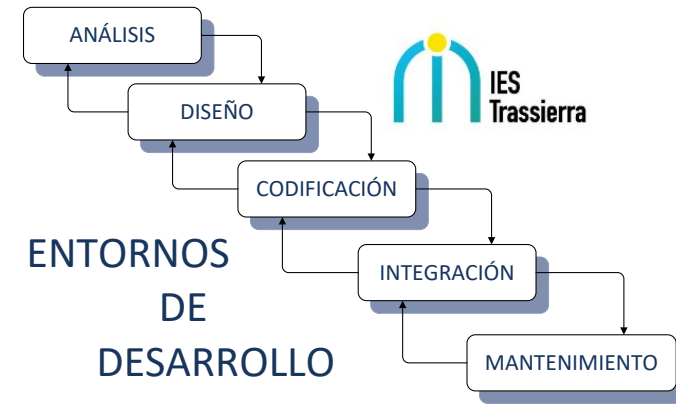




## BIBLIOGRAFÍA:

### UNIDAD 1: (TEMA DISPONIBLES EN MOODLE)

- Cabrera, G, Montoya, G : Análisis y Diseño Detallado de Aplicaciones Informáticas de Gestión. Mc Graw-Hill.
- Piattini, Calvo, Cervera, Fernández. Análisis y Diseño Detallado de Aplicaciones Informáticas de Gestión. Ra-Ma
- Pressman R.S. Ingeniería del Software. Un Enfoque Práctico. McGraw-Hill
- Sommerville, I. Ingeniería de Software. (6ª Edición), Addison Wesley.
- Larman C. UML y Patrones. (2ª Edición), Pearson Prentice Hall.
- Carlos Casado Iglesias: Entornos de Desarrollo. RA-MA
- Juan Carlos Moreno Pérez: Entornos de Desarrollo. Editorial SINTESIS
- RAMOS MARTÍN, Alicia (2014). Entornos de desarrollo. Madrid: Editorial Garceta.
- Alicia Ramos Martín, Mª Jesús Ramos Martín: Entornos de Desarrollo. Garceta grupo editorial
- Juan Carlos Moreno Pérez. Entornos de Desarrollo. Editorial SINTESIS
- Carlos Casado Iglesias. Entornos de Desarrollo. Ra-Ma



## UT 2: Entornos de desarrollo de aplicaciones multiplataforma y web

Profesora: Elena Fernández Chirino  
DEPARTAMENTO DE INFORMÁTICA

<http://www.iestrassierra.com>



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).

# RESULTADOS DE APRENDIZAJE






---

Al finalizar esta unidad valoraremos que el alumno:

- R.A.2: Evalúe entornos integrados de desarrollo analizando sus características para editar código fuente y generar ejecutables.

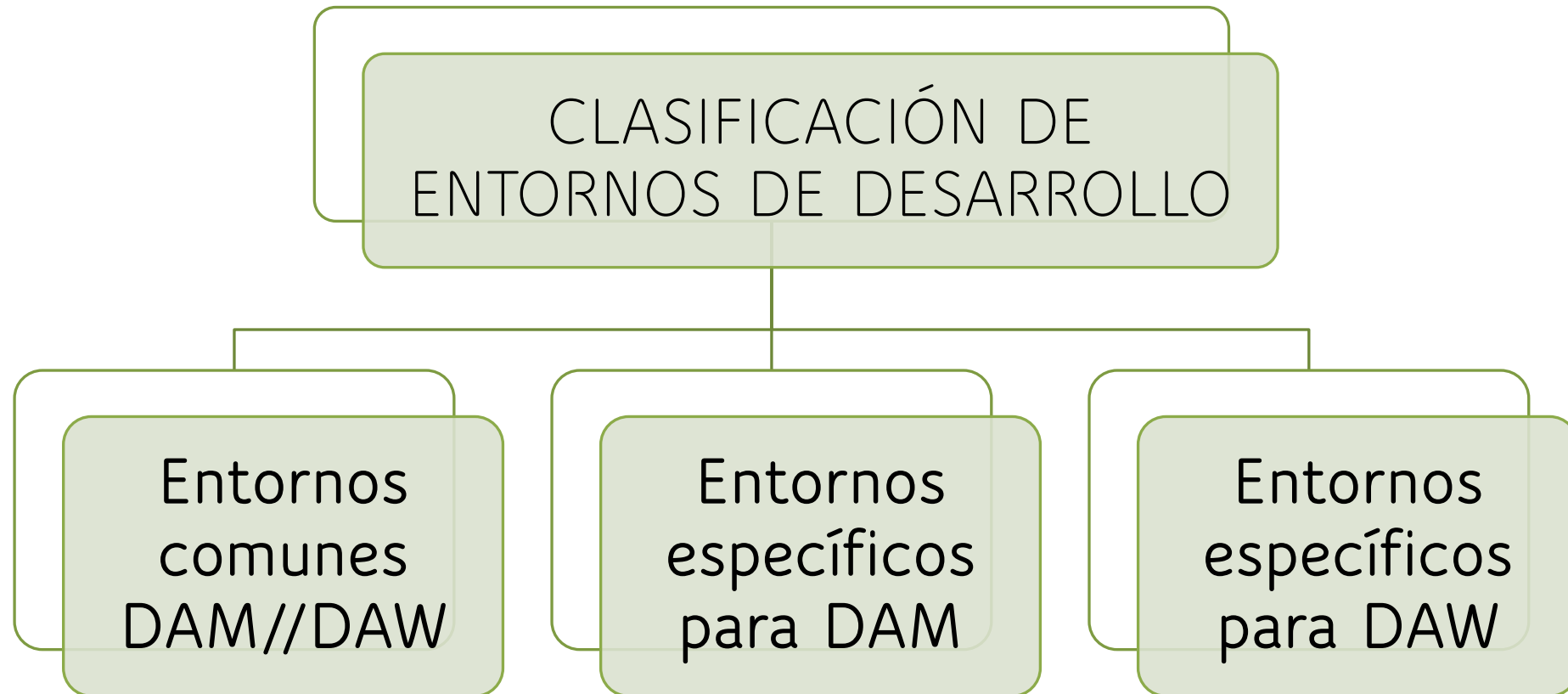


# 0.- ¿Qué aprenderemos en esta unidad?

- 
- Las características que debe tener un IDE y en cuáles nos tenemos que fijar a la hora de elegir nuestro IDE.
- 
- Haremos un recorrido global sobre las diferentes funcionalidades de los IDE.
- 
- Distinguiremos entre aquellos que permiten tanto el DAM como DAW y aquellos que están orientados exclusivamente a DAM o DAW.
- 
- Al final de la unidad deberemos ser capaces de evaluar y conocer las necesidades, de manera que sea posible escoger el IDE apropiado a cada proyecto.
- 
- Para el desarrollo de este módulo, utilizaremos Neatbeans como IDE de soporte en nuestro aprendizaje.

