

Apresentação da disciplina

Tópicos Especiais em Desenvolvimento de Sistemas

Prof. Manuel Fernández Paradela Ledón
manuel.ledon@cruzeirosul.edu.br

Profs. Alcides, Gobbato, Luiz Reis, Ana Paula

Professor da disciplina

Prof. MSc. Manuel Fernández Paradela **Ledón**



- Graduação em Engenharia Elétrica (Computação)
- Mestrado em Informática Aplicada na Engenharia e Arquitetura

Universidad Tecnológica de La Habana José Antonio Echeverría. Diplomas revalidados na Universidade de São Paulo, USP.

Contato

manuel.ledon@cruzeirosul.edu.br

mfpledon@gmail.com

<https://sites.google.com/site/mfpledon/info-geral>

<http://mfpledon.com.br>

Contatos da Coordenação



Coordenação do Curso de CCP

Prof. Dr. Alcides Teixeira Barboza Jr.

alcides.junior@cruzeirodosul.edu.br

Coordenação do Curso de ADS

Prof^ª. Dr^ª. Cristiane Camilo Hernandez

cristiane.camilo@cruzeirodosul.edu.br

Coordenação de Informática

Campus São Miguel Paulista, 5º andar do bloco

B. Secretaria: F. 2037-5751



Horário das aulas

Manhã			
1h15m por módulo	1º Módulo	Intervalo (10 min)	2º Módulo
	8h30 às 9h45	9h45 às 9h55	9h55 às 11h10

Noturno			
1h15min por módulo	1º Módulo	Intervalo (10 min)	2º Módulo
	19h10 às 20h25	20h25 às 20h35	20h35 às 21h50

EMENTA da disciplina Tópicos Especiais em Desenvolvimento de Sistemas

Estudo de novas tecnologias, paradigmas e plataformas para desenvolvimento de sistemas de informações.

Objetivos da disciplina

OBJETIVOS

Cognitivos

a) Aprofundar nos paradigmas de desenvolvimento de aplicações que processam bancos de dados; b) Desenvolver aplicativos para Internet/Web; c) Desenvolver o interesse pela pesquisa e pelo conhecimento de novas tecnologias; d) Conhecer processos de elaboração de aplicações e explorar outras plataformas de desenvolvimento;

Habilidades

a) Utilizar os conceitos teóricos estudados em domínios de aplicação para tecnologias específicas dedesenvolvimento; b) Identificar os tipos de ferramentas de desenvolvimento de software e utilizar em problemas reais; c) Utilizar os novos conceitos e ser capaz de acomodá-los de acordo com o próprio conhecimento; d) Desenvolver diferentes soluções utilizando outras plataformas de desenvolvimento.

Atitudes

a) Ter desenvoltura e segurança na utilização das linguagens de programação e tecnologias estudadas; b) Ser crítico, ético, receptivo e estar preparado para o trabalho em equipes ou coletivos de pesquisadores e programadores; c) Ser analítico e responsável; d) Solidificar o pensamento abstrato; e) Ser criativo e ter iniciativa diante da solução de problemas.

UNID.	C/H	CONTEÚDO
I	3	Apresentação da Disciplina - Apresentação e discussão do Plano de Ensino, focando objetivos, conteúdos, estratégias, avaliação e bibliografia
II	6	Introdução Visual Studio - Conceitos da plataforma .NET- Apresentação do Visual Studio .NET- Introdução à linguagem C#
III	9	Interface do usuário e Orientação a objetos - Interface gráfica - Classes e Objetos gráficos
IV	9	Manipulação de banco de dados - O modelo ADO.NET- Classes e Modos de acesso - Aplicações utilizando conexão a banco de dados
V	12	Desenvolvimento web com Visual Studio - Aplicativos para Web - Web Applications e ASP.NET- Cookies, Session e Application- Acesso a banco utilizando ASP.NET. Web Services.
VI	6	Framework e mapeamento objeto-relacional - Introdução ao desenvolvimento utilizando recursos avançados de programação para mapear objeto relacional tais como: Entity Framework e NHibernate;- LINQ: Criação de Queries
VII	9	Computação em nuvem - Fundamentos de Cloud Computing: terminologias e conceitos- Elasticidade, Resiliência, On-Demand- Benefícios, Desafios e Riscos das Plataformas e Serviços- Administração e Regras- Software as a Service (SaaS)- Platform as a Service (PaaS) - Infrastructure as a Service (IaaS)- Cloud Delivery Models-Modelos de computação em nuvem (Amazon AWS, Google Cloud, Microsoft Cloud, etc)- Migração de Aplicações para Nuvem.
VIII	6	Avaliações da disciplina - Prova Parcial e Prova Regimentoal presenciais

