I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

UNIDAD 2
EVALUACIÓN DE ENTORNOS INTEGRADOS DE DESARROLLO

En esta unidad se estudiarán cuáles son las funciones y componentes de un entorno de desarrollo y se procederá a la evaluación de algún entorno de desarrollo existente en el mercado. Se explicará el mecanismo de actualización del entorno de desarrollo, y como se editan los programas y se generan los ejecutables.

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

Contenido

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

1 CONCEPTO DE ENTORNO DE DESARROLLO

En la unidad anterior hablábamos de las fases en el proceso de desarrollo de software. Una de ellas era la fase de codificación, en la cual se hacía uso de algún lenguaje de programación para pasar todas las acciones que debía llevar a cabo la aplicación a algún lenguaje que la máquina fuera capaz de entender y ejecutar.

También se hizo alusión a herramientas de apoyo al proceso de programación. En esta unidad vamos a analizar, instalar y ejecutar estas herramientas para entender su acción y efecto.

Muchas personas aprender a programar utilizando un editor de texto simple, compilador y depurador. Pero la mayoría, finalmente, terminan haciendo uso de algún entorno de desarrollo integrado (IDE) para crear aplicaciones.

Un entorno integrado de desarrollo (IDE), es un tipo de software compuesto por un conjunto de herramientas de programación. En concreto, el IDE se compone de:

- Editor de código de programación.
- Compilador.
- Intérprete.
- Depurador.
- Interfaz gráfico de usuario (GUI, Graphical User Interface)

Los IDE proporcionan un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, PHP, Python, Java, C#, Delphi, Visual Basic, etc. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, en donde se permite utilizar el lenguaje de programación

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

en forma interactiva, sin necesidad de trabajo orientado a archivos de texto.

Los primeros entornos de desarrollo integrados nacen a principios de los años 70, y se popularizan en la década de los 90. Tienen el objetivo de ganar fiabilidad y tiempo en los proyectos de software. Proporcionan al programador una serie de componentes con la misma interfaz gráfica, con la consiguiente comodidad, aumento de eficiencia y reducción de tiempo de codificación.

Normalmente, un IDE está dedicado a un determinado lenguaje de programación. No obstante, las últimas versiones de los IDEs tienden a ser compatibles con varios lenguajes (por ejemplo, Eclipse, NetBeans, Microsoft Visual Studio...) mediante la instalación de plugins adicionales.

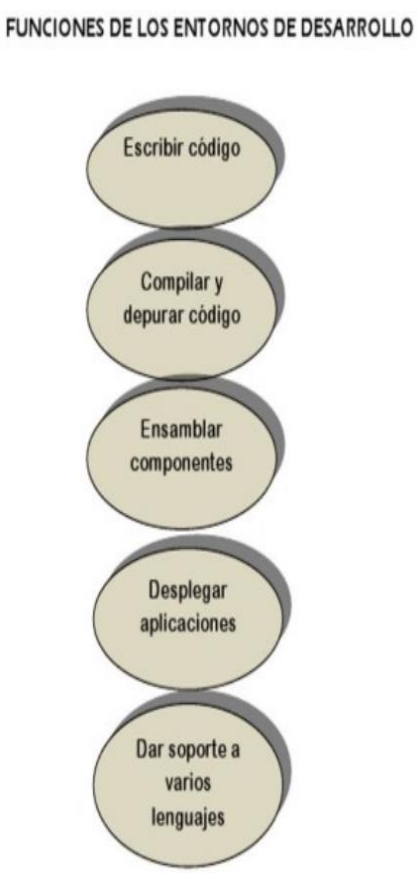
I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

2 FUNCIONES DE UN ENTORNO DE DESARROLLO

Como sabemos, los entornos de desarrollo están compuestos por una serie de herramientas software de programación, necesarias para la consecución de sus objetivos. Estas herramientas son:

- Un editor de código fuente.
- Un compilador y / o un intérprete.
- Un depurador.
- Un constructor de interfaz gráfica.