# 1.- Introducción a los entornos de desarrollo. Clasificación

- Los entornos de desarrollo son un tipo de software que nos permite desarrollar aplicaciones facilitando la labor a través de herramientas y tareas predefinidas.
- 



## Entornos comunes DAM//DAW

Entornos de Desarrollo	Descripción
Netbeans	Es el entorno multilenguaje más extendido y el más integrado con Java. Actualmente lo puedes descargar desde la página oficial de proyecto NetBeans.org: <a href="#">página oficial de Netbeans</a> .
Eclipse	Uno de los entornos mas populares multilenguaje y que además esta preparado para desarrollar entornos de desarrollo. Fue el primero en integrar la SDK de Android para desarrollo de aplicaciones móviles. Actualmente está perdiendo protagonismo, debido a que <b>Oracle, propietario de Java, ha elegido a Netbeans como el IDE oficial de dicho lenguaje</b> , y Android ha apostado por su propio IDE, Android Studio. <a href="#">Para descargar la última versión se puede acceder a la página oficial del proyecto.</a>
Visual Studio	Microsoft Visual Studio es un potente IDE multiplataforma que permite desarrollar aplicaciones en múltiples lenguajes y framework, tales como Node.js, javascript, C++, PHP y .NET entre otros. Podemos <a href="#">descargar Visual Studio de forma gratuita desde la página oficial del proyecto</a> .
Oracle SQL Developer	Oracle SQL Developer es un entorno de desarrollo integrado que simplifica el desarrollo y la gestión de bases de datos de Oracle. Con dicho IDE podemos desarrollar aplicaciones PL/SQL, realizar consultas a una base de datos y gestionar nuestras bases de datos a través del lenguaje SQL. <a href="#">El IDE lo podemos descargar desde la página oficial de Oracle.</a>
Monodevelop	Mono es un proyecto de código abierto en el que se desarrollan herramientas basadas en Linux que permiten desarrollar utilizando la tecnología .NET. Lo puedes descargar desde <a href="#">aquí</a> . Además, desde <a href="#">aquí</a> puedes descargar el IDE Monodevelop para Mono.

# Entornos comunes DAM

Entornos de Desarrollo	Descripción
Android Studio	El entorno de desarrollo implementado por Google para aplicaciones Android. Actualmente es la referencia para el desarrollo de este tipo de aplicaciones. Para más información ir a la <a href="#">página web oficial</a> .
XCode	XCode es el entorno de desarrollo para macOS y contiene las herramientas necesarias para desarrollar software para los distintos sistemas operativos de la marca Apple (macOS, iOS, watchOS y tvOS). Aunque es gratuito, solo puede instalarse en computadoras con sistema operativo macOS. Para más información <a href="#">podemos consultar la web oficial del proyecto</a> .
APPinventor	Permite el desarrollo de aplicaciones sencillas para Android con un entorno muy intuitivo. <a href="#">Podemos encontrar más información así como interesantes tutoriales en su página oficial</a> .
Unity	Unity es, además de un motor de videojuegos, una plataforma de desarrollo. Aunque la mayoría de versiones son de pago, podemos utilizar la "Personal" para poder desarrollar nuestros propios videojuegos. La mayoría de juegos que se desarrollan hoy en día para móviles y para escritorio, están desarrollados bajo Unity. <a href="#">En la página oficial podemos encontrar tutoriales, además de descargarnos el software necesario para comenzar a desarrollar en Unity</a> .
JMonkey	JMonkey es un motor de videojuegos libre orientado al desarrollo de juegos en tres dimensiones. Está escrito en Java y tiene su propio IDE de desarrollo. Lo podemos descargar desde <a href="#">aquí</a> .

## Entornos de Desarrollo de Aplicaciones Web

Entornos de Desarrollo	Descripción
Atom	Es un IDE de código abierto que fue lanzado a través de la plataforma GitHub. Al tratarse de una herramienta OpenSource (Código abierto) está provocando que muchos programadores lo estén mejorando y que obtenga el apoyo de una importante comunidad de desarrolladores. Esta disponible para la mayoría de las plataformas (Windows, macOS, GNU/Linux) es fácil de configurar y personalizar y facilita la integración con el framework Node.js. <a href="#">En la página oficial del proyecto podemos encontrar más información sobre el mismo, así como descargarlo.</a>
PhpStorm	PhpStorm es un IDE comercial orientado al desarrollo de aplicaciones web en php. Una de sus principales ventajas radica en la integración con framework de desarrollos de php como Symfony. Aunque sea de pago, podemos obtener una licencia gratuita para estudiantes y profesores. <a href="#">En la página oficial de la aplicación podemos encontrar más información y solicitar nuestra licencia libre como estudiantes/profesores.</a>

