedad TipoLicencia.

Implementación

```
using plboo.Interfaces;

10 referencias
public class Chofer : Piloto
{
    private string tipoLicencia;
    5 referencias
    public string Nombre { get; set; }
    5 referencias
    public int Edad { get; set; }
    6 referencias
    public string TipoLicencia
    {
        get { return tipoLicencia; }
        set
        {
            ValidarLicencia(value, Edad);
            tipoLicencia = value;
        }
    }
}

2 referencias
public Chofer(string nombre, int edad, string tipoLicencia)
{
    Nombre = nombre;
    Edad = edad;
    TipoLicencia = tipoLicencia; // Esto llamará al setter y validará la licencia
}

1 referencia
private void ValidarLicencia(string licencia, int edad)
{
    if (edad < 18 && licencia != "M")
    {
        throw new ArgumentException("Los menores de 18 años solo pueden tener licencia tipo M.");
    }
    // Agrega más validaciones según sea necesario
}

0 referencias
public void MostrarInformacion()
{
    Console.WriteLine("Nombre: {0}", this.Nombre);
    Console.WriteLine("Edad: {0}", this.Edad);
    Console.WriteLine("Licencia Tipo: {0}", this.TipoLicencia);
}
}
```

CHAT CON: **COMPILOT**