PROCESOS DE SOFTWARE, MODELOS Y COMPARATIVA

CRISTHIAN URREGO SALAZAR

CENTRO DE ESTUDIOS TECNOLÓGICOS DEL NORTE DEL VALLE COTECNOVA

TECNOLOGÍA EN SISTEMAS DE INFORMACIÓN

CARTAGO VALLE

2017

PROCESOS DE SOFTWARE, MODELOS Y COMPARATIVA

CRISTHIAN URREGO SALAZAR

NOMBRE PROFEROS

DOCENTE DISEÑO DE SISTEMAS DE INFORMACIÓN

CENTRO DE ESTUDIOS TECNOLÓGICOS DEL NORTE DEL VALLE COTECNOVA

TECNOLOGÍA EN SISTEMAS DE INFORMACIÓN

CARTAGO VALLE

2017

**QUE ES UN PROCESO DE SOFTWARE**

La meta de la ingeniería de software es construir productos de software, o mejorar los existentes; en ingeniería de procesos, la meta es desarrollar o mejorar procesos.

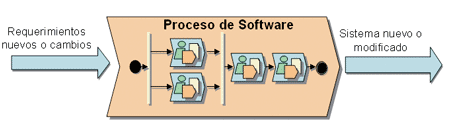
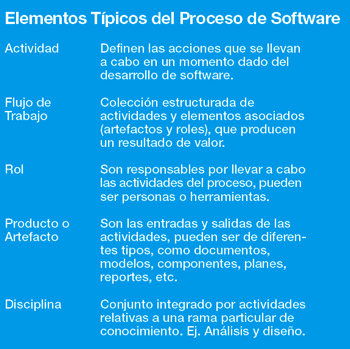
Un proceso de desarrollo de software es un conjunto de personas, estructuras de organización, reglas, políticas, actividades y sus procedimientos, componentes de software, metodologías, y herramientas utilizadas o creadas específicamente para definir, desarrollar, ofrecer un servicio, innovar y extender un producto de software.

Un proceso de software efectivo habilita a la organización a incrementar su productividad al desarrollar software:

* Permite estandarizar esfuerzos, promover reusó, repetición y consistencia entre proyectos.
* Provee la oportunidad de introducir mejores prácticas de la industria.
* Permite entender que las herramientas deben ser utilizadas para soportar un proceso.
* Establece la base para una mayor consistencia y mejoras futuras.

Un proceso de software mejora los esfuerzos de mantenimiento y soporte:

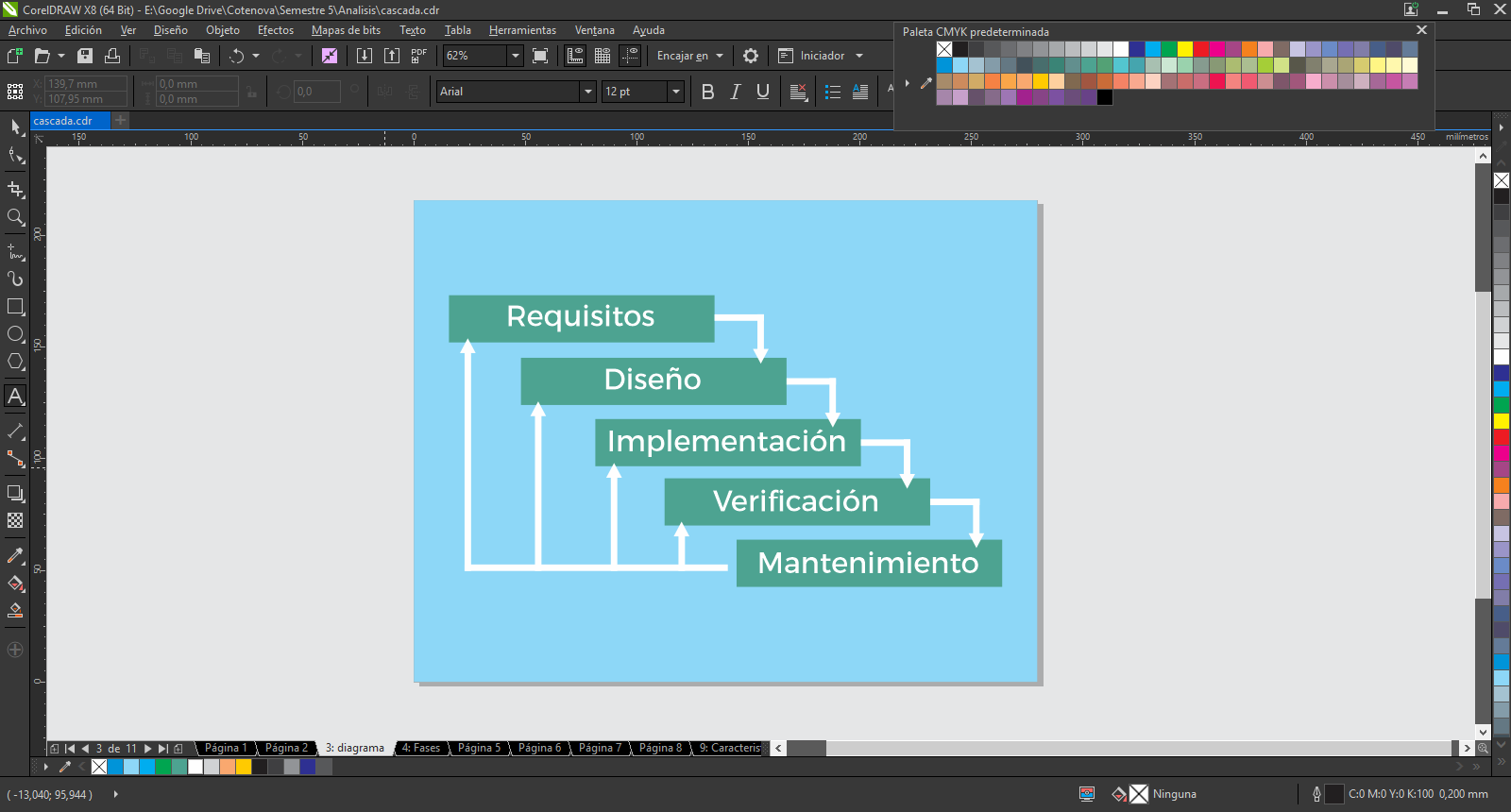
* Define cómo manejar los cambios y liberaciones a sistemas de software existentes.
* Define cómo lograr la transición del software a la operación, y cómo ejecutar los esfuerzos de operación y soporte.

