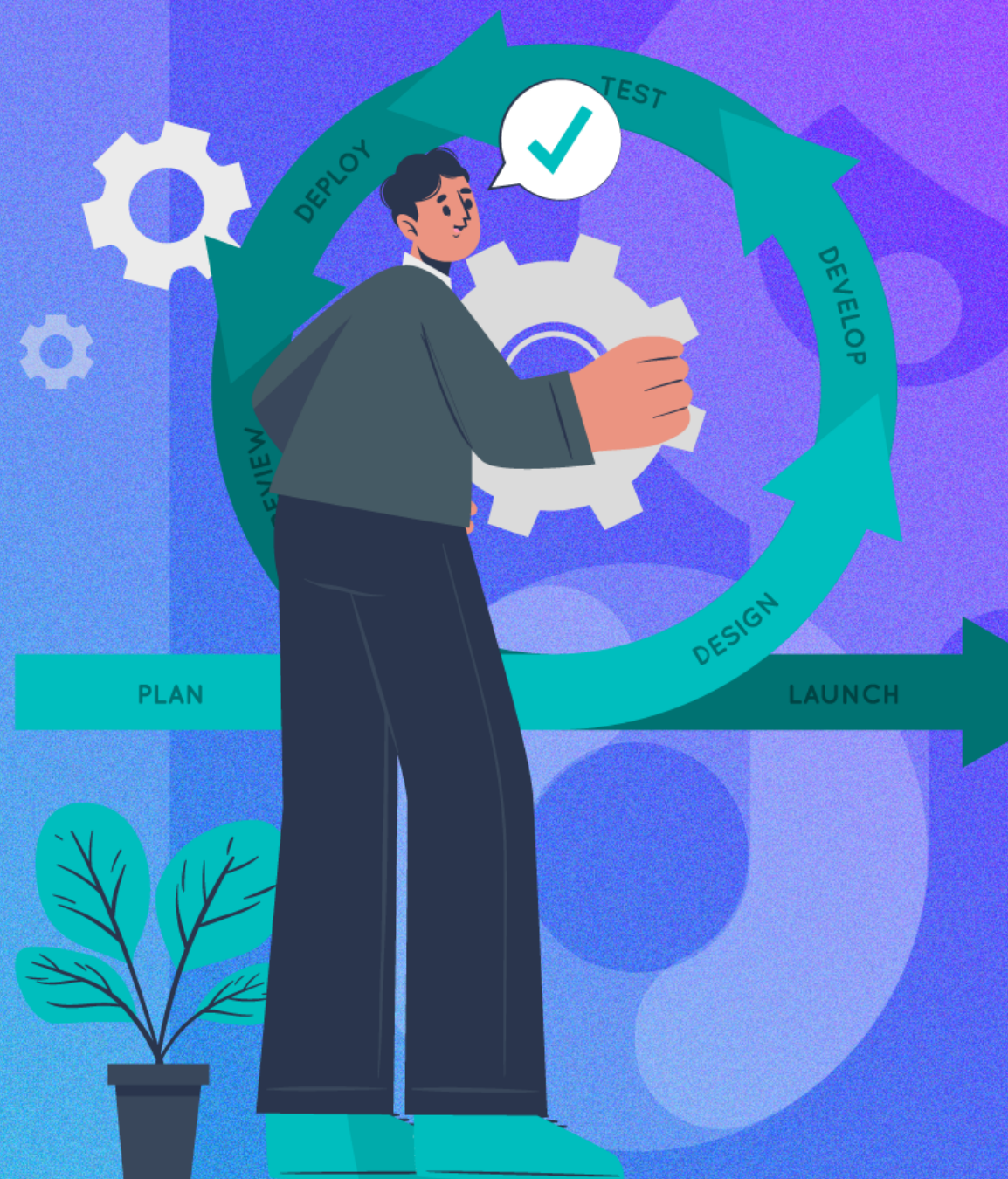


E-book

Guia definitivo para implementar **metodologias de projetos ágeis**





SUMÁRIO

Para quem é esse ebook?	03
O que são metodologias de projetos ágeis?	04
Qual é a diferença entre metodologias preditivas e ágeis	07
Quais são as principais metodologias ágeis?	10
Metodologia SCRUM	12
Metodologia Extreme Programming (XP)	19
Metodologia Lean	22
Metodologia Feature-Driven Development (FDD)	25
Metodologia Dynamic Systems Development Method (DSDM)	28
Ferramentas essenciais para gestão de projetos ágeis	33
Quais são as principais competências de um gestor de projetos ágeis	36
Atualize seus conhecimentos	37
Sobre o Artia	39
Sobre o autor	40



PARA QUEM É ESSE E-BOOK?

Profissionais iniciantes

Para quem está estudando sobre as técnicas fundamentais de gestão de projetos e quer aprimorar seus conhecimentos.

Profissionais intermediários

Para quem já atua em projetos e tem uma boa compreensão das práticas de gerenciamento, mas precisa melhorar a padronização de processos, gestão de recursos, riscos, qualidade e controle financeiro das iniciativas.

Profissionais avançados

Para quem já domina as melhores práticas e gerencia múltiplos projetos, mas precisa se atualizar nas tendências emergentes e quer aprimorar a visão estratégica e acelerar a inovação.

O QUE SÃO METODOLOGIAS DE PROJETOS ÁGEIS?

As metodologias ágeis são um conjunto de práticas e princípios de gestão de projetos que enfatizam a flexibilidade, tendo um foco maior no valor da entrega do que no planejamento em si. Assim, suas principais características são a colaboração entre equipes, a eficiência na entrega de resultados e a capacidade de se adaptar rapidamente a mudanças.

Esse tipo de metodologia surgiu no início do século XX com o manifesto ágil. Primeiramente, ela foi aplicada exclusivamente ao contexto de desenvolvimento de software, pois representava uma resposta às limitações das metodologias

preditivas em lidar com a natureza volátil e iterativa da criação de software. No entanto, hoje, a aplicação dessas práticas se expandiu para além dessa área, sendo utilizadas em diversos tipos de projetos e indústrias.



