



CEU

*Fundación San Pablo
Andalucía*

CENTRO DE ESTUDIOS PROFESIONALES

Glorieta Ángel Herrera Oria, s/n, 41930 Bormujos, Sevilla

HERRAMIENTAS CASE



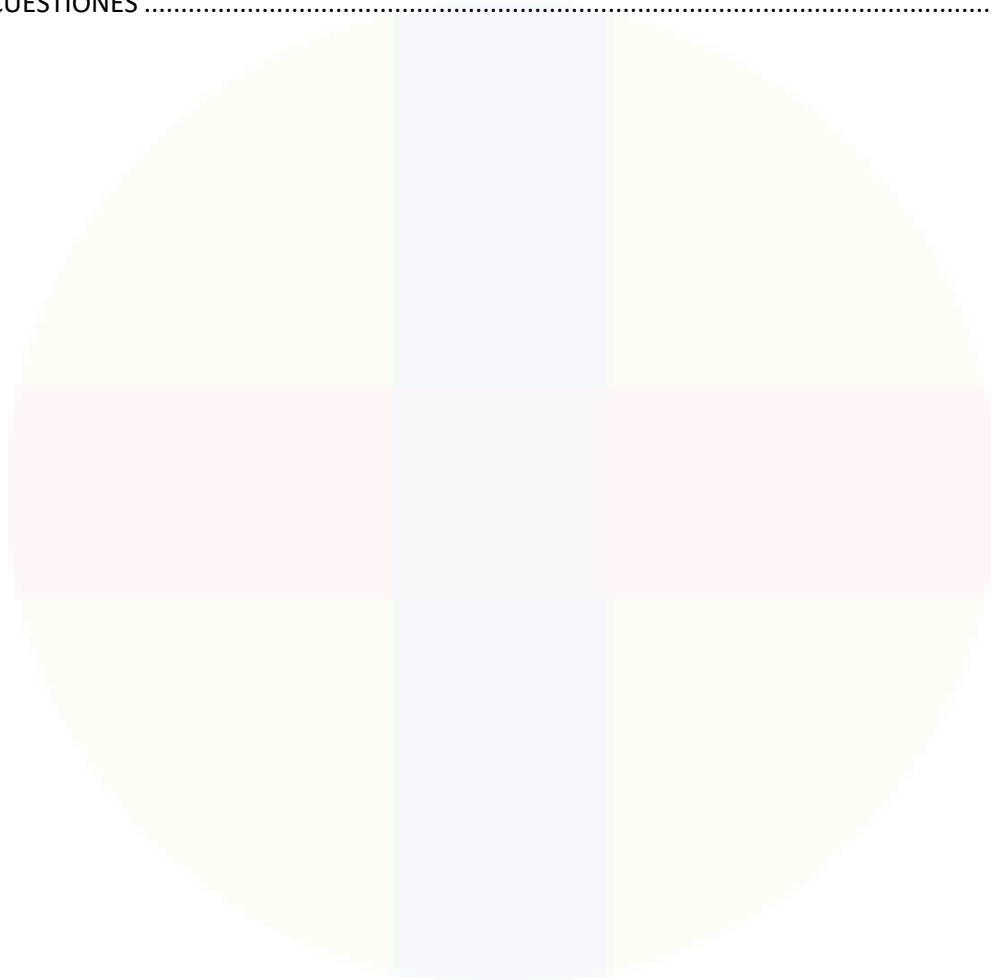
Realizado por:

Jhonatan Guzmán Panozo

TABLA DE CONTENIDOS

Contenido

1.	HERRAMIENTA CASE Y SU EXPLICACIÓN.....	1
2.	CUESTIONES	2



1. HERRAMIENTA CASE Y SU EXPLICACIÓN

Selecciona un ejemplo concreto de herramientas CASE y explícala en 750 palabras como máximo.

Las herramientas CASE son un conjunto de aplicaciones informáticas, utilizadas para automatizar actividades del ciclo de vida de desarrollo de sistemas. Son usadas por los directores de proyectos de software, analistas e Ingenieros para desarrollar sistemas de software.

El uso de las Herramientas CASE acelera el desarrollo del proyecto con tal de producir los resultados deseados y ayuda a encontrar imperfecciones antes de proseguir con la siguiente etapa del desarrollo de Software.

En este caso, hablaré de una en concreto, la herramienta CASE Diagrama. Esta herramienta se usa para representar componentes del sistema, datos, y controlar la fluidez de varios componentes y estructura del software de manera gráfica.

Este componente consiste en dar soporte para la creación de los diagramas más utilizados para el análisis y diseño del software.

