Tarea Fundamentos de Software

Conceptos

Software:

Definición propia: El software es un conjunto de programas, datos y otros elementos que permiten que un dispositivo electrónico (como una computadora, un teléfono móvil, etc.) realice tareas específicas de acuerdo con las instrucciones dadas por el usuario.

Definición formal: El software se refiere a los programas de computadora, datos y documentación asociada que se utilizan para realizar tareas específicas en un dispositivo electrónico. Puede ser tanto el sistema operativo que controla el hardware de la computadora como las aplicaciones de software que los usuarios utilizan para realizar diversas funciones.

Ingeniería de Software:

Definición propia: La ingeniería de software es una disciplina que se enfoca en aplicar principios de ingeniería para el desarrollo, mantenimiento y gestión de software de alta calidad de manera sistemática y eficiente.

Definición formal: La ingeniería de software es una disciplina que se ocupa de todos los aspectos del desarrollo de software, desde la concepción de las necesidades del cliente hasta el mantenimiento del software una vez que está en funcionamiento. Utiliza principios de ingeniería para garantizar la calidad, fiabilidad, eficiencia y mantenibilidad del software desarrollado.

Git:

Definición propia: Git es un sistema de control de versiones distribuido que permite a los desarrolladores colaborar en proyectos de software, rastrear cambios en el código y coordinar el trabajo de manera eficiente.

Definición formal: Git es un sistema de control de versiones distribuido de código abierto, diseñado para gestionar proyectos de software de cualquier tamaño de manera eficiente. Permite a los desarrolladores rastrear cambios en el código, colaborar en proyectos y mantener un historial completo de todos los cambios realizados en el código fuente.

Programación:

Definición propia: La programación es el proceso de escribir, probar y mantener el código fuente de un programa de computadora para que realice una determinada tarea o función.

Definición formal: La programación es el proceso de escribir, probar y mantener el código fuente de un programa de computadora utilizando un lenguaje de programación específico. Implica la traducción de algoritmos y especificaciones en un lenguaje comprensible por la computadora para que pueda ejecutar tareas específicas.

Lenguaje de Programación:

Definición propia: Un lenguaje de programación es un conjunto de reglas y símbolos que se utilizan para escribir instrucciones que una computadora puede entender y ejecutar para realizar una tarea específica.

Definición formal: Un lenguaje de programación es un conjunto de reglas sintácticas y semánticas que definen la estructura y el significado de las instrucciones utilizadas para escribir programas de computadora. Los lenguajes de programación pueden ser de alto nivel, como Python o Java, o de bajo nivel, como el lenguaje ensamblador.

Compilador:

Definición propia: Un compilador es un programa que traduce el código fuente escrito en un lenguaje de programación a un código ejecutable que una computadora puede entender y ejecutar ósea un lenguaje binario.

Definición formal: Un compilador es un programa que traduce el código fuente escrito en un lenguaje de programación de alto nivel a un código de máquina o código objeto que puede ser ejecutado por un procesador de computadora. El compilador realiza varias etapas, como el análisis léxico, el análisis sintáctico, la generación de código intermedio y la optimización del código, para producir el código ejecutable final.

Ciclo de Desarrollo de Software y sus Componentes:

Definición propia: El ciclo de desarrollo de software es el proceso que sigue un proyecto de desarrollo de software desde la concepción de la idea hasta su entrega final al cliente, pasando por etapas como la planificación, el diseño, la implementación, las pruebas y el mantenimiento.

Definición formal: El ciclo de desarrollo de software es un proceso sistemático que sigue un proyecto de desarrollo de software desde la identificación de los requisitos del cliente hasta la entrega del producto final. Sus componentes principales incluyen la planificación, el análisis de requisitos, el diseño, la implementación, las pruebas, la integración, la implementación y el mantenimiento del software. Este ciclo puede seguir diferentes modelos, como el modelo en cascada, el modelo en espiral, el modelo iterativo e incremental, entre otros.