Características das metodologias ágeis

Colaboração intensiva

As equipes trabalham em estreita colaboração, com comunicação constante para garantir que o produto atenda às necessidades dos usuários.

Entrega contínua de valor

As funcionalidades são entregues ao longo de todo o projeto, agregando valor ao cliente de forma contínua.

Iterações curtas e incrementais

O projeto ágil é dividido em ciclos curtos, chamados iterações ou sprints, permitindo ajustes frequentes e redirecionamentos ao longo da execução do projeto.

Flexibilidade e adaptabilidade

A abordagem ágil permite mudanças no escopo do projeto em qualquer estágio, facilitando a adaptação às necessidades do cliente ou a novas informações que surgem.

Foco no cliente

As necessidades e o feedback do cliente são coletados a cada ciclo de execução, garantindo que o produto seja relevante e esteja alinhado com os valores desejados.

Conheça o Manifesto Ágil

O Manifesto Ágil foi publicado em fevereiro de 2001 por um grupo de 17 desenvolvedores de software que buscavam uma maneira mais eficiente e flexível de criação. Assim, esse é um documento que apresenta os valores e princípios fundamentais para uma abordagem mais adaptável e centrada nas pessoas.

O Manifesto ágil é composto por 4 valores. Conheça o [documento completo aqui](#).

Indivíduos e interações mais que processos e ferramentas

Importância do trabalho em equipe e da comunicação direta entre os stakeholders.

Software em funcionamento mais que documentação abrangente

Prioriza as entregas funcionais do software funcional, de forma a atender às necessidades do cliente acima da criação de extensas documentações.

Colaboração com o cliente mais que negociação de contratos

Valoriza a colaboração contínua com o cliente e a adaptação às suas necessidades e feedbacks.

Responder a mudanças mais que seguir um plano

Reconhece a importância de ser flexível e capaz de responder corrigir o planejamento para garantir a entrega do valor.