## 1.1.- Funciones de un entorno de desarrollo

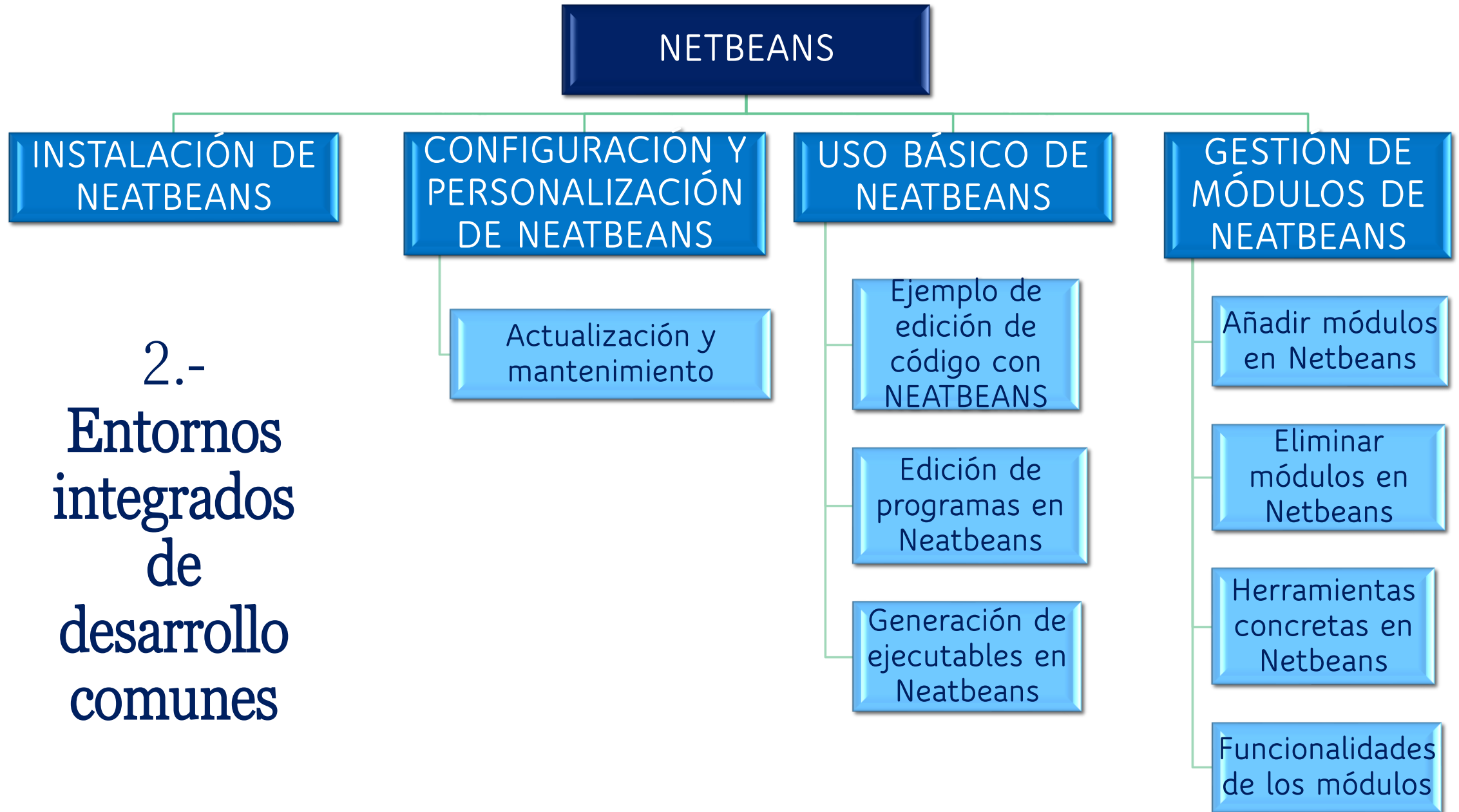


## 1.2.- Estructura de un entorno de desarrollo

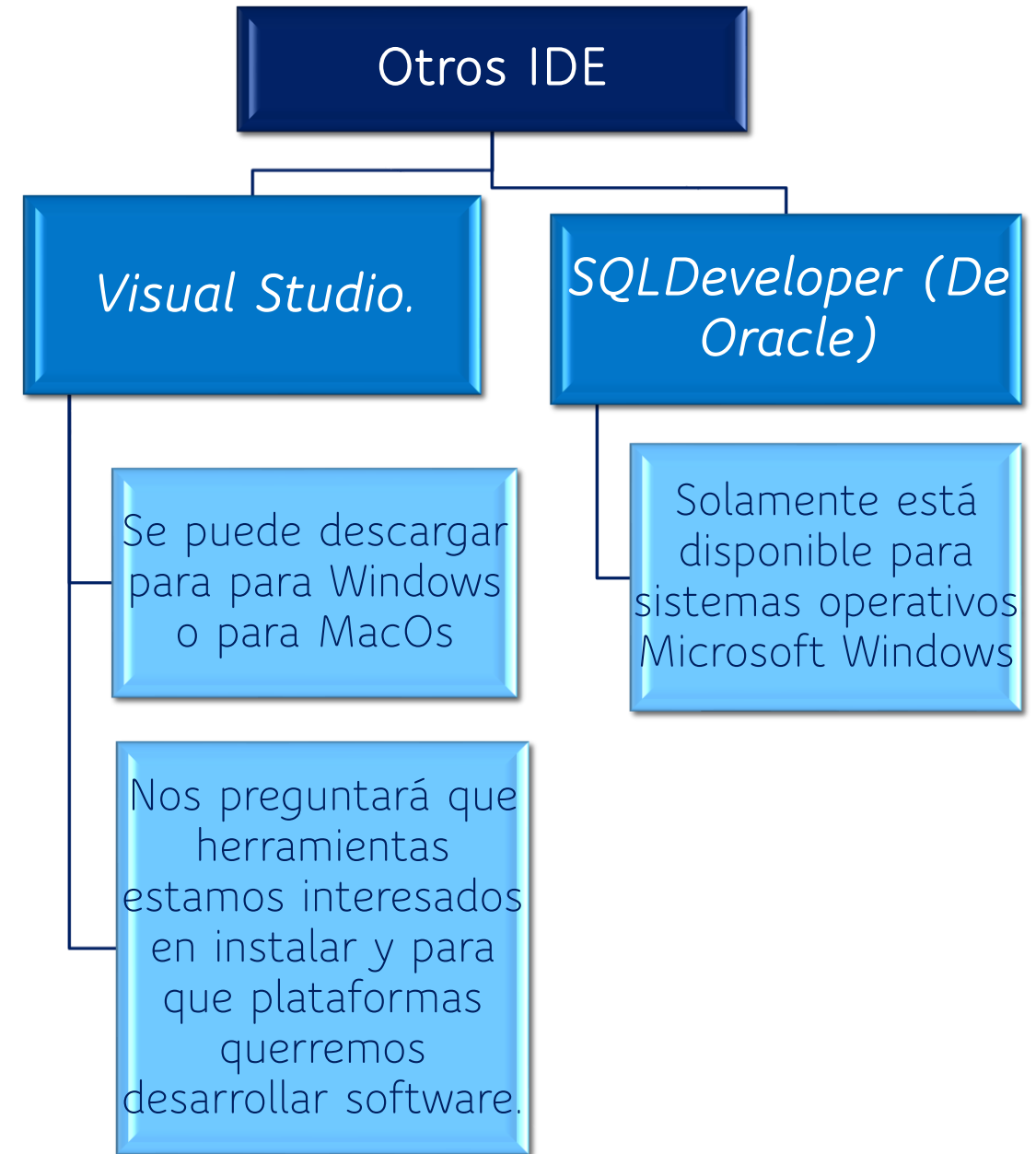
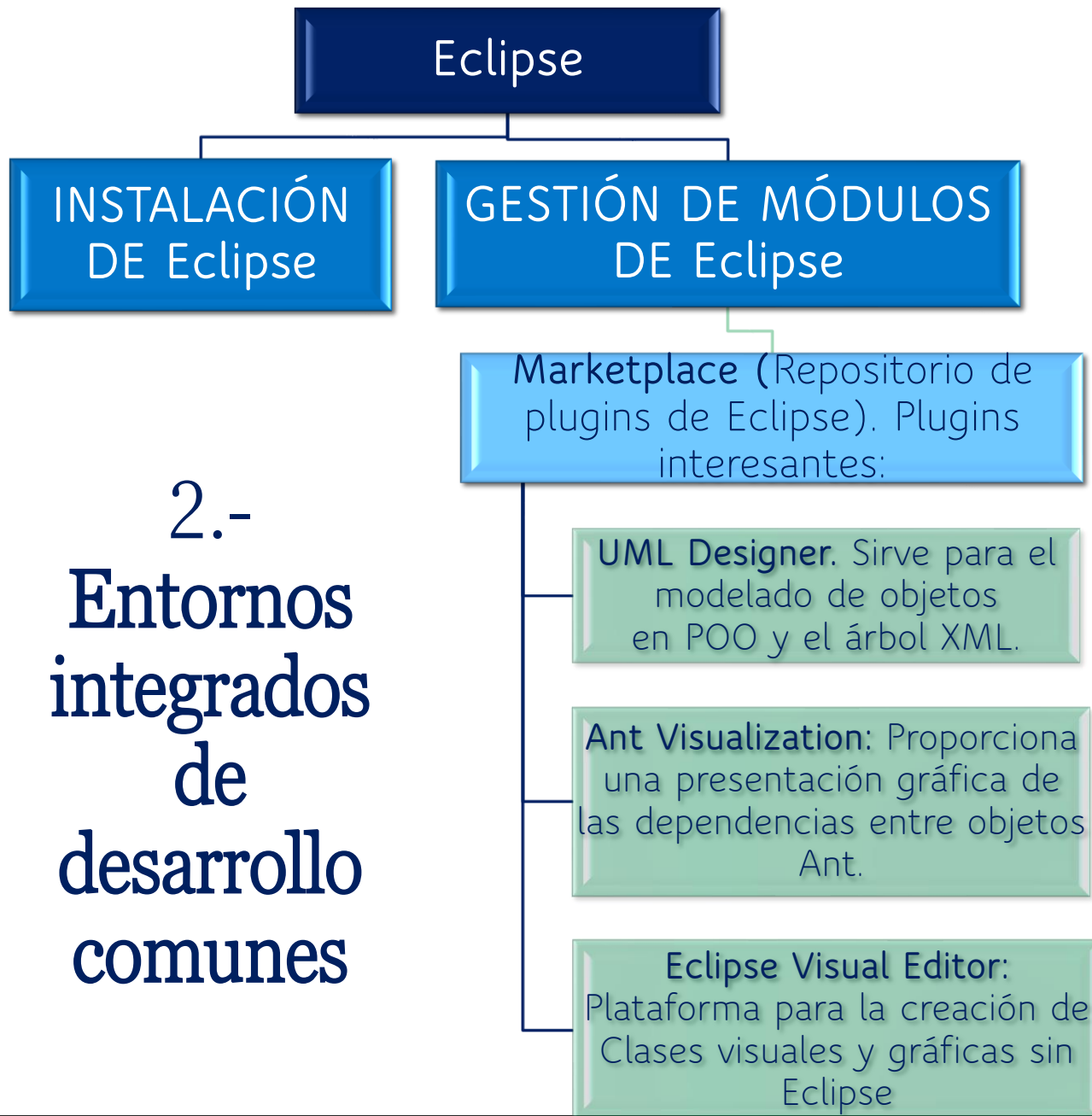




