**Exemplo Prático / Estudo de Caso FDD**

***Alguns exemplos reais e práticos do FDD são:***

**Exemplo real:**

Em 1997 a United Overseas Bank (UOB), um banco de singapura decidiu reformular por completo a sua plataforma de empréstimos para empresas, comércios e consumidores em geral. Aqui vemos a metodologia sendo usada na prática pela primeira vez.

Por ser um projeto grande e complexo, foi exigido fosse feito por equipes especializadas, e divididas em funcionalidades (features), para que a plataforma não ficasse inativa durante a reforma.

[https://pt.wikipedia.org/wiki/Feature\_Driven\_Development?utm\_source](https://pt.wikipedia.org/wiki/Feature_Driven_Development?utm_source=chatgpt.com)

**Exemplo prático para um app de venda de roupas:**

***Passo 1***: vamos entender a estrutura desse aplicativo:

Essas são chamadas de “**subject areas”.** Em outras palavras, são as principais áreas do app.

* Usuários (clientes)
* Produtos (roupas)
* Carrinho de compras
* Pagamento
* Avaliações

***Passo 2:*** Em seguida vamos construir uma lista de funcionalidades

|  |  |  |
| --- | --- | --- |
|  | **Funcionalidade (Feature)** | **Descrição / Valor para o usuário** |
| 1 | Usuário se cadastra no site | Permite que o usuário tenha conta e acesse funcionalidades do site. |
| 2 | Usuário adiciona produtos ao carrinho | Permite selecionar produtos antes de finalizar a compra. |
| 3 | Usuário finaliza compra com pagamento | Permite que o usuário conclua a compra e gere receita. |
| 4 | Usuário avalia produtos | Permite dar feedback, aumentar confiança e ajudar outros usuários. |
| 5 | Usuário pesquisa produtos por categoria ou tamanho | Facilita encontrar produtos rapidamente, melhorando a experiência do usuário. |

***Passo 3:*** *Planejamento de ciclos:*

Aqui vamos definir a ordem da entrega de cada funcionalidade do software.

|  |  |  |
| --- | --- | --- |
|  | **Funcionalidade entregue** | **Por que essa ordem** |
| 1 | Usuário se cadastra | Sem cadastro, não é possível comprar |
| 2 | Adicionar produtos ao carrinho | Essencial para começar a compra |
| 3 | Finalizar compra / pagamento | Permite gerar receita real |
| 4 | Avaliação de produtos | Feedback social e confiança para outros usuários |
| 5 | Pesquisa e filtros de produtos | Facilita encontrar itens rapidamente |

***Passo 4:*** detalhar e atribuir as funcionalidades aos dev:

Aqui iremos designar os dev as suas funções para o desenvonvimento do software:

* Jean → cadastro de usuário
* Gustavo → adicionar produtos ao carrinho
* João Pedro → finalização de compra / pagamento
* Gabriel → avaliações de produtos
* Ramon → pesquisa e filtros

***Passo 5:*** incrementação por ciclos:

Aqui iremos incrementar de acordo com os ciclos citados no **passo 3**

* **Ciclo 1:** Usuários se cadastram → software já tem interação mínima.
* **Ciclo 2:** Usuários adicionam produtos ao carrinho → software já começa a funcionar como loja.
* **Ciclo 3:** Usuários finalizam a compra → gera receita.
* **Ciclo 4:** Avaliações → adiciona engajamento social.
* **Ciclo 5:** Pesquisa e filtros → melhora experiência do usuário e torna o site completo.

**Comparação com outras metodologias:**

Primeiro vamos fazer uma comparação entre o FDD e o SCRUM.

No FDD, você entrega features independentes e funcionais. A vantagem disso, é que o usuário tem ideias reais de como ficará seu software.

Enquanto no SCRUM, você entrega sprints inteiros, que nem sempre são funcionais isoladamente.

Outra vantagem do FDD em comparação ao SCRUM, é que como você está entregando features funcionais, isso faz com que diminua o risco de problemas futuros.

Tudo depende do tempo determinado, da urgência daquele software, e do que você planeja entregar para seu cliente. Mas basicamente, o FDD deve ser escolhido em projetos grandes, onde temos muitas dependências.