QUAL É A DIFERENÇA ENTRE METODOLOGIAS PREDITIVAS E ÁGEIS?

	Metodologia Preditiva	Metodologia Ágil
Planejamento e Execução	Planejamento detalhado, com etapas sequenciais e predeterminadas.	Planejamento adaptativo, com execução iterativa e incremental.
Foco	Enfatiza o cumprimento do escopo, prazo e custos definidos.	Concentra-se na satisfação do cliente e na entrega contínua de valor.
Abordagem de Gestão	Gerenciamento top-down, com decisões tomadas pela liderança e um caminho claro definido desde o início.	Abordagem colaborativa, com equipes autogerenciáveis que tomam decisões baseadas em feedback contínuo.
Resposta a Mudanças	As mudanças são vistas como um risco a ser minimizado, com foco em seguir o plano inicial.	As mudanças são vistas como oportunidades para melhorar o resultado do projeto.
Ideal Para Projetos	Projetos com requisitos bem definidos e pouco propensos a mudanças, como construção e manufatura.	Projetos em ambientes dinâmicos e incertos, especialmente no desenvolvimento de software.

E a metodologia híbrida?

Projetos híbridos combinam elementos e práticas das metodologias ágeis e preditivas para criar uma abordagem **flexível, adaptativa e estruturada**. Eles são projetados para aproveitar os pontos fortes de ambos os mundos: a **rapidez e adaptabilidade** das metodologias ágeis com a **previsibilidade e a estrutura** das abordagens preditivas.

O objetivo da metodologia híbrida é oferecer uma solução mais personalizada, que se adapte às necessidades específicas da organização. Assim, há a percepção de que não existe uma única abordagem que funcione melhor em todos os cenários e que os processos podem ser alternados.

Os projetos híbridos são especialmente úteis em ambientes onde partes do projeto são previsíveis e outras partes necessitam de adaptação.



Quando optar pela metodologia ágil?



Mudanças frequentes no escopo

Projetos que exigem adaptação do projeto ao longo do progresso.



Necessidade de entrega rápida

Quando há urgência na entrega de resultados e o cliente exige entregas de valor de forma contínua.



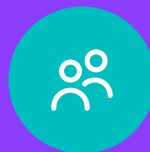
Requisitos incertos ou complexos

Projetos onde não é possível definir todos os requisitos detalhadamente no início e é necessário explorar soluções.



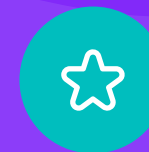
Ambientes de inovação e experimentação

O Ágil permite explorar novas ideias, testar hipóteses e aprender com os resultados.



Equipes multidisciplinares e colaborativas

Quando há forte dependência da colaboração entre diferentes especialistas.



Entrega de valor

Projetos que têm como foco principal a satisfação do cliente e necessitam de feedback contínuo para aperfeiçoamento.

QUAIS SÃO AS PRINCIPAIS METODOLOGIAS ÁGEIS?



Scrum

O Scrum é a metodologia ágil mais popular, conhecida por sua estrutura de sprints, reuniões regulares e papéis definidos. É particularmente eficaz em projetos complexos, em que os requisitos não são claramente definidos.

Extreming Programming (XP)

O XP é uma metodologia focada no desenvolvimento de software de alta qualidade e na capacidade de se adaptar a mudanças nos requisitos do cliente. Inclui práticas como programação em pares, desenvolvimento orientado a testes, integração contínua, e refatoração do código.

Lean Software Development

Inspirado pelos princípios de manufatura enxuta da Toyota, o Lean Software Development foca na entrega de valor ao cliente, eliminando desperdícios (atividades que não agregam valor), otimizando o processo de desenvolvimento, e encorajando uma cultura de melhoria contínua.

Feature-Driven Development (FDD)

O FDD é uma metodologia centrada no cliente e orientada a recursos, projetada para ser escalável e aplicável a equipes de todos os tamanhos. Enfatiza entregas frequentes, modelos de domínio para cada recurso, e propriedade clara de cada recurso ou área do projeto.