Total de horas da disciplina: 60 h/a

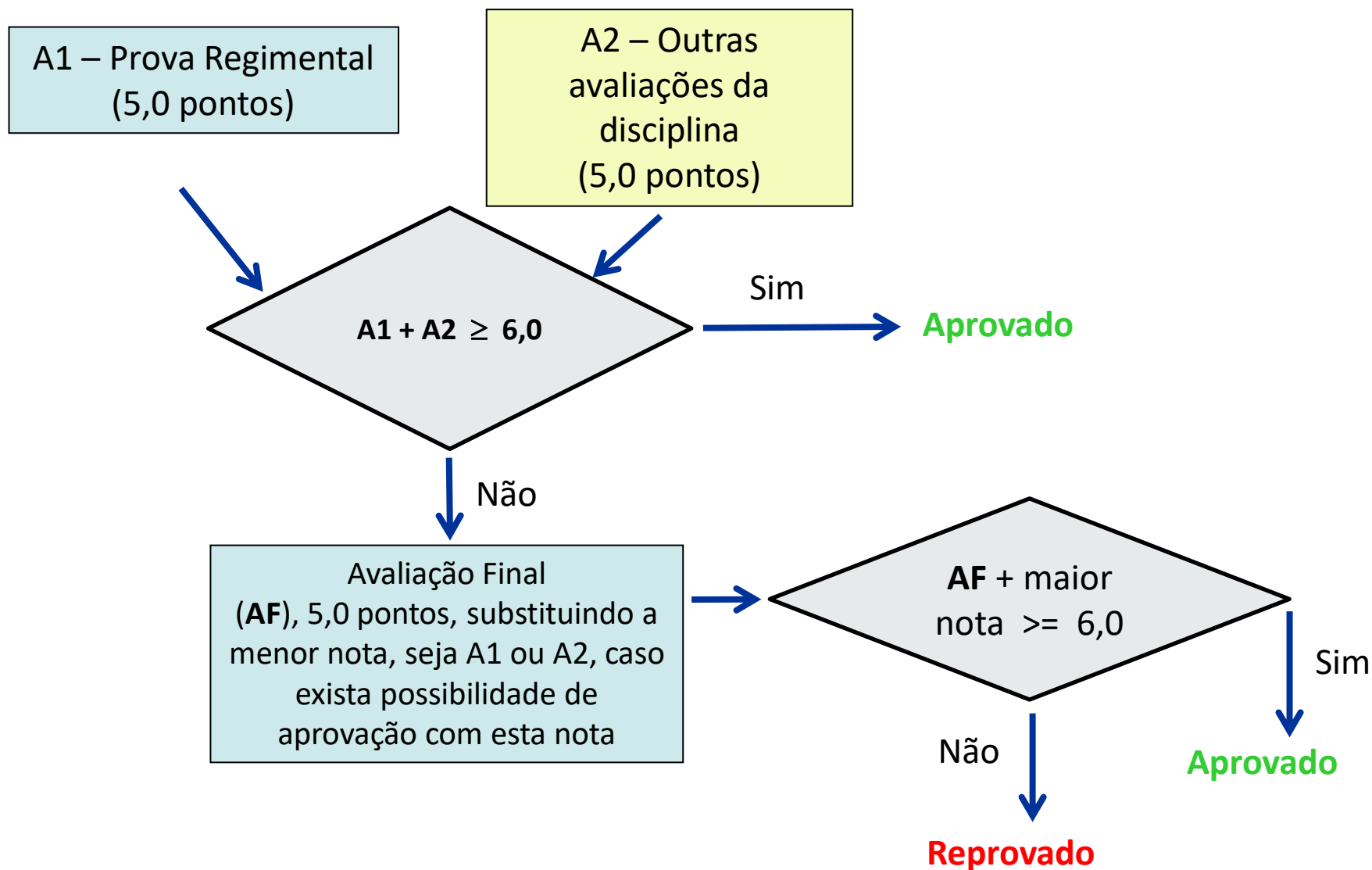
Bibliografia

BIBLIOGRAFIA BÁSICA	BIBLIOGRAFIA COMPLEMENTAR
<p>LEDUR, C. L. Desenvolvimento de sistemas com C#. Porto Alegre: SAGAH, 2018. E-book.</p> <p>PICHETTI, R. F. V.; CORTES, E. da S.; PAIXÃO, V. S. M. Banco de dados. Porto Alegre: SAGAH, 2020. E-book.</p> <p>DEITEL, P.; DEITEL, H. Java: como programar. 10. ed. São Paulo: Pearson Education do Brasil, 2016. E-book.</p>	<p>Deitel, Paul J.; Deitel, Harvey M. Ajax, Rich Internet Applications e Desenvolvimento Web. Editora Pearson. Edição: 1º (2008). Idioma: Português. ISBN: 9788576051619 (e-book)</p> <p>LOPES MARINHO, A. et. al. Desenvolvimento de aplicações para Internet. Editora Pearson. Edição: 2ª (2019). Idioma: Português. ISBN: 9786550110604 (e-book)</p> <p>Kalbach, James. Design de Navegação Web - Otimizando a Experiência do Usuário Ed. Bookman, Artmed. Porto Alegre: 2009. (e-book)</p> <p>Miletto, Evandro M.; Bertagnolli, Silvia C. Desenvolvimento de Software II - Introdução ao Desenvolvimento Web com HTML, CSS, JavaScript e PHP EAN: 9788582601969. Editora Bookman. Porto Alegre: 2014 (e-book)</p> <p>Paz, Mónica. Webdesign. Editora Intersaberes. Edição: 1º (2021) Idioma: Português. ISBN: 9786555179378 (e-book)</p>

Avaliação da disciplina

- **A1** (Prova Regimental) - 5,0 pontos
- **A2** (restantes avaliações) - 5,0 pontos, divididos em:
 - ☐ Prova Parcial - 3,0 pontos
 - ☐ Projeto Interdisciplinar: disciplina Tópicos Especiais e a disciplina Análise e Projeto de Sistemas II - 1,0 ponto
 - ☐ Exercícios para entregar pelo Blackboard - 1,0 ponto





Obs: mínimo de 75% de frequência para aprovação

Softwares que serão utilizados

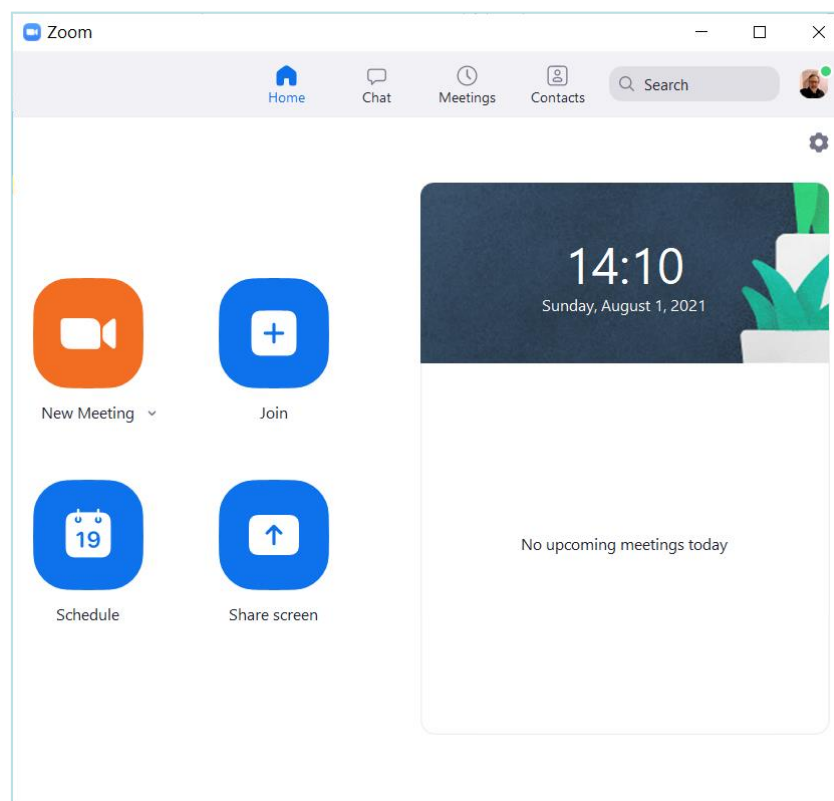
Nesta disciplina vamos utilizar os seguintes softwares:

- **Visual Studio Community 2022**
- MySQL
- SQLyog, MySQL Workbench, Microsoft Access, SQL Server, conector MySQL para .NET.
- Outros...



bb.cruzeirodosulvirtual.com.br

Zoom



A plataforma .NET da Microsoft

A plataforma .NET.
O .NET framework.
Linguagens de programação.
Componentes do .NET framework.

Documentação sobre o .NET

<https://docs.microsoft.com/pt-br/dotnet/>



.NET

.NET is a free, cross-platform, open source developer platform for building many different types of applications.

With .NET, you can use multiple languages, editors, and libraries to build for web, mobile, desktop, gaming, and IoT.