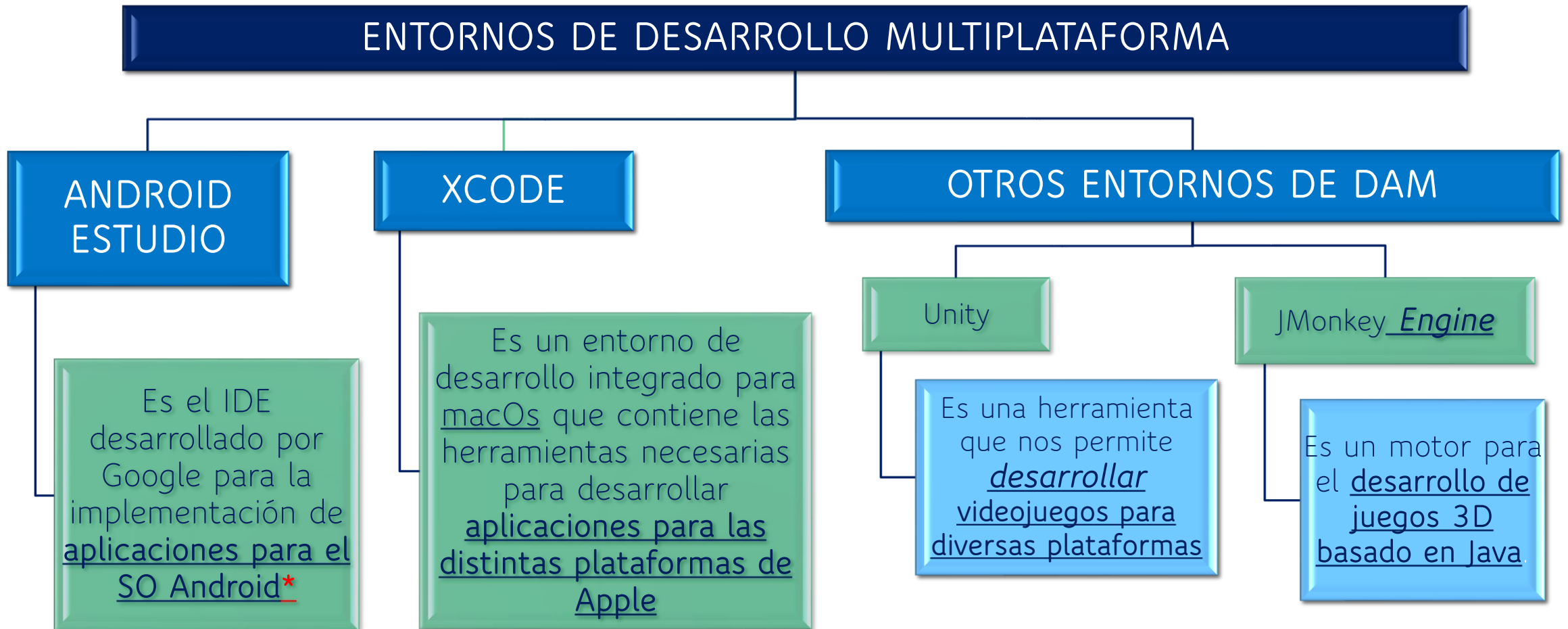
## 2.- Entornos integrados de desarrollo comunes



## 2.- Entornos integrados de desarrollo comunes



### 3.- Entornos de desarrollo multiplataforma



\* Con el editor de código podrás escribir un código más eficaz, trabajar más rápido y aumentar tu productividad, ya que te permite completar, refactorizar y analizar código de manera avanzada. A medida que escribes, Android Studio proporciona sugerencias en una lista desplegable.

## 4.- Entornos de desarrollo WEB

### ENTORNOS DE DESARROLLO WEB

```
graph TD; A[ENTORNOS DE DESARROLLO WEB] --> B[Aptana Studio]; A --> C[PHP Storm]; B --> D[Es un entorno de desarrollo integrado de software libre basado en eclipse y que funciona bajo Windows, MacOS y GNU/Linux. Soporta una amplia variedad de lenguajes de programación como PHP, Python, Ruby, CSS, HTML y JavaScript.]; C --> E[Es un IDE comercial para PHP. Soporta la mayoría de framework basados en PHP. A pesar de tratarse de software comercial,, existe la posibilidad de obtener una licencia gratuita para miembros de la comunidad educativa.];
```

#### Aptana Studio

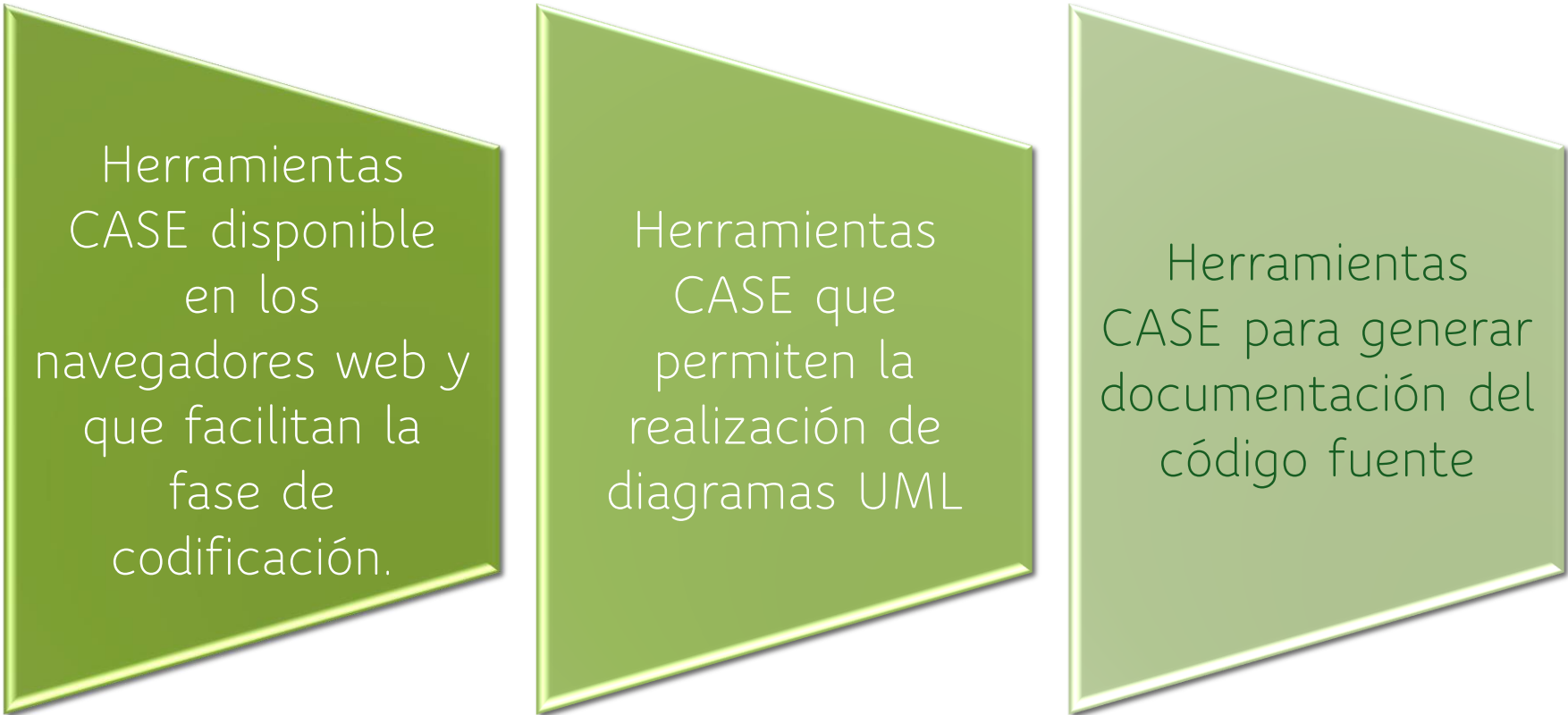
Es un entorno de desarrollo integrado de software libre basado en eclipse y que funciona bajo Windows, MacOS y GNU/Linux. Soporta una amplia variedad de lenguajes de programación como PHP, Python, Ruby, CSS, HTML y JavaScript.

