

## UML PARA DESARROLLADORES I - ANÁLISIS y DISEÑO

Duración: 24 hrs.

Código: UMLAN

### Curso:

### Descripción del curso

Ingeniería de Requerimientos es la disciplina para desarrollar una especificación completa, consistente y no ambigua, la cual servirá como base para acuerdos comunes entre todas las partes involucradas y en dónde se describen las funciones que realizará el sistema. En este curso el instructor presentará los principios y técnicas necesarios de la Ingeniería de Requerimientos como pieza inicial de un proyecto exitoso que termine en un "contrato" que se pueda cumplir en un tiempo, presupuesto y alcance inicialmente establecidos, utilizando estándares mundialmente conocidos y aceptados por empresas líderes en el desarrollo de software (IEEE, CMMI, RUP).

### Dirigido a:

- Estudiante.
- Profesional.

### Objetivos:

El Participante al finalizar el curso será capaz de:

Entender la estructura de la metodología RUP (Rational Unified Process) y la notación UML (Unified Modeling Language) en el diseño de software. Analizar los requerimientos del cliente y ordenarlos mediante la estructura que ofrecen los casos de uso para llevarlo al mínimo detalle (especificación)

### REQUISITOS MÍNIMOS

Tener experiencia en la elaboración de sistemas informáticos.

Ingeniería de Reuqerimientos



## CONTENIDO

### Sesión 1

#### VISIÓN GLOBAL DEL PROCESO UNIFICADO DE DESARROLLO, RUP (ANÁLISIS)

Las cuatro fases que componen RUP son Inicio, Elaboración, Construcción y Transición. Cada una de estas fases está compuesta de iteraciones que pueden ser consideradas como pequeños proyectos que combinados representan la ruta que permitirá obtener una solución global para desarrollar una aplicación de software. En este módulo el instructor intentará determinar los conocimientos del grupo con respecto a gestión de proyectos y metodologías de análisis y diseño. El objetivo de este módulo es el que los participantes obtengan un claro entendimiento sobre RUP y las ventajas que ofrece sobre las metodologías.

#### INTRODUCCIÓN A LA ORIENTACIÓN A OBJETOS (DISEÑO)

Para iniciar el taller el instructor aplica una encuesta para evaluar el nivel de conocimientos sobre orientación a objetos de los participantes. Dependiendo de los resultados de dicha encuesta se explicarán y/o afianzarán los conceptos asociados a la orientación a objetos (encapsulación, herencia, polimorfismo y otros) mediante ejemplos.

### Sesión 2

#### CASOS DE USO (ANÁLISIS)

El modelo de casos de uso integra actores, casos de uso y las relaciones entre ellos. Los actores representan todas aquellas entidades que interactúan con el sistema, incluyendo lo que tradicionalmente se denomina usuarios. Cuando un actor interactúa con el sistema está ejecutando un caso de uso. Una buena especificación de caso de uso es una secuencia de transacciones que generan un resultado medible para el actor. El conjunto de casos de uso representan la funcionalidad del sistema. En este módulo el instructor ayudará a los asistentes a aplicar procesos para identificar actores, dibujar diagramas de contexto, identificar, diagramar, categorizar, extender y reutilizar casos de uso.

#### DIAGRAMA DE SECUENCIA (DISEÑO)

El diseño orientado a objetos permite construir los productos desarrollados durante la fase de diseño mediante la transformación de objetos candidatos en clases, la definición de protocolos de intercambio de mensajes entre todos los objetos, la definición de estructuras de datos y procedimientos, y la proyección de todo lo anterior sobre un lenguaje de programación orientado a objetos.

## CONTENIDO

### Sesión 3

#### CASOS DE USO (RELACIONES)

**(ANÁLISIS)**

En esta sesión el instructor mostrará al estudiante las relaciones que hay entre actores, casos de uso y entre ambos, poniendo ejemplos de aplicación para cada uno:

- Generalización
- Asociación:
  - I. Communicate
  - II. Include
  - III. Extend

#### DIAGRAMA DE COLABORACIÓN

**(DISEÑO)**

Este diagrama que puede ser obtenido a partir del diagrama de secuencia nos permite ver cada una de las clases con todas sus interacciones, ayudándonos a determinar de esta manera a modelar los atributos y operaciones que debe soportar de acuerdo al tipo de clase.

