

# APLICACIONES I

```
3 require File.expand_path("../support/spec_helper", __FILE__)
4 # Prevent database truncation if the environment is production
5 abort("The Rails environment is running in production mode!") if Rails.env.production?
6 require 'spec_helper'
7 require 'rspec/rails'
8
9 require 'capybara/rspec'
10 require 'capybara/rails'
11
12 Capybara.javascript_driver = :webkit
13 Category.delete_all; Category.create(:name => "Category")
14 Shoulda::Matchers.configure do |config|
15   config.integrate do |with|
16     with.test_framework :rspec
17     with.library :rails
18   end
19 end
20
21 # Add additional requires here so the controller and model stubs have access to
22 # the required constants
23 # Requires supporting ruby files with constants like :supporting_constants
24 # spec/support/ and its subdirectories. These files will be loaded by default
25 # run as spec files by default. This can be overridden by running
26 # in _spec.rb will both be required and loaded.
27 # run twice. It is recommended to use the :require option to
28 # end with _spec.rb. You can override this by using the :load option.
29 # rubicon on the command line or by using the :load option.
30
31 No results found for 'mongoid'
```

# **APLICACIÓN QUE REALIZA OPERACIONES MATEMATICAS**

## **Estudiante**

Steven Andres Navarro G

## **Docente**

David Bohórquez Caro

**POLITECNICO INTERNACIONAL  
DESARROLLO DE SOFTWARE Y APLICACIONES MOVILES  
APLICACIONES I  
MAYO 2024  
BOGOTA**

## Contenido

OBJETIVOS .....	4
Objetivo General .....	4
Objetivos específicos .....	4
PROBLEMÁTICA .....	5
SOLUCION .....	6
Diseño de Formulario. ....	6
Configurar los controles .....	6
Escribir el código.....	7
Detalles del código.....	7
Ejecución de la aplicación .....	8
CONCLUSIONES .....	9

## OBJETIVOS

### Objetivo General

Desarrollar una aplicación de Windows Forms en C# que permita realizar la suma algebraica de dos números ingresados por el usuario.

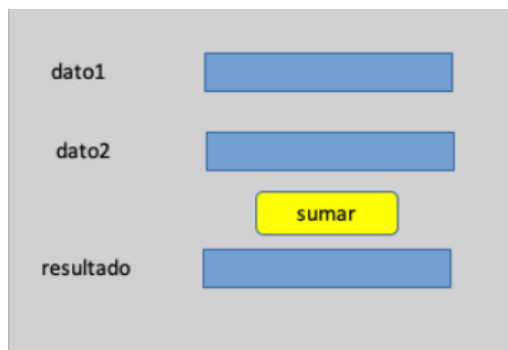
### Objetivos específicos

- Facilitar el aprendizaje y uso de Windows Forms y C#. A través del desarrollo de esta aplicación, se proporciona una experiencia práctica y útil para aquellos interesados en aprender sobre el desarrollo de aplicaciones de escritorio utilizando Windows Forms y el lenguaje C#.
- Promover el desarrollo de aplicaciones útiles. La aplicación no solo tiene un propósito educativo, sino que también puede servir como base para desarrollar aplicaciones más complejas que requieran operaciones matemáticas o procesamiento de datos.
- Desarrollar habilidades de validación de datos. Al incluir la validación de entrada de datos, se fomenta la importancia de crear aplicaciones robustas y seguras, evitando errores comunes que pueden surgir del ingreso de datos incorrectos.

## PROBLEMÁTICA

La empresa Edmsoft -de desarrollo de software- quiere incursionar en el desarrollo de aplicaciones en el lenguaje de programación C#. La empresa ya conoce cómo construir aplicaciones de consola en dicho lenguaje de programación, pero quiere averiguar cómo utilizar interfaces gráficas de usuario Windows Forms para realizar aplicaciones sencillas. Para ir conociendo un poco más el tema de las interfaces gráficas de usuario Windows Forms, la empresa Edmsoft quiere construir una pequeña aplicación con interfaz gráfica de usuario que realice la suma de dos números ingresados en cajas de texto que hacen parte de la interfaz gráfica.

A continuación, se muestra una imagen de un diseño ejemplo de lo que quiere construir la empresa Edmsoft. Básicamente se trata de un formulario que cuenta con unas cajas de texto para recibir los dos datos a sumar, un botón que al presionarse realice la suma, y una tercera caja de texto que muestre el resultado de la suma.



El diagrama muestra una interfaz gráfica de usuario (GUI) con un fondo gris. En la parte superior izquierda, hay una etiqueta "dato1" seguida de una caja de texto azul. Debajo de ella, hay una etiqueta "dato2" seguida de otra caja de texto azul. En el centro, hay un botón amarillo con el texto "sumar". En la parte inferior, hay una etiqueta "resultado" seguida de una tercera caja de texto azul.

