

# UD 02 - INSTALACIÓN Y USO DE ENTORNOS DE DESARROLLO

## 1. Introducción y ejemplos de IDEs

Los **entornos de desarrollo** son un tipo de software que permite desarrollar aplicaciones facilitando la labor a través de herramientas y tareas predefinidas. Los hay generales, pudiendo abarcar diversos lenguajes y tipos de aplicaciones y también específicos.

Los entornos de desarrollo se pueden clasificar:

- en función del tipo de licencia: software libre o software privado
- en función de la plataforma: multiplataforma o no
- en función del lenguaje: PHP, Java, múltiples...

### Comunes DAM / DAW

- **Netbeans**: Entorno multilenguaje más extendido y más integrado con Java.
- **Eclipse**: Multilenguaje. Primero en integrar la SDK de Android. Perdiendo protagonismo porque Oracle ha elegido Netbeans como IDE y Android apuesta por Android Studio.
- **Visual Studio**: Aplicaciones en múltiples lenguajes Node.js, Javascript, C++, PHP, .NET
- **Oracle SQL Developer**: Desarrollo y gestión de BBDD de Oracle. Desarrollar aplicaciones PL/SQL, realizar consultas a BBDD y gestionar con lenguaje SQL.
- **Monodevelop**: Proyecto de código abierto con herramientas basadas en Linux que permiten desarrollar usando tecnología .NET.

### Propios de DAM

- **Android Studio**: Por Google para aplicaciones Android
- **Xcode**: Desarrollo para macOS (macOS, iOS, watchOS, tvOS). Es gratuito pero solo se puede instalar en computadoras con macOS.
- **APPinventor**: Aplicaciones sencillas para Android
- **Unity**: Motor de videojuegos y plataforma de desarrollo. La mayoría son versiones de pago; pero existe una "Personal" para desarrollo de videojuegos propios. La mayoría de videojuegos de móvil y PC están en Unity.
- **JMonkey**: Motor de videojuegos libre en tres dimensiones. Escrito en Java, tiene su propio IDE.

### Propios de DAW

- **Atom**: IDE de código abierto lanzado a través de GitHub. Es OpenSource. Disponible para muchas plataformas. Integración con Node.js
- **PhpStorm**: IDE comercial para desarrollo de aplicaciones en PHP. Integración con frameworks como Symfony. Hay licencia gratuita de estudiantes

### Otras herramientas

- **Notepad++**: Editor de código. Ficheros en múltiples lenguajes
- **Sublime Text**: Editor de código ligero. Es de pago pero tiene versión de prueba trial.
- **Adobe Dreamweaver**: Desarrollo de páginas web usando editor WYSIWYG
- **Brackets.io**: Editor de texto open source desarrollado por Adobe. Permite ver en tiempo real en el navegador cómo quedan las modificaciones.

## Conceptos: IDE vs Framework

- IDE: Entorno de desarrollo integrado que nos facilita el proceso de desarrollo. Tiene un editor de texto, herramientas de desarrollo automático, un depurador y, si lo necesita el lenguaje, un compilador. También tiene entornos de ejecución para validar el software desarrollado (ej. Android Studio tiene terminal virtual de Android)
- Framework: Entorno de trabajo que proporciona herramientas, librerías y patrón base que facilita el proceso de desarrollo. Estructura de código base a partir del cual se puede desarrollar un código mucho más complejo usando las herramientas proporcionadas. Ej.: Angular, Symfony, Bootstrap.

## 1.1. Funciones de los IDEs

Ya se ha comentado que un IDE tiene:

- Editor de código fuente
- Compilador y/o intérprete
- Automatización de generación de herramientas
- Depurador

Las funciones de los IDEs son:

- Editor de código: coloración de la sintaxis (construcción válida de sentencias en un lenguaje)
- Autocompletado, de código: atributos y métodos de clases
- Identificación automática de código
- Herramientas de concepción visual para crear y manipular componentes visuales
- Asistentes de gestión y generación de código
- Archivos fuentes en unas carpetas y compilados en otras
- Compilación de proyectos complejos en un solo paso
- Control de versiones: Único almacén de archivos compartido por los desarrolladores con mecanismo de autorecuperación a estados anteriores estables
- Cambios de varios usuarios de forma simultánea
- Generación de documentación integrado
- Detección de errores de sintaxis en tiempo real
- Refactorización de código
- Introducir tabulaciones y espaciados
- Depuración: Seguimiento de variables, puntos de ruptura, mensajes de error
- Aumento de funcionalidades con gestión de módulos y plugins
- Administración de las interfaces de usuario (menús y barras)
- Administración de las configuraciones de usuario

