



## **Curso Taller de Arquitectura de Software usando UML**

### **Presentación:**

Este curso comprende las técnicas necesarias para el modelamiento de sistemas a través de los diagramas definidos por UML (Unified Modelling Language), que hoy en día es el estándar más utilizado y aceptado para realizar modelos más completos, pensando en las nuevas tendencias del desarrollo orientado por objetos y a componentes.

### **Objetivos:**

Desarrollar capacidades de análisis y diseño orientado a objetos junto con el aprendizaje de la notación y los elementos que componen los diagramas estáticos y dinámicos de UML y utilizarlos dentro de las etapas de análisis y diseño del desarrollo de software, de forma tal que se puedan obtener los elementos de calidad deseados en el sistema a desarrollar.

### **Contenido:**

#### **INTRODUCCIÓN**

- Qué es UML
- Antecedentes de UML
- La importancia del Modelado Visual

#### **MODELO DE CASOS DE USO**

- Actores
- Casos de Uso
- Diagrama de Casos de Uso
- Paquetes de Casos de Uso
- Relaciones <<include>> y <<extend>>
- Puntos de extensión
- Paquetes de Casos de Uso



## **ESPECIFICACIÓN DE CASOS DE USO (FLUJOS DE EVENTOS)**

- Documentación de un Caso de Uso
- Caso de Uso de Alto Nivel
- Flujos Primarios, Alternos y Excepcionales
- Precondiciones y postcondiciones
- Requerimientos especiales del caso de uso
- Escenarios
- Las Pruebas y los Casos de Uso

## **MODELO CONCEPTUAL**

- Conceptos
- Atributos
- Relación de Asociación
- Diagrama del Modelo Conceptual
- Identificación de conceptos mediante un análisis de Casos de Uso

## **DIAGRAMAS DE SECUENCIA**

- Clases y Objetos
- Línea de Vida
- Foco de Control
- Mensajes y Operaciones
- Diagrama de Secuencia
- Diagrama de Colaboración
- Diferencias entre el Diagrama de Colaboración y de Secuencia
- Impacto del Diagrama de Interacción en el Diagrama de Clases

## **PATRONES DE ASIGNACIÓN DE RESPONSABILIDADES**

- Qué son los patrones
- Patrones para la Asignación de Responsabilidades
- Alta Cohesión y Bajo Acoplamiento
- Diseño en 3 Capas



## **DIAGRAMAS DE CLASES**

- Clases
- Atributos
- Operaciones
- Alcance de Atributos y Operaciones
- Relaciones de Asociación, Agregación y Dependencia
- Generalización: la implementación de la herencia
- Visibilidad entre Clases
- Navegabilidad
- Multiplicidad
- Completando el diagrama de clases mediante el diagrama de interacción
- Paquetes de clases

## **DIAGRAMAS DE COMPONENTES**

- Componentes
- Interfases
- La interfase en el diagrama de clases
- La interfase en el diagrama de componentes
- Relación de Realización
- Tipos de Componentes
- Dependencias

## **DIAGRAMAS DE DISTRIBUCIÓN**

- Nodos
- Asociaciones entre Nodos
- Dispositivos
- Diagrama de Distribución

## **IMPLEMENTACIÓN EN EL LENGUAJE SELECCIONADO**

- Interpretación del Diagrama de Clases
- Interpretación del Diagrama de Secuencia
- Interpretación del Diagrama de Componentes

## **GENERACIÓN DE CÓDIGO**

- Uso de herramientas CASE para la Generación de Código
- Generación de Código
- Ingeniería Inversa
- Round Trip Engineering

## **SEGUNDO CASO PRÁCTICO**

- Segundo Caso Práctico para Repasar los Conceptos Aprendidos en una simulación de proyecto y donde además se incluyen nuevos artefactos de UML:
- Planeación del caso práctico bajo RUP: Identificación de fases, actividades y planeación de tiempos
- Entrevista de requerimientos
- Modelado de negocios
- Modelo de casos de uso (a partir de los procesos analizados en el diagrama de actividad)
- Modelo Conceptual
- Diagrama de estados
- Diagrama de interacción
- Diagrama de clases
- Diagrama de componentes
- Diagrama de despliegue
- Codificación
- Generación de código e ingeniería inversa

## **DIAGRAMAS DE ACTIVIDAD (VISTO DENTRO DEL SEGUNDO CASO PRÁCTICO)**

- Actividades
- Transiciones
- Decisiones
- Carriles y responsabilidades dentro del proceso
- Trabajo en paralelo (barras de sincronización)



- Paso de las actividades al modelo de casos de uso
- Modelado de casos de uso con diagramas de actividad
- Modelado de procesos de negocio con diagramas de actividad

## **DIAGRAMAS DE ESTADO (VISTO DENTRO DEL SEGUNDO CASO PRÁCTICO)**

- Estados
- Transiciones
- Eventos
- Acciones
- Condiciones de guardia
- Superestados
- Historia
- Modelado de análisis y diseño con un diagrama de estados

### **Habilidades previas recomendadas:**

Con la finalidad de que el participante obtenga el máximo aprovechamiento de este curso de UML, es recomendable que cuente con los conceptos fundamentales de la orientación a objetos. También es bastante deseable que el profesional ostente experiencia previa en el uso de alguno de los lenguajes modernos de programación orientado a objetos.

**Duración: 20 horas**