Algunas de las funciones de un IDE son:



- ✓ Editor de código avanzado:
  - Coloración de sintaxis
  - Auto-completado de código
  - Auto-indentación
  - Asistentes y utilidades de gestión y generación de código
- ✓ Compilación de proyectos complejos en un solo paso.
- ✓ Depuración de código:
  - Seguimiento e inspección de variables
  - Puntos de ruptura
  - Ejecución paso a paso
- ✓ Control de versiones:
  - Almacén de archivos compartido por todos los colaboradores de un proyecto.

- Mecanismo de auto-recuperación a un estado anterior estable al producirse un error.

✓ Generador de documentación integrado.

✓ Creación automatizada de bancos de pruebas

✓ Refactorización de código: cambios menores en el código que facilitan su legibilidad sin alterar su funcionalidad (por ejemplo, cambiar el nombre a una variable).

✓ Aumento de funcionalidades a través de la gestión de módulos y plugins.

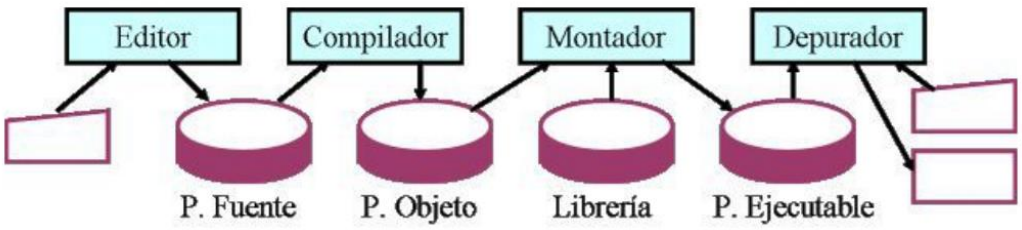
✓ Administración de las configuraciones de usuario (menús y barras de herramientas).

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

3 COMPONENTES DE UN ENTORNO DE DESARROLLO

En las primeras etapas de la informática, la preparación de programas se realizaba mediante una cadena de operaciones tales como las que se muestra en la figura para un lenguaje procesado mediante compilador. Cada una de las herramientas debía invocarse manualmente por separado. En estas condiciones no podía hablarse propiamente de un entorno de desarrollo, ya que:

- El editor es un editor de texto simple
- El compilador sólo traduce cada fichero de código fuente a código objeto
- El montador (linker / builder / loader) combina varios ficheros objeto para generar un fichero ejecutable
- El depurador maneja información en términos de lenguaje de máquina



Un entorno de desarrollo propiamente dicho combina herramientas como éstas, mejoradas e integradas. Los componentes cuya evolución ha sido más aparente son los que realizan la interacción con el usuario:

- El editor ya no es un simple editor de texto, sino que tiene una clara orientación al lenguaje de programación usado (reconoce y maneja determinados elementos sintácticos).
- El depurador no presenta información en términos del lenguaje de máquina, sino del lenguaje fuente.

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

- El editor está bien integrado con las demás herramientas, se posiciona directamente en los puntos del código fuente en los que hay errores de compilación, o que se están ejecutando con el depurador en un momento dado.

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

4 TIPOS DE ENTORNOS DE DESARROLLO

Resulta complicado establecer una clasificación de los entornos de desarrollo. Se podrían establecer clasificaciones atendiendo a las características de los mismos:

- En función del número de lenguajes soportados: multilenguaje o dedicados (monolenguaje)
- En función del número de plataformas soportadas: multiplataforma o dependientes de la plataforma.
- En función de la licencia: libres y propietarios.

El OS para el que está pensado el entorno de desarrollo

Haciendo eso de esta última característica, trataremos de clasificar los IDEs actuales, enumerando así mismo el resto de características.

4.1. ENTORNOS LIBRES

Son aquellos con licencia de uso público. No hay que pagar por ellos, y aunque los más conocidos y utilizados son Eclipse y NetBeans, hay bastantes más. En los últimos años destaca IntelliJ IDEA y Android Studio. El aspecto de la licencia del IDE que se elija para el desarrollo de un proyecto es una cuestión de vital importancia. En su elección prevalecerá la decisión de los supervisores del proyecto y de la dirección de la empresa.

