

Universidad Tecnológica Nacional

Facultad Regional Avellaneda



Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos

Materia: Laboratorio de Programación II

Apellido:		Fecha:	18-05-2021
Nombre:		Docente ⁽²⁾ :	Dávila
División:	2°D	Nota ⁽²⁾ :	
Legajo:		Firma ⁽²⁾ :	
Instancia ⁽¹⁾ :	<div style="display: flex; justify-content: space-around;"> PP RPP X SP RSP FIN </div>		

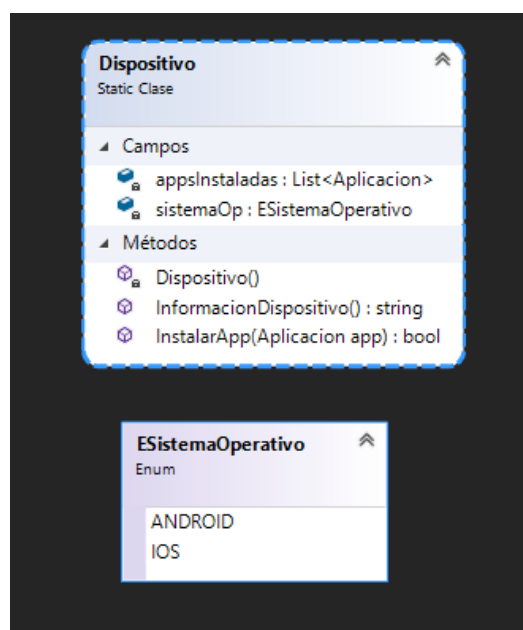
(1) Las instancias validas son: 1^{er} Parcial (PP), Recuperatorio 1^{er} Parcial (RPP), 2^{do} Parcial (SP), Recuperatorio 2^{do} Parcial (RSP), Final (FIN). Marque con una cruz.

(2) Campos a ser completados por el docente.

IMPORTANTE:

- 2 (dos) errores en el mismo tema anulan su puntaje.**
- La correcta documentación y reglas de estilo de la cátedra serán evaluadas.
- Colocar sus datos personales en el nombre de la carpeta principal y la solución:
Apellido.Nombre.Div. Ej: Pérez.Juan.2D. No se corregirán proyectos que no sea identificable su autor.
- No se corregirán exámenes que no compilen.
- Reutilizar** tanto código como crean necesario.
- Colocar nombre de la clase (en estáticos), **this** o **base** en todos los casos que corresponda.
- Aplicar los principios de los 4 pilares de la POO.

1. Crear un proyecto del tipo Biblioteca de Clases y colocar el siguiente esquema de clases:



Aplicacion

Abstract Clase

Campos

- nombre : string
- sistemaOp : ESistemaOperativo
- tamanoMb : int

Propiedades

- SistemaOperativo { get; } : ESistemaOperativo
- Tamano { get; } : int

Métodos

- Aplicacion(string nombre, ESistemaOperativo sistOp, int tamano)
- implicit operator Aplicacion(List<Aplicacion> listaApp) : Aplicacion
- InfoApp() : string
- operator !=(List<Aplicacion> listaApp, Aplicacion app) : bool
- operator +(List<Aplicacion> listaApp, Aplicacion app) : bool
- operator ==(List<Aplicacion> listaApp, Aplicacion app) : bool
- ToString() : string



AppJuegos

Clase

→ Aplicacion

Propiedades

- Tamano { get; } : int

Métodos

- AppJuegos(string nombre, ESistemaOperativo sistOp, int tamano)



AppMusical

Clase

→ Aplicacion

Campos

- listaCanciones : List<string>

Propiedades

- Tamano { get; } : int

Métodos

- AppMusical(string nombre, ESistemaOperativo SistOp, int tamaniolnicial)
- AppMusical(string nombre, ESistemaOperativo SistOp, int tamaniolnicial, List<string> lista)
- InfoApp() : string

2. Clase **Aplicación**:

- a. Las propiedades marcadas en cursiva son abstractas.
- b. InfoApp retornará los datos de la aplicación utilizando StringBuilder.
- c. El operador implícito recibirá una lista de aplicaciones y devolverá la instancia de la aplicación que más tamaño tenga.
- d. Método ToString(): Devolverá el nombre de la aplicación.
- e. La sobrecarga del operador == buscará si una aplicación existe en la lista recibida por parámetro comparando por su nombre. Devolverá true si ya existe.
- f. La sobrecarga del operador + añadirá una aplicación a la lista de aplicaciones pasada por parámetro siempre y cuando sea no exista previamente. Reutilizar código.

3. Clase **AppJuegos**:

- a. Propiedad Tamano retorna el tamaño de la aplicación.

4. Clase **AppMusical**:

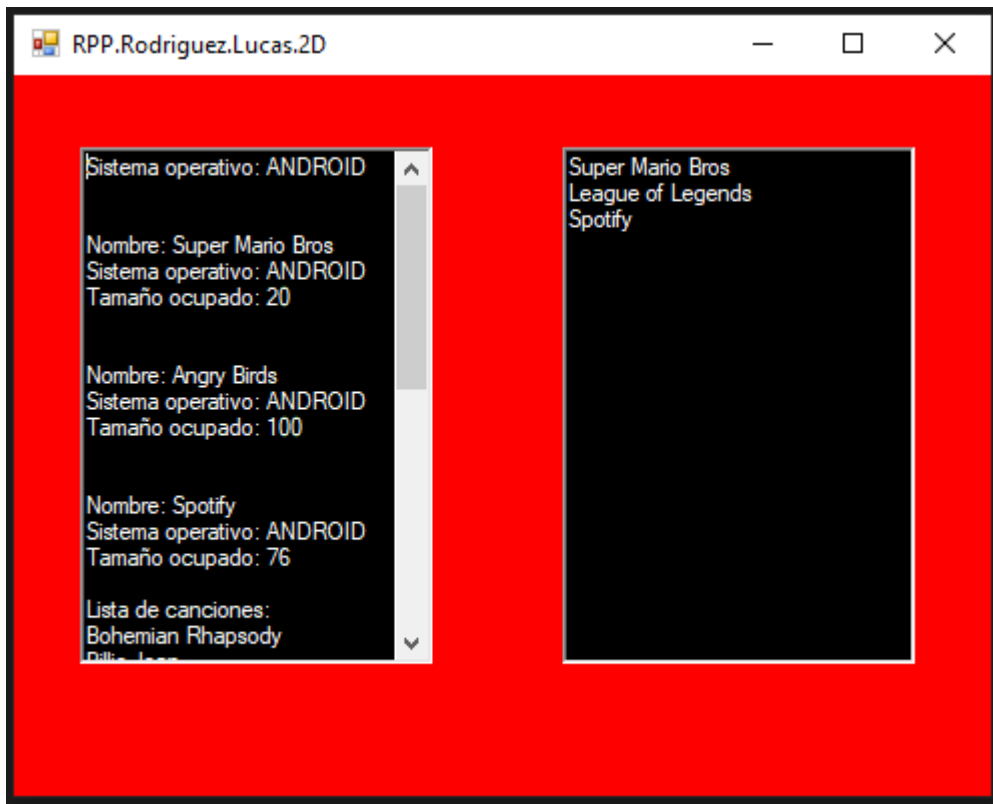
- a. Propiedad Tamano retorna el tamaño de la app, el cual crecerá en base a la cantidad de canciones que tenga cargada. Por cada canción cargada, su tamaño crecerá en 2mb más.
Por ej: Si la aplicación pesa 50, y tiene 10 canciones, entonces se retornará 70.
- b. InfoApp mostrará también la lista de canciones.

5. Clase **Dispositivo**:

- a. Es una clase estática.
- b. El constructor estático instanciará la lista estática appsInstaladas, y seteará el sistema operativo de este dispositivo en ANDROID.
- c. El método InstalarApp recibirá por parámetro una aplicación para ser instalada. Antes de intentar instalarla, debe comprobarse que sea compatible para ese sistema operativo. Una vez comprobado que es compatible, intentará añadir esa aplicación a la lista de aplicaciones. Devolverá true si logra añadirla. Reutilizar código.
- d. InformacionDispositivo mostrará toda la información del dispositivo, y el detalle de sus aplicaciones instaladas.

6. Formulario **FormPrincipal**:

- a. El formulario debe iniciar centrado en la pantalla con el siguiente formato:



- b. Colocar el BackColor en un tono de Rojo.
- c. Colocar el BackColor de ambos del RichTextBox en Negro.
- d. Colocar el ForeColor de ambos RichTextBox en Blanco.
- e. El nombre del formulario debe ser RPP.Apellido.Nombre del alumno.
- f. El RichTexbox izquierdo se llamará rtbAppInstaladas.
- g. El RichTextBot derecho se llamará rtbAppNoInstaladas.
- h. Agregar el siguiente código al evento **Load** del formulario.

```

List<Aplicacion> listaApps = new List<Aplicacion>()
{
    new AppJuegos("Super Mario Bros", ESistemaOperativo.ANDROID, 20),
    new AppJuegos("Super Mario Bros", ESistemaOperativo.ANDROID, 20),
    new AppJuegos("Angry Birds", ESistemaOperativo.ANDROID, 100),
    new AppJuegos("League of Legends", ESistemaOperativo.IOS, 500),
    new AppMusical("Spotify", ESistemaOperativo.ANDROID, 50,
        new List<string>()
        {
            "Bohemian Rhapsody - Queen",
            "Billie Jean - Michael Jackson",
            "Another Brick In The Wall - Pink Floyd",
            "Romeo And Juliet - Dire Straits",
            "Where The Streets Have No Name - U2",
            "Let It Be - The Beatles",
            "Could You Be Loved - Bob Marley",
            "Every Breath You Take - The Police",
            "November Rain - Guns n Roses",
            "Crazy Train - Ozzy Osbourne",
            "Volver - Carlos Gardel",
            "En La Ciudad De La Furia - Soda Stereo",
            "Gualicho - Patricio Rey",
            "Paint It Black - Rolling Stones"
        }
    ),
    new AppMusical("Spotify", ESistemaOperativo.ANDROID, 50),
    new AppMusical("YouTube Music", ESistemaOperativo.ANDROID, 30)};

foreach (Aplicacion app in listaApps)
    if (!Dispositivo.InstalarApp(app))
        rtbAppNoInstaladas.Text += string.Concat(app.ToString(), "\n");

this.rtbAppInstaladas.Text = Dispositivo.InformacionDispositivo();

```