#### PHP Storm

Es un IDE comercial para PHP. Soporta la mayoría de framework basados en PHP. A pesar de tratarse de software comercial,, existe la posibilidad de obtener una licencia gratuita para miembros de la comunidad educativa.

## 5.- Herramientas CASE para el desarrollo, prueba y documentación del código

Las **herramientas CASE** (Computer Aided Software Engineering) son aplicaciones software que nos facilitan el desarrollo del software, reduciendo el tiempo y el coste del mismo. Facilitan la automatización del ciclo de vida del software, completamente o en alguna de sus fases. En este apartado se estudian:



Herramientas CASE disponible en los navegadores web y que facilitan la fase de codificación.

Herramientas CASE que permiten la realización de diagramas UML

Herramientas CASE para generar documentación del código fuente

## 5.1.- Herramientas de los navegadores web.

- Actualmente, las últimas versiones de los navegadores web más populares: Mozilla Firefox, Google Chrome, Safari y Microsoft Edge, contienen herramientas para los desarrolladores web.
- A través de dichas herramientas podemos trabajar con nuestros desarrollos web de forma dinámica, permitiéndonos habilitar y deshabilitar distintos lenguajes utilizados, como JavaScript y CSS, modificar la hoja de estilos en tiempo real e incluso modificar líneas de código HTML de forma dinámica, para ver como quedaría nuestra web con dichas modificaciones y con la ventaja de que los cambios serán deshechos al recargar la página.
- Veremos las herramientas de ayuda a los desarrolladores web que contienen los navegadores web:

