

# Metodología RUP

---

## *Acerca de la Metodología RUP*

**RUP (Rational Unified Process):** Es un proceso de ingeniería de software que proporciona un enfoque disciplinado para la asignación de tareas y responsabilidades dentro de una organización de desarrollo, con el fin de garantizar la producción de software de alta calidad que satisfaga las necesidades de sus usuarios finales, dentro de un lapso de tiempo y presupuesto predecibles.

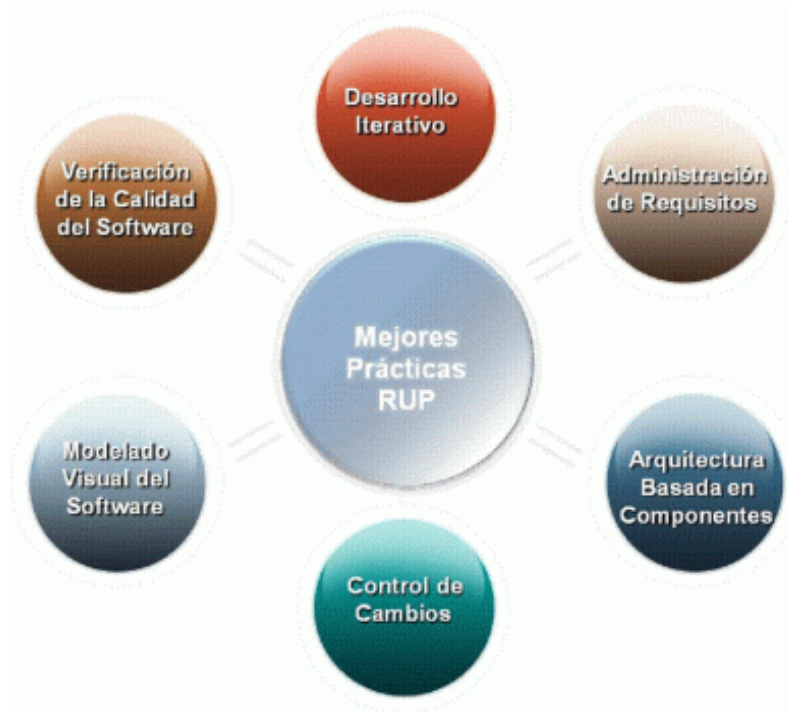
### **Antecedentes:**

- \* 1967 Surge la Metodología Ericcson (Ericcson Approach) elaborada por Ivar Jacopson. Componentes y Casos de Uso.
- \* 1987 – 1995 Jacobson fundó la compañía Objectory AB. Proceso de desarrollo Objectory (abreviación de Object Factory).
- \* 1995 – 1997 Rational Software Corporation adquiere Objectory AB y se desarrolla Rational Objectory Process (ROP) a partir de Objectory 3.8 y del Enfoque Rational (Rational Approach). UML.
- \* 1998 Rational Unified Process.
- \* 2003 IBM Adquiere Rational Software.

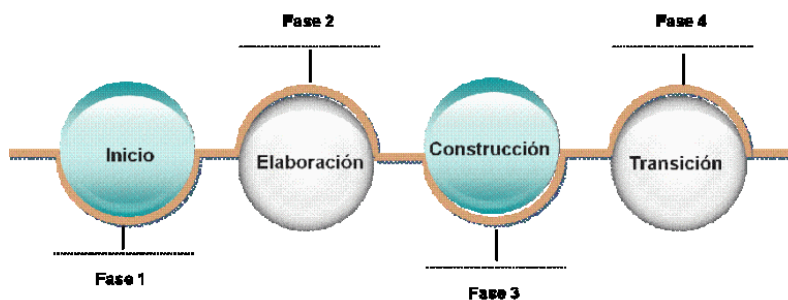
### **Características Principales**



### **Las Seis Mejores Prácticas**



## Ciclo de Vida



**Inicio:** Alcanzar un acuerdo entre todos los interesados respecto a los objetivos del ciclo de vida para el proyecto, generando el ámbito del proyecto, el caso de negocio, síntesis de arquitectura posible y el alcance del proyecto.

**Elaboración:** Establecimiento de la línea base para la Arquitectura del sistema y proporcionar una base estable para el diseño y el esfuerzo de implementación de la siguiente fase, mitigando la mayoría de los riesgos tecnológicos.

**Construcción:** Completar el desarrollo del sistema basado en la línea base de la arquitectura.

**Transición:** Garantizar que el software está listo para entregarlo a los usuarios.

## Iteraciones:

\* De Proceso:

1. Modelado del Negocio
2. Requerimientos
3. Análisis y Diseño
4. Implementación
5. Pruebas
6. Despliegue.

\* De Apoyo o Soporte:

1. Entorno
2. Gestión del Proyecto.
3. Gestión de Configuración y Cambios.

### **Artefactos:**

\* Inicio:

1. Documento Visión
2. Especificación de Requerimientos

\* Elaboración:

1. Diagrama de Casos de Uso.

\* Construcción:

1. Diagrama de clases.
2. Modelo E-R.
3. Diagrama de Secuencia.
4. Diagrama de Estados.
5. Diagrama de Colaboración.
6. Modelo del Dominio.

**Roles en RUP:** Un rol define el comportamiento y responsabilidades de un individuo, o de un grupo de individuos trabajando juntos como un equipo. Una persona puede desempeñar diversos roles, así como un mismo rol puede ser representado por varias personas.

Las responsabilidades de un rol son tanto el llevar a cabo un conjunto de actividades como el ser el 'dueño' de un conjunto de artefactos.

Los roles se distribuyen entre los miembros del proyecto y que definen las tareas de cada uno y el resultado (artefactos) que se espera de ellos.

Actividades: Las actividades tienen un objetivo concreto, normalmente expresado en terminos de crear o actualizar algún producto.

Productos: Los productos son los resultados tangibles del proyecto, las cosas que va creando y usando hasta obtener el producto final.

### **Analistas:**

- \* Analista de procesos de negocio.
- \* Diseñador del negocio.
- \* Analista de sistema.
- \* Especificador de requisitos.

### **Desarrolladores:**

- \* Arquitecto de software.
- \* Diseñador
- \* Diseñador de interfaz de usuario
- \* Diseñador de cápsulas.
- \* Diseñador de base de datos.
- \* Implementador.
- \* Integrador. Gestores:
- \* Jefe de configuración.
- \* Jefe de pruebas
- \* Jefe de despliegue
- \* Ingeniero de procesos
- \* Revisor de gestión del proyecto
- \* Gestor de pruebas.

### **Apoyo:**

- \* Documentador técnico
- \* Administrador de sistema
- \* Especialista en herramientas
- \* Desarrollador de cursos
- \* Artista gráfico

### **Especialista en pruebas:**

- \* Especialista en Pruebas (tester)
- \* Analista de pruebas
- \* Diseñador de pruebas

### **Otros roles:**

- \* Stakeholders
- \* Revisor
- \* Coordinación de revisiones
- \* Revisor técnico

## Grupo de Trabajo y sus Roles en RUP