IDE	Lenguajes que soporta	Sistema Operativo
NetBeans	C/C++, Java, JavaScript, PHP, Python	Windows, Linux, Mac OS X
Eclipse	Ada, C/C++, Java, JavaScript, PHP	Windows, Linux, Mac OS X
IntelliJ IDEA	Java, JavaScript, PHP, Python, Kotlin	Windows, Linux, Mac OS X
Android Studio	Java, C/C++, Kotlin	Windows, Linux, Mac OS X
Gambas	Basic	Linux
Anjuta	C/C++, Python, Javascript	Linux
Geany	C/C++, Java	Windows, Linux, Mac OS X
GNAT Studio	Fortran	Windows, Linux, Mac OS X

4.2. ENTORNOS PROPIETARIOS

Son aquellos entornos integrados de desarrollo que necesitan licencia. No son free software, hay que pagar por ellos. El más conocido y utilizado es Microsoft Visual Studio, que usa el framework .NET y es desarrollado por Microsoft.

IDE	Lenguajes que soporta	Sistema Operativo
Microsoft Visual Studio	Basic, C/C++, C#	Windows
FlashBuilder	ActionScript	Windows,Mac OS X
C++ Builder	C/C++	Windows
Turbo C++ professional	C/C++	Windows
JBuilder	Java	Windows
JCreator	Java	Windows
Xcode	C/C++, Java	Windows, Linux, Mac OS X

4.3. ENTORNOS DE DESARROLLO MÁS POPULARES

- **C++ Builder:** Software propietario. Desarrollado por Borland, apareció en la década de los 90, a partir del compilador Turbo C++.
- **Eclipse:** Software libre. Es uno de los entornos Java más utilizados a nivel profesional. El paquete básico de Eclipse se puede expandir mediante la instalación de plugins para añadir funcionalidades a medida que se vayan necesitando.
- **JBuilder:** Software propietario. Versión Java de C++ Builder. Se pueden obtener versiones de prueba o versiones simplificadas gratuitas en la web, buscando en la sección de productos y desarrollo de aplicaciones. Permite desarrollos gráficos.
- **JCreator:** Software propietario. Se pueden obtener versiones de prueba o versiones simplificadas gratuitas en la web. Este IDE está escrito en C++ y omite herramientas para desarrollos gráficos, lo cual lo hace más rápido y eficiente que otros IDEs.
- **NetBeans:** Software libre. Otro de los entornos Java muy utilizados,

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

también expandible mediante plugins. Facilita bastante el diseño gráfico asociado a aplicaciones Java.

- **Visual Studio:** Software propietario. Desarrollado por Microsoft para plataformas Windows, soporta programación en múltiples lenguajes al igual que entornos de desarrollo web como ASP.NET
- **IntelliJ IDEA:** Desarrollado por JetBrains (anteriormente conocido como IntelliJ), está disponible en dos ediciones: Community Edition (de código abierto, gratuita, orientada al desarrollo Java/Android) y Ultimate Edition (de pago, orientada al desarrollo web). Es multiplataforma, multilenguaje (Java, Python, PHP, HTML, …) y soporta de forma nativa herramientas como GIT y Subversion.
- **Android Studio:** Software Libre. Desarrollado por Google a partir de IntelliJ, reemplazó a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android. Disponible de manera gratuita a partir de la licencia Apache 2.0, soporta Java, C++ y Kotlin (nuevo lenguaje desarrollado por JetBrains sobre la máquina de Java).

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

5 FRAMEWORKS Y RTE's

5.1 FRAMEWORKS

Un framework (*plataforma, entorno, marco de trabajo del desarrollo rápido de aplicaciones*) es una estructura de ayuda al programador, en base a la cual podemos desarrollar proyectos sin partir desde cero.

Se trata de una plataforma software donde están definidos programas soporte, bibliotecas, lenguaje interpretado, etc., que ayuda a desarrollar y unir los diferentes módulos o partes de un proyecto.

Con el uso de framework podemos pasar más tiempo analizando los requerimientos del sistema y las especificaciones técnicas de nuestra aplicación, ya que la tarea laboriosa de los detalles de programación queda resuelta.

Las ventajas de utilizar un framework son evidentes:

- **Desarrollo rápido** de software.
- **Reutilización** de partes de código para otras aplicaciones
- **Diseño uniforme** del software
- **Portabilidad** de aplicaciones de un equipo a otro, ya que los bytecodes que se generan a partir del lenguaje podrán ser ejecutados en cualquier máquina virtual.

