

MP_0487. Entornos de desarrollo

UF1. Desarrollo de software

**1.4. CASE, Herramientas
para el modelado**

Índice

☰	Objetivos	3
☰	Herramientas CASE	4
☰	Clasificación	5
☰	Funcionalidad	6
☰	Beneficios para las organizaciones	7
☰	Herramientas CASE en el mercado actual	9
☰	Resumen	11

Objetivos

Los objetivos de esta lección son:

- 1 Conocer qué son las herramientas CASE.
- 2 Conocer cuáles son los tipos de herramientas CASE .
- 3 Conocer cuáles son las funcionalidades de las herramientas CASE.
- 4 Conocer cómo se benefician las organizaciones al usar herramientas CASE.
- 5 Conocer las ventajas e inconvenientes de las herramientas CASE.

Herramientas CASE

Las herramientas CASE sirven para ayudar a los ingenieros de software en el desarrollo de proyectos en las fases y tareas que se define dentro del ciclo de vida del software.

CASE significa en inglés: **Computer Aided Software Engineering**. Es decir, ingeniería del software asistida por ordenador.

Las herramientas CASE tienen como objetivo final, mejorar la gestión de un proyecto de desarrollo de software, siendo muy útiles para la correcta planificación y automatización de procesos.

Podemos englobar sus funciones en dos tipos:

Automatizar

Automatizar los procesos, como pueden ser: la gestión del proyecto, el desarrollo de software, la documentación, detección de errores y generación de código.

Permitir

Además permiten: facilitar el uso de las metodologías de la ingeniería de software, la reutilización y la portabilidad del software o la estandarización de la documentación.

Clasificación

Las herramientas CASE cubren un amplio abanico de funcionalidades dentro del ciclo de vida del software.

Algunas, las más ambiciosas, llegan desde el diseño pasando por el desarrollo y acabando en la fase de mantenimiento. Aunque esto no es lo normal, pues la mayoría se enfoca a una de las fases del desarrollo. Las herramientas CASE podrían clasificarse por el tipo de plataforma a la que están asociada. Por ejemplo, si Oracle o Microsoft fabrican este tipo de herramientas seguramente estarán orientadas a dar soporte a sus lenguajes de programación.

Una forma muy común de clasificarlas está relacionada por el área que cubren:



- **Integrated CASE (I-CASE):** este tipo de herramientas permiten hacer el proceso completo a la hora de desarrollar un software. Van desde la fase inicial de análisis hasta que se implementa.
- **Upper CASE (U-CASE):** estas herramientas están enfocadas al apoyo en las fases de planificación y análisis, siendo muy útil para crear los diagramas UML.
- **Lower CASE (L-CASE):** en este caso hablamos de herramientas que nos ayudan a semi-automatizar la creación del código. Son muy útiles para crear la documentación del desarrollo y para la detección de errores, además de soportar la depuración de programas y pruebas.

Funcionalidad

Las herramientas CASE se pueden agrupar si tienen una funcionalidad similar, actividades de proceso y la capacidad de ser integradas con otras herramientas.

Si las agrupáramos por funcionalidad, serían estas:

1

Análisis y modelado de negocio: herramientas de modelado gráfico.

Por ejemplo, E / R de modelado, modelado de objetos, etc.

2

Desarrollo e implementación: diseño y construcción de las fases del ciclo de vida. Entornos de depuración. Por ejemplo, depurador de GNU.

3

Verificación y validación: analizar el código y las especificaciones para la corrección, el rendimiento, etc.

4

Gestión de la configuración: controlar el registro de entrada y de salida de los objetos del repositorio y archivos. Por ejemplo, CCSC, CMS.

5

Mediciones y métricas: analizar el código para la complejidad, la modularidad (por ejemplo, no "Ir a la"), el rendimiento, etc.

6

Gestión de proyecto: manejo de planes de proyectos, asignación de tareas, programación.