Dynamic Systems Development Method (DSDM)

O DSDM é uma abordagem que enfatiza a entrega frequente de produtos, o envolvimento ativo do usuário, equipes capacitadas para tomar decisões e a garantia de que os principais critérios de qualidade são atendidos. Essa é uma metodologia versátil e que pode ser integrada com outras práticas ágeis como Scrum e XP.

Metodologia SCRUM

O Scrum é um framework de gestão de projetos focado na entrega de valor e na melhoria contínua dos processos. Para isso, ele funciona através de entregas em ciclos, as chamadas Sprints.

A lógica por traz dessa metodologia é permitir que o produto evolua de forma iterativa e incremental. Isso significa que o produto é desenvolvido e entregue em partes, com cada sprint resultando em um incremento funcional.

Atenção! Para que tudo isso funcione na prática, o Scrum estabelece um conjunto de **papéis, cerimônias e artefatos**, que devem ser seguidos.

ARTEFATOS DO SCRUM

Product Backlog

O Product Backlog é uma lista priorizada de todos os requisitos, funcionalidades, melhorias e correções que precisam ser feitas no produto a ser entregue. Este artefato é fundamental porque, quanto mais importante for a entrega, maior a sua urgência em começar a desenvolvê-la.

Incremento de Produto

O incremento de produto é o resultado do trabalho realizado pela equipe de desenvolvimento durante o sprint. Ou seja, é um conjunto de funcionalidades completas e testáveis que podem ser entregues ao cliente.

Sprint Backlog

O Sprint Backlog é uma sublista do Product Backlog, que definirá as funcionalidades que serão implementadas no próximo Sprint.

Este artefato é criado durante a reunião de planejamento de sprint, onde a equipe seleciona os itens do backlog do produto que serão incluídos no sprint com base na capacidade e nas prioridades.

PAPÉIS DO SCRUM



Product Owner (PO)

Product Owner, pela tradução, é “Dono do Produto”. Isso significa que essa é a pessoa encarregada de manter a qualidade do produto durante a execução e organizar as tarefas para que a entrega seja feita.

Responsabilidades:

- Gerenciar e manter o Product Backlog atualizado;
- Tomar decisões sobre o que será desenvolvido e em que ordem;
- Planejar a Sprint;
- Definir critérios de aprovação das tarefas;
- Maximizar o valor do produto e do trabalho da equipe.

Scrum Master

O Scrum Master é a pessoa que mais conhece a metodologia Scrum no time, sendo responsável por nortear a equipe e oferecer suporte tanto ao PO quanto ao time de desenvolvimento.

Responsabilidades:

- Ajudar a equipe a entender e seguir as práticas ágeis;
- Facilitar a comunicação entre as partes interessadas;
- Dar suporte ao Product Owner;
- Remover impedimentos que possam atrapalhar o progresso da equipe.

Time de Desenvolvimento

O Time de Desenvolvimento é formado por colaboradores responsáveis pela entrega dos incrementos do produto.

Responsabilidades:

- Executar a Sprint;
- Autogerenciar as tarefas para atingir os objetivos da sprint;
- Entregar os incrementos de produto funcionais a cada sprint.

CERIMÔNIAS DO SCRUM

Sprint Planning (planejamento da Sprint)

A Sprint Planning marca o início de cada sprint. Esta reunião tem como objetivo definir o que será entregue ao final da Sprint e como as tarefas serão organizadas ao longo dela.

Aqui, o Product Owner apresenta os itens prioritários do Product Backlog e a equipe de desenvolvimento seleciona os itens que consegue entregar durante o sprint.

Daily Scrum (reunião diária)

Ao longo da execução, a equipe se reúne diariamente para avaliar o andamento das tarefas e compartilhar seus progressos e dificuldades. Geralmente as dailys levam cerca de 15 minutos e buscam responder três perguntas de forma objetiva:

- O que eu fiz ontem?
- O que eu farei hoje?
- Estou enfrentando algum obstáculo?

