

HERRAMIENTAS EN EL PROCESO DE DESARROLLO DE SW

1. CONCEPTOS GENERALES

Desde el inicio de la escritura del software, ha existido un conocimiento de la necesidad de herramientas automatizadas para ayudar al diseñador del software. Inicialmente la concentración estaba en herramientas de apoyo a programas como traductores, compiladores, ensambladores, y procesadores de macros.

Este conjunto de aplicaciones que pueden informatizarse, aumentó rápidamente en un breve espacio de tiempo, causando una gran demanda por nuevo software a desarrollar. A medida que se escribía nuevo software, había ya en existencia infinidad de líneas de código que necesitaban ser mantenidas y actualizadas.

Esto causó a la industria de las computadoras muchos problemas, debido a que no podían cubrir el incremento de la demanda con los métodos que se estaban usando. Esto fue reconocido como una crisis del software. Para superar este problema en el proceso de desarrollo de software, se introdujeron metodologías para intentar crear estándares de desarrollo, acompañando a esto se mejoraron también herramientas informáticas usadas en el proceso de desarrollo de software.

Al día de hoy la industria del software se ha ayudado a superar las dificultades de uso de esta tecnología disponible. La industria de Software ha desarrollado un soporte automatizado para el desarrollo y mantenimiento de software.

Se pretende contar entonces con un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del Ciclo de Vida de desarrollo de un Software. Los estados en el Ciclo de Vida de desarrollo de un Software son:

- Investigación Preliminar,
- Análisis,
- Diseño,
- Implementación, e
- Instalación.

El uso de Herramientas en el proceso de desarrollo de Software puede ser tan sencilla como herramienta que preste su apoyo para una única actividad de ingeniería de software. O bien puede ser compleja como todo entorno que abarque herramientas, una base de datos, personas, hardware, una red, sistemas operativos, estándares y otros componentes más.

1. 1 Beneficios

Estas herramientas pueden proveer muchos beneficios de acuerdo de acuerdo a las etapas del proceso de desarrollo de software que den soporte, algunas de ellos son:

- Verificar el uso de todos los elementos en el sistema diseñado,
- automatizar el dibujo de diagramas,
- ayudar en la documentación del sistema,
- ayudar en la creación de relaciones en la Base de Datos, y

- generar estructuras de código.

La principal ventaja de la utilización de una Herramienta Informáticas aplicadas al proceso de desarrollo de software, es la mejora de la calidad de los desarrollos realizados y en segundo término, el aumento de la productividad. Para conseguir estos dos objetivos es conveniente contar con una organización y una metodología de trabajo, además de la propia herramienta.

La mejora de calidad se consigue reduciendo sustancialmente muchos de los problemas de análisis y diseño, inherentes a los proyectos de mediano y gran tamaño (lógica del diseño, coherencia, consolidación, etc.). La mejora de productividad se consigue a través de la automatización de determinadas tareas, como la generación de código y la reutilización de objetos o módulos.

2. CLASIFICACIÓN DE LAS HERRAMIENTAS BASADO EN SU FUNCIONALIDAD

- **Herramientas de gestión de proyectos:** la mayoría de las herramientas de gestión de proyectos se centran en un elemento específico de la gestión del proyecto, en lugar de proporcionar un soporte global para la actividad de gestión.

Utilizando un conjunto seleccionado de estas herramientas se pueden realizar estimaciones de esfuerzo, coste y duración, hacer un seguimiento continuo del proyecto, estimar la productividad y la calidad, etc.

Existen también herramientas que permiten al comprador del desarrollo de un sistema, hacer un seguimiento que va desde los requisitos del pliego de prescripciones técnicas inicial, hasta el trabajo de desarrollo que convierte estos requisitos en un producto final. Se incluyen dentro de las herramientas de control de proyectos las siguientes:

- Planificación de proyectos.
- Seguimiento de requisitos.
- Gestión y medida.

- **Herramientas de análisis y diseño:** permiten al desarrollador crear un modelo del sistema que se va a construir y también la evaluación de la validez y consistencia de este modelo. Proporcionan un grado de confianza en la representación del análisis y ayudan a eliminar errores con anticipación. Estas herramientas cubren las siguientes funcionalidades:

- Análisis y diseño (Modelamiento).
- Creación de prototipos y de simulación.
- Diseño y desarrollo de interfaces.