Fonte: <https://dotnet.microsoft.com/learn/dotnet/what-is-dotnet>

Microsoft .NET

- A Microsoft define .NET como um conjunto de tecnologias que abrangem diversas plataformas.
“The .NET Framework provides a comprehensive programming model for building all kinds of applications on Windows, from mobile to web to desktop.”
<https://www.microsoft.com/net/download/framework>
- Como veremos, trata-se de um conjunto de ferramentas, bibliotecas e tecnologias que permitem construir aplicativos para Windows (Windows Applications), aplicativos em modo de interface de linha de comando (console applications), aplicativos para a Web (ASP.NET Web applications, Web Components, Web Services etc.), aplicativos para mobile multiplataforma e outros projetos mais específicos.

Languages

You can write .NET apps in C#, F#, or Visual Basic.

- C# is a simple, modern, object-oriented, and type-safe programming language.
- F# is a cross-platform, open-source, functional programming language for .NET. It also includes object-oriented and imperative programming.
- Visual Basic is an approachable language with a simple syntax for building type-safe, object-oriented apps.



Fonte: <https://dotnet.microsoft.com/learn/dotnet/what-is-dotnet>

Cross Platform

Whether you're working in C#, F#, or Visual Basic, your code will run natively on any compatible OS. Different .NET implementations handle the heavy lifting for you:

- **.NET Core** is a cross-platform .NET implementation for websites, servers, and console apps on Windows, Linux, and macOS.
- **.NET Framework** supports websites, services, desktop apps, and more on Windows.
- **Xamarin/Mono** is a .NET implementation for running apps on all the major mobile operating systems.



.NET Standard é um conjunto base de APIs que são comuns a todas as implementações do .NET.
<https://dotnet.microsoft.com/platform/dotnet-standard>

.NET Core

O .NET Core é uma **implementação multiplataforma do .NET**, projetado para lidar com cargas de trabalho de servidor e na nuvem em escala. Ele é executado no Windows, no Linux e no macOS. O ASP.NET Core é executado no .NET Core.

.NET Framework

O .NET Framework é a implementação original do .NET que existe desde 2002. É o mesmo .NET Framework que os desenvolvedores do .NET sempre usaram. Ele contém APIs adicionais específicas do Windows, como APIs para desenvolvimento de área de trabalho do Windows com o Windows Forms e o WPF. O .NET Framework é otimizado para a compilação de aplicativos da área de trabalho do Windows.

.NET Standard é um conjunto base de APIs que são comuns a todas estas implementações do .NET. Veja em <https://dotnet.microsoft.com/platform/dotnet-standard>.

Mono

O Mono é uma implementação do .NET que é usada principalmente quando um pequeno tempo de execução é necessário. É o tempo de execução que impulsiona aplicativos Xamarin no Android, Mac, iOS, tvOS e watchOS e concentra-se principalmente em um impacto pequeno. O Mono também é plataforma para jogos criados com o mecanismo Unity.

UWP (Plataforma Universal do Windows)

A UWP é uma implementação do .NET que é usada para criar aplicativos do Windows modernos e sensíveis ao toque, bem como software para a IoT (Internet das Coisas). Ela foi projetada para unificar os diferentes tipos de dispositivos, incluindo PCs, tablets, phablets, telefones e até mesmo o Xbox.

.NET Framework

É um grupo de bibliotecas e recursos, que permitem desenvolver aplicativos .NET, independentemente das linguagens de programação utilizadas. Existem versões como a 4.7 (a versão 4.5 do .NET Framework já vinha instalada junto com o Windows 8.1). Você pode usar o .NET Framework para desenvolver os seguintes tipos de aplicativos e serviços:

- Aplicativos de Console. Veja [Criando aplicativos de console](#).
- Aplicações Windows GUI (Windows Forms). Veja [Windows Forms](#).
- Aplicações Windows Presentation Foundation (WPF) Veja [Introdução ao Windows Presentation Foundation](#).
- Aplicações ASP.NET. Consulte [Desenvolvendo aplicativos Web com o ASP.NET](#).
- Serviços Windows. Veja [Introdução aos Aplicativos de Serviço do Windows](#).
- Aplicativos orientados à Serviços usando Windows Communication Foundation (WCF). Consulte [Desenvolvendo aplicativos orientados a serviços com WCF](#).
- Aplicativos habilitados para fluxo de trabalho usando Windows Workflow Foundation (WF). Consulte [Criando fluxos de trabalho no .NET Framework](#).

Visual Studio .NET ou simplesmente **Visual Studio** (diferente do Visual Studio Code).

É um ambiente para desenvolvimento de programas (**IDE**), que permite usar recursos de compilação, edição, depuração, ajuda etc.

Alguns componentes tradicionais do .NET Framework

<i>ASP.NET</i>	System.Web (suporte para formulários e serviços Web)
<i>User Interface</i>	System.Windows.Forms e System.Drawing (interface de usuário)
<i>ADO.NET</i>	System.Data (modelo ADO.NET) e System.XML (suporte para XML)
<i>.NET Framework Class Library</i>	System , System.IO , System.NET e outras classes de base
<i>Common Language Runtime</i>	Exception manager, Garbage collection, debug engine, etc.

- Na figura mencionamos alguns dos componentes do .NET Framework e os *namespaces* relacionados com estes componentes. Um "namespace" (espaço de nome, por exemplo: Forms ou Data) é uma forma de agrupar elementos (classes, interfaces, módulos) relacionados logicamente dentro de uma estrutura monolítica, que permita uma melhor compreensão e organização. Um *namespace* poderá conter várias classes, módulos de procedimentos e funções e interfaces.
- Veja uma lista completa da Biblioteca de classes do .NET Framework em: [http://msdn.microsoft.com/pt-br/library/gg145045\(v=VS.110\).aspx](http://msdn.microsoft.com/pt-br/library/gg145045(v=VS.110).aspx)

Terminologia relacionada com a plataforma .NET

- **CLR** - Sigla de Common Language Runtime. Base comum a todas as linguagens .NET, como C#, VB etc. O CLR é o ambiente que gerencia a execução de código escrito em qualquer linguagem. Faz parte do framework.
- **Framework** - É o modelo e grupo de bibliotecas da plataforma .NET para construir, instalar e rodar qualquer aplicação, no desktop ou na Internet. Para executar um programa .NET, é preciso ter o framework instalado.
- **MSIL** - Microsoft Intermediate Language. Quando se compila uma aplicação .NET, ela é convertida para uma linguagem intermediária. A MSIL é um conjunto de instruções independentes de CPU. Na hora de executar o programa, um novo compilador, chamado Just-in-time (JIT) Compiler, o converte para o código nativo, ou seja, específico para o processador da máquina.