Sprint Review (revisão da Sprint)

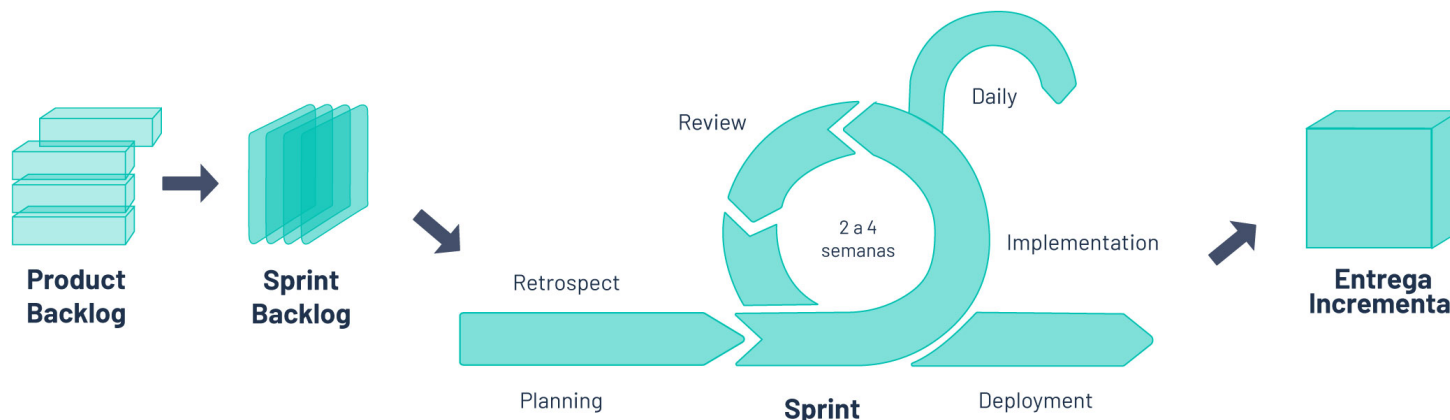
A Sprint Review é realizada ao final de cada sprint com o objetivo de inspecionar os resultados obtidos e adaptar o Product Backlog caso algo não tenha sido concluído. Esta reunião ajuda o PO a compreender os itens entregues e permite fazer ajustes no backlog do produto.

Sprint Retrospective (retrospectiva da Sprint)

A Sprint Retrospective é realizada após a Sprint Review e antes da próxima reunião de planejamento de Sprint. O propósito dessa reunião é refletir sobre o último sprint e identificar:

- O que funcionou bem;
- O que deu errado;
- Quais ações podem ser tomadas para melhorar o próximo sprint.

LÓGICA DO SCRUM



- O trabalho é dividido em sprints curtas, geralmente de 2 a 4 semanas.
- Cada sprint tem um objetivo claro e um conjunto específico de funcionalidades a serem entregues.
- No início da sprint, a equipe planeja o que será feito e como o trabalho será realizado.

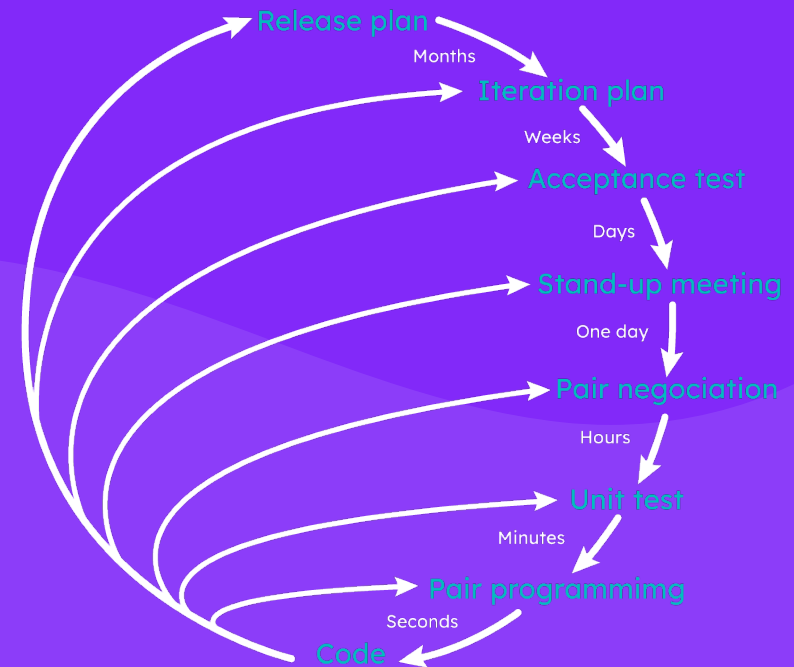
- Durante a sprint, a equipe se reúne diariamente para acompanhar o progresso e identificar impedimentos.
- Ao final da sprint, a equipe demonstra o que foi feito ao cliente e recebe um feedback.
- Antes de finalizar a sprint, revisar o processo para identificar oportunidades de melhoria.