Las herramientas de análisis y diseño cubren una parte importantísima del proceso de desarrollo de software. Su principal objetivo es ayudar a la definición de requisitos del sistema y sus propiedades.

Dentro de esta categoría destacan las herramientas que permiten crear y modificar:

- Diagramas de clases.
- Diagramas de estructura de cuadros.
- Diagramas E/R.
- Diagramas de flujo de datos.

También son muy importantes las herramientas de prototipado como:

- Los diseñadores de pantallas.
- Generadores de menús.
- Generadores de informes y lenguajes de especificación ejecutables.

Un aspecto importante es la capacidad de análisis y verificación de especificaciones que soporta la herramienta, no sólo sintáctica sino también semántica, como por ejemplo la capacidad de normalizar un diagrama de datos.

Todas estas herramientas operan sobre un repositorio donde se va almacenando la información necesaria para el funcionamiento de la misma herramienta.

- **Herramientas de programación:** se engloban aquí los compiladores, los editores y los depuradores de los lenguajes de programación convencionales. Ejemplos de estas herramientas son:
 - Codificación convencional.
 - Codificación de cuarta generación.
 - Programación orientada a objetos.

A partir de las especificaciones del diseño se puede generar código (C, JAVA, PHP y C#) como los esquemas de bases de datos (sentencias de definición en SQL) convenientes.

Actualmente las herramientas CASE ofrecen interfaces con diversos lenguajes de cuarta generación para la construcción de sistemas de manera rápida.

- **Herramientas de integración y prueba:** sirven de ayuda a la adquisición, medición, simulación y prueba de los equipos lógicos desarrollados. Entre las herramientas más utilizadas están:
 - Análisis estático.
 - Codificación de cuarta generación.
 - Programación orientada a los objetos.

- **Herramientas de gestión de prototipos:** los prototipos son utilizados ampliamente en el desarrollo de aplicaciones, para la evaluación de especificaciones de un sistema de información, o para un mejor entendimiento de cómo los requisitos de un sistema de información se ajustan a los objetivos perseguidos.

En la fase de Diseño, su propósito, basándose en los requerimientos previamente obtenidos, es mostrar las ventanas, su navegación, interacción, controles y botones al usuario y obtener una retroalimentación que nos permite mejorar el Diseño de Interfaz

- **Herramientas de mantenimiento:** la categoría de herramientas de mantenimiento se puede subdividir en:
 - Ingeniería inversa. (reverse engine de proceso, datos, interfaz)
 - Reestructuración y análisis de código.
 - Reingeniería. (reverse engine y fordware engine de todo el software)

En cuanto herramientas de Ingeniería inversa este tipo cubren las siguientes funcionalidades:

- **Ingeniería inversa de datos:** son capaces de extraer la información del código fuente que describe la estructura de los elementos de datos, construyendo así diagramas Entidad-Relación partiendo de esquemas relacionales en red, o incluso archivos.
 - **Ingeniería inversa de procesos:** permiten aislar la lógica de las entidades y las reglas del negocio a partir del código.
 - **Reestructuración de código fuente:** modifican su formato o implantan un formato estándar.
 - **Redocumentación:** permiten generar diagramas a fin de que se comprenda mejor el código.
 - **Análisis de código:** sus funcionalidades van desde la indentación automática del código fuente hasta la posibilidad de ir visualizando dinámicamente las llamadas del mismo.
- **Herramientas de soporte:** se engloban en esta categoría las herramientas que recogen las actividades aplicables en todo el proceso de desarrollo, como las que se relacionan a continuación:
 - Documentación: creación automatizada de un conjunto muy variado de documentación que va desde la descripción textual de un pseudocódigo, hasta diagramas más o menos complejos.
 - Software de sistemas.
 - Control de calidad.
 - Bases de datos.
 - Herramientas de gestión de configuración.

La **Gestión de Configuración**, se hace imprescindible la incorporación de una herramienta capaz de gestionar la configuración de los sistemas. Este tipo de herramientas ofrece distintos tipos de capacidades:

- *Control de versiones:* capacidad de proporcionar almacenamiento y acceso controlado a los datos, así como de registrar los cambios sobre los mismos y poder recuperar versiones anteriores.
- *Construcción:* consiste en automatizar las tareas asociadas a la compilación y enlace de un sistema con el fin de generar ejecutables.
- *Trazabilidad de requisitos y análisis de impacto:* permiten rastrear un requisito hasta su implementación, y conocer los elementos del sistema que se ven afectados ante un cambio.