CLR - Common Language Runtime

- Executa as aplicações .NET;
- Interage com o Sistema Operacional;
- Abstrai das aplicações .NET detalhes obscuros do Sistema Operacional;
- Torna possível que aplicações .NET sejam indiferentes ao hardware e sistema operacional;
- O .NET Framework tem este ambiente de tempo de execução chamado de Common Language Runtime, que executa o código e fornece serviços que tornam o processo de desenvolvimento mais fácil.

MSIL - Microsoft Intermediate Language

- O .Net Framework não está preso a uma linguagem de programação específica;
- O compilador converte o código fonte em Microsoft Intermediate Language (MSIL), que é um conjunto de instruções independentes de CPU, e que poderá ser eficientemente convertido no código nativo de uma máquina específica.
- Antes de executar, o MSIL deverá ser convertido no código específico da CPU, geralmente por um [Compilador Just-In-Time \(JIT\)](#).
- Como o Common Language Runtime fornece um ou mais compiladores JIT para cada arquitetura de computador para a qual oferece suporte, o mesmo conjunto de MSIL pode ser compilado e executado em qualquer arquitetura suportada.

Assembly

- É o código utilizado para disponibilização de aplicativos;
- Utilizado para versionamento e segurança: o uso do assembly especifica qual é a versão utilizada;
- Assemblies formam a unidade fundamental de implantação, controle de versão, reutilização, ativação de escopo e permissões de segurança para um aplicativo de base .NET. Assemblies tomam a forma de um arquivo executável (.exe) ou arquivo de biblioteca de links dinâmicos (.dll) e são os blocos de construção do .NET Framework. Eles fornecem o Common Language Runtime com as informações de que precisam para estarem cientes das implementações de tipo.

Tudo o que você precisa para
criar ótimos aplicativos.
Gratuito.

<https://visualstudio.microsoft.com/pt-br/free-developer-offers/>



Visual Studio Community

O melhor IDE abrangente para desenvolvedores .NET e C++ no Windows. Totalmente empacotado com uma bela matriz de ferramentas e recursos para elevar e aprimorar cada estágio de desenvolvimento de software.

[Saiba mais →](#)

Download gratuito



Visual Studio para Mac

Um IDE abrangente para desenvolvedores .NET nativo do macOS. Inclui suporte de alto nível para Web, nuvem, dispositivos móveis e desenvolvimento de jogos.

[Saiba mais →](#)

Leia mais sobre [ativação da sua licença](#)

Download gratuito



Visual Studio Code

Um editor de código-fonte independente que roda no Windows, macOS e Linux. A melhor escolha para desenvolvedores de JavaScript e Web, com extensões para dar suporte a praticamente qualquer linguagem de programação.

[Saiba mais →](#)

Ao usar o Visual Studio Code, você concorda com sua [licença](#) e [política de privacidade](#)

Download gratuito ▾

Download do Microsoft **Visual Studio**

<https://visualstudio.microsoft.com/pt-br/vs/features/net-development/>

Visual Studio Community 2022

IDE gratuito e completo para estudantes, desenvolvedores individuais e de software livre

Saiba mais →

Download gratuito

Visual Studio Professional 2022

Ferramentas profissionais para desenvolvedores, serviços e benefícios de assinatura para pequenas equipes

Saiba mais →

Avaliação gratuita

Visual Studio Enterprise 2022

Solução de ponta a ponta para atender às necessidades exigentes de qualidade e escala de equipes de todos os tamanhos

Saiba mais →

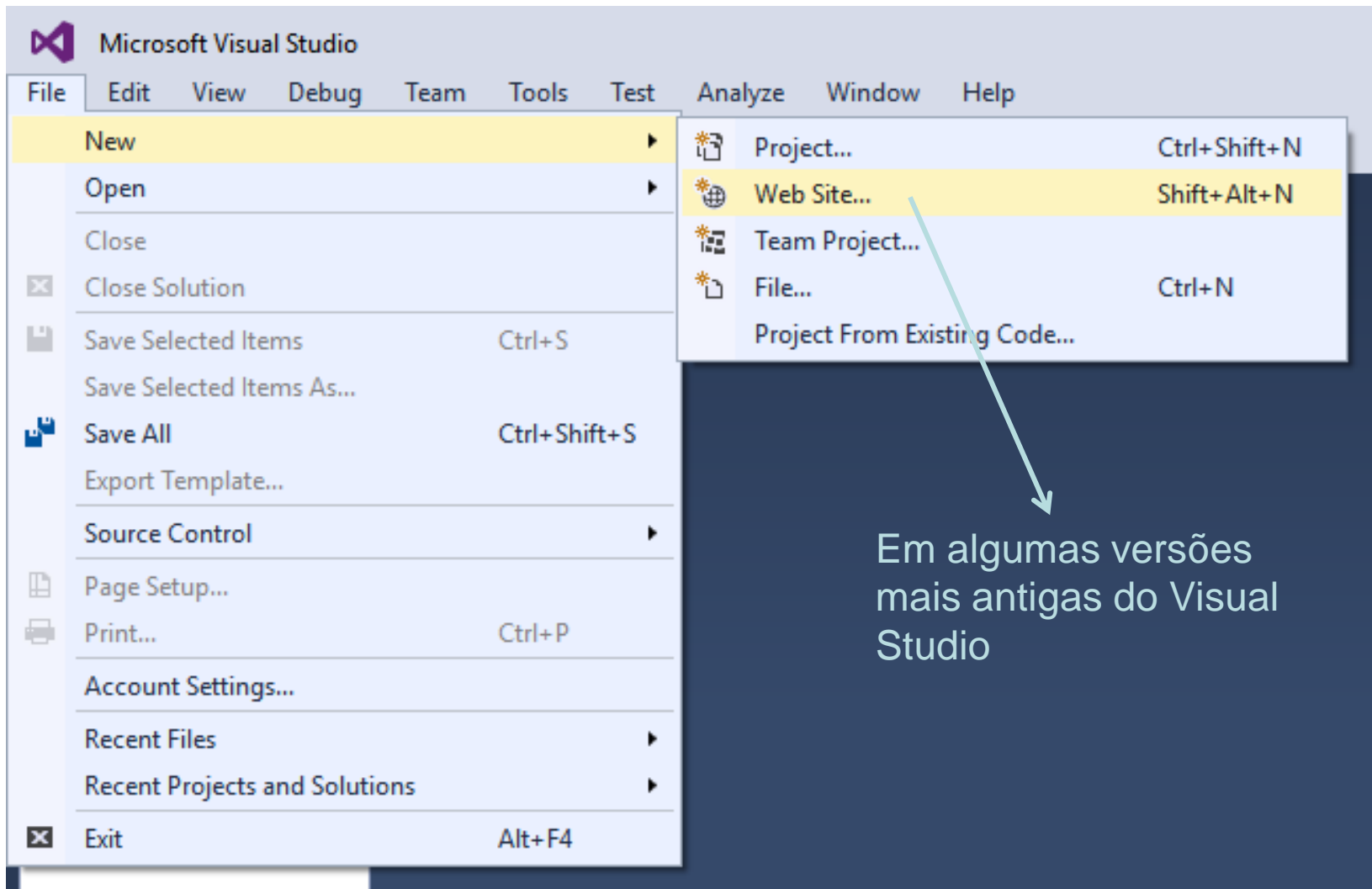
Avaliação gratuita



.NET

.NET

Usando o Visual Studio e o .NET, você pode desenvolver aplicativos para área de trabalho, Web, celulares, jogos e IoT. Você pode escrever aplicativos .NET nas linguagens C#, F# ou Visual Basic.