## 1.2. Estructura de los IDEs

- **Editor de texto:** Resulta y colorea la sintaxis, autocompleta código, lista parámetros de funciones y métodos, inserta automáticamente paréntesis, corchetes, tabulaciones y espaciados
- **Compilador/intérprete:** Detección de errores de sintaxis en tiempo real
- **Depurador:** Botón de ejecución y traza, puntos de ruptura y seguimiento de variables. Posibilidad de depurar en servidores remotos.
- **Generador automático de herramientas:** Para visualización, creación y manipulación de componentes visuales
- **Interfaz gráfica:** Programar en varios lenguajes en un mismo IDE. Interfaz agradable con innumerables bibliotecas y plugins.

## 2. Entornos de desarrollo comunes

### 2.1. Netbeans

- Desarrollado principalmente para Java, aunque con módulos permite desarrollar en otros como PHP
- Desarrollado por Sun Microsystems y después propiedad de Oracle

- Soporta desarrollos de Java J2EE, EJB, aplicaciones móviles
- Permite instalarle módulos para generación de XML, modelado UML, soporte de PHP, C++...
- Para ser instalado necesita el JDK
- Se tienen parámetros configurables del entorno: Carpetas donde se aloja el proyecto, descripción del proyecto para mejor localización, carpetas de almacenamiento de paquetes fuente y pruebas, opciones de compilación, opciones de empaquetado, opciones de generación de documentación, opciones de combinación de teclas en teclado..... Fuentes, Bibliotecas, Generación de código: Compilación, Empaquetado, Ejecución...
- Actualización: De forma online. Hay un Auto Update Services ya incorporado. En eclipse hay un "Check for updates" para actualización automática. Los IDEs se actualizan para incluir y modificar las funcionalidades del entorno.

## Pasos adecuados:

Para generar ejecutables debemos, una vez que esté libre de errores sintácticos: COMPILAR, DEPURAR y EJECUTAR. Así se ejecutan los programas en el propio IDE.

Si quiere generarse un ejecutable: "Clean and Build", el .jar está en la carpeta **dist**

## Módulos en Netbeans

- Componente software que contiene clases de Java que pueden interactuar con las API del entorno de desarrollo y el manifest file que es un archivo especial que lo identifica como módulo.
- Los módulos pueden construirse y desarrollarse de forma independiente, lo que posibilita su reutilización y que las aplicaciones se puedan construir a través de la inserción de módulos con finalidades concretas. Una aplicación puede ser extendida con la adición de módulos nuevos que aumenten su funcionalidad.
- Existen multitud de módulos disponibles para todas las versiones de los IDEs más usados.

Pueden añadirse módulos:

- Que Netbeans instala por defecto
- Descargar un módulo de algún sitio web y añadirlo
- Instalarlo online desde el entorno.

**Adición offline:** Lo usual es añadirlos desde la web oficial de Netbeans (se descarga en formato .nbm, propio de los módulos de Netbeans). Desde nuestro IDE se cargan e instalan esos plugins

**Adición online:** Se pueden instalar on-line sin salir del IDE y sin tener que descargarlos previamente.

El módulo puede ser desactivado o desinstalado cuando ya no lo necesitamos.

## Herramientas concretas en Netbeans

- Importador de proyectos: Para trabajar en lenguajes como **JBuider**
- Servidor de aplicaciones Glassfish
- Soporte para Java EE
- NetBeans Swing GUI Builder: Creación de interfaces de usuario
- NetBeans Profiler: Eficiencia de trozo de software
- Editor WSDL: Servicios web basados en XML
- Editor XML Schema Editor. Refinar aspectos de los documentos XML
- Aseguramiento de seguridad de datos mediante Sun Java System Access Manager
- Soporte beta de UML
- Soporte bidireccional que sincroniza los modelos de desarrollo con los cambios en el código conforme avanzamos por el ciclo de vida de la aplicación.

## Funcionalidades de los módulos

- Construcción de código
- Bases de datos

- Depuradores
- Aplicaciones
- Edición
- Documentación de aplicaciones
- Interfaz gráfica de usuario
- Lenguajes de programación y bibliotecas
- Refactorización
- Aplicaciones web
- Prueba

(Las funcionalidades de Netbeans serán más o menos interesantes según la tarea a realizar y el nivel de usuario)

## 2.2. Eclipse

Plataforma de desarrollo software que dispone de IDEs para los lenguajes de programación más usados (Java, PHP, C++) y una gran comunidad de desarrolladores que respaldan el proyecto. Son de código abierto, gratuitos y multiplataforma.

