



PRÁCTICA 1

TEMA 2

MARIO JIMÉNEZ MARSET

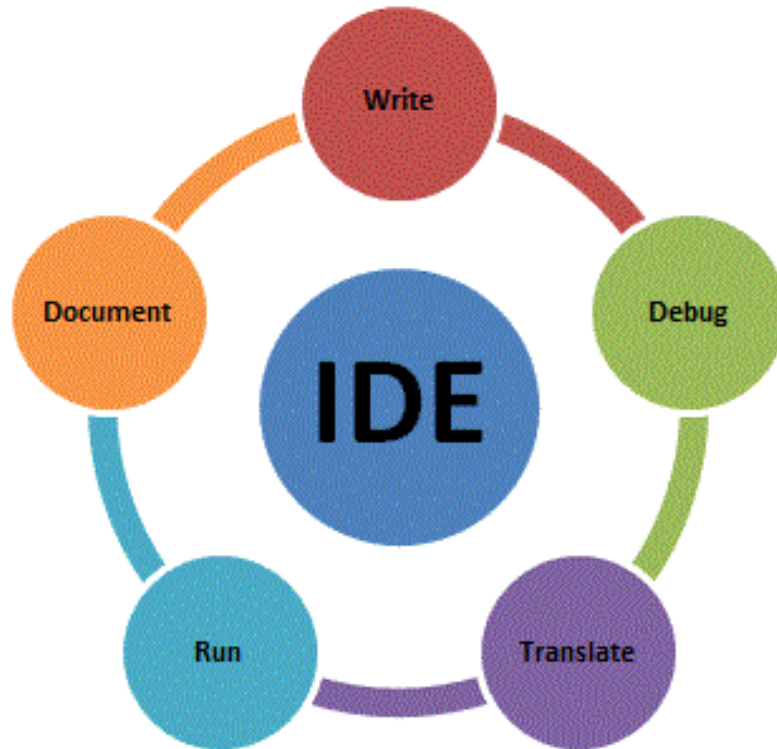
ÍNDICE

1. ¿QUÉ ES UN IDE?	3
2. ECLIPSE	4
3. NETBEANS.....	5
4. VISUAL STUDIO	7
5. WEBGRAFÍA	9

1. ¿QUÉ ES UN IDE?

Un IDE (**Integrated Development Environment**) es un programa que está compuesto por un conjunto de **herramientas** que se usan por un programador.

Los IDE se crearon para **facilitar** un programa de forma única dentro de todo el **desarrollo de un software** o sistema, lo cual conlleva al aumento de la **productividad** de los desarrolladores.



Pueden funcionar como aplicaciones por sí solas, o también formar parte de **aplicaciones existentes**. En muchos casos se dedican a un solo lenguaje de programación y otros múltiples lenguajes; permite opciones como **editar, compilar, depurar** y desarrollar interfaces gráficas de usuario.

Existen bastantes ejemplos de IDEs. Los más utilizados son **Eclipse, NetBeans y Visual Studio**.

2. ECLIPSE

El programa “**Eclipse**” es una plataforma de desarrollo, diseñada para ser extendida de forma indefinida a través de *plugins*. No es un lenguaje específico, sino que es un **IDE genérico**. Tiene mucha popularidad entre la comunidad de desarrolladores del lenguaje de programación Java, puesto que usan el *plugin* “**JDT**” que viene incluido en la distribución estándar del IDE.



CARACTERÍSTICAS:

- **Perspectivas, editores y vistas:** el trabajo está basado en las perspectivas, que no es otra cosa que una pre-configuración de ventanas y editores. Permiten trabajar en un determinado entorno de trabajo de forma **óptima**.
- **Gestión de proyectos:** el desarrollo de este programa se basa en los **proyectos**, que son el conjunto de recursos relacionados entre sí (ejemplos: código fuente, documentación, ficheros, árbol de directorios...). Se proporcionan **asistentes** y ayudas para la creación de proyectos.
- **Depurador de código:** potente depurador, de uso fácil e intuitivo, que ayuda a mejorar el código. El procedimiento a seguir es simplemente ejecutar el programa en **modo depuración**. Se presenta una nueva perspectiva específica para esta depuración de código: la **perspectiva depuración**, la cual muestra de forma ordenada toda la información necesaria para realizar una función determinada.
- **Extensa colección de *plugins*:** disponibles en gran cantidad, unos publicados por Eclipse y otros por terceros. Es una colección muy grande; los hay gratuitos, de pago, con licencias...