New Project

Recent












Installed

- Templates
 - Visual C#
 - Windows
 - Windows 8
 - Universal
 - Windows
 - Windows Phone
 - Classic Desktop**
 - Web
 - Android
 - Cloud
 - Extensibility
 - iOS
 - Mobile Apps
 - Reporting
 - Silverlight
 - Test
 - WCF
 - Workflow
 - Visual Basic
 - Visual C++
 - Visual F#

Online

.NET Framework 4.5.2 Sort by: Default

Search Installed Templates (Ctrl+E)

	Windows Forms Application	Visual C#
	WPF Application	Visual C#
	Console Application	Visual C#
	Shared Project	Visual C#
	Class Library	Visual C#
	Class Library (Portable)	Visual C#
	WPF Browser Application	Visual C#
	Empty Project	Visual C#
	Windows Service	Visual C#
	WPF Custom Control Library	Visual C#
	WPF User Control Library	Visual C#

[Click here to go online and find templates.](#)

Type: Visual C#
A project for creating an application with a Windows Forms user interface

Name: WindowsFormsApplication2

Location: c:\users\mfpledon\documents\visual studio 2015\Projects

Solution name: WindowsFormsApplication2

Browse...

☒ Create directory for solution

☐ Add to source control

OK Cancel

Novo Projeto

Recentes







Instalados

Visual C#

- Universal do Windows
- Windows Desktop
- Web
 - Versões Anteriores**
 - .NET Core
 - .NET Standard
 - Cloud
 - Teste
 - WCF

Não está encontrando o que procura?
[Abrir Instalador do Visual Studio](#)

Classificar por: Padrão

	Site ASP.NET Vazio	Visual C#
	Site de Web Forms do ASP.NET	Visual C#
	Site ASP.NET (Razor v3)	Visual C#
	Site de Entidades de Dados Dinâmicos do ASP.NET	Visual C#
	Site de Linq to SQL de Dados Dinâmicos do ASP.NET	Visual C#
	Serviço WCF	Visual C#

Pesquisar (Ctrl+E)

Tipo: Visual C#

Um site da Web vazio. Este modelo não produz um arquivo de projeto e tem suporte do MSBuild limitado.

Nome: WebSite4

Local: C:\Users\mfple\source\repos

Solução: Criar nova solução

Nome da Solução: WebSite4

Estrutura: .NET Framework 4.6.1

Procurar...





☒ Criar diretório para solução

☐ Adicionar ao controle do código-fonte

OK Cancelar

Criar um novo projeto

Modelos de projeto recentes

-  Aplicativo do Windows Forms C#
-  Android App (Xamarin) C#
-  Aplicativo Web ASP.NET (.NET Framework) C#
-  Aplicativo Web ASP.NET (.NET Framework) Visual Basic

C#

[Limpar tudo](#)

Todos os idiomas

Todas as plataformas

Todos os tipos de pr...



Mobile App (Xamarin.Forms)

A multiproject template for building apps for iOS and Android with Xamarin and Xamarin.Forms.

C#

Android

iOS

Windows

Celular



Aplicativo do Console (.NET Framework)

Um projeto para criar um aplicativo de linha de comando

C#

Windows

Console



Aplicativo Web ASP.NET (.NET Framework)

Modelos de projeto para criar aplicativos ASP.NET. Você pode criar aplicativos de API Web, MVC ou ASP.NET Web Forms e adicionar muitos outros recursos no ASP.NET.

C#

Windows

Nuvem

Web



Biblioteca de Classes (.NET Framework)

Um projeto para criar uma biblioteca de classes do C# (.dll)

C#

Windows



















Biblioteca

[Voltar](#)

[Próximo](#)

Criar um novo projeto


Modelos de projeto recentes

 Aplicativo Web ASP.NET (.NET Framework)	C#	
 Aplicativo Web do ASP.NET Core (Model-View-Controller)	C#	
 Aplicativo Web ASP.NET Core	C#	
 ASP.NET Core Vazio	C#	
 Aplicativo Blazor WebAssembly	C#	
 Aplicativo Blazor Server Vazio	C#	
 Aplicativo Blazor Server	C#	
 Aplicativo do Windows Forms	C#	
 Aplicativo do Console	C#	

Sugestão de alguns modelos interessantes para criar projetos .NET (modelos ou templates, com scaffolding de código para começar)

Criar um novo projeto

Modelos de projeto recentes

 Aplicativo Web ASP.NET Core

C#

Pesquisar modelos (Alt+S)



[Limpar tudo](#)

Todos os idiomas

Todas as plataformas

Web



Aplicativo Web ASP.NET Core

Um modelo de projeto para criar um aplicativo ASP.NET Core com um exemplo ASP.NET Core Razor Pages

C#

Linux

macOS

Windows

Nuvem

Serviço



Aplicativo Blazor Server

Um modelo de projeto para criar um aplicativo Blazor Server lado do servidor em um aplicativo ASP.NET Core e manipular um usuário em uma conexão SignalR. Esse modelo pode ser usado com UIs (interfaces do usuário) completas e dinâmicas.

C#

Linux

macOS

Windows

Blazor

Nuvem



API Web do ASP.NET Core

Um modelo de projeto para criar um aplicativo ASP.NET Core com um Controlador de exemplo para um serviço HTTP RESTful. Esse modelo também pode ser usado para Controladores e Exibições do ASP.NET Core MVC.

