

Unit Testing



Español

Entries

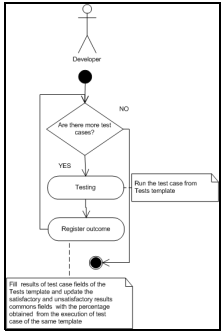
- Testing.xls

Exit

- Testing.xls

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

- [Writing Functional Tests Pattern](#)
- [Integrate Pattern](#)
- [Modify Test Cases Pattern](#)
- [Writing Story Pattern](#)

Quality Controllers

- None

Templates

- Pruebas.xls

Examples

- None

Support Tools

- It will need a text editor such as [OpenOffice Writer](#) o [Microsoft Word](#).
- It will require a spreadsheet editor such as [OpenOffice Calc](#) o [Microsoft Excel](#)
- 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 should be tested separately tasks to be implemented according to a set of test cases previously identified.

This indicates the programmer what to do when he codes. The requirements appear as unit evidence. You know when it is over because all unit test are exceeded .The benefit that have about it the design, is that in the unit test is set what is important to the customer.



Result Context

Developers get a series of unit results for each test case associated with a task.



Problem

Developers should be able to get unit testing test cases, separately, satisfactory to continue the process.

A small problem is that the test itself may have errors and this leads to unit tests do not perform their activities properly.



Restricciones (*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 you need urgent application or dispose of some of their funcionalidades.



Roles

- Developers (2-12)



Lessons Learned

- If the unit tests do not work must be corrected as soon as possible. This is the most important part of team work. Because of this, balance must be kept in the time devoted to the performance of the tests and the correction of non satisfactory.
- The pair of developer carried on each unit test cases separate tests.
- The Pruebas.xls file is a generic document for all test types with unit testing box checked, which brings together all the test cases for a task.



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/assignaturas/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://ie.fing.edu.uy/~josej/docs/XP%20-%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>
- Newkirk, James y Martin, Robert C.(2001), La programación Extrema en la Práctica. Ed Addison Wesley.