Metodologia Extreme Programming (XP)

O Extreme Programming (XP) é uma metodologia ágil de desenvolvimento de software que enfatiza a excelência técnica, a capacidade de resposta às mudanças nas necessidades dos clientes e a comunicação efetiva dentro das equipes. Foi criada por Kent Beck no final dos anos 90 como uma resposta aos desafios enfrentados em projetos de grande escala e alta volatilidade.

O objetivo do XP é melhorar a qualidade do software e a capacidade de resposta ao mudar os requisitos do cliente através de feedback frequente, desenvolvimento iterativo, trabalho em equipe e um ciclo de vida de desenvolvimento mais sustentável.

Planning / feedback loops



PRINCÍPIOS DO EXTREME PROGRAMMING (XP)

Planejamento incremental

Os requisitos são divididos em pequenas tarefas que podem ser desenvolvidas rapidamente, permitindo ajustes frequentes no planejamento.

Programação em pares

Dois programadores trabalham juntos em um único computador, um escreve o código enquanto o outro revisa cada linha de código conforme é escrita. Isso melhora a qualidade do software e dissemina o conhecimento pela equipe.

Desenvolvimento Orientado a Testes (TDD – Test-Driven Development)

Os testes são escritos antes do código que faz os testes passarem, garantindo que o software tenha apenas o código necessário para atender aos requisitos e que cada funcionalidade seja testada.

Integração Contínua

O código é integrado e testado várias vezes ao dia, reduzindo o risco de problemas na integração.

Design simples

O sistema é projetado para ser simples e suficiente para a tarefa atual, sem tentar prever requisitos futuros, reduzindo a complexidade e facilitando mudanças.

Refatoração

O código é constantemente melhorado para aumentar a clareza e reduzir a complexidade, facilitando a manutenção e a extensão no futuro.

Propriedade coletiva do código

Todo o código é de propriedade da equipe, permitindo que qualquer membro da equipe faça melhorias em qualquer parte do código.

Padrões de codificação

A equipe segue um conjunto consistente de padrões de codificação e práticas para garantir que o código seja fácil de ler e manter por todos.



Metodologia Lean

A metodologia Lean Software Development (Desenvolvimento Enxuto de Software) é uma abordagem de desenvolvimento de software inspirada pelos princípios da manufatura enxuta, originalmente desenvolvida pelo sistema de produção Toyota. O objetivo principal do desenvolvimento enxuto de software é maximizar o valor para o cliente enquanto minimiza o desperdício, ou seja, tudo aquilo que não agrega valor ao produto.

Agile Lean Software Development



PRINCÍPIOS DO LEAN

Eliminar desperdício

Identificar e remover atividades que não agregam valor ao cliente, como burocracia excessiva, funcionalidades desnecessárias, atrasos no processo etc.

Amplificar o aprendizado

Encorajar a experimentação e o aprendizado contínuo dentro da equipe, por meio de iterações rápidas, feedback constante e práticas como desenvolvimento orientado a testes (TDD).

Decidir o mais tarde possível

Adiar decisões importantes até o último momento, para maximizar a flexibilidade e asserividade às mudanças de requisitos.