Así mismo, también existen inconvenientes:

- Gran dependencia del código respecto al framework utilizado (si cambiamos de framework, habrá que reescribir gran parte de la aplicación).

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

- La instalación del framework en nuestro equipo consume bastantes recursos del sistema.

Algunos ejemplos de Frameworks son:

- **.NET** es un framework para desarrollar aplicaciones sobre Windows. Ofrece el IDE "Visual Studio .Net" que nos da facilidades para construir aplicaciones y su motor es ".Net framework" que permite ejecutar dichas aplicaciones. Es un componente que se instala sobre el sistema operativo.

- **Spring** es un framework de código abierto para desarrollar aplicaciones para la plataforma Java.

5.2. ENTORNO DE EJECUCIÓN (RTE, RUNTIME ENVIRONMENT)

Un entorno de ejecución es un servicio de máquina virtual que sirve como base software para la ejecución de programas. En ocasiones pertenece al propio sistema operativo, pero también se puede instalar como software independiente que funcionará por debajo de la aplicación. Es decir, es un conjunto de utilidades que permiten la ejecución de programas.

Durante la ejecución, los entornos se encargarán de:

- Configurar la memoria principal disponible en el sistema
- Enlazar los archivos del programa con las bibliotecas existentes y con los subprogramas creados. Llamamos biblioteca al conjunto de subprogramas que sirven para desarrollar o comunicar componentes software, pero que ya están previamente desarrollados, y los subprogramas serán aquellos componentes que hemos creado a propósito para el programa.
- Depurar los programas: comprobar la existencia (o no) de errores

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

semánticos del lenguaje (los sintácticos ya se detectaron en la compilación).

El Entorno de Ejecución está formado por la máquina virtual y las API's (bibliotecas de clases estándar) necesarias para que la aplicación, escrita en algún lenguaje de programación, pueda ser ejecutada. Estos dos componentes se suelen distribuir conjuntamente, porque necesitan ser compatibles entre sí.

El entorno funciona como intermediario entre el lenguaje fuente y el sistema operativo, y consigue ejecutar aplicaciones. Sin embargo, si lo que queremos es desarrollar nuevas aplicaciones, no es suficiente con el entorno de ejecución. Para desarrollar aplicaciones necesitamos un IDE.

6 .NET FRAMEWORK

.NET es una plataforma desarrollada por Microsoft que permite escribir programas de forma sencilla, e incluso permite combinar código escrito en diferentes lenguajes, de forma que se puedan ejecutar de forma nativa en cualquier sistema operativo. El sistema .NET incluye herramientas, lenguajes de programación y bibliotecas para el desarrollo de software de alto rendimiento.

Algunas de las tareas que realiza son:

- Traducir el código del lenguaje de programación en instrucciones que el dispositivo de comunicación pueda procesar.
- Definir un conjunto de tipos de datos básicos para almacenar información como texto, números, etc.

Existen varias implementaciones de .NET. Cada implementación permite que el código .NET se ejecute en diferentes sistemas operativos: Windows, Linux, macOX, Android, etc.

.NET Framework es la implementación original de .NET.

Los principales componentes de .NET Framework son:

- **Common Language Runtime (CLR)**: Es el motor de ejecución para ejecutar las aplicaciones.
- **La biblioteca de clases:** proporciona APIs para leer y escribir archivos, conectarse a bases de datos, etc. y tipos de datos como cadenas, fechas, números, etc.