Algunos de los diagramas y modelos utilizados con mayor frecuencia son:

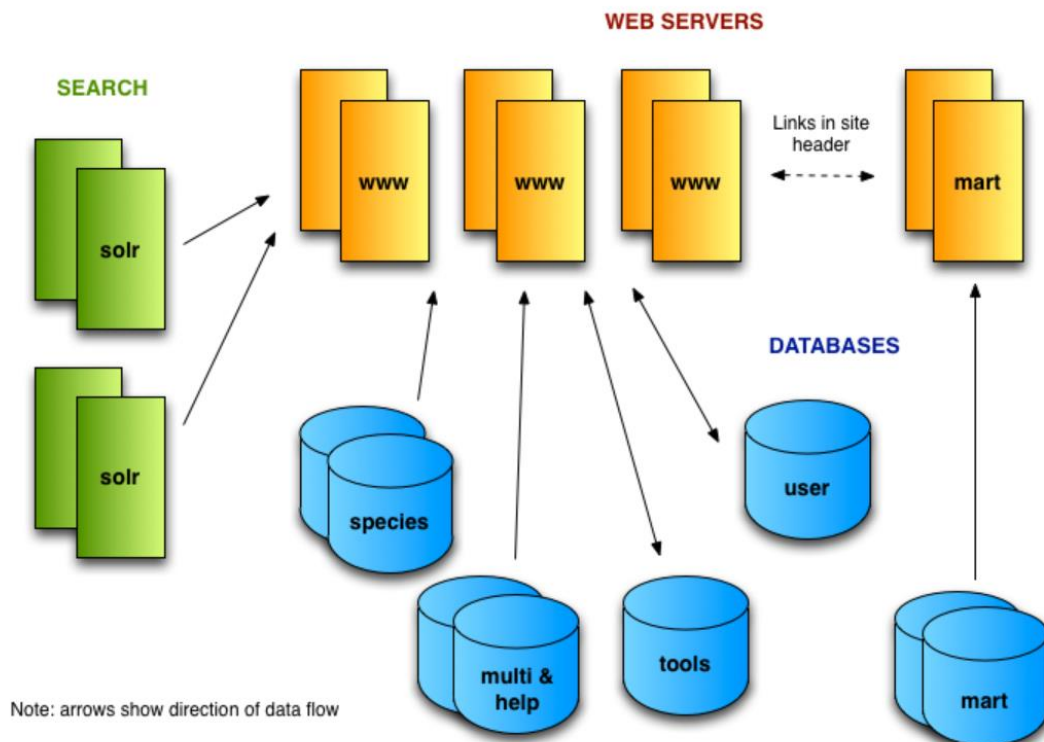
- Diagrama de flujo de datos.
- Modelo entidad - interrelación.
- Historia de la vida de las entidades.
- Diagrama Estructura de datos.
- Diagrama Estructura de cuadros.
- Técnicas matriciales.

Por ejemplo, la herramienta "System Architect" sirve para crear los más novedosos Diagramas de flujos. Esta herramienta posee un repositorio único que integra todas las herramientas, y metodologías usadas. En la elaboración de los diagramas, el System Architect conecta directamente al diccionario de datos, los elementos asociados, comentarios, reglas de validaciones, normalización, etc.

Posee control automático de diagramas y datos, normalizaciones y balanceamiento entre diagramas "Padre e Hijo", además de balanceamiento horizontal, que trabaja integrado con el diccionario de datos, asegurando la compatibilidad entre el Modelo de Datos y el Modelo Funcional.

El siguiente diagrama muestra una versión simplificada de la arquitectura del hardware, omitiendo detalles de la red como los balanceadores de carga.

SYSTEM ARCHITECTURE FOR WWW.ENSEMBL.ORG



2. CUESTIONES

Contesta las siguientes cuestiones y razona la respuesta.

1. Diferencias entre compilador e interprete.

Un compilador es un programa que transforma el código fuente de un programa a su equivalente en otro lenguaje de programación de más bajo nivel, el objetivo de los compiladores es obtener una traducción del programa fuente a otro lenguaje.

Un intérprete es un programa que ejecuta directamente las instrucciones escritas en un lenguaje de programación dado, tienen como objeto la obtención de los resultados del programa. Para ello deben realizar dos tareas: analizar su entrada y llevar a cabo las acciones especificadas por ella.

2. ¿Cuáles son los diferentes estados por los que pasa el código de un programa desde que se escribe hasta que se ejecuta por el ordenador?

Antes de que un programa se ejecute, en el proceso de compilación el código pasa por diferentes estados.

Código fuente: Conjunto de sentencias entendibles por el programador que componen el programa o una parte de ello. Se suele almacenar en un fichero de tipo texto. El código fuente se escribe en un lenguaje de programación concreto.

Código objeto: Conjunto de instrucciones y datos escritos en un lenguaje que entiende el ordenador directamente: binario o código máquina. Es la traducción de cierto código fuente, es un fragmento del programa final y es específico de la plataforma de ejecución.

Código ejecutable: Reúne diferentes códigos u objetos generados por los programadores junto con las “librerías de uso general” (propias del entorno o del lenguaje de programación) componiendo el programa final. Éste es el código que ejecutan los usuarios del sistema, y es específico para una plataforma concreta: Windows, Linux, Mac OS, o ciertos sistema hardware.

3. Cuáles de las siguientes afirmaciones sobre la JVM es cierta:

a) Un fichero .class contiene el código en lenguaje máquina.

b) La JVM toma y traduce el bytecode en código binario.

c) Los ficheros .class pueden ser ejecutados en Linux y en Windows.

d) A la hora de instalar el entorno de ejecución de la JVM necesitamos saber en qué sistema operativo se va a instalar.