

# Modify Integration

 Español

 **Entries**

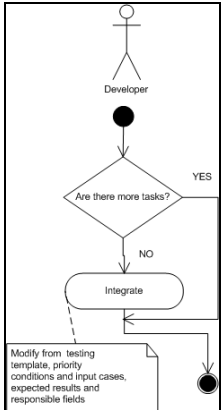
- Testing

 **Exit**

- Testing


 **Solution**

 **Process**



 **Development time**

- ◆ To acquire the necessary knowledge to develop the software product:
- ◆ To create the Product Pattern: 45 minutes.
- ◆ To apply the Product Pattern:

 **Explanatory video**

- ◆ Not applicable

 **Related Patterns**


- [Integration Test Pattern](#)

 **Quality Controllers**

- None

 **Templates**

- Integración\_Continua.doc

 **Examples**

- None

 **Support Tools**

- It will need a text editor such as [OpenOffice Writer](#) o [Microsoft Word](#).
- As well as a tool [Visual Paradigm](#) for UML for performing exposed diagrams.



## Initial Context

This product can be used in any project in which it is present the concept of continuous integration and collective property, by which any member of the team can modify it by another member.



## Result Context

Developers get conclusions of the integration tests that were unsatisfactory or can be simplified.



## Problem

Developers should keep the principles of continuous integration so that you never lose too much time in identifying a problem caused by a mistake made in recent weeks. In the integration should be taken into account that the entire system (code and test) must be reviewed by any team member.



## Restrictions (*Forces*)

- **Characteristics of organizations:** This pattern can be used in existing projects in any company.
- **System Type to develop** This product can be used in projects in which user requirements are changing.
- **Type of Customer:** It must exist or be achieved, the target area development business being involved in achieving it.
- **Heuristics of use:** If it need urgent application or dispose of some of their funcionalidades.



## Roles

- Developers (2-12)



## Lessons Learned

- Recode continually has the effect of splitting the system into lots of small items and lots of small methods. This diminishes the likelihood that two pairs of developers change the same class or change the same method at the same time. If they do, the effort required to reconcile changes is small, because one of each members represent only a few hours of development.
- The modification of integration based on unsatisfactory evidence or identifying elements can be simplified even more.
- The tests file is a generic document for all test types with unit testing box checked, which bind all test cases together for a task. We must pay special attention to tests with unsatisfactory integration because they have to be modified. In this document their responsible and history associated is reflected. Corresponding to the test results and associated comments field is filled.



## Capability Level

- Not applicable.



## Basic Knowledge and Skills



### Knowledge

- Knowledge of coding standard that defines the shared code ownership and the rules for writing and documenting code and communication between different pieces of code developed by different teams. Programmers have to follow the so that the code in the system look like if it had been written by one person.
- Knowledge of the common vision of how the program works in which the activities take place.



## Abilities

- Ability to work in group. All on an XP computer contribute in any way they can.
- Predicting what will be completed by the deadline, and determining what to do next.
- Programming capability in pairs. Besides to generate better code and tests, used to communicate knowledge through teams.



## Information Resources

- Álvarez, José R. y Arias Manuel. Método Extreme programming. Recuperado el 2010-03-05 de <http://www.ia.uned.es/ia/asignaturas/adms/GuiaDidADMS/node61.html>
  - Anaya Villegas, Adrian. A proposito de programación extrema XP(extreme Programming). Recuperado el 2010-02-10 de <http://www.monografias.com>
  - Beck, K.(2000), Una explicación de la programación extrema. Aceptar el cambio. Ed. Addison Wesley.
  - De Seta, Leonardo. Una introducción a Extreme Programming. Recuperado el 2010-03-02 de <http://www.dosideas.com/noticias/metodologias/822-una-introduccion-a-extreme-programming.html>
  - Extreme Programming: A gentle introduction. Recuperado el 2010-03-15 de <http://www.extremeprogramming.org/>
  - Joskowicz, José. Reglas y prácticas en Xtreme Programming. Recuperado el 2010-03-15 de <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>
  - Letelier, Patricio y Panadés M<sup>a</sup> Carmen. Metodologías Ágiles en el desarrollo de software: extreme programming. Recuperado el 2010-03-15 de <http://www.willydev.net/descargas/masyxp.pdf>
  - Newkirk, James y Martin, Robert C.(2001), La programación Extrema en la Práctica. Ed Addison Wesley.
-