

# Diagrama de Secuencia del Sistema



English



## Entradas

- Diagrama de Casos de Uso
  - ◆ ¿Se puede emplear gestión de la configuración?: **Si**
- Documento de Especificación de Requisitos
  - ◆ ¿Se puede emplear gestión de la configuración?: **Si**



## Salidas

- Diagramas de Secuencia del Sistema
  - ◆ ¿Se puede emplear gestión de la configuración?: **Si**



## Solución



## Proceso



## Tiempo de Desarrollo

- ◆ Para adquirir el conocimiento necesario para desarrollar el producto software: 4 horas
- ◆ Para crear el Patrón de Producto: 2 horas
- ◆ Para aplicar el Patrón de Producto: 2 días.



## Video Explicación



## Patrones Relacionados

- Contrato de Operación
- Diagrama de Casos de Uso
- Diagrama de Casos de Uso de Alto Nivel
- Diagrama de Casos de Uso en Formato Expandido



## Controladores de Calidad

- Ninguno



## Plantillas

- Elementos del diagrama de secuencia del sistema
- Plantilla del diagrama de secuencia del sistema



## Ejemplos

- Ejemplo de diagramas de secuencia del sistema "Login", "Cerrar Sesión" y "Modificar datos" en el contexto de una red social



## Herramientas de Soporte

- Argo UML (Open Source Project)
- Dia (GNOME)
- Rational Software Modeler (IBM)
- StarUML (Open Source Project)
- Visual Paradigm for UML (Visual Paradigm)



## Contexto Inicial

Se está desarrollando un sistema software empleando el Método de Craig Larman, ya se ha creado el diagrama de casos de uso y se quieren detallar más explícitamente los casos de uso y las funcionalidades del sistema que representan.



## Contexto Resultante

Se cuenta con una representación gráfica que muestra una ejecución real de cada caso de uso, con las posibles bifurcaciones y alternativas resueltas de forma particular.



## Problema

Se desea representar, de manera más explícita, la interacción de los actores (identificados en el diagrama de casos de uso) con el sistema software que se está desarrollando, desde la perspectiva de cada uno de los casos de uso.



## Restricciones (*Forces*)

- **Tipo de Organización:** PyMEs, Grandes Empresas.
- **Tipo de Sistema:** Aplica a todos los tipos de sistemas.
- **Paradigma de Programación:** Orientado a Objetos.



## Roles

- Analista
- Cliente
- Jefe de Proyecto
- Usuarios del Sistema



## Lecciones Aprendidas

- Ninguna



## Nivel de Madurez

- Nivel de Madurez 2 del CMMI.



## Conocimientos y Habilidades Básicos



### Conocimientos

- Definición de requisitos software (funcionales y no funcionales).
- [Método de Craig Larman](#)
- Nociones de UML, en especial de la técnica de modelado de casos de uso y de diagramas de secuencia.



### Habilidades

- Capacidad de análisis.
- Capacidad de abstracción.



## Recursos de Información

- Ferré Grau, X & Sanchez-Segura, M. (2004). *Desarrollo Orientado a Objetos con UML*. Recuperado el 2009-11-26 de <http://www.clikear.com/manuales/uml/index.aspx>.
  - Fowler, M. (2004). *UML distilled: a brief guide to the standard object modelling language*. Addison-Wesley.
  - Larman, C. (2001). *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process* (2da. ed.). Prentice Hall.
  - Jacobson, I. & Booch, G. (1999). *The unified software development process*. Addison-Wesley.
  - Rumbaugh, J. & Jacobson, I. (2005). *The unified modeling language reference manual*. Addison-Wesley.
  - Rumbaugh, J. (2005). *Object-Oriented Modelling and Design*. Prentice Hall.
-