

General

Bienvenidos

Estimados alumnos; soy la profesora Yesenia Rodríguez Zamora, técnica en computación e informática, especialista en base de datos, con más de 15 años de experiencia en desarrollo de sistemas de información; les doy la bienvenida al curso de Ingeniería de Software Avanzada.

El curso de Ingeniería de software avanzada; es un curso que proporciona las bases del conocimiento de las tecnologías de información

El software constituye hoy en día uno de los recursos importantes de las organizaciones, que les permite la realización y registro de sus operaciones del día a día, apoyo en los procesos de toma de decisiones, etc.

El proceso de fabricación de software ha alcanzado una madurez y ha desarrollado un cuerpo de conocimiento tanto en los aspectos técnicos del desarrollo así como en los aspectos administrativos del proceso en sí. Es por ello importante tomar conocimiento de estos conceptos para así comprender el proceso de desarrollo de software propiamente dicho; así como también analizar las diferentes técnicas que pueden ser aplicadas para la planificación, administración y ejecución de los proyectos.

El valor de un producto de TI, se ve incrementado en función del uso adecuado de las herramientas que conlleven a un producto confiable y de calidad.



INDICACIONES

Debido a la coyuntura que vive nuestro país y el resto del mundo; nos vemos obligados a esforzarnos un poco mas por compartir nuestros conocimientos y experiencias de una manera diferente; por lo que solicito de parte de ustedes compromiso y dedicación.

Durante este periodo publicaré cada tema de acuerdo a nuestro horario regular de clases, estos deberán ser desarrollados con cuidado y al final de cada uno habrá tareas que deberán cumplir según las indicaciones y en el plazo designado en cada caso.

De acuerdo a nuestro horario, los iré invitando a reuniones virtuales, para poder resolver preguntas, consideraré puntaje de participación; tendremos foros cuya participación también tendrá un puntaje sobre su nota permanente.

Cualquier duda podremos resolverla por este medio o a través de un mensaje a mi correo: y.rodriguez.zamora@isur.edu.pe

- Avisos
- Sílabo

Calendario

junio 2020

Dom	Lun	Mar	Mié	Jue	Vie
(Domingo)	(Lunes)	(Martes)	(Miércoles)	(Jueves)	(Viernes)
	1	2	3	4	5
7	8	9	10	11	12
14	15	16	17	18	19
21	22	23	24	25	26
28	29	30			

- Ocultar eventos de sitio
- Ocultar eventos de categoría
- Ocultar eventos de curso
- Ocultar eventos de grupo
- Ocultar eventos de usuario

Tema 7 Arquitectura de Aplicaciones y Diseño Orientado a Objetos

Arquitectura de aplicaciones

Con todo lo que hemos visto a cerca de infraestructura y los diferentes elementos que pueden ser combinado para utilizarlos de una manera óptima; veremos como a través de las aplicaciones intermediarias y middleware podremos armar las estructuras que necesitamos:

En el caso de una arquitectura de aplicaciones en web y móviles, también están basadas en este tipo de combinaciones.