Entregar o mais rápido possível

Reduzir o tempo de ciclo do desenvolvimento para fornecer valor ao cliente rapidamente e receber feedback que pode ser incorporado em futuras iterações.

Empoderar a equipe

Dar à equipe de desenvolvimento a autonomia para tomar decisões importantes, promovendo a inovação e a responsabilidade.

Construir qualidade

Incorporar qualidade no processo de desenvolvimento desde o início, através de técnicas como integração contínua e desenvolvimento orientado a testes.

Otimizar o todo

Focar na otimização do processo de desenvolvimento como um sistema integrado, em vez de otimizar partes individuais em detrimento do todo.



Metodologia Feature-Driven Development (FDD)

O Feature-Driven Development (FDD) é uma metodologia ágil de desenvolvimento de software que enfatiza a construção de um sistema de forma iterativa e incremental, focando em características específicas (“features”) do software. Desenvolvida originalmente por Jeff De Luca no final dos anos 90, a FDD combina princípios de modelagem orientada a objetos e práticas ágeis para entregar funcionalidades tangíveis e de valor agregado em um cronograma regular.



PRINCÍPIOS DO FEATURE-DRIVEN DEVELOPMENT (FDD)

Desenvolvimento de um modelo global

Inicia com a criação de um modelo de alto nível do domínio do projeto, envolvendo clientes e desenvolvedores em sessões de modelagem para entender o contexto e as necessidades do negócio.

Construção de uma lista de características

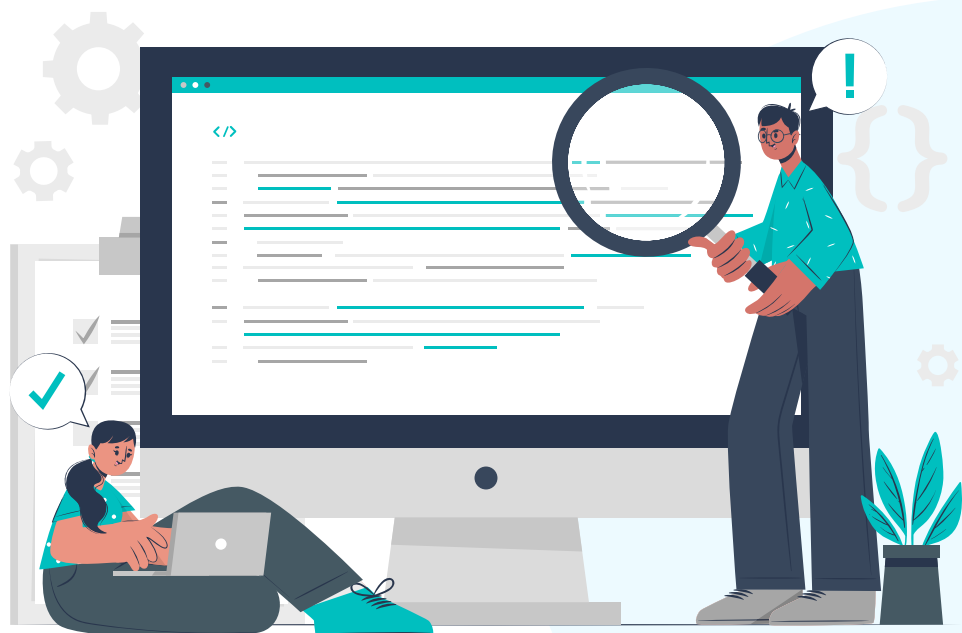
A partir do modelo global, uma lista detalhada de características (pequenas funções de valor para o cliente) é criada. Essas características são pequenos pedaços de funcionalidades que podem ser desenvolvidas em poucos dias.

Planejamento por característica

As características são agrupadas em categorias e planejadas para desenvolvimento. Esta etapa envolve a priorização das características e a alocação de recursos.

Design por característica e construção por característica

Cada característica é desenhada detalhadamente e depois construída. Este processo é rápido, com cada característica passando por design, revisão de código, e integração em um curto período.



