

METODOLOGÍA ÁGIL: FEATURE DRIVEN DEVELOPMENT

Miguel Albertí Pons
Sofía Almeida Bruno
Pedro Manuel Flores Crespo
María Victoria Granados Pozo
Lidia Martín Chica

Índice

1. Introducción	2
2. Procesos	2
2.1. Construir lista de características (<i>Build Feature List</i>)	2
2.2. Planificado por características	2
3. Comparación con otros métodos	3
4. Ventajas	4
5. Roles y responsabilidades	4
6. Bibliografía	5

1. Introducción

2. Procesos

La metodología FDD consta de cinco fases en su proceso: Son las siguientes:

2.1. Construir lista de características (*Build Feature List*)

Esta etapa recibe como entrada el modelo de objetos y los requerimientos (*feature list informal*) obtenidos en la etapa anterior. Estos son agrupados según el área de dominio. Cada grupo de se denomina *Major List Sets*. Esta lista a su vez es dividida en otros sub-conjuntos según la funcionalidad. Posteriormente, cada característica o funcionalidad es priorizada y por último aquellas más complejas son divididas en otras más pequeñas. Esta etapa nos aporta como salida la *Feature List* que es revisada por los usuarios para su aprobación y validación. Las tareas llevadas a cabo en esta etapa se resumen en la siguiente tabla.

Tarea	Responsables
Formación del <i>Feature-List Team</i>	<i>Project Manager</i> y <i>Development Manager</i>
Identificación de funcionalidades y formación del <i>Feature Set</i>	<i>Feature-List Team</i>
Priorización de las funcionalidades	<i>Feature-List Team</i>
División de las funcionalidades complejas	<i>Feature-List Team</i>

Cuadro 1: Tareas etapa de construcción de la lista de características.

2.2. Planificado por características

El resultado final, de este paso, es un plan dedesarrollo, sujeto a revisión y aprobación por parte del gestor de desarrollo y del jefe de la arquitectura, la identificación de la clase y la identificación de los propietarios sobre el conjunto de características. Este paso gira entorno a la planificación el orden en el que se van a implementar las características, la forma de hacerlo y quien las va implementar, para ello se asignarán a los desarrolladores. Obviamente mientras planificamos consideramos diferentes aspectos como riesgos, complejidad, dependencias, la carga de trabajo, etc, para evitar que surjan problemas complejos.

En esta etapa se incluye la creación de un plan de alto nivel, en el cual la features list es ordenada en base a la prioridad y a la dependencia entre cada feature. Además,

las clases identificadas en la primera etapa son asignadas a cada programador.

En base a las características de la lista de la anterior etapa, el gestor del proyecto, el desarrollador y el programador jefe establecen hitos y diseñan un cronograma de diseño y construcción.

Finalmente se realiza una verificación del plan teniendo en cuenta la opinión de todos los miembros del equipo. La salida es un plan de desarrollo incluyendo la fecha de finalización, para cada conjunto de características y cada característica indicar el programador jefe asignado, y las fechas de inicio y fin. Por último para cada clase indicar el correspondiente propietario de la clase

3. Comparación con otros métodos

FDD es una mezcla entre eXtreme Programming y Scrum, añadiendo técnicas de Domain Driven Design

FDD vs XP

Feature Driven Development da una forma de controlar la naturaleza iterativa e incremental de los proyectos ágiles, pero sin decir nada sobre cómo implementar esas características.

Estas características se pueden implementar de diversas formas con distintas técnicas. En un modelo de desarrollo donde se usa FDD para los detalles de la iteración y en donde las características se tratan como tareas a realizar. Usamos métodos ágiles para modelar las tareas que se van a realizar, y usamos XP para implementar los requisitos. Además notemos que en el análisis explícito tiene lugar el diseño o el trabajo de modelado para determinar las características que habría que implementar o dividir en varias tareas. XP utiliza el trabajo en parejas y FDD no, pero se puede incluir. FDD tiene como objetivo controlar proyectos adaptativos, ágiles e iterativos que pueden o no estar basados en XP.

Comunicación

Los métodos ágiles se centran bastante en la comunicación entre los miembros del equipo y el resto de personas interesadas en el proyecto.

En XP y Scrum la documentación también es importante,

Centro los usuarios

Duración del sprint

Reuniones

Tamaño del proyecto

FDD vs Scrum

Puntos en comun: Mejorar la comunicación, enfatizar las cualidades de los componentes del grupo, aumentar la colaboración.

Por otro lado, FDD se centra en las prácticas de ingenierías específicas Scrum no especifica ninguna práctica de ingeniería en particular. Además, mientras que FDD tiene bucles de retroalimentación más largos, Scrum tiene ciclos más cortos.

4. Ventajas

5. Roles y responsabilidades

6. Bibliografía