

# Fundamentación de la tecnología

Primera entrega 26/6/2019

Ruta en GitLab: /Actividades/Prog01001/



Todo comenzó con Visual Basic (VB), el cual es un lenguaje de programación dirigido por eventos. Este lenguaje de programación es un dialecto de BASIC, con importantes agregados. Su primera versión fue presentada en 1991, con la intención de simplificar la programación utilizando un ambiente de desarrollo.

En 2001 Microsoft propuso abandonar el desarrollo basado en la API Win32 y pasar a un framework o marco común de librerías, independiente de la versión del sistema operativo .NET Framework, a través de Visual Basic .NET fue el sucesor de Visual Basic 6.

Ahora centrándonos en Visual Basic .NET (VB.NET), es un lenguaje de programación orientado a objetos, característica que podemos aprovechar enormemente ya que la aplicación que vamos a desarrollar se basa en una realidad en la que se presentan una gran cantidad de objetos, por lo tanto tener las herramientas que nos brinda este lenguaje con respecto a la programación orientada a objetos va a facilitar la realización de la aplicación. Se considera una evolución de Visual Basic implementada sobre el framework .NET, la cual brinda nuevas características como el control estructurado de excepciones, la programación orientada a objetos, o los Genéricos, facilitando así el desarrollo de aplicaciones más avanzadas con herramientas modernas. La gran mayoría de programadores de VB.NET utilizan el entorno de desarrollo integrado Microsoft Visual Studio, en alguna de sus versiones, el cual es uno de los mejores IDE para este lenguaje ya que posee funciones como:

**Team Foundation:** un componente que forma parte integral de Windows también compatible con la compilación y ejecución de las aplicaciones futuras, así como con Servicios Web XML. Con el objetivo de proporcionar un entorno que se pueda entender la programación orientada a objetos, proporcionar un entorno de ejecución de código que pueda reducir lo máximo posible la implementación de software y los conflictos de las versiones, ofrecer un entorno de ejecución de código que incentive la ejecución segura del mismo y también de la creación por terceras personas, ofrecer al programador una experiencia entre tipos de aplicaciones muy diferentes como en la web o Windows.

**Windows Presentation Foundation:** tecnología la cual es un conjunto de tipos .NET Framework que se pueden utilizar para crear la visualización de aplicaciones cliente de Windows. WPF está compuesto de características como el lenguaje de marcado de aplicaciones extensible el cual tenemos a XAML.

**Formularios Windows Forms,** con los cuales se puede desarrollar aplicaciones gráficamente buenas y fáciles de implementar y actualizar. Además, las aplicaciones de formularios Windows Forms pueden tener acceso a los recursos en el equipo local de una manera más segura que las aplicaciones tradicionales basadas en Windows.

**ASP.NET:** se genera en las clases de programación en .NET Framework y proporciona un modelo de aplicación web y un conjunto de controles y una infraestructura que hace que la compilación de aplicaciones web resulte más sencilla.

Aunque existen otras alternativas, como SharpDevelop (que además es libre, lo que es una ventaja en cuanto a presupuesto), sus características principales son

- Incorpora un diseñador de Windows Forms
- Completado de código. Soporta el uso de la combinación de teclas Ctrl + Espacio
- Depurador incorporado
- Herramientas para "Ir a Definición", "Encontrar referencias" y "renombrado"
- Títulos para títulos y para depuración
- Conversor bidireccional entre C# y Visual Basic .NET, y unidireccional hacia Boo
- Escrito enteramente en C#

- Compilación de código directamente dentro del entorno de desarrollo integrado.
- Complementos para ILAsm y C++
- Integración con herramientas de pruebas unitarias NUnit y MbUnit
- Analizador para ensamblado FxCop
- Pre visualización de documentación XML
- Gran integración con plantillas a la hora de añadir o crear ficheros, proyectos o compiladores
- Escritura de código C#, ASP.NET, ADO.NET, XML y HTML
- Coloreado de sintaxis para los lenguajes C#, HTML, ASP, ASP.NET, VBScript, Visual Basic .NET, y XML
- Llaves inteligentes en la escritura de código
- Gestión de marcadores (favoritos)
- Soporte para plantillas de código
- Extensible mediante herramientas externas, o complementos
- Las ventajas principales que Visual Basic.Net tiene con respecto a otros lenguajes son:
- Completamente orientado a objetos.
- Controles de usuario y personalizados.
- Facilita el mantenimiento de grandes aplicaciones.
- Incremento de velocidad de respuesta del servidor.
- Mayor velocidad.
- Mayor seguridad.
- Posee una curva de aprendizaje muy rápida.
- Integra el diseño e implementación de formularios de Windows.
- Permite usar con facilidad la plataforma de los sistemas Windows, dado que tiene acceso prácticamente total a la API de Windows, incluidas librerías actuales.
- Es uno de los lenguajes de uso más extendido, por lo que resulta fácil encontrar información, documentación y fuentes para los proyectos.
- Fácilmente extensible mediante librerías DLL y componentes ActiveX de otros lenguajes.
- Posibilita añadir soporte para ejecución de scripts, VBScript o JScript, en las aplicaciones mediante Microsoft Script Control.
- Tiene acceso a la API multimedia de DirectX (versiones 7 y 8). También está disponible, de forma no oficial, un componente para trabajar con OpenGL 1.1.8
- Existe una versión, VBA, integrada en las aplicaciones de Microsoft Office, tanto Windows como Mac, que permite programar macros para extender y automatizar funcionalidades en documentos, hojas de cálculo y bases de datos (Access).
- Si bien permite desarrollar grandes y complejas aplicaciones, también provee un entorno adecuado para realizar pequeños prototipos rápidos.