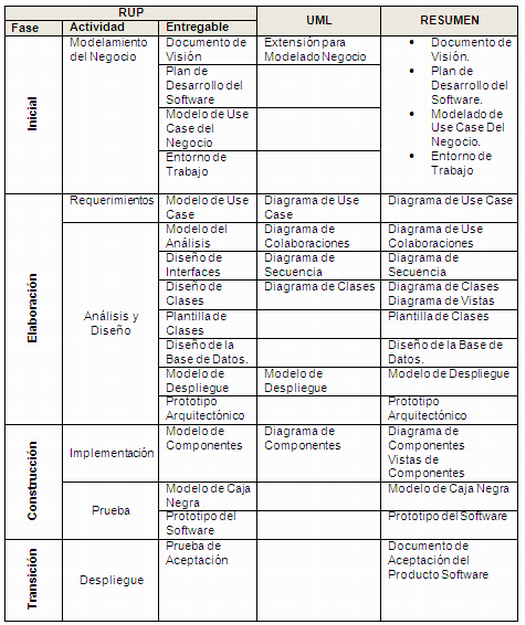
**Metodología RUP**



**Plantillas:**

**Inicio**

* **Documento de visión:** Un documento de visión es el principal artefacto, en el cual se define el alcance de alto nivel y propósito de un programa, producto o proyecto. Es una declaración clara del problema, la solución propuesta, y las características de alto nivel de un producto que ayudan a establecer las expectativas y reducir los riesgos de efecto del mismo. Para el desarrollo de una visión el analista de productos propietario o un negocio trabaja con las partes interesadas para desarrollar un documento de visión que define el ámbito de alto nivel y la finalidad del producto.
* **Plan de desarrollo de software:** El Plan de Desarrollo del Software describe el plan global usado para el desarrollo de un sistema. El detalle de las iteraciones individuales se describe en los planes de cada iteración, documentos que se aportan en forma separada. Durante el proceso de desarrollo en el artefacto “Visión” se definen las características del producto a desarrollar, lo cual constituye la base para la planificación de las iteraciones.

**Elaboración**

**Requisitos**

* **Casos de uso:** Los casos de uso se crean para refinar un conjunto de requisitos de acuerdo con una función o tarea. En lugar de la tradicional lista de requisitos que quizá no trate de forma directa el uso de la solución, los casos de uso reúnen requisitos comunes basados en el tipo de función u objetivo. Los casos de uso definen qué harán los usuarios o funciones en la solución y un proceso empresarial define cómo realizarán esas funciones.

**Análisis y diseño**

* **SAD** (Documento de arquitectura de software): Este Documento de Arquitectura de Software, abarca absolutamente a todo el sistema. Habitualmente se organiza por vistas, a través de las cuales pueden especificarse los distintos aspectos técnicos y funcionales, así como también las decisiones involucradas en cada incumbencia. Es un documento "vivo" e iterativo, las Vistas pueden ser escritas en distintas fases del proyecto.

Dentro de este documento se contemplan los siguientes entregables:

- Diseño de clases

- Diseño de interfaces

- Diseño de bases de datos

- Modelo de despliegue

**Construcción**

**Implementación**

* **Modelo de componentes:** Un diagrama de componentes representa cómo un sistema de software es dividido en componentes y muestra las dependencias entre estos componentes. Los componentes físicos incluyen componentes, interfaces, relaciones de dependencia, generalización, asociación y paquetes o subsistemas.

**Prueba**

* **Prototipo de software:** Los prototipos son una representación limitada de un producto, permite a las partes probarlo en situaciones reales o explorar su uso, creando así un proceso de diseño de iteración que genera calidad.

**Transición**

* **Prueba de aceptación:** Se explican los propósitos y enfoques de las pruebas, el plan de trabajo, los procedimientos operacionales, las herramientas necesarias y las responsabilidades.

**Documento de visión.**

LÍNEA BASE

INICIO

**Plan de desarrollo de**

**Software.**

ELABORACIÓN

**Casos de uso**

**DAS**

CONSTRUCCIÓN

**Modelo de componentes**

**Prototipo**

TRANSICIÓN

**Pruebas de aceptación**