C#

Linux

macOS

Windows

Nuvem

Serviço

Web

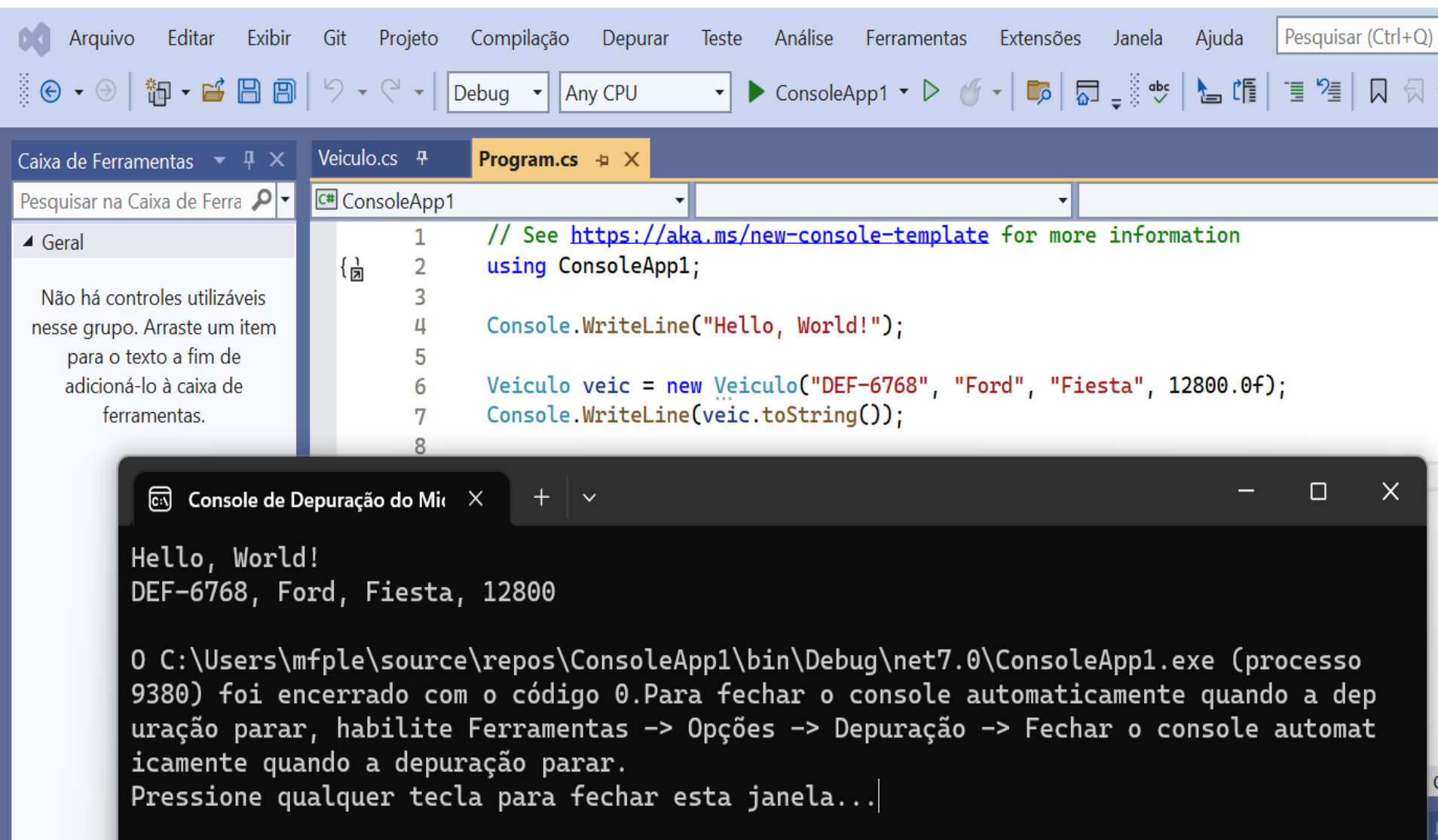
WebAPI



ASP.NET Core Vazio

Um modelo de projeto vazio para a criação de um aplicativo ASP.NET Core. Esse modelo não tem nenhum conteúdo.

Próximo



The image shows the Visual Studio IDE with a C# console application named 'ConsoleApp1'. The code in 'Program.cs' includes a comment linking to a new console template, a 'using' statement for 'ConsoleApp1', and a 'Main' method that writes 'Hello, World!' and the details of a 'Veiculo' object to the console. The 'Veiculo' object is instantiated with license plate 'DEF-6768', brand 'Ford', model 'Fiesta', and price '12800.0f'. The debug console shows the output of the program, followed by a message indicating the application has been closed and instructions on how to configure the IDE to close the console automatically.

Arquivo Editar Exibir Git Projeto Compilação Depurar Teste Análise Ferramentas Extensões Janela Ajuda Pesquisar (Ctrl+Q)

Caixa de Ferramentas

Pesquisar na Caixa de Ferramentas

Veiculo.cs Program.cs

ConsoleApp1

```
1 // See https://aka.ms/new-console-template for more information
2 using ConsoleApp1;
3
4 Console.WriteLine("Hello, World!");
5
6 Veiculo veic = new Veiculo("DEF-6768", "Ford", "Fiesta", 12800.0f);
7 Console.WriteLine(veic.toString());
8
```

Console de Depuração do Mic

Hello, World!
DEF-6768, Ford, Fiesta, 12800

O C:\Users\mfple\source\repos\ConsoleApp1\bin\Debug\net7.0\ConsoleApp1.exe (processo 9380) foi encerrado com o código 0. Para fechar o console automaticamente quando a depuração parar, habilite Ferramentas -> Opções -> Depuração -> Fechar o console automaticamente quando a depuração parar.
Pressione qualquer tecla para fechar esta janela...

WindowsFormsApplication1 - Microsoft Visual Studio

FILE EDIT VIEW PROJECT BUILD DEBUG TEAM FORMAT TOOLS TEST ARCHITECTURE ANALYZE WINDOW

Manuel Fernández Paradela Ledón

Quick Launch (Ctrl+Q)

Start Debug Any CPU

Form1.cs [Design]

Search Toolbox

- FolderBrowserDi...
- FontDialog
- GroupBox
- HelpProvider
- HScrollBar
- ImageList
- Label
- LinkLabel
- ListBox
- ListView
- (.) MaskedTextBox
- MenuStrip
- MessageQueue
- MonthCalendar
- NotifyIcon
- NumericUpDown
- OpenFileDialog
- PageSetupDialog
- Panel
- PerformanceCo...
- PictureBox
- PrintDialog
- PrintDocument
- PrintPreviewCon...
- PrintPreviewDial...

Cálculos especiais

Dado1:

Dado2:

Dado3:

Cálculo1 Cálculo2

Resultados:

Search Solution Explorer (Ctrl+ç)

Solution 'WindowsFormsApplication1' (1 project)

- WindowsFormsApplication1
 - Properties
 - References
 - App.config
 - Database1.mdf
 - Database1_log.ldf
 - Form1.cs
 - Form1.Designer.cs
 - Form1.resx
 - Form1
 - Program.cs

Code Analysis Solution Explorer Team Explorer Class View

Properties

Form1 System.Windows.Forms.Form

Size	409; 356
SizeGripStyle	Auto
StartPosition	WindowsDefaultLocation
Tag	
Text	Cálculos especiais
TopMost	False