### Sesión 4

#### ESPECIFICACIÓN DE CASOS DE USO

**(ANÁLISIS)**

Los casos de uso encierran diversos escenarios del negocio que requieren ser detallados para su adecuada programación, caso contrario se contaría sólo con un caso general llevando al error de alcance y tiempo que no se contemplaría en el cronograma además de contar con casos de uso ambiguos. Esta ambigüedad que genera retrabajo y sobrecostos puede ser mitigada con una adecuada especificación de casos de uso. En este módulo el instructor mostrará a los asistentes la plantilla de RUP correspondiente a una buena especificación de casos de uso, sus secciones y usos tanto individuales como conjuntos.

#### DIAGRAMA DE CLASES

**(DISEÑO)**

Una vez obtenidas las clases que participarán en el sistema en los diagramas de interacción previos (secuencia y colaboración), procedemos a revisar la interacción entre las clases apuntando a separarlas en grupos de clases que se convertirán posteriormente en las capas del sistema.



## CONTENIDO

### Sesión 5

#### **PROTOTIPOS**

**(ANÁLISIS)**

El uso de prototipos ayuda a presentar una imagen previa de cómo será el software a desarrollar brindando una visión de "maqueta" al cliente de tal forma que nos permite fijar con una imagen lo que es difícil para el cliente indicar con palabras. En este módulo el instructor mostrará los alcances del prototipo y su relación con la especificación de casos de uso y pruebas preliminares. Se mostrará el uso de herramientas libres y de pago para trabajo colaborativo, tradicionales de análisis y diseño funcional.

#### **DIAGRAMA DE CLASES (APLICACIÓN)**

**(DISEÑO)**

### Sesión 6

#### **DIAGRAMAS DE ESTADO**

**(ANÁLISIS)**

Permite mostrar el comportamiento de un objeto a lo largo de su vida, esto nos permitirá saber cuáles son las entidades del sistema identificadas desde el negocio que necesitan que les hagamos un "seguimiento", atendiendo a los valores que toma (estados) y qué los "empuja" a cambiar de estado.

#### **DIAGRAMA DE ESTRUCTURA COMPUESTA**

**(DISEÑO)**

El diagrama de estructura compuesta, introducido en UML 2.0, describe los participantes y las relaciones internas de un clasificador individual, como un objeto, guión de uso, clase o colaboración. Un diagrama de estructura compuesta a menudo se adjunta a un diagrama más grande para describir a estructura o contexto internos de un elemento único en el diagrama padre. El diagrama muestra los roles y relaciones entre las instancias de objeto en un componente de sistemas.



## CONTENIDO

### Sesión 7

#### **DIAGRAMAS DE TIEMPO**

**(ANÁLISIS)**

El propósito primario del diagrama de tiempos es mostrar los cambios en el estado o la condición de una línea de vida (representando una Instancia de un Clasificador o un Rol de un clasificador) a lo largo del tiempo lineal. El uso más común es mostrar el cambio de estado de un objeto a lo largo del tiempo, en respuesta a los eventos o estímulos aceptados. Los eventos que se reciben se anotan, a medida que muestran cuándo se desea mostrar

#### **IMPLEMENTACIÓN DEL SISTEMA**

**(DISEÑO)**

El sistema a implementar está conformado por una gran cantidad de clases y otros elementos que se agrupan en componentes dando lugar a modelos de "n" capas, los cuales son instanciados en servidores de acuerdo a las necesidades del negocio para buscar un adecuado equilibrio del uso de los recursos que nos lleven a una máxima performance; esto se comunica a través de los diagramas de componentes y distribución.

### Sesión 8

#### **DIAGRAMA DE ACTIVIDADES**

**(ANÁLISIS)**

Describe la lógica de un procedimiento, un proceso de negocio o workflow, esto no sólo nos permite revisar la secuencia de actividades, sino también qué entidades son las responsables de ellas. El diseño de este diagrama nos permite describir cómo se lleva a cabo el proceso que debemos automatizar.

#### **EVALUACIÓN FINAL**

**(DISEÑO)**


 CONTENIDO


 Sesión 9


 EVALUACIÓN FINAL

(ANÁLISIS)

Como evaluación final el participante expondrá su trabajo de análisis sobre un caso real que se desarrollará desde la primera sesión.


 EVALUACIÓN

La evaluación de cursos será totalmente práctica. Se realizarán entre 4 y 5 prácticas de las cuales se eliminará la nota más baja y se obtendrá un promedio (PP). Durante la última sesión se realizará un examen final (EF), el cual se promediará con la nota de prácticas y de esta manera se tendrá la calificación final.


 PROMEDIO DE PRACTICAS

$$PP = \frac{(PR1 + Pr2 + Pr3 + PR4) - \text{Menor (PR)}}{3}$$

## Nota Final:

$$NF = \frac{(PP + EF)}{2}$$