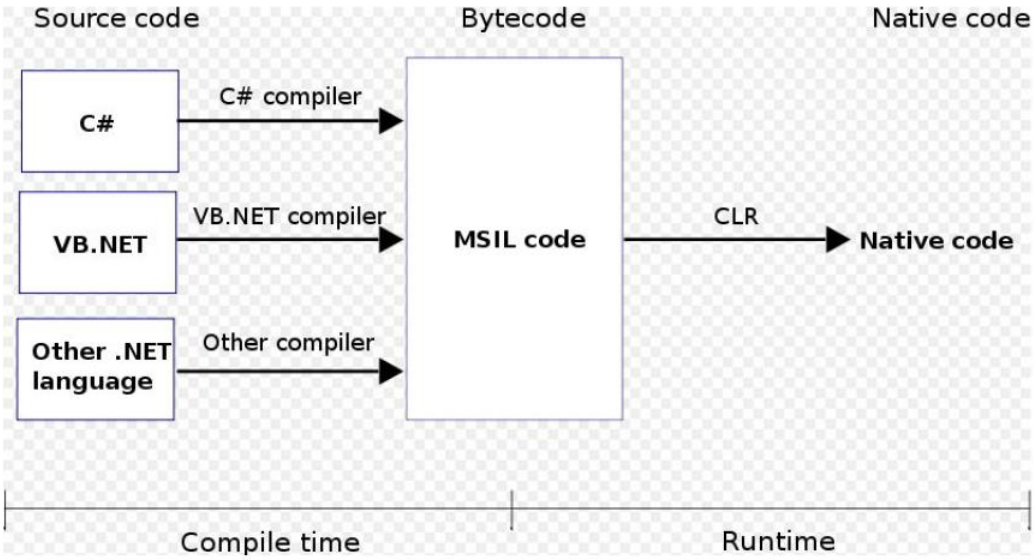
I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

Algunos de los lenguajes desarrollados para .NET Framework son: C#, Visual Basic, Delphi (Object Pascal), C++, J#, F#, Perl, Python, Fortran y Cobol.NET.

El Common Language Runtime o CLR ("entorno en tiempo de ejecución de lenguaje común") es un entorno de ejecución para los códigos de los programas que corren sobre la plataforma Microsoft .NET. El CLR es el encargado de compilar una forma de código intermedio llamada Common Intermediate Language (CIL, anteriormente conocido como MSIL, por Microsoft Intermediate Language), al código de maquina nativo, mediante un compilador en tiempo de ejecución.

Los desarrolladores que usan CLR escriben el código fuente en un lenguaje compatible con .NET, como C# o Visual Basic .NET. En tiempo de compilación, un compilador .NET convierte el código fuente a CIL. Para generarlo, el compilador se basa en la especificación CLS (Common Language Specification) que determina las reglas necesarias para crear el código MSIL compatible con el CLR. En tiempo de ejecución, el compilador del CLR (JIT -Just In Time-) convierte el código CIL en código máquina nativo para el sistema operativo. De esta forma se consigue con .NET independencia de la plataforma de hardware. La compilación JIT la realiza el CLR a medida que el programa invoca métodos. El código ejecutable obtenido se almacena en la memoria caché del ordenador, siendo recompilado de nuevo sólo en el caso de producirse algún cambio en el código fuente.

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es



La especificación Common Language Specification (CLS) define los mínimos estándares que deben satisfacer los lenguajes y desarrolladores si desean que sus componentes y aplicaciones sean ampliamente utilizados por otros lenguajes compatibles con .NET. Además, permite a los desarrolladores crear aplicaciones como parte de un equipo que utiliza múltiples lenguajes con la seguridad de que no habrá problemas con la integración de los diferentes lenguajes.

Cuando un dispositivo cliente con la plataforma .NET lanza la aplicación en Visual C# .NET, se ejecuta en el lenguaje máquina del sistema cliente y puede integrarse totalmente e interactuar con otras aplicaciones y servicios basados en .NET independientemente del lenguaje en el que hayan sido desarrollados.

La Biblioteca de Clases Base (BCL por sus siglas en inglés) maneja la mayoría de las operaciones básicas que se encuentran involucradas en el desarrollo de aplicaciones, incluyendo entre otras:

- Interacción con los dispositivos periféricos
- Manejo de datos (ADO.NET)
- Administración de memoria

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu, s/n
☎ 965936520 • 965936521
03008 ALICANTE
iesmarenostrum.edu.gva.es
03012736@edu.gva.es

- Administración de componentes Web que corren tanto en el servidor como en el cliente (ASP.NET)
- Manejo y administración de excepciones
- Manejo del sistema de ventanas
- Herramientas de despliegue de gráficos (GDI+)
- Interacción con otras aplicaciones
- Manejo de cadenas de caracteres y expresiones regulares
- Operaciones aritméticas
- Manipulación de fechas, zonas horarias y periodos de tiempo
- Manipulación de archivos de imágenes
- Generación de código
- Manejo de idiomas
- Auto descripción de código
- Compilación de código