VENTAJAS:

1. El IDE de **Eclipse** emplea **módulos** para proporcionar toda su funcionalidad, a diferencia de otros entornos monolíticos donde las funcionalidades están todas incluidas.
2. Este mecanismo de módulos es una plataforma **ligera** para componentes de software. A Eclipse se le permite trabajar con lenguajes para **procesado de texto**, como “**LaTeX**” o sistemas de gestión de base de datos.
3. La arquitectura **plug-in** permite escribir cualquier extensión, como sería la “Gestión de la configuración”. Se provee **soporte** para Java y CVS en el SDK de Eclipse.
4. Eclipse provee al programador con “**frameworks**” ricos para el desarrollo de **aplicaciones gráficas**, la **manipulación de modelos de software**...
5. El SDK de Eclipse incluye las **herramientas de desarrollo Java**, ofreciendo un IDE con un compilador de Java interno y un modelo completo de los archivos fuente de Java, permitiendo técnicas avanzadas de refactorización y análisis de código.
6. Eclipse como tal dispone de un **editor de texto** con resaltador de sintaxis.
7. La compilación es en tiempo **real**.
8. Hay un **control de versiones** con “CVS”.

Sin embargo, tiene inconvenientes. Por ejemplo, tiene un **alto consumo de recursos** del sistema. Además, carece de mucho **soporte** para “webapps” (.war, .jsp y servlets).

3. NETBEANS

Es uno de los IDE más famosos para programar en Java. Fue creado por la empresa “**Sun Microsystems**”, en el año 2000. Es un entorno de desarrollo que permite programar tanto con Java como a través de **plugins y módulos**, permitiendo la programación con otros lenguajes como “C”, “C++”, “PHP”...

Este IDE ha ido perteneciendo a diferentes empresas, siendo la actual “**Apache Software Foundation**”.

NetBeans facilita la creación de aplicaciones “**Java Swing**”. Como programa, se puede utilizar en todas las **plataformas** para las que está disponible una máquina virtual compatible. Es un entorno de desarrollo integrado **libre**, existiendo además un número importante de módulos para extenderlo.

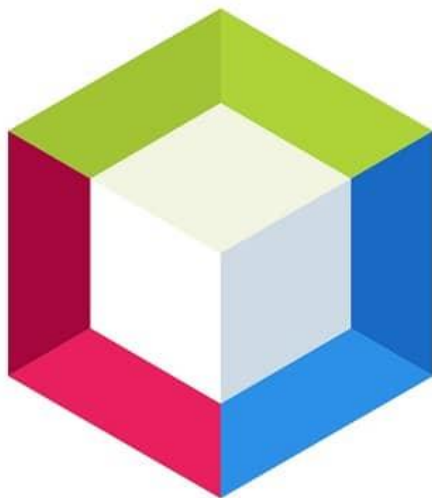
Es un proyecto de código abierto con una gran base de usuarios y una comunidad en constante crecimiento. Permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos (archivo Java que contiene clases de Java escritas para interactuar con las **APIs** de NetBeans y un archivo especial “**manifest file**”, el cual lo identifica como módulo).

Las aplicaciones construidas a partir de módulos pueden ser **extendidas** agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

Características:

- Permite la creación y administración de **interfaces de usuario**, como por ejemplo menús o barras de herramientas.
- Hay una **gestión de ventanas** (organiza el IDE a gusto del programador).
- Contiene una colección de widgets en la **biblioteca visual** de NetBeans.
- Permite la gestión de la **configuración del usuario** (adición del autor en las clases, configuración de los comentarios...).
- Hay un marco para el almacenamiento de **datos generales** (para guardar o cargar datos).
- Herramientas para el **desarrollo** de programas.

Una de las grandes ventajas que tiene NetBeans frente a otros IDE en los que se puede programar en Java es el **framework** de “Java Swing”, el cual simplifica muchísimo el desarrollo de aplicaciones con entorno gráfico (ventanas de escritorio), ya que ofrece muchas **facilidades** a la hora de crear el entorno y poder programar las funcionalidades.



Apache NetBeans

VENTAJAS:

- Es un lenguaje **multiplataforma**. El código que es escrito en Java es leído por un **intérprete**, por lo que su programa estará en cualquier plataforma.
- Cuenta con un **manejo automático** de la **memoria**, utilizando el “garbage collector”.
- Desarrolla aplicaciones web **dinámicas**.
- Sirve para desarrollar aplicaciones de **servidor** para foros en línea, encuestas, procesamientos de formularios html...
- Es **gratis**.