Diseño Orientado a Objetos

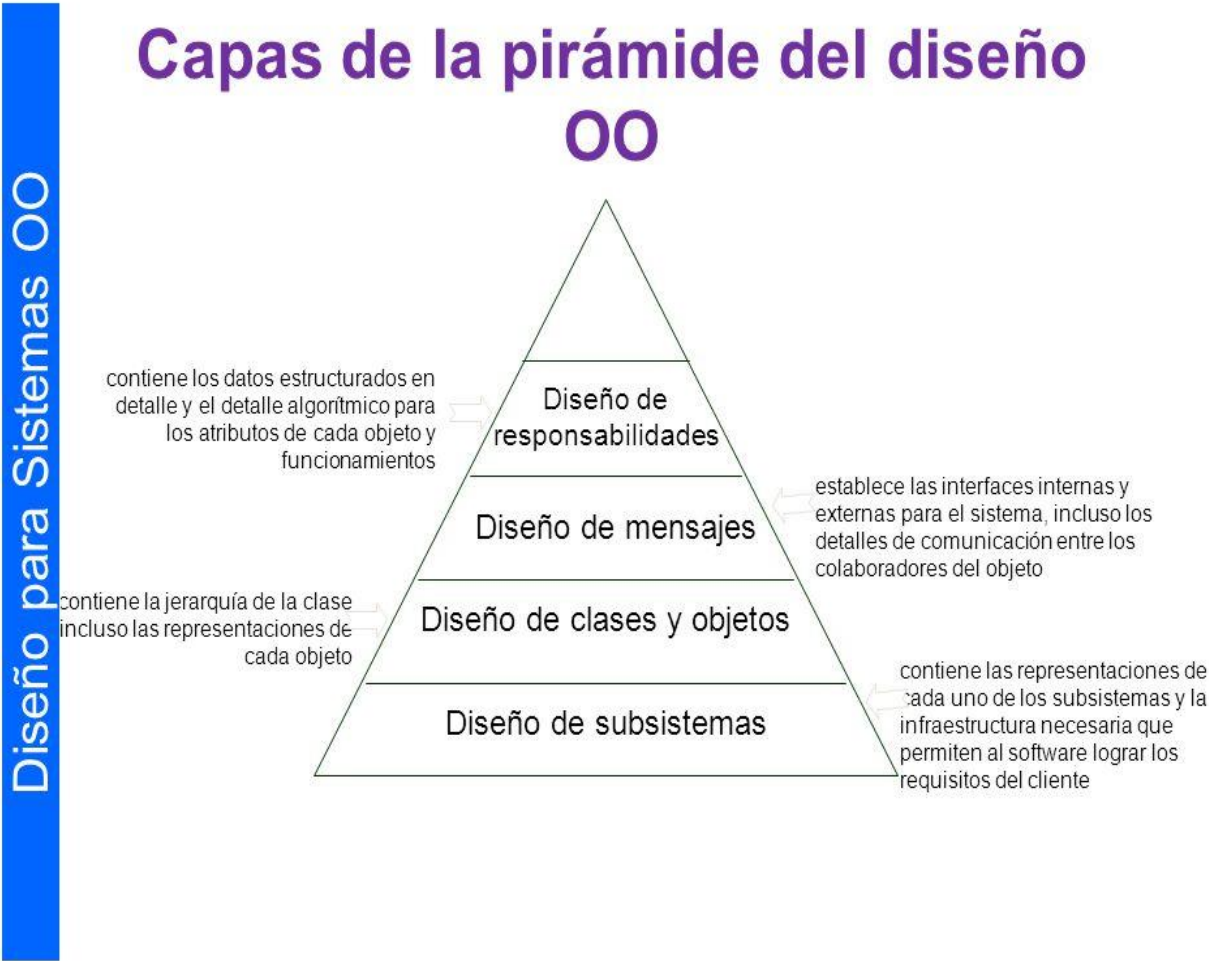
El diseño Orientado a Objetos (DOO) difiere considerablemente del diseño estructurado ya que en DOO no se realiza un problema en términos de tareas (subrutinas) ni en términos de datos, sino (como ya se vio en la introducción) se analiza el problema como un sistema de objetos que interactúan entre sí. Un problema desarrollado con técnicas orientadas a objetos requiere, en primer lugar saber cuales son los objetos del programa. Como tales objetos son instancias de clases, la primera etapa en el desarrollo orientado a objetos requiere de la identificación de dichas clases (atributos y comportamiento), así como las relaciones entre éstas y su posterior implementación en un lenguaje de programación. Existen numerosos métodos de diseño orientado a objetos: Booch, Yourdon-Coad, Martín, Shlaer & Mellor, Rumbaugh, por citar algunos. Pero en general como ocurre en cualquier proyecto estructurado, un proyecto software OO se compone de las siguientes etapas:

- Análisis Orientado a Objetos (AOO)
- Diseño Orientado a Objetos (DOO)
- Programación Orientada a Objetos (POO)

Aunque no siempre están bien delimitadas las etapas de análisis y diseño en la OO, se pueden sintetizar de alguna forma las ideas claves de las distintas tecnologías existentes dentro del desarrollo orientado a objetos al que denominaremos diseño. El método de Booch considera que las etapas del proceso en un desarrollo orientado a objetos son:

1. Identificar las claves y objetos en un nivel dado de abstracción
2. Identificar la semántica de estas clases y objetos
3. Identificar las relaciones entre clases y objetos
4. Especificar la interfaz y la implementación de estas clases y objetos.

Estas etapas suelen seguirse por la mayoría de los métodos de diseño OO existentes. De hecho, para los sistemas orientados a objetos se define el siguiente diseño en pirámide que contempla el método de Booch.



La capa del subsistema.-Contiene una representación de cada uno de los subsistemas que le permiten al software conseguir los requisitos definidos por el cliente e implementar la infraestructura técnica que los soporta.

La capa de clases y Objetos.-Contiene las jerarquías de clase que permiten crear el sistema usando generalizaciones y especializaciones mejor definidas. Esta capa también contiene representaciones de diseño para cada objeto.

La capa de mensajes.-Contiene los detalles que le permiten a cada objeto comunicarse con sus colaboradores. Esta capa establece las interfaces externas e internas para el sistema.