Eclipse ofrece:

- Creación de nuevos proyectos (Requiere de un workspace; Al contrario que en NetBeans que son proyectos)
- Permite importación/exportación de proyectos
- Funciona con proyectos Android
- Potencial por los plugins
- Sistema de depuración del proyecto
- Uso para desarrollo de aplicaciones web
- Actualización automática
- Traducción a 47 idiomas (proyecto Babel)
- Generación de Javadoc
- Soporte JUnit

Con la opción genérica (Eclipse Neon) nos preguntará qué lenguaje vamos a usar para instalar el Eclipse IDE que mejor se adapte.

Los módulos de Eclipse se instalan con el Marketplace. También pueden buscarse plugins en otros marketplaces como RedHat y Obeo.

Algunos plugins recomendables:

- UML Designer: Modelado de objetos en POO y XML
- Ant visualization: Presentación gráfica de dependencias entre objetos Ant
- Eclipse visual editor: Plataforma para crear clases visuales y gráficas sin Eclipse. Soporta WYSIWYG.

## 2.3. Visual Studio

IDE desarrollado por Microsoft y disponible para Windows y macOS. Soporta múltiples lenguajes como C++, C#, Java, Visual Basic, .NET, PHP, Python. También hay versión Express gratuita, que luego en 2017 se ha llamado Visual Studio Community.

Desarrollar aplicaciones en Android, iOS, macOS, Windows y Web. Permite navegar por el código, escribirlo y modificarlo fácilmente.

**Visual Studio Code** es un IDE que permite escribir código de forma sencilla y en múltiples lenguajes. Se descarga desde la página oficial de visualstudio.

## 2.4. SQLDeveloper

Gestión de BBDD Oracle. Desarrollo de aplicaciones en PL/SQL. Solo disponible para Windows.

## 3. Entornos de desarrollo multiplataforma

### 3.1. Android Studio

Desarrollado por Google

Tiene:

- Edición de código inteligente
- Integración GitHub
- Desarrollo multidispositivo
- Dispositivos virtuales en todos los tamaños y orientaciones
- Generación de APK basado en Gradle
- Requisitos: 2 GB Ram mínimo, 4 GB recomendado, 400 GB espacio en disco duro, 1 GB Android SDK, 1200x800 de resolución, JDK...  
Instalar Android Studio, SDK Build Tools, Google repository, Google API.

Quizás al principio no detecte bien el SDK y haya que indicárselo.

### 3.2. XCode

Poco que decir.

Android copa cerca del 88% del mercado de dispositivos móviles. Android Studio es el IDE por excelencia en desarrollo multiplataforma.

Desarrollar aplicaciones para iOS es mucho más rentable.

### 3.3. Unity

Se puede descargar versión gratuita (pero con limitación no comercial).

Elegir al crear el nuevo proyecto si lo queremos en 2D o 3D.

### 3.4. JMonkey Engine

Motor para desarrollo de juegos 3D basado en Java. Compatible con LWJGL (Lightweight Java Game Library) y tiene comunidad de desarrolladores que lo respalda.

Tiene su propio IDE para trabajar con JMonkeyEngine. Se descarga de manera conjunta con el motor de videojuegos y las librerías de desarrollo.

## 4. Entornos de desarrollo web

Inicialmente se basaba en un bloc de notas que, con código basado en etiquetas, generaba archivo interpretado que los navegadores podían entender.

### 4.1. Aptana Studio

- Basado en Eclipse
- Funciona en Windows, macOS, GNU/Linux
- Lenguajes PHP, Python, Ruby, CSS, HTML, Javascript.
- Soporte librerías de Javascript (jQuery, Ext JS, DOJO, YUI)

### 4.2. PHP Storm

- IDE comercial para PHP
- Soporta Symfony, Drupal, Wordpress, Zend Framework, Yii, CakePHP...
- Uno de los más completos para trabajar con PHP (149 \$ al año pero con licencia de estudiante gratis)
- Soporta tecnologías frontend (HTML5, CSS, Typescript, Javascript) y permite visualizar cambios en caliente, depurarlo, hacer test de pruebas, reformatear...

## 5. Herramientas CASE para desarrollo, prueba y documentación

### 5.1. De los navegadores

Firefox, Chrome, Safari, Edge tienen herramientas para modificarlos de forma dinámica:

- Habilitar / Deshabilitar lenguajes
- Modificar CSS, HTML en tiempo real...