DESVENTAJAS:

- Su **lentitud** a la hora de ejecutar las aplicaciones.
- Requiere de un **intérprete**.
- Algunas implementaciones y librerías pueden tener código algo **rebuscado**.
- Una mala implementación de un programa en Java, pudiendo resultar muy **lento**.
- Algunas herramientas cuentan con un **coste** adicional.
- Poca existencia de **plugins** para la plataforma.
- No posee un **editor de código** html.

4. VISUAL STUDIO

Es un IDE desarrollado por Microsoft. Se trata de un conjunto de **herramientas** que permiten desarrollar aplicaciones de escritorio, aplicaciones móviles, aplicaciones web y servicios web XML. Es **compatible** con lenguajes de programación varios como C#, C++, Visual Basic.net, Java, Python, Ruby, PHP o F#.

Fue presentado por primera vez en el año 1997, lanzándose al mercado con la edición **Profesional** y la **Enterprise**. En el año 1998, lanzó la siguiente versión “**Visual Studio 6.0**”. En 2002, se lanzó “**Visual Studio.NET**”, produciéndose un gran cambio, ya que se introdujo la plataforma .net, además de los lenguajes de programación “C#” y “J#”.

En 2005, se lanzó “**Visual Studio 2005**”, comercializándose de forma online y de forma física en tiendas, incluyendo mejoras y el soporte para plataformas de arquitectura de 64 bits.

Las siguientes versiones (2008, 2010, 2012, 2013, 2015, 2017, 2019) incorporan mejoras a nivel del **framework.net**.

Visual Studio es una herramienta de desarrollo muy completa que permite escribir código de forma **sencilla**.

CARACTERÍSTICAS:

- Ofrece opciones para **depurar código**, **generar perfiles** y emitir **diagnósticos** fácilmente.
- Permite **integración de pruebas**.
- Permite acceder a un **marketplace** para instalar extensiones.
- Permite utilizar el **control de versiones** de Git para rastrear y guardar los cambios en los ficheros.
- Permite desarrollar e implementar **bases de datos** de SQL Server y Azure SQL fácilmente.
- Permite desarrollar **extensiones propias**.
- Permite crear, administrar e implementar aplicaciones de escala de **nube** en Azure.
- Permite desarrollar tanto aplicaciones **nativas** como **híbridas** para Android, iOS y Windows.

*VENTAJAS:*

- Es **multiplataforma**.
- Permite desarrollar con **múltiples** lenguajes de programación.
- Utiliza la tecnología **"IntelliSense"**.
- Utiliza gráficos de **vanguardia**.
- Tiene acceso a casi todas las **API** de Windows y puede usar la plataforma de sistemas Windows.
- Añade soporte para ejecutar **"scripts"** en las aplicaciones con VBScript o JScript, gracias a Microsoft Script Control.7.
- Al ser de Windows, integra su **diseño** e **implementación** de formularios.

DESVENTAJAS:

- En sus librerías **"runtime.dll"**, tiene problemas de versionado.
- Pese a ser una programación orientada a objetos, tiene un **soporte escaso**.
- Para crear aplicaciones multihilo, es necesario realizar llamadas a la **API** de Windows.

5. WEBGRAFÍA

- <https://lovtechnology.com/que-es-entorno-de-desarrollo-integrado-ide/>
- <https://blog.educacionit.com/2014/01/16/eclipse-ide-principales-carateristicas/>
- https://www.ecured.cu/Eclipse_entorno_de_desarrollo_integrado
- <http://evergarzadam.blogspot.com/2016/06/caracteristicas-ventajas-y-desventajas.html>
- <https://www.pcrresumen.com/menu-software/25-entornos-de-desarrollo/21-netbeans-ide>
- <https://tecnologiandroid.com/que-es-netbeans-entorno-de-desarrollo/>
- <http://javanetbeansjhjdfdfndfn.blogspot.com/2017/02/ventajas-y-desventajas.html>
- <https://conectasoftware.com/apps/visual-studio/>
- <https://fp.uoc.fje.edu/blog/visual-basic-ventajas-y-desventajas/>

9.1

Índice de comentarios

9.1 hubiera sido perfecto haber añadido algunas capturas de los diferentes IDEs