

# FDD

## Desarrollo Basado en Funciones

01

### Introducción a FDD

Feature-Driven Development (FDD) es una metodología ágil enfocada en la entrega incremental de funcionalidades que aportan valor directo al usuario final. Nacida en 1997 de la colaboración entre Jeff De Luca y Peter Coad durante un proyecto financiero en Singapur, FDD se destaca por su enfoque estructurado y su capacidad para gestionar proyectos grandes y complejos.

El proceso de FDD se organiza en torno a cinco actividades clave:

1. Desarrollar un modelo general del sistema.
2. Construir una lista de características necesarias.
3. Planificar el desarrollo basado en dichas características.
4. Diseñar cada funcionalidad antes de implementarla.
5. Construir y entregar cada funcionalidad de manera iterativa.

02

## Fases de FDD

### Desarrollo Modelo General

1

Se establece el modelo del sistema a desarrollar a alto nivel, todo esto con la presencias de



2

### Construcción Lista de Funcionalidades

Se identifican funcionalidades claves del modelo y se consignan en una lista. Cada funcionalidad debe ser manejable y pequeña.

3

### Planificación por Funcionalidades

Se identifican funcionalidades claves del modelo y se consignan en una lista. Cada funcionalidad debe ser manejable y pequeña.



4

### Diseño por Funcionalidades

Realización del diseño de cada funcionalidad antes de su implementación, se establecen las especificaciones técnicas necesarias así como también los diagramas necesarios.

5

### Desarrollo por Funcionalidades

Se realiza el desarrollo de cada funcionalidad basado en el diseño, posteriormente se realiza la inspección del producto y la integración al producto.



03

### Ejemplos de casos de éxito y fracaso

#### EJEMPLO CASO DE ÉXITO

Tras la fusión con Merrill Lynch, Bank of America necesitaba integrar los sistemas de ambas compañías, sin ocasionar grandes daños o molestias a sus clientes.

Proceso:

- Desarrollo del Modelo General: Se integraron los sistemas financieros y de gestión de clientes.
- Construcción de la Lista de Funcionalidades: Se identificaron y priorizaron funcionalidades clave como la sincronización de datos y consolidación de cuentas.
- Planificación por Funcionalidad: Se diseñó un plan detallado para minimizar el impacto en operaciones diarias.
- Diseño por Funcionalidad: Se adaptaron las funcionalidades a las diferencias entre sistemas y nuevas necesidades.
- Construcción por Funcionalidad: Se desarrollaron en ciclos cortos, permitiendo integración y pruebas continuas.

**Resultado:** La metodología FDD ayudo a manejar la complejidad del proyecto y asegurar una integración exitosa, manteniendo la calidad del servicio al cliente y minimizando el tiempo de inactividad.

VS

#### EJEMPLO CASO DE FRACASO

Una empresa de software utilizó FDD para un proyecto que requería adaptaciones frecuentes, lo que terminó siendo un desafío para aplicar la metodología.

Proceso:

- Desarrollo del Modelo General: Se construyó un modelo sin considerar adecuadamente los cambios frecuentes en los requisitos.
- Construcción de la Lista de Funcionalidades: Las funcionalidades se definieron con requisitos inestables, generando problemas en la planificación.
- Planificación por Funcionalidad: La planificación fue rígida y se volvió obsoleta rápidamente debido a la falta de adaptabilidad.
- Diseño por Funcionalidad: El diseño fue interrumpido por cambios constantes, lo que afectó la coherencia.
- Construcción por Funcionalidad: La construcción sufrió retrasos y errores por la necesidad de revisar continuamente las especificaciones.

**Resultado:** La falta de flexibilidad en FDD, junto con la naturaleza cambiante del proyecto, resultó en un software que no cumplía con las expectativas del cliente, llevando a la cancelación del contrato.

04

### Funcionamiento de FDD en la Práctica

En la práctica se hacen iteraciones en las cuales se desarrolla una funcionalidad cada 2 semanas, siendo el máximo tiempo de trabajo que ha de tomar cada funcionalidad. Recordar que el ciclo de trabajo que se repite es:

1. Modelar -> 2. Listar funciones -> 3. Planificar -> 4. Diseñar -> 5. Construir -> 6. Revisar -> 7. Entregar

De esta forma se repite hasta llegar a la iteración final donde se han agregado todas las funcionalidades a la compilación principal.



05

## Ventajas y Desventajas de FDD

### Ventajas

- Alcance enfocado
- Entrega temprana y frecuente
- Comunicación mejorada
- Mayor flexibilidad
- Mejor gestión de proyectos

### Desventajas

- Complejidad
- Tiempo y recursos
- Dependencias
- Visibilidad limitada
- Control limitado

#### Grupo 1

Santiago Sánchez Moya - 20211020032

Pablo Andrés Pérez Ipia 20202020042

Julián David Pérez Chaparro 20192020017

Juan José Aguilar González 20201020063

Angie Catalina Callejas Rueda 20182007065

#### Bibliografía

Rising, L., & Janoff, N. S. (2000). The Scrum Software Development Process for Small Teams. IEEE Software, 17(4), 26-32

Agile Software Development: The Cooperative Game, Alistair Cockburn

GeeksforGeeks. (2023, April 27). FDD full form. GeeksforGeeks. <https://www.geeksforgeeks.org/fdd-full-form/>

Stanke,b.(2022).Feature-Driven Development: The Pros, Cons, and How It Compares to Scrum.Bob Stanke Blog.