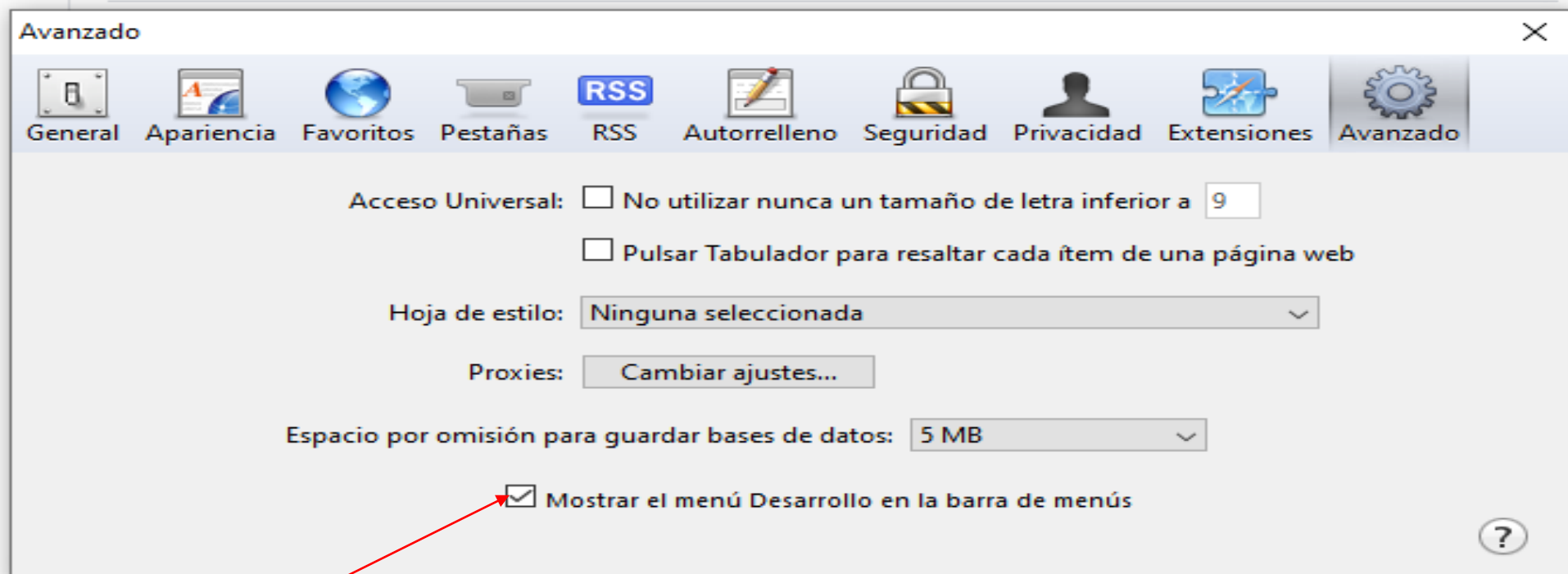


## 5.1.1.- Safari

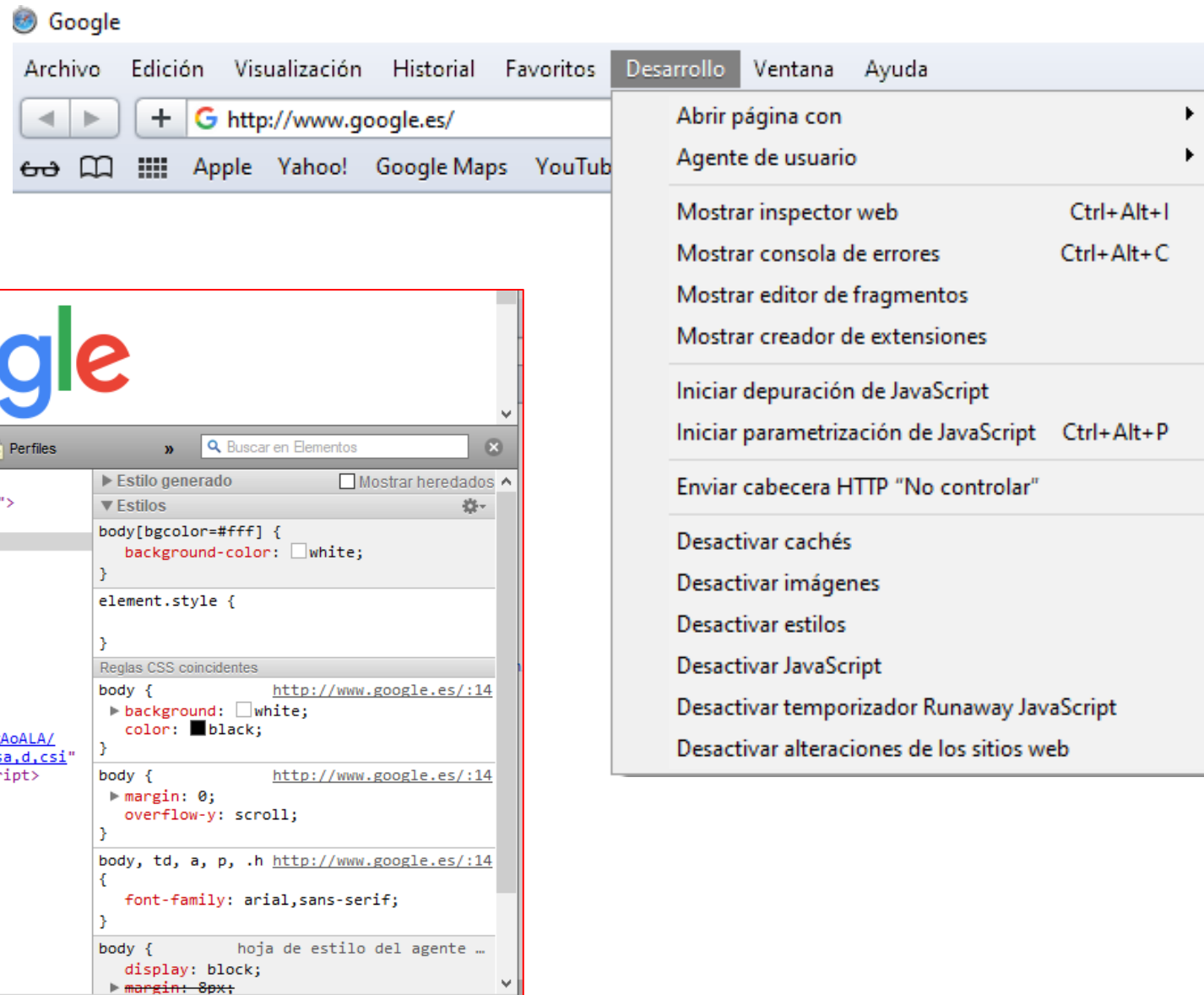
**Safari** es el navegador web desarrollado por Apple. Dispone de un amplio conjunto de herramientas para desarrolladores web.

**Los pasos para ello son los siguientes:**

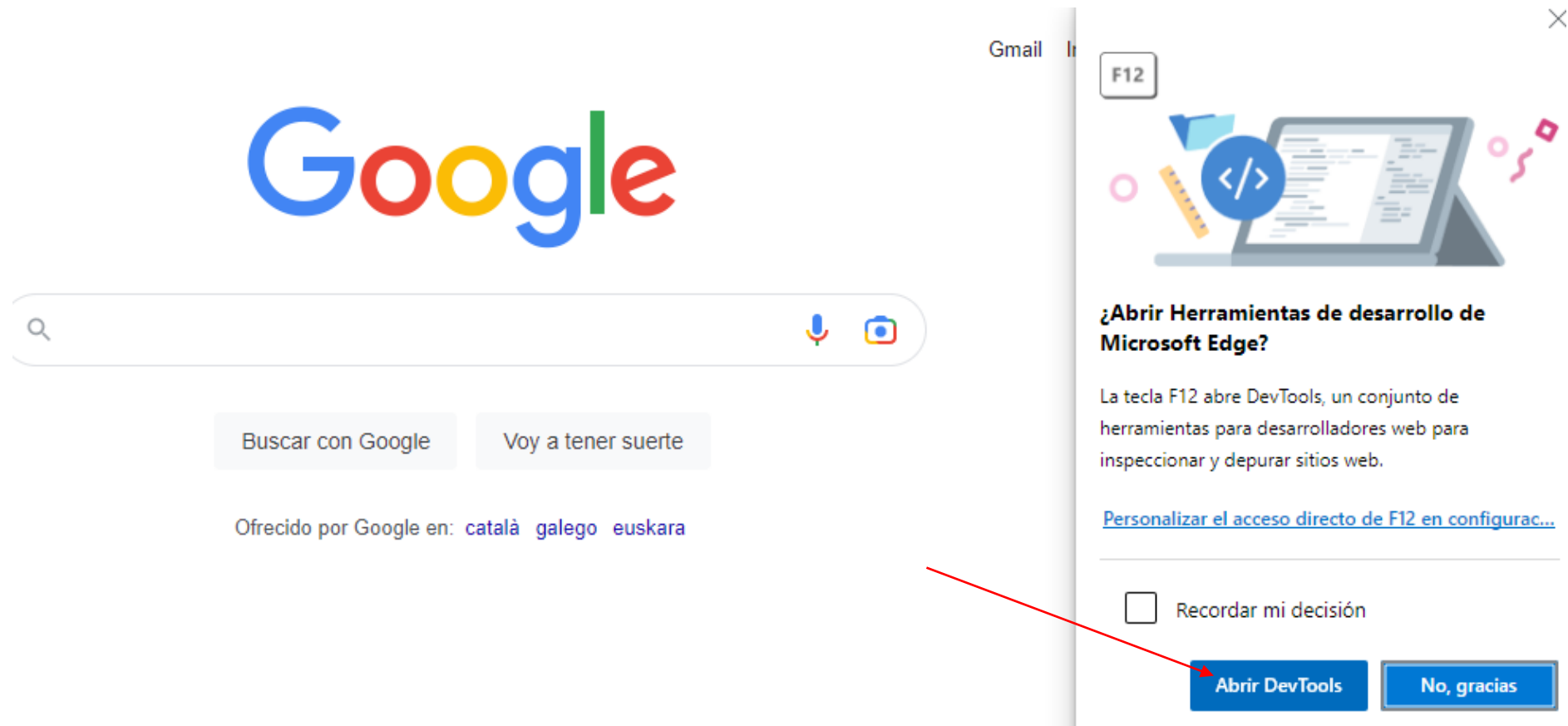
1. Abrimos la app **Safari** en nuestro Mac.
2. En el menú **Safari** elegimos Preferencias.
3. Entramos en la **pestaña** Avanzado.
4. Activamos la opción **Mostrar el menú Desarrollo** en la barra de menús.



