

High Level Use Case Diagram



Entries

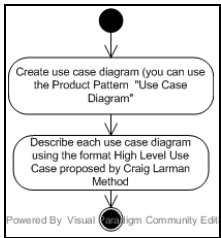
- Use Case Diagram
 - ◆ Can it be used configuration management?: **Yes**
- Requirements Specification Document
 - ◆ Can it be used configuration management?: **Yes**

Exit

- High Level Use Case Diagram (using [Craig Larman Method](#))
 - ◆ Can it be used configuration management?: **Yes**

Solution

Process



Development time

- ◆ To acquire the necessary knowledge to develop the software product: 2 hours
- ◆ To create the Product Pattern: 2 hours
- ◆ To apply the Product Pattern: 10 hours

Explanatory video

- ◆ Not applicable



Related Patterns

- [Use Case Diagram](#)
- [Use Case Diagram in Expanded Format](#)
- [Requirements Specification](#)

Quality Controllers

- Not applicable

Templates

-  [High Level Use Case Format](#)
-  [Template High Level Use Case](#)

Examples

-  [Example High Level Use Case "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 of the system..



Result Context

It has been described in general terms all use cases of the use case diagram of the system, using the format proposed in **Craig Larman Method**.



Problem

It is desired to create an overview of all use cases diagram, including its type and the actors interacting in each.



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

- Benefits of using this pattern
 - ◆ A graphic description of how the software system works and how it will be used is provided.
 - ◆ It is possible to describe the interactions between the system and its actors.
 - ◆ It has a representation that is useful to show the client what will be the system's functionality and roles are involved.
 - ◆ It has a high level description of the purpose of each use case diagram.



Capability Level

- Capability Level 2 CMMI.



Basic Knowledge and Skills



Knowledge

- Definition of software requirements (functional and non-functional).
- Craig Larman Method.
- Notions of UML, especially the technique of modeling of use cases.



Abilities

- Capacity of abstraction.



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.
-