

Diagrama de Casos de Uso



English



Entradas

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



Salidas

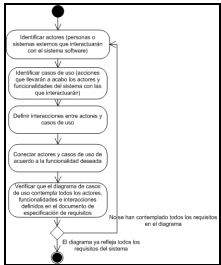
- Diagrama de Casos de Uso (empleando UML)
 - ◆ ¿Se puede emplear gestión de la configuración?: **Si**



Solución



Proceso



Tiempo de Desarrollo

- ◆ Para adquirir el conocimiento necesario para desarrollar el producto software: 1 día
- ◆ Para crear el Patrón de Producto: 1 día
- ◆ Para aplicar el Patrón de Producto: 1 día



Video Explicación

- ◆ No aplica



Patrones Relacionados

- Diagrama de Casos de Uso de Alto Nivel
- Diagrama de Casos de Uso en Formato Expandido
- Especificación de Requisitos



Controladores de Calidad

- Ninguno



Plantillas

- Notación UML de los Casos de Uso
- Plantilla para documentar los casos de uso
- Plantilla diagrama casos de uso



Ejemplos

- Ejemplo de caso de uso "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 cuenta con el documento de especificación de requisitos, y tomándolo como punto de partida, se desea crear una representación que muestre de manera global el funcionamiento de un sistema software; se desea incluir en dicha representación las interacciones entre el sistema software que se va a desarrollar y sus actores (ya sean personas o sistemas externos).



Contexto Resultante

Se cuenta con un diagrama basado en el estándar UML que representa la funcionalidad completa de un sistema software satisfaciendo la especificación de requisitos, detallando la forma en que el sistema podrá ser utilizado e indentificando las interacciones entre el sistema software y sus actores.



Problema

Se desea crear una representación del funcionamiento de un sistema software desde la perspectiva del usuario. Dentro de dicha representación se desea detallar los distintos tipos de usuarios y sistemas externos que interactuarán con el sistema software que se pretende desarrollar. El grupo de desarrollo ya cuenta con el documento de especificación de requisitos.



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

- Beneficios de utilizar este patrón
 - ◆ Se ofrece una descripción gráfica del funcionamiento del sistema software y de la forma en que será utilizado.
 - ◆ Es posible describir las interacciones entre el sistema y sus actores.
 - ◆ Se cuenta con una representación que es de gran utilidad para mostrar al cliente cual será la funcionalidad del sistema y que roles están involucrados.



Nivel de Madurez

- Nivel de Madurez 2 del CMMI.



Conocimientos y Habilidades Básicos



Conocimientos

- Definición de requisitos software (funcionales y no funcionales).
- Nociones de UML, en especial de la técnica de modelado de casos de uso.



Habilidades

- Capacidad de abstracción.



Recursos de Información

- Amescua A., et al. (2003). *Análisis y Diseño Estructurado y Orientado a Objetos del Sistema Informáticos*. McGraw Hill/Interamericana de España, S.A.U.
 - 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.
 - 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.
-