## 5.1.1.- Safari



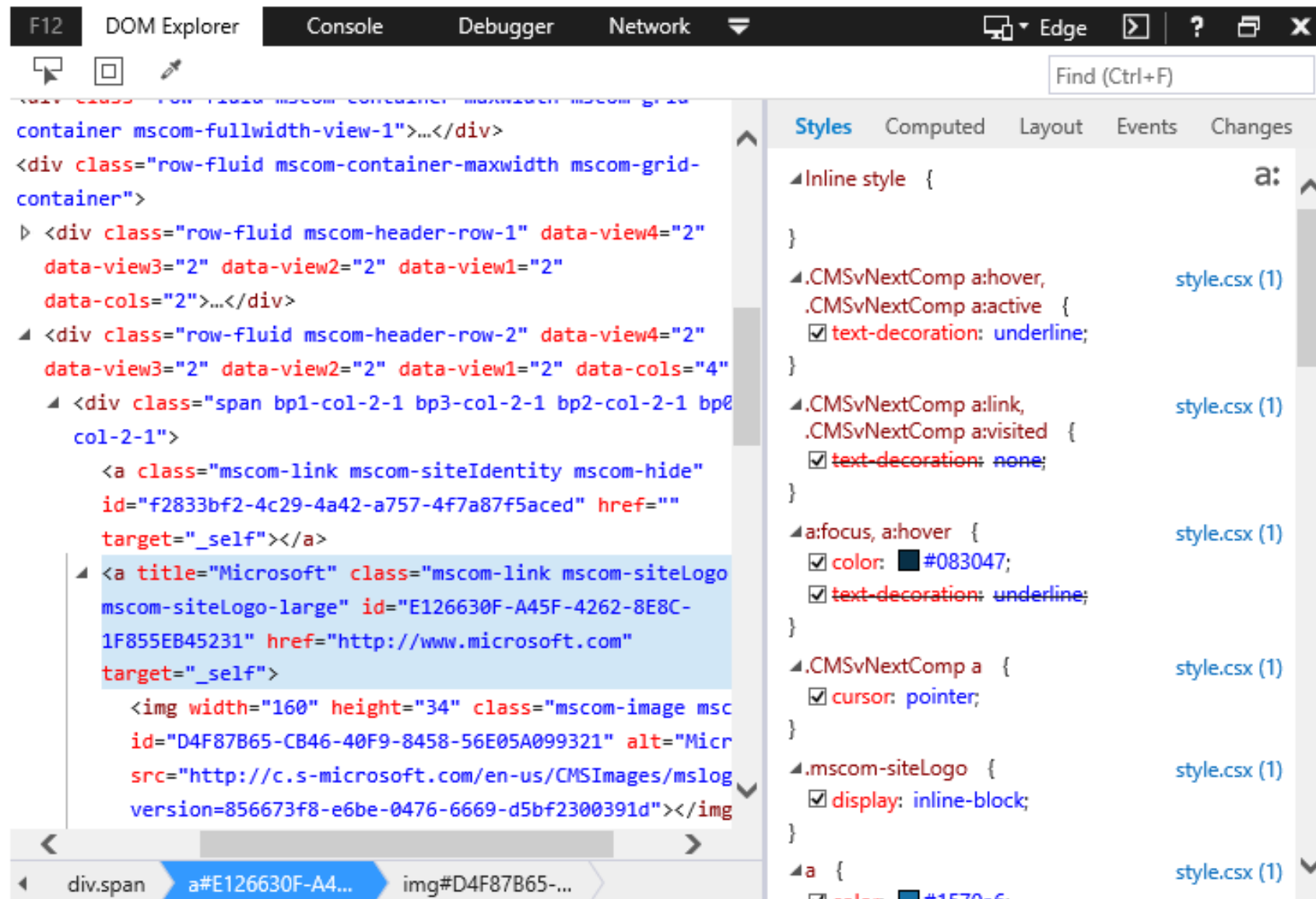




## 5.1.2.- Microsoft Edge

Es el nuevo navegador web de Microsoft Windows. Disponible a partir de la versión 10 de Windows. Para acceder a las herramientas de desarrollo basta con pulsar la tecla F12 y se abrirá el menú de herramientas para desarrolladores.

# Microsoft Edge





## 5.1.3.- Mozilla Firefox

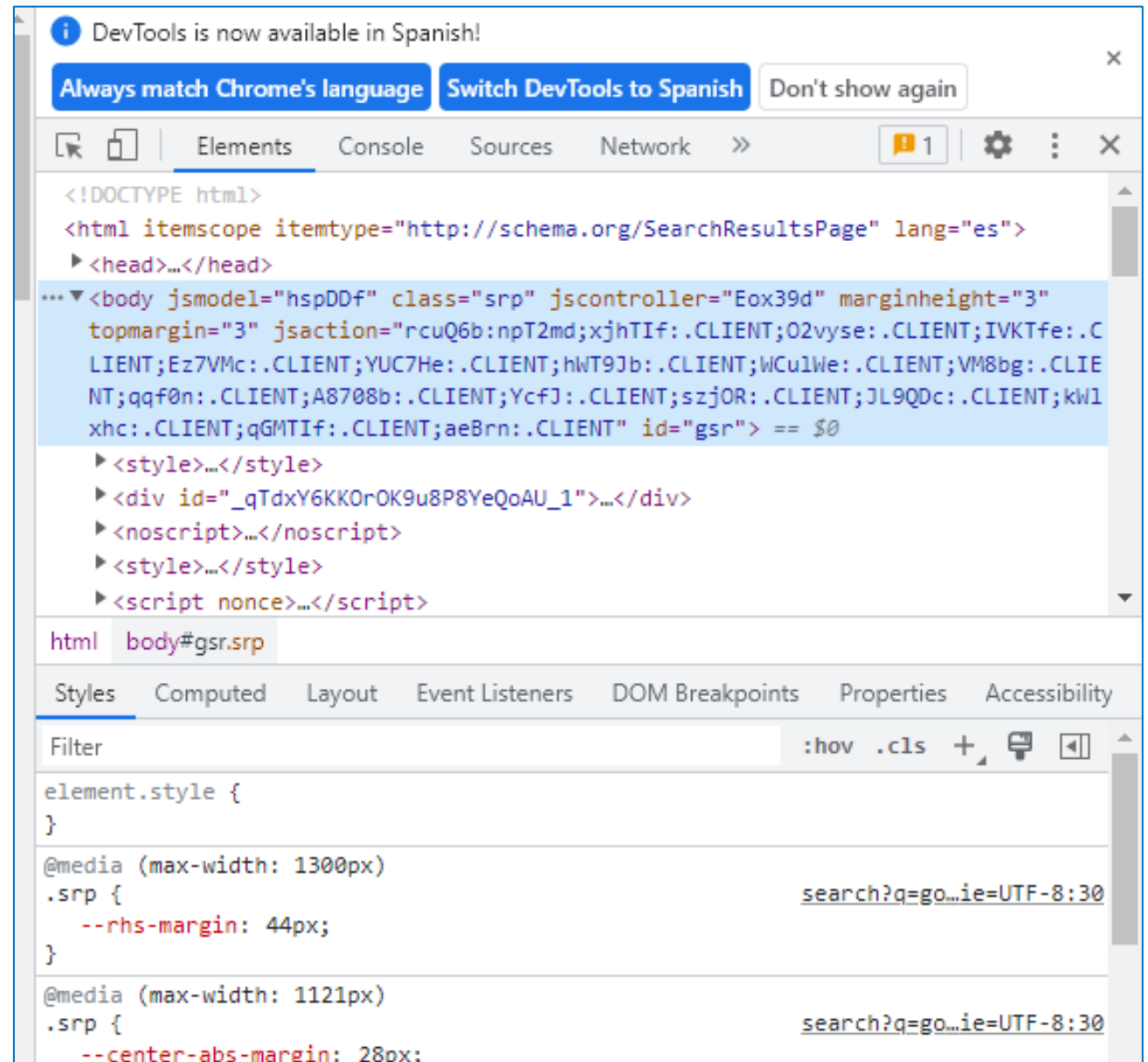
Originalmente se utilizaba como herramienta para inspeccionar el código fuente en HTML y CSS de la web la extensión Firebug. En las últimas versiones, Mozilla Firefox ya incluye como utilidad nativa herramientas para el desarrollo de la web, entre las que se encuentra un inspector de código con la misma funcionalidad que la extensión Firebug.

Se puede acceder al menú con las opciones de desarrollo accediendo a la pestaña Herramientas->Desarrollador web.

## 5.1.4.- Google Chrome

Chrome es un navegador que provee de diversas herramientas para la edición y depuración de la web en tiempo real. Esto quiere decir que podemos incluir modificaciones y visualizarlas o ejecutarlas en tiempo real.

