

Integrate



Español



Entries

- Testing



Exit

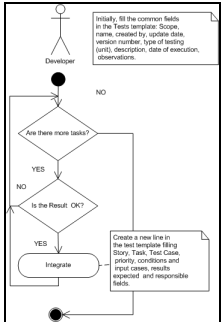
- Testing



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

- Unit Testing Pattern
- Modify Test Cases Pattern
- Integration Test Pattern



Quality Controllers

- None



Templates

- Pruebas.xls



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 after the design or implementation code or modification should be integrated with other work done by the team.



Result Context

The work done by a team member works 100% with the rest of work done by the whole team.



Problem

Developers should maintain balanced and integrated their work with the rest of the team. Continuous integration means to be integrated individual work every few hours (1 day maximum).

In the integration the small software modules that we have joint together and it seems to work, but those little pieces have just passed unit tests which have not had to interact with other modules. The problem is that there are errors that only occur in this interaction and that is precisely the problem of integration.



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 for use:** If you need urgent application or dispose of some of its functionality.



Roles

- Developers (2-12)



Lessons Learned

- It is important to have an integration tool that supports rapid integration build / test cycle. The group of test should run in a few minutes. Moreover, collisions should require relatively little effort. It dramatically reduces project risk. If two people have different ideas about the form a function of a part of the code they will know within hours. They never devoted many hours to pursue a mistake that was made sometime in the last few weeks.
- We need to integrate each time a class or method is developed for balance and unity of the whole team work.
- 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 those cases of unsatisfactory test, it must be modified. In this document their responsible and history associated is reflected. You can add or remove test cases as deemed appropriate.



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%20Jose%20Joskowicz.pdf>
- Letelier, Patricio y Panadés M^a Carmen. Metodologías Ágiles en el desarrollo de software: extreme programming. Recuperado el 2010-03-15 de <http://www.willydev.net/descargas/masyxp.pdf>