## SOLUCION

### Diseño de Formulario.

En el IDE Visual Studio, creamos un proyecto nuevo (Windows Forms), al crear el proyecto nos aparece una ventana en blanco en donde vamos a proceder a realizar el diseño del formulario.

En el Diseñador de Windows Forms, agregamos los controles necesarios desde el cuadro de herramientas:

- Agregamos tres controles TextBox para que el usuario ingrese los números y el tercero para mostrar el resultado de la operación.
- Agregamos tres controles Button. Uno para realizar la suma, el segundo para limpiar el formulario, y el tercero para salir de la aplicación.
- Agregamos los controles Label para darle nombre a los TextBox y a los Button.

### Configurar los controles

Asigna nombres descriptivos a los controles para facilitar la escritura del código. Por ejemplo:

- textBox1 para el primer número.
- textBox2 para el segundo número.
- textBox3 para mostrar el resultado.
- Button1 para el botón que realiza la suma.
- Button2 para el botón que limpia el formulario.
- Button3 para el botón que sale de la aplicación

Escribir el código.

```
}

1 referencia
private void button1_Click(object sender, EventArgs e)
{
    double dato1, dato2;

    dato1 = Convert.ToDouble(textBox1.Text);
    dato2 = Convert.ToDouble(textBox2.Text);

    textBox3.Text = Convert.ToString(dato1 + dato2);
}

1 referencia
private void textBox3_TextChanged(object sender, EventArgs e)
{
}

1 referencia
private void button2_Click(object sender, EventArgs e)
{
    textBox1.Text = "";
    textBox2.Text = "";
    textBox3.Text = "";
}

1 referencia
private void button3_Click(object sender, EventArgs e)
{
    Close();
}
}
```

## Detalles del código

- **Obtención de Valores:** Los valores ingresados en los TextBox se obtienen como cadenas de texto.
- **Conversión de Valores:** Se utiliza `Convert.ToDouble` para intentar convertir las cadenas a números de tipo double. Esta función devuelve true si la conversión es exitosa.
- **Realización de la Suma:** Si ambas conversiones son exitosas, se realiza la suma de los dos números.
- **Mostrar Resultado:** El resultado de la suma se muestra en el textBox3.
- **Manejo de Errores:** Si alguna de las conversiones falla, se muestra un mensaje de error indicando que se deben ingresar números válidos.

## Ejecución de la aplicación

Guardamos todos los cambios y hacemos clic en el botón de "Iniciar" en Visual Studio para compilar y ejecutar la aplicación. Esto abrirá la ventana del formulario donde podemos ingresar dos números y ver el resultado de su suma al presionar el botón.

Con esta guía, tenemos una aplicación Windows Forms funcional que realiza la suma algebraica de dos números ingresados por el usuario.



## CONCLUSIONES

**Desarrollo Exitoso de la Aplicación:** Se logró desarrollar una aplicación de Windows Forms en C# que permite a los usuarios ingresar dos números, realizar su suma algebraica y mostrar el resultado de manera clara y precisa. Este proyecto demostró la eficacia del uso de Windows Forms para crear aplicaciones de escritorio interactivas y fáciles de usar.

**Interfaz de Usuario Intuitiva y Funcional:** La interfaz de usuario diseñada cumplió con los objetivos de ser intuitiva y accesible. Los controles utilizados (TextBox, Button, y Label) se integraron de manera efectiva para proporcionar una experiencia de usuario fluida, permitiendo ingresar datos y obtener resultados de forma eficiente.

**Adquisición y Aplicación de Buenas Prácticas de Programación:**

A lo largo del desarrollo del proyecto, se aplicaron buenas prácticas de programación, como el uso de nombres descriptivos para variables y métodos, y la documentación del código. Esto no solo facilitó el desarrollo y la depuración, sino que también mejoró la mantenibilidad del código.

**Documentación Completa del Proceso:** La documentación detallada de cada etapa del desarrollo del proyecto proporciona una guía útil para futuros desarrollos y estudios. La documentación abarca desde el diseño de la interfaz hasta la implementación del código y las pruebas, ofreciendo una referencia completa y estructurada.

### Resumen Final

El proyecto de creación de una aplicación Windows Forms en C# para la suma algebraica de dos números demostró ser un ejercicio valioso y exitoso. Los objetivos planteados se cumplieron satisfactoriamente, proporcionando una herramienta funcional y práctica, al mismo tiempo que se fomentó el aprendizaje y la aplicación de buenas prácticas de desarrollo de software. La experiencia adquirida y la documentación elaborada servirán como una sólida base para futuros proyectos y estudios en el campo del desarrollo de aplicaciones.