Se accede a Herramientas ⇨ Mas herramientas ⇨ herramientas para desarrolladores



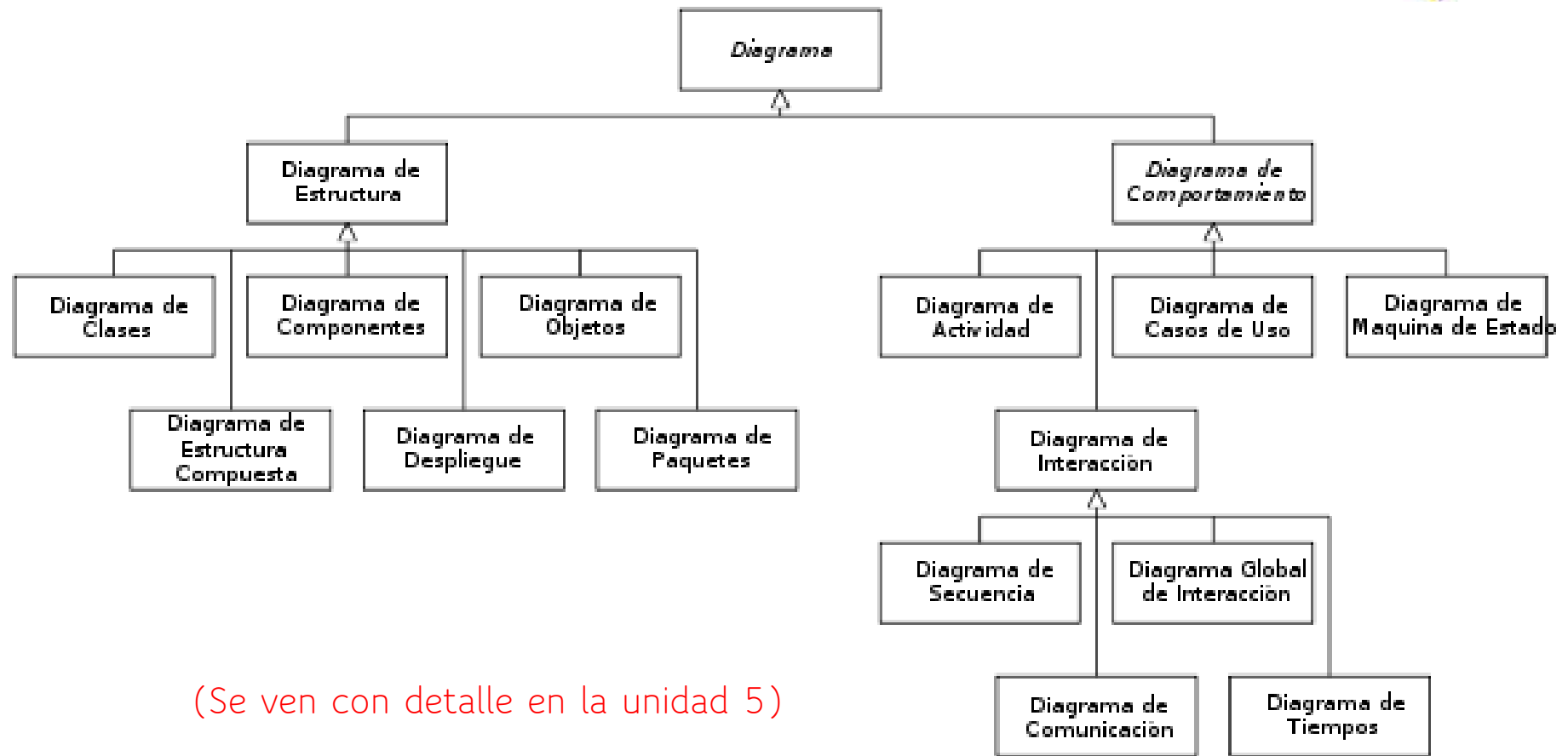
## 5.2.- Plugin UML para netbeans

NetBeans permite programación orientada a objetos como Java. Ahora bien, siendo un IDE muy bueno, debería permitirnos manejar UML para modelar nuestros objetos ¿cierto? Pues bien, se puede instalar un plugin en NetBeans que nos permita crear UML.



### ¿Qué es UML?

UML (Lenguaje de modelado unificado), es un lenguaje de modelado de sistemas orientados a objetos. Este lenguaje permite construir una serie de modelos, a través de diagramas de diferentes visiones de un proyecto.



(Se ven con detalle en la unidad 5)

## 5.3.- Software para generar documentación del código fuente.

- Uno de los principales objetivos que debe de satisfacer el software desarrollado es su reutilización. Uno de los axiomas principales radica en no reinventar la rueda. Para ello es fundamental reutilizar el software que ya hemos desarrollado, o bien nosotros u otros programadores, siempre y cuando su licencia lo permita.
- Por este motivo es muy importante desarrollar código fuente bien documentado. Además de la documentación propia del código fuente, es importante que exista una documentación de las clases y de ficheros fuentes desarrollados.
- Existen herramientas software que facilitan el proceso de generar la documentación a partir de nuestro código fuente. En este epígrafe vamos a centrarnos en JavaDoc, que permite generar la documentación del código fuente escrito en Java y está disponible en NetBeans. Decir que existen programas orientados a otros lenguajes de programación como PHPDoc.

## 5.3.- Software para generar documentación del código fuente.

JavaDoc nos permite obtener la documentación a partir de nuestro código fuente.

Para ello en el código tenemos que insertar una serie de etiquetas, las cuales vamos a ver a continuación:

- **@author** - El nombre del autor del proyecto por ejemplo Javier Mejías.
- **@version** - La versión del proyecto, por ejemplo 2.3.
- **@see** - Añade una referencia a una clase, método o enlace web.
- **@param** - Nombre de parámetro utilizado en una función indicando para que sirve.
- **@return** - El resultado devuelto por la función.
- **@exception** - Excepciones que lanza nuestro función o clase.
- **@throws** - Nombre de la excepción junto con su descripción.
- **@deprecated** - Añade una alerta al usuario de que el método que sigue a continuación ya no debe usarse y que será eliminado en versiones posteriores.

Introduciremos como comentario al comienzo de cada función y/o clase las etiquetas anteriores que nos interesen. Veamos el siguiente ejemplo.

```
    * and open the template in the editor.
    */
package javaejemplo;

/**
 *
 * @author javiermejias
 * @version - 1.0 mayo de 2017
 */
public class Usuario {
    private String login;
    private String password;

    /**
     * @return el login del usuario. Variable de tipo String.
     */
    public String getLogin() {
        return login;
    }
}
```



Una vez tengamos generado todo nuestro proyecto, tendremos que ir a la opción Ejecutar (Execute) ⇒ Generar JavaDoc (Generate JavaDoc) y de forma automática se nos generará toda la documentación de nuestro proyecto.

All Classes

JavaEjemplo

Usuario

PACKAGECLASSUSE TREE DEPRECATED INDEX HELP

PREV CLASSNEXT CLASSFRAMESNO FRAMES

SUMMARY: NESTED | FIELD | CONSTR | METHODDETAIL: FIELD | CONSTR | METHOD

javaejemplo

Class Usuario

java.lang.Object

javaejemplo.Usuario

public class Usuario

extends java.lang.Object

Constructor Summary

Constructors

Constructor and Description

Usuario()

Method Summary

All MethodsInstance MethodsConcrete Methods

Modifier and Type	Method and Description
java.lang.String	getLogin()
java.lang.String	getPassword()
void	setLogin(java.lang.String login)
void	setPassword(java.lang.String password)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail