

Design Thinking: Uma Abordagem Centrada no Usuário para Desenvolvimento de Software

Design Thinking é uma metodologia de resolução de problemas que coloca o usuário no centro do processo de desenvolvimento de software, priorizando a empatia, a criatividade e a iteração.



Profa. Lauriana Paludo

lauriana.paludo@ifpr.edu.br



Etapas do Design Thinking

-
- ```
graph TD; A[1 Fase 1: Empatia] --- B[2 Fase 2: Definição]; B --- C[3 Fase 3: Ideação]; C --- D[4 Fase 4: Prototipagem]; D --- E[5 Fase 5: Teste]; E --- F[6 Fase 6: Implementação]
```
- 1 Fase 1: Empatia  
Compreender profundamente as necessidades, desejos e frustrações dos usuários-alvo do software.
  - 2 Fase 2: Definição  
Definir o problema a ser resolvido com clareza e precisão, focando nas necessidades do usuário.
  - 3 Fase 3: Ideação  
Gerar ideias inovadoras e soluções criativas para o problema identificado, explorando diferentes perspectivas.
  - 4 Fase 4: Prototipagem  
Criar protótipos rápidos e tangíveis das soluções, permitindo testes e validação com os usuários.
  - 5 Fase 5: Teste  
Obter feedback dos usuários sobre os protótipos, identificando áreas de melhoria e validando a viabilidade das soluções.
  - 6 Fase 6: Implementação  
Implementar a solução final, incorporando os aprendizados e aprimoramentos obtidos durante o processo.

# Empatia: Entendendo o Usuário



## Escuta Ativa

Realizar entrevistas, observações e pesquisas para compreender as necessidades, desafios e expectativas dos usuários.



## Criação de Personas

Criar representações fictícias de usuários-alvo, com base em dados e informações coletadas, para facilitar a compreensão de suas necessidades.



## Mapeamento da Jornada do Usuário

Visualizar as etapas e interações que os usuários têm com o software, identificando pontos de dor e oportunidades de otimização.

# Definição: Identificando o Problema

## 1 Enquadramento do Problema

Definir o problema de forma clara e concisa, utilizando uma linguagem que seja compreensível para todos os membros da equipe.

## 2 Priorização

Identificar e priorizar os problemas mais relevantes e impactantes para os usuários, focando nas necessidades mais urgentes.

## 3 Enunciado do Problema

Criar um enunciado do problema que seja preciso e objetivo, definindo o que precisa ser resolvido.



# Ideação: Gerando Soluções Criativas

## Brainstorming

Gerar ideias de forma livre e espontânea, sem censurar ou julgar as sugestões, buscando a quantidade e a diversidade.

## Pensamento Lateral

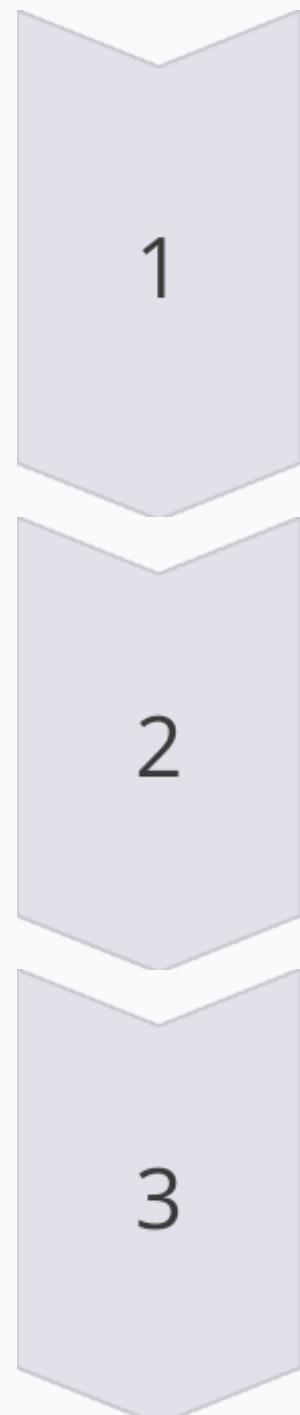
Explorar soluções inovadoras que desafiem as convenções e abram novas possibilidades para resolver o problema.

## Combinar Ideias

Combinar e integrar ideias diferentes para criar soluções mais completas e eficazes, aproveitando o melhor de cada proposta.



# Prototipagem: Testando as Soluções



Protótipo de Baixa Fidelidade

Criar um protótipo simples e rápido, utilizando materiais acessíveis, para testar a usabilidade e a funcionalidade da solução.

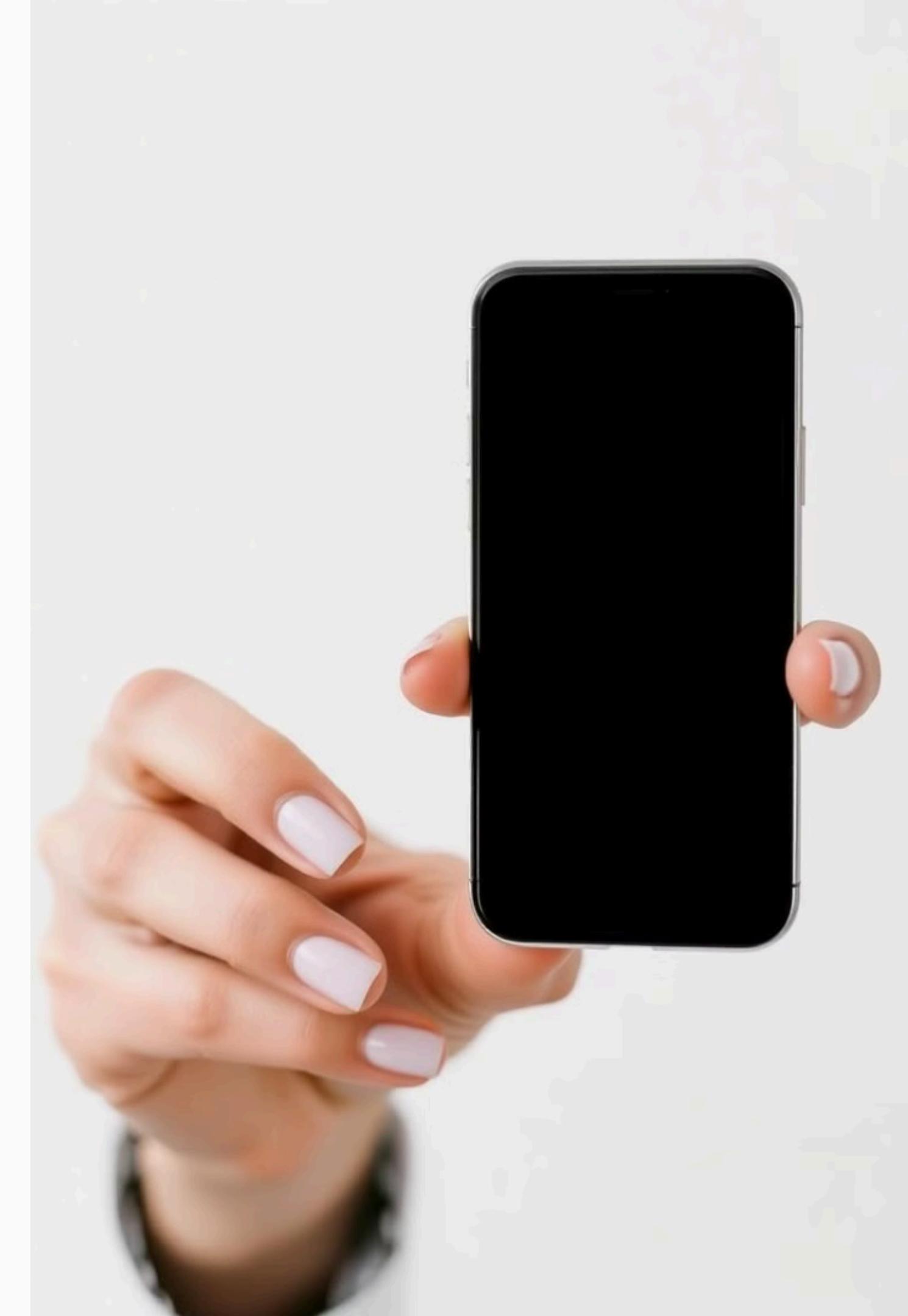
Teste de

Usabilidade

Observar e analisar a interação dos usuários com o protótipo, identificando dificuldades e pontos de melhoria.

Iteração

Ajustar e aprimorar o protótipo com base no feedback dos usuários, realizando novas iterações de teste e refinamento.



# Implementação: Colocando em Prática



## Planejamento

Definir o escopo da implementação, o cronograma de atividades e os recursos necessários.

## Desenvolvimento

Desenvolver o software com base no protótipo validado e nos aprendizados obtidos durante o processo.

## Teste

Realizar testes rigorosos para garantir a qualidade, a performance e a segurança do software.

## Lançamento

Lançar o software para os usuários, com base em um plano de comunicação e divulgação.

# Conclusão e Benefícios do Design Thinking



## Soluções Centradas no Usuário

O Design Thinking garante que o software atenda às necessidades e expectativas dos usuários, tornando-o mais útil e relevante.



## Inovação e Criatividade

Incentiva a criação de soluções inovadoras e criativas, que superem as expectativas dos usuários.



## Melhoria da Comunicação

Promove a colaboração entre as equipes de desenvolvimento e os usuários, melhorando a comunicação e o entendimento mútuo.



## Redução de Riscos

Reduz os riscos de desenvolver um software que não atenda às necessidades dos usuários, garantindo maior sucesso do projeto.

