

System Sequence Diagram



Español



Entries

- Use Case Diagram
 - ◆ Can it be used configuration management ? : **Yes**
- Requirements Specification Document
 - ◆ Can it be used configuration management ? : **Yes**



Exit

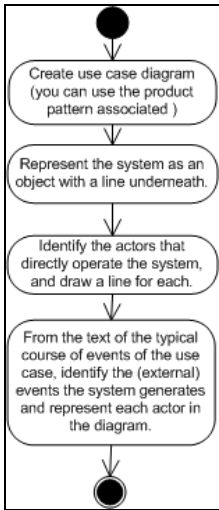
- System Sequence Diagrams
 - ◆ Can it be used configuration management ? : **Yes**



Solution



Process



Development time

- ◆ To acquire the necessary knowledge to develop the software product: 4 hours
- ◆ To create the Product Pattern: 2 hours
- ◆ To apply the Product Pattern: 2 days



Explanatory video



Related Patterns

- [Operation Contract](#)
- [Use Case Diagram](#)
- [High Level Use Case Diagram](#)
- [Use Case Diagram in Expanded Format](#)



Quality Controllers

- None



Templates

- [Basic elements of the sequence diagram.](#)
- [Template System Sequence Diagram](#)



Examples

- [Example System Sequence Diagram "Login", "LogOut" and "ModifyData" in the context of a social network.](#)



Support Tools

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



Initial Context

It is developing a software system using the Craig Larman Method, it is already created the use case diagram and it wanted to detail more explicitly, the use cases and features of the system they represent.



Result Context

It has a graphical representation showing a real execution of each use case, with the possible bifurcations and resolved alternatives in particular way.



Problem

It is desired to represent, more explicitly, the interacción of actors (identified in the use case diagram) with the software system being developed from the perspective of each of the use cases.



Restrictions (*Forces*)

- **Type of Organization:** SMEs, Large Companies.
- **System Type:** It applies to all types of systems.
- **Programming Paradigm:** OO (Object Oriented).



Roles

- Analyst
- Customer
- Project Manager
- System Users



Lessons Learned

- None



Capability Level

- Capability Level 2 CMMI.



Basic Knowledge and Skills



Knowledge

- Definition of software requirements (functional and non-functional).
- [Craig Larman Method](#)
- UML notions, especially the technique of modeling use cases and sequence diagrams.



Abilities

- Capacity of abstraction.
- Capacity of Analysis.



Information Resources

- 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.
-