Text

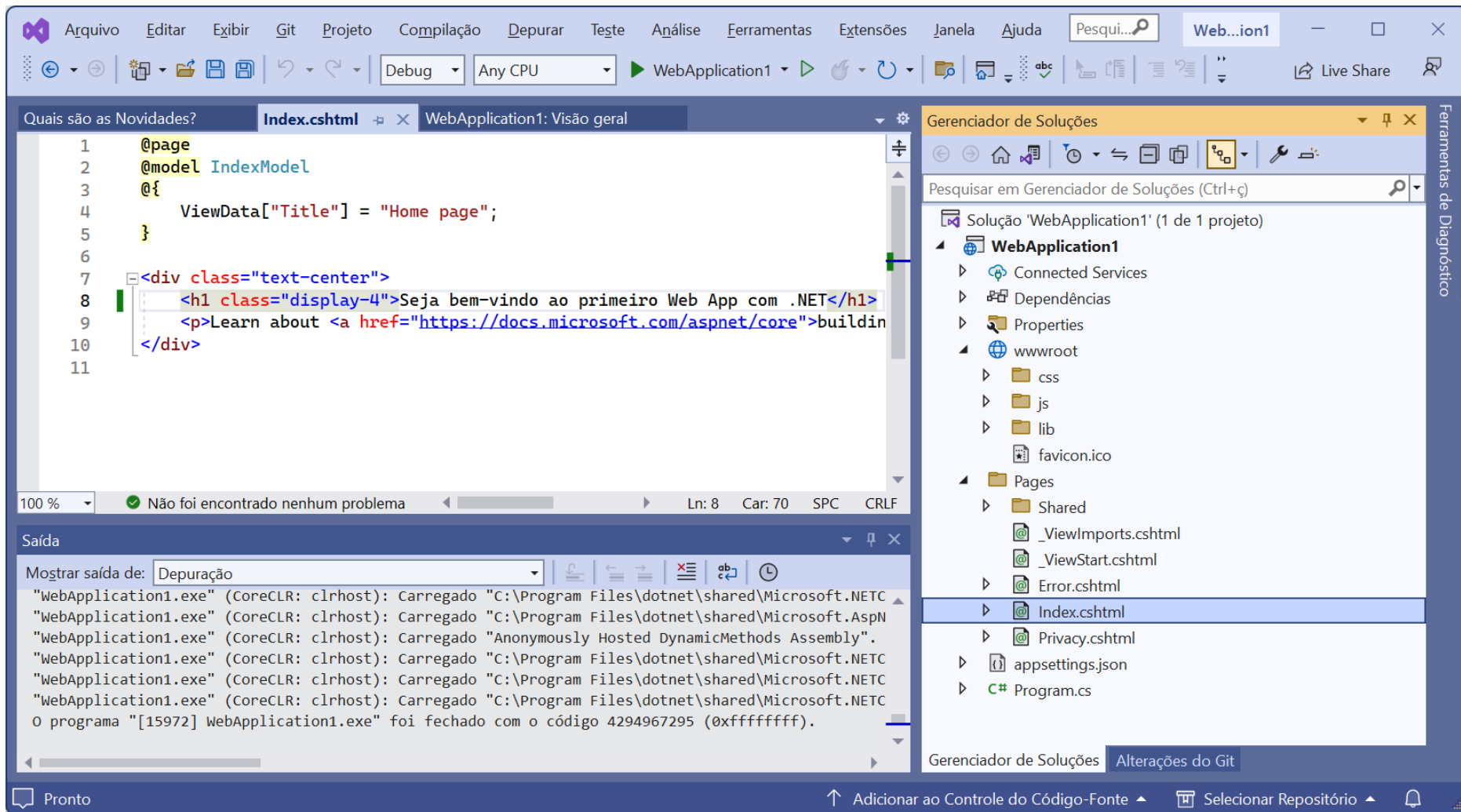
The text associated with the control.

Output

Show output from: Solution

Error List Output

Item(s) Saved



The screenshot displays the Visual Studio IDE with a .NET Web Application project named 'WebApplication1'. The main editor shows the 'Index.cshtml' file, which contains a Razor view. The view includes a page directive, a model declaration, and a layout section. The layout section contains a heading and a paragraph with a link to the ASP.NET Core documentation.

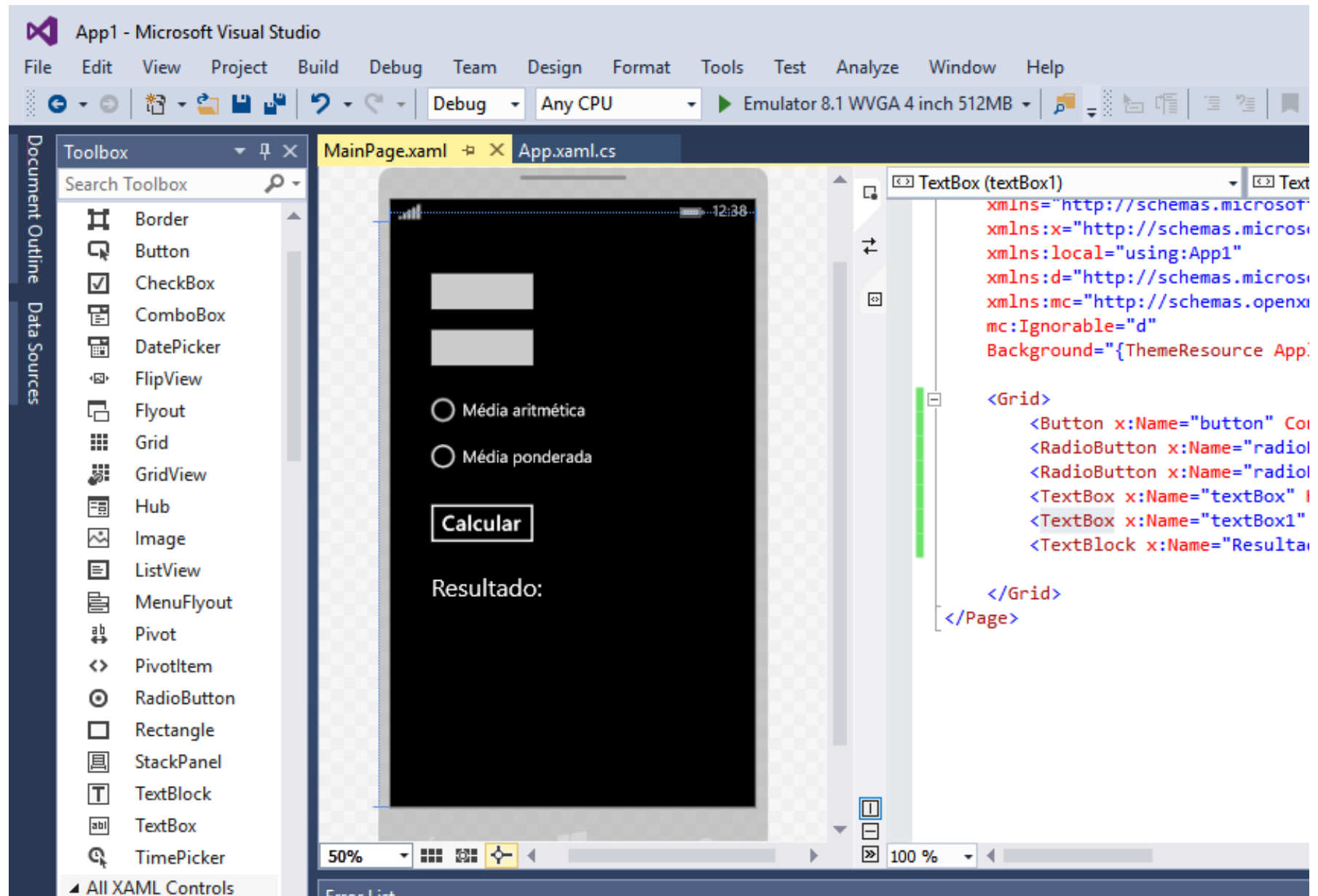
```
1 @page
2 @model IndexModel
3 @{
4     ViewData["Title"] = "Home page";
5 }
6
7 <div class="text-center">
8     <h1 class="display-4">Seja bem-vindo ao primeiro Web App com .NET</h1>
9     <p>Learn about <a href="https://docs.microsoft.com/aspnet/core">buildin</a>
10 </div>
```

