

1. NORMAS DE ESTILO

Líneas 30-31. Falta incluir las llaves, es necesario intentar no omitirlas.

```

30
31 // RMB2324 Se incluyen las llaves
32 if (pwd==null || pwd.Length <= 0)
33 {
34     return -1; // Si la contraseña es nula o vacía, devolvemos un código de error
35 }
36
37 if (pwd.Length < 6)
38 {
39     return 0; // No tiene la longitud mínima, error
40 }

```

Líneas 42, 43, 44, y 45

Es buena praxis declarar las variables al inicio del método.

```

26 public int test(string p)
27 {
28     pwd = p;
29
30     //RMB2324 Se declaran las variables al inicio del método
31     bool mins = false;
32     bool mays = false;
33     bool nums = false;
34     bool length = false;
35

```

Línea 35.

Los nombres de los métodos deben ser PasCal.

```

34 //RMB2324 Los nombres de los métodos deben ser PasCal
35 1 referencia | 0 cambios | 0 autores, 0 cambios
36 public int Test(string p)
37 {

```

2. DISEÑO DE PRUEBAS (CAJA NEGRA)

Parámetro password

A1. password → Vacía o nula → Error

A2. password < 6 → Error

A3. password > 12 → Ok

A3.1. fortaleza 1 → contraseña débil

A3.2. fortaleza 2 → contraseña normal

A3.3. fortaleza 3 → contraseña fuerte

A3.4. password contiene al menos 4 minúsculas, 4 mayúsculas y 4 números → Contraseña muy fuerte

Clases de equivalencia	Parámetro password	Salida
A1	""	Error
A2	1234	Error
A3	123456	Ok
A3.1	aaaaaa	Contraseña débil
A3.2	aaaaa1	Contraseña normal
A3	aaAA11	Contraseña fuerte
A3.4	AAAAaaaa1111	Contraseña muy fuerte

4. REFACTORIZACIÓN

1. Cambiar nombres → dar nombres autoexplicativos.

Se realiza a través de Editar > Refactorizar > Cambiar nombre

```
//RMB2324 Poner nombres autoexplicativos
private bool minúsculas;
private bool mayúsculas;
private bool números;
private bool length;
```

2. Encapsular campo.

Se realiza a través de Editar > Refactorizar > Encapsular campo.

Se hace el campo privado y se generan las propiedades.

```
//RMB2324 Encapsular campo
private string pwd;
```

3. Reemplazar número mágico por constante simbólica.

Se crean las constantes.

```
//RMB2324 Declarar constantes
public const string ERR_CONTRASEÑA_NULA_O_VACIA = "";
public const string ERR_CONTRASEÑA_CORTA = "";
```

Se sustituyen los números mágicos por constantes.

```
if (Password==null || Password.Length <= 0)
{
    //RMB2324 Se sustituyen los números mágicos por las constantes
    throw new ArgumentOutOfRangeException(ERR_CONTRASEÑA_NULA_O_VACIA);
}

if (Password.Length < 6)
{
    throw new ArgumentOutOfRangeException(ERR_CONTRASEÑA_CORTA);
}
```