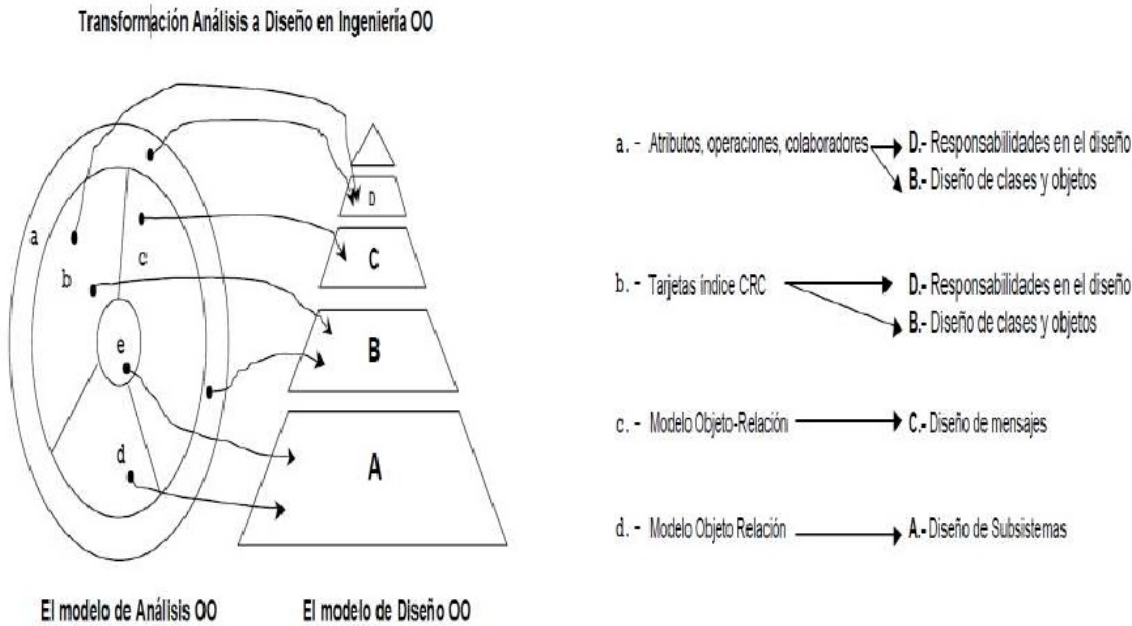
La capa de responsabilidades.-Contiene las estructuras de datos y el diseño algorítmico para todo los atributos y operaciones de cada objeto.Esta pirámide de diseño se centra entonces en el diseño de un producto o sistema específico

EL ENFOQUE CONVENCIONAL Y EL ENFOQUE OO

Los enfoques convencionales para el diseño del software aplican notaciones y heurísticas diferentes para establecer correspondencias entre el modelo de análisis y el diseño. Si recordamos la ingeniería de software clásica (imperativa), veremos que cada elemento del modelo de análisis convencional tiene correspondencia con una o más capas del modelo de diseño tal como lo ilustra la siguiente figura



Al igual que el diseño de software convencional, el DOO aplica diseño de datos (cuando se representan atributos), diseño de interfaces (cuando se presenta el intercambio de mensajes) y diseño procedimental (en el diseño de operaciones), no obstante el diseño arquitectónico es diferente.La arquitectura de diseño OO se centra más en las colaboraciones entre los objetos que con el flujo de control de datos. De esta manera las capas de la pirámide se renombran para reflejar de forma más exacta la naturaleza del DOO. La siguiente figura muestra ahora la correspondencia entre el AOO con las correspondientes capas de la pirámide de diseño OO.



EL DISEÑO: Bertrand Meyer sugiere los siguientes criterios para poder juzgar la capacidad que poseé un método de diseño en poder lograr ciertos elementos importantes tales como la modularidad:

Descomponibilidad.- Facilidad con la cual un método de diseño ayuda al diseñador a descomponer un gran problema en subproblemas más sencillos de resolver.

Componibilidad.- Grado con el cual un método de diseño asegura que los componentes de un programa (módulos), una vez diseñados y construidos, pueden reusarse para crear otros sistemas.

Comprensibilidad.- Facilidad de comprensión de un componente de programa sin referencia a otra información o módulos.

Continuidad.- Facilidad de hacer pequeños cambios en un programa y hacer que estos se manifiesten por sí mismos en cambios correspondientes solamente en no o unos pocos módulos más.

Protección.-Característica arquitectónica que reducirá la propagación de efectos colaterales si ocurre un error en un módulo dado.

Éstos criterios y principios de diseño presentados por Meyer pueden aplicarse a cualquier método de diseño (incluyendo diseño estructurado), no obstante el método de diseño orientado a objetos alcanza cada uno de los principios de manera más eficiente que otros enfoques y el resultado final es una arquitectura modular que permite cumplir con todos los principios de modularidad de una manera más eficiente.


Revisen en mapa conceptual de OO que comparto a continuación.

Lo conversaremos mas tarde.

Saludos

 Mapa Conceptual de OO



 Sesión de arquitectura de aplicaciones y diseño OO





Navegación