#### MODELOS DE PROCESOS

**MODELO CASCADA**

Su visión es muy simple: el desarrollo de software se debe realizar siguiendo una secuencia de fases. Cada etapa tiene un conjunto de metas bien definidas y las actividades dentro de cada una contribuyen a la satisfacción de metas de esa fase o quizás a una subsecuencia de metas de la misma.



**FASES**

**Requisitos**

**Ingeniería y Análisis del Sistema:** Debido a que el software es siempre parte de un sistema mayor, el trabajo comienza estableciendo los requisitos de todos los elementos del sistema y luego asignando algún subconjunto de estos requisitos al software.

**Análisis de los requisitos del software:** el proceso de recopilación de los requisitos se centra e intensifica especialmente en el software. El ingeniero de software debe comprender el ámbito de la información del software así como la función, el rendimiento y las interfaces requeridas.

**Diseño**

El diseño del software se enfoca en cuatro atributos distintos del programa; la estructura de los datos, la arquitectura del software, el detalle procedimental y la caracterización de la interfaz. El proceso de diseño traduce los requisitos en una representación del software con la calidad requerida antes de que comience la codificación.

**Implementación**

El diseño debe traducirse en una forma legible para la máquina. Si el diseño se realiza de una manera detallada, la codificación puede realizarse mecánicamente.

**Verificación**

Una vez que se ha generado el código comienza la prueba del programa. La prueba se centra en la lógica interna del software y en las funciones externas, realizando pruebas que aseguren que la entrada definida produce los resultados que realmente se requieren.

**Mantenimiento**

El software sufrirá cambios después de que se entrega al cliente. Los cambios ocurrirán debidos a que se haya encontrado errores, a que el software deba adaptarse a cambios del entorno externo (sistema operativo o dispositivos periféricos) o a que el cliente requiera ampliaciones funcionales o del rendimiento.

**Características**

•Cada fase genera documentación para la siguiente. Esta documentación debe ser aprobada.

•Una fase no comienza hasta que la anterior ha terminado.

•Requiere disponer de unos requisitos completos y precisos al principio del desarrollo.

•Es una visión del proceso de desarrollo de software como una sucesión de etapas que producen productos intermedios.

•Para que el proyecto tenga éxito deben desarrollarse todas las fases.

Si se cambia el orden de las fases, el producto final será de inferior calidad.

**Ventajas**

•Se tiene todo bien organizado y no se mezclan las fases.

•Ayuda a localizar errores en las primeras etapas del proyecto a un bajo costo.

•Ayuda a minimizar los gastos de la planificación porque permite realizarla sin problemas.

**Desventajas**

•Gran dependencia en los requerimientos iníciales

•Difícilmente un cliente va a establecer al principio todos los requerimientos necesarios, por lo que provoca un gran atraso trabajando en este modelo, ya que este es muy restrictivo y no permite movilizarse entre fases.

•El modelo genera pocos signos visibles de progreso hasta el final. Esto puede dar la impresión de un desarrollo lento, existe la incertidumbre de los clientes si sus proyectos serán entregados a tiempo.

•Inicio de la codificación muy tarde en el ciclo de vida del proyecto

**BIBLIOGRAFIA**

EduRed. (25 de Agosto de 2017). *EduRed*. Obtenido de https://www.ecured.cu/Modelo\_en\_cascada

Ruvalcaba, M. (s.f.). *Procesos de Software*. Recuperado el 23 de 08 de 2017, de SG Buzz: https://sg.com.mx/revista/1/procesos-software#.WZ3alVFYGUk

Soloriio, M. (25 de Agosto de 2017). *metodologiaencascada* . Obtenido de blogspot: http://metodologiaencascada.blogspot.com.co/

Zulay, M. (25 de Agosto de 2017). Obtenido de http://modelosprocesosdesoftware.blogspot.com.co/p/modelo-en-cascada-o-lineal-secuencial.html