Safari: Inspector web "Mostrar inspector web"

Edge: F12, Menú de herramientas para desarrolladores

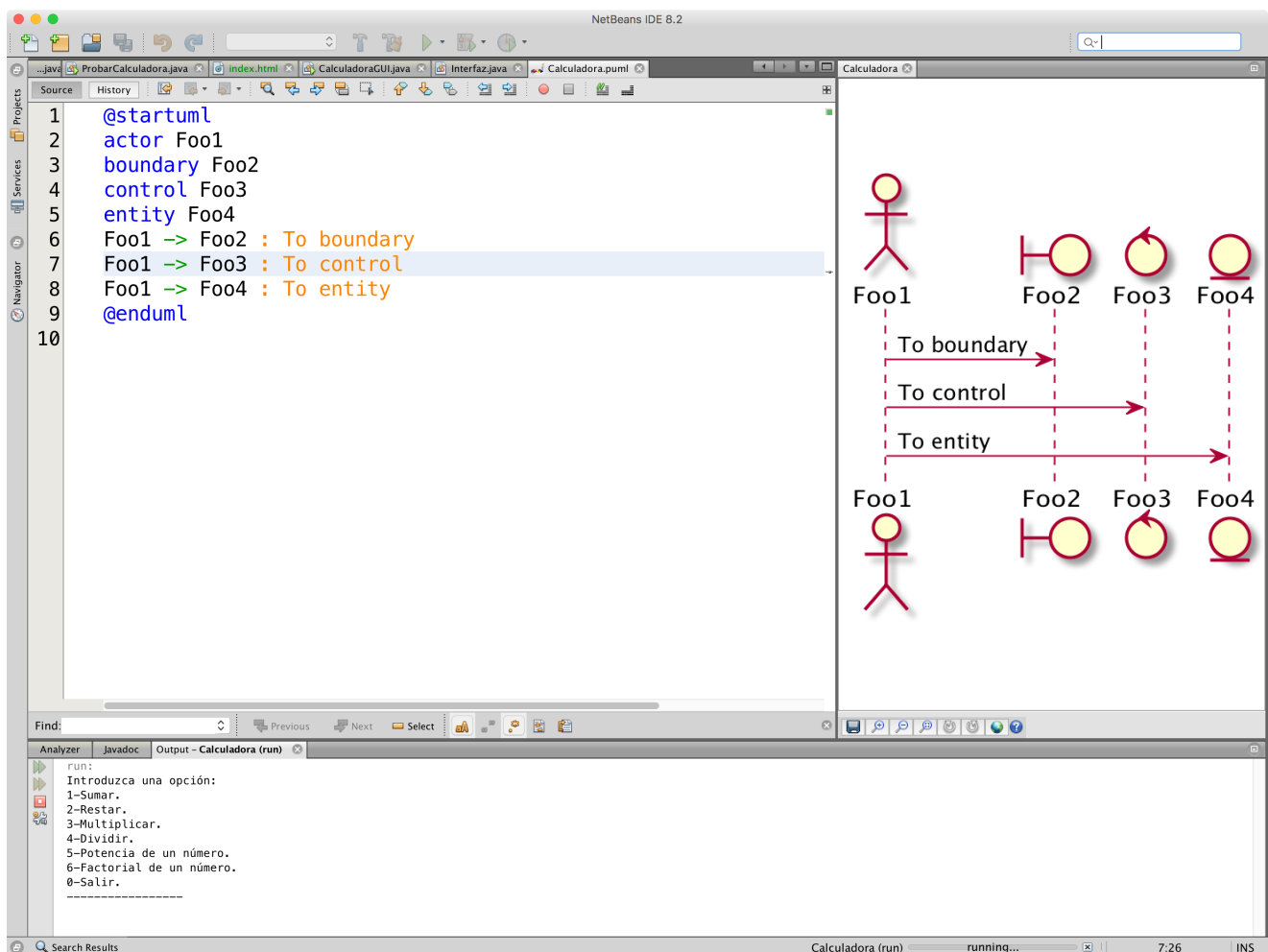
Firefox: Extensión Firebug, pero ahora tiene utilidad nativa de Desarrollador web

Chrome: Tiene también herramientas responsive Responsive Web Design Tester

### 5.2. Plugin UML para Netbeans

PlantUML se puede añadir en Netbeans.

Usando sintaxis se puede generar diagramas UML. Por ejemplo... un diagrama de comportamiento



### 5.3. Documentación de código fuente

#### JavaDoc

En Netbeans Execute -> Generate JavaDoc