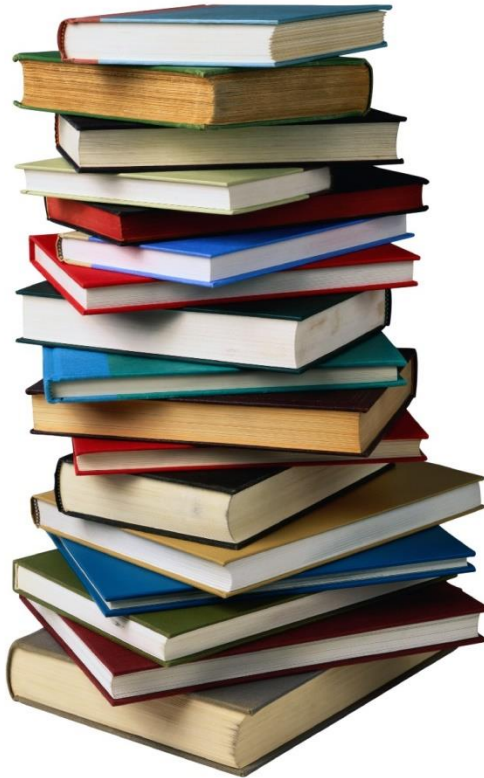
The Solution Explorer on the right shows the project structure, including folders for 'css', 'js', 'lib', 'Pages', and 'Shared', and files for '_ViewImports.cshtml', '_ViewStart.cshtml', 'Error.cshtml', 'Index.cshtml', 'Privacy.cshtml', 'appsettings.json', and 'Program.cs'.

The Output window at the bottom shows the debug output, indicating that the application was successfully built and executed.

Gerenciador de Soluções

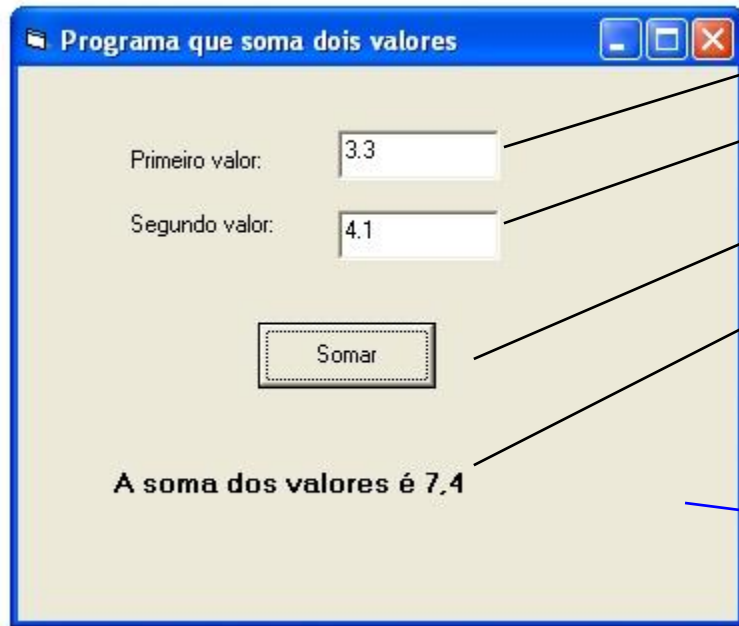
Alterações do Git





Pré-requisitos

Os conceitos da Programação Orientada a Objetos (**POO**), a programação visual na elaboração de aplicativos com interfaces gráficas, Bancos de Dados, projeto de interfaces e outros serão úteis no desenvolvimento com a plataforma .NET.



objetos que fazem parte da interface desta janela

Objetos criados utilizando as **classes** Button, TextBox, Label etc.

Painel de conteúdo ou área interna da janela

Esta janela **MeuForm** foi criada como uma **classe derivada da classe Form** (System.Windows.Forms.Form)

Uma janela com interface gráfica e os conceitos da POO: herança e composição.

Public Class **MeuForm**
Inherits System.Windows.Forms.**Form**

Herança

‘Declaração dos objetos componentes da janela (composição):

```
Friend WithEvents TextBox1 As System.Windows.Forms.TextBox  
Friend WithEvents TextBox2 As System.Windows.Forms.TextBox  
Friend WithEvents Label1 As System.Windows.Forms.Label  
Friend WithEvents Label2 As System.Windows.Forms.Label  
Friend WithEvents Label3 As System.Windows.Forms.Label  
Friend WithEvents Button1 As System.Windows.Forms.Button
```

.....
‘Como seria criado ou instanciado um objeto:
Me.Button1 = New System.Windows.Forms.Button()

End Class

Composição

Programação visual: o IDE (por exemplo, o Visual Studio .NET) gera programação em forma automática, completamente **orientada a objetos (OO)**

Código (exemplo Visual Basic.NET) gerado automaticamente
(programação visual) pelo Visual Studio .NET

Download do Visual Studio Community 2022

<https://visualstudio.microsoft.com/pt-br/vs/features/net-development>

Introdução ao Visual Studio

<https://www.visualstudio.com/pt-br/vs/getting-started/>

Veja orientações para a instalação dos softwares que utilizaremos na disciplina, no material em:

'Instalar Visual Studio, MySQL e outros softwares.pdf'

What's New for .NET 2019

<https://docs.microsoft.com/en-us/visualstudio/ide/whats-new-visual-studio-2019?view=vs-2019>

Desenvolvimento móvel multiplataforma no Visual Studio 2019

<https://docs.microsoft.com/pt-br/visualstudio/cross-platform/cross-platform-mobile-development-in-visual-studio?view=vs-2019>

Picking between Xamarin and Apache Cordova

<http://blogs.msdn.com/b/nzdev/archive/2014/12/22/picking-between-xamarin-and-apache-cordova.aspx>

Demonstração rápida utilizando o IDE Visual Studio Community 2022

Obrigado!

