

Feature Driven Development

Metodologías de desarrollo Ágiles



Alejandro Manuel Hernández Recio

Juan Pablo García Sánchez

Daniel Matilla Bastero

José María Gómez García

Mónica Cordovilla Gámez

Índice

1. Qué es FDD
2. Pasos de diseño:
 - a. Desarrollar un modelo superficial
 - b. Crear lista de funcionalidades
 - c. Planificar por funcionalidad
 - d. Diseñar por funcionalidad
 - e. Construir por funcionalidad
3. Roles
4. Fundamentos
5. Ventajas y desventajas frente a SCRUM
6. Conclusión

Que es FDD

Metodología ágil

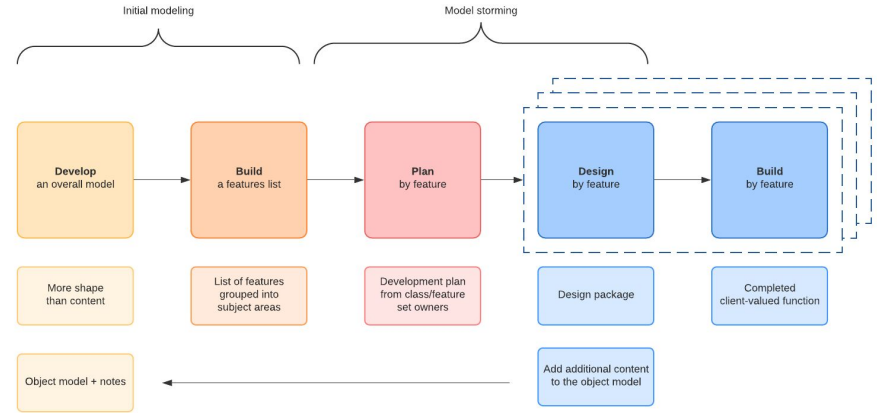
Iterativa

Primer proyecto: 15 meses y 50 personas

Proyectos grandes con equipos numerosos

Seguir las buenas prácticas del software

Seguimiento continuo con el cliente



Pasos de diseño



1. Desarrollar un modelo superficial

- Definir el esquema que abarca el dominio del modelo
- Equipos de desarrollo y arquitecto definen el alcance y contexto.
- Los equipos proponen cada uno su modelo, se elige uno o la unión de todos ellos y se usa este como modelo final.
- El resultado: imagen clara del proyecto.



2. Crear lista de funcionalidades

- Se identifican la lista de funcionalidades descomponiendo el dominio en pequeñas funciones.
- Si una función no puede ser desarrollada en un plazo de dos semanas, debería ser disgregada en funciones más pequeñas y manejables.



3. Planificar por funcionalidad

- Se analiza la complejidad de las características y se planean tareas.
- Se crea un orden y se asignan a los equipos.
- Se identifican “dueños de las clases”.



4. Diseñar por funcionalidad

1. Escoger funciones, diseñarlas, construirlas y priorizarlas
2. Componer equipos y adjudicar un rol temporal
3. Prestar atención al desarrollo y mantenimiento

Tareas de los equipos:

- A. Revisión de diseño
- B. Codificación
- C. Pruebas unitarias
- D. Integración
- E. Revisión de código



5. Construir por funcionalidad

- Se implementan los ítems necesarios para lograr el diseño de las características.
- Se diseña la UI y se hace y testea un prototipo. Si pasan las pruebas, se pueden pasar a los clientes.
- Cualquier característica que sobrepase las dos semanas de longitud, se descompone en otras más pequeñas para cumplir con este periodo.



Roles

El equipo que forma parte activa de la ejecución de la metodología FDD está jerarquizado en los siguientes roles:

- Project manager
- Chief architect
- Development manager
- Chief programmer
- Class owner
- Domain expert



Fundamentos

Se debe tener en cuenta una serie de principios o fundamentos para implementar correctamente Feature Driven Development, y son los siguientes:

- Modelado de objetos del dominio
- Desarrollo por función
- Equipo de características
- Inspecciones
- Programa de compilación regular
- Informes de progreso

Ventajas

- Varios equipos → Menor tiempo
- Documentación estructurada
- Proyectos grandes
- Experiencia de desarrolladores

Desventajas

- Proyectos pequeños
 - Roles de programador jefe
 - Varios roles → errores humanos
 - Comunicación formal
-

Conclusiones

Feature Driven Development es una buena opción cuando se trata de un equipo lo bastante grande como para no poder ejecutar el método Scrum de una forma efectiva. Esta metodología ágil es adecuada para proyectos a largo plazo que cambian que agregan características continuamente en iteraciones regulares.

FDD funciona en equipos grandes y pequeños ya que está diseñado para enfocarse siempre en lo que el cliente necesita.

Bibliografía

Garzás, J. (2012, Septiembre 4). *Un resumen de la metodología ágil FDD*. javiergarzas.com.

<https://www.javiergarzas.com/2012/09/metodologia-gil-fdd-1.html>

Globalluxsoft. (2017, Octubre 18). *5 Popular Software Development Models with their Pros and Cons*. medium.

<https://medium.com/globalluxsoft/5-popular-software-development-models-with-their-pros-and-cons-12a486b569dc>

Lucidchart Content Team. (n.d.). *Why (and How) You Should Use Feature-Driven Development*. Lucidchart.

<https://www.lucidchart.com/blog/why-use-feature-driven-development>

Lynn, R. (n.d.). *What is FDD in Agile?* planview. <https://www.planview.com/resources/articles/fdd-agile/>

Rychlý, M., & Tichá, P. (2008). *A Tool for Supporting Feature-Driven Development*. (B. Meyer, J.R. Nawrocki, & B. Walter, Eds.).

Springer, 196-207. https://link.springer.com/content/pdf/10.1007%2F978-3-540-85279-7_16.pdf

Bibliografía

Tirumala, S. S., Ali, S., & Anjan Babu, G. (2016, Diciembre). A Hybrid Agile model using SCRUM and Feature Driven Development. *International Journal of Computer Applications*.

https://www.researchgate.net/profile/Sreenivas_Sremath_Tirumala/publication/311775064_A_Hybrid_Agile_model_using_SCRUM_and_Feature_Driven_Development/links/588b1800a6fdcc225a341c72/A-Hybrid-Agile-model-using-SCRUM-and-Feature-Driven-Development.pdf

Valesca. (2012, Junio 12). *Metodología FDD*. Metodología FDD – Feature Driven Development / Desarrollo Basado en Funciones. <http://metodologiafdd.blogspot.com/>

W. Ambler, S. (n.d.). *Feature Driven Development (FDD) and Agile Modeling*. Agile Modeling (AM). <http://agilemodeling.com/essays/fdd.htm>

Feature Driven Development