Beneficios para las organizaciones

El uso de estas herramientas resulta imprescindible hoy en día para las empresas de desarrollo. A continuación se describen algunos de los beneficios que proporcionan el uso de estas herramientas por los equipos de desarrollo:

Facilitar una metodología de diseño única en la empresa

Las herramientas CASE ayudan a las organizaciones a estandarizar el proceso de desarrollo. También facilitan el desarrollo coordinado. La integración se hace fácil a medida que se adopta una metodología común.

Desarrollo rápido de aplicaciones

Para mejorar la velocidad y la calidad del desarrollo de sistema en las organizaciones se utilizan herramientas CASE.

Pruebas

Las herramientas CASE ayudan a mejorar el proceso de prueba mediante un control automatizado y un mantenimiento más sencillo del programa.

Documentación

En un proceso de desarrollo de software tradicional la calidad de la documentación en las diversas etapas depende de la persona. En varias etapas de las herramientas CASE SDLC se mejora la calidad y la uniformidad de la documentación en cada una de ellas. También asegura la integridad de la documentación.

Gestión de proyectos

Mejora la actividad en la gestión de proyectos y en cierta medida automatiza varias actividades involucradas en dicha gestión.

Reducir costos de mantenimiento

El uso de herramientas CASE hace que el software sea fácil de mantener y, por tanto, reducir los costes de mantenimiento.

Aumentar la productividad

La automatización de las diversas actividades de los procesos de desarrollo y de gestión del sistema aumenta la productividad del equipo de desarrollo.

Herramientas CASE en el mercado actual

Actualmente existe una gran variedad de **herramientas CASE** (**Computer Aided Software Engineering o Ingeniería de Software Asistida por Ordenador**) para el proceso de desarrollo de software, por lo que resulta complicado definir las mejores, pues depende del tipo de proyecto que estemos realizando.

A continuación puedes encontrar los enlaces a las más extendidas en las empresas:

Netbeans

Potente entorno de desarrollo, multiplataforma y con soporte para múltiples lenguajes de programación.

[IR A SITIO WEB](#)

Argo UML

ArgoUML es la herramienta líder de modelado UML de código abierto e incluye soporte para todos los diagramas UML. Se ejecuta en cualquier plataforma Java y está disponible en diez idiomas.

[IR A SITIO WEB](#)

Eclipse

El entorno de desarrollo más extendido actualmente, con soporte para muchos lenguajes y con capacidades de ampliación mediante la instalación de plugins.

[IR A SITIO WEB](#)

Dia

Editor para diagramas de estructura estáticos UML.

[IR A SITIO WEB](#)**SQL Developer**

Entorno para trabajar con bases de datos.

[IR A SITIO WEB](#)

Resumen

Has finalizado esta unidad, veamos los puntos más importantes que hemos tratado.

En esta unidad hemos visto cómo se definen y se clasifican las herramientas que utilizaremos para ayudarnos a realizar los desarrollos de software en nuestra profesión.

Las herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Ordenador) son aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software, reduciendo el costo de las mismas en términos de tiempo.

La siguiente clasificación es la más habitual, basada en las fases del ciclo de desarrollo que cubren:

Upper CASE (U-CASE)

Herramientas que ayudan en las fases de planificación, análisis de requisitos y estrategia del desarrollo usando, entre otros, diagramas UML.

Middle CASE (M-CASE)

Herramientas para automatizar tareas en el análisis y diseño de la aplicación.

Lower CASE (L-CASE)

Herramientas que semi-automatizan la generación de código, crean programas de detección de errores, soportan la depuración de programas y pruebas. Además automatizan la documentación completa de la aplicación. Aquí pueden incluirse las herramientas de desarrollo rápido de aplicaciones.

Según funcionalidad:

- Herramientas de generación semiautomática de código.
- Editores UML.
- Herramientas de refactorización de código.
- Herramientas de mantenimiento, como los sistemas de control de versiones.



PROEDUCA