

# Clasificar por Valor



English



## Entradas

- Historias de Negocio



## Salidas

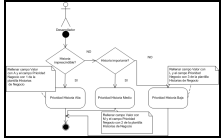
- Historias de Negocio



## Solución



## Proceso



## Tiempo de Desarrollo

- ◆ Para adquirir el conocimiento necesario para desarrollar el producto software:
- ◆ Para crear el Patrón de Producto: 15 minutos.
- ◆ Para aplicar el Patrón de Producto:



## Video Explicación

- No aplica



## Patrones Relacionados

- Patrón Estimar Historia
- Patrón Dividir Historia
- Patrón Establecer Velocidad



## Controladores de Calidad

- Ninguno



## Plantillas

Historias\_Negocio.doc  
Tareas\_Historias\_Ver\_X\_DyD\_Y.doc



## Ejemplos

- Ninguno



## Herramientas de Soporte

- Será necesario un editor de texto como [OpenOffice Writer](#) o [Microsoft Word](#).
- Además de una herramienta como [\[Visual Paradigm for UML\]](#) para la realización de los diagramas expuestos.



## Contexto Inicial

Este producto puede usarse en cualquier proyecto que requiera una priorización por parte del área de negocio de las historias escritas por ellos mismos y posteriormente estimadas por el área de tecnología. Esta priorización esta marcada por el valor que la nueva funcionalidad aportara al negocio cuando sea desarrollada. Dentro de esta clasificación se establecen tres grupos, las historias que describen funcionalidades imprescindibles, las importantes y las misceláneas.



## Contexto Resultante

Se obtendrá el conjunto de historias estimadas por tecnología pero ordenadas según el valor aportado al negocio. Dentro del ese conjunto se tendrán 3 subconjuntos (imprescindibles, importantes y miscelánea).



## Problema

El área de negocio debe tener muy clara la función que representa cada historia en el aplicativo a desarrollar y no dejarse llevar por las historias que representan funcionalidades irrelevantes (miscelánea) en vez de las imprescindibles, por ejemplo.



## Restricciones (*Forces*)

- **Características de las organizaciones:** Este patrón puede utilizarse en los proyectos existentes en cualquier tipo de compañía.
- **Tipo de sistema a desarrollar:** Este producto puede utilizarse en proyectos en los que los requerimientos de usuario sean cambiantes.
- **Tipo de Cliente:** Debe existir, o debe conseguirse, que el área de negocio destinataria del desarrollo se implique en la consecución del mismo.
- **Heurísticas de uso:** El usuario debe haber realizado estudios formales de viabilidad funcional con los pros y contras de las funcionalidades descritas por cada una de las historias para poder lanzar una opinión fiable. Lo primero en realizar debe ser lo imprescindible.



## Roles

- Usuarios del área de negocio (2 como mucho)



## Lecciones Aprendidas

- La visión del aplicativo que se desea obtener debe ser clara. Las funcionalidades mas importantes deben clasificarse por encima de las que son meramente decorativas.
- Hay que rellenar el campo indicando la prioridad de la historia desde el punto de vista de negocio. Existen tres categorías (A-(alto)Imprescindible, M-(medio)Necesario, B-(bajo)Miscelanea).



## Nivel de Madurez

- Este Patrón de Producto no se relaciona con ningún nivel de madurez(N/A).



## Conocimientos y Habilidades Básicos



### Conocimientos

- Conocimiento del estándar de codificación que define la propiedad del código compartido así como las reglas para escribir y documentar el código y la comunicación entre diferentes piezas de código desarrolladas por diferentes equipos. Los programadores las han de seguir de tal manera que el código en el sistema se vea como si hubiera estado escrito por una sola persona.
- Conocimiento de la visión común de cómo funciona el programa en el que se desarrollan las actividades.



### Habilidades

- Capacidad de trabajo en grupo. Todos en un equipo XP contribuyen de la manera que pueden.
- Predicción de qué se habrá terminado para la fecha de entrega, y determinación de qué hacer después.
- Capacidad de programación de a pares. Además de generar mejor código y pruebas, sirve para comunicar el conocimiento a través de los equipos.



## Recursos de Información

- 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/>
  - Jaskowicz, José. Reglas y prácticas en Xtreme Programming. Recuperado el 2010-03-15 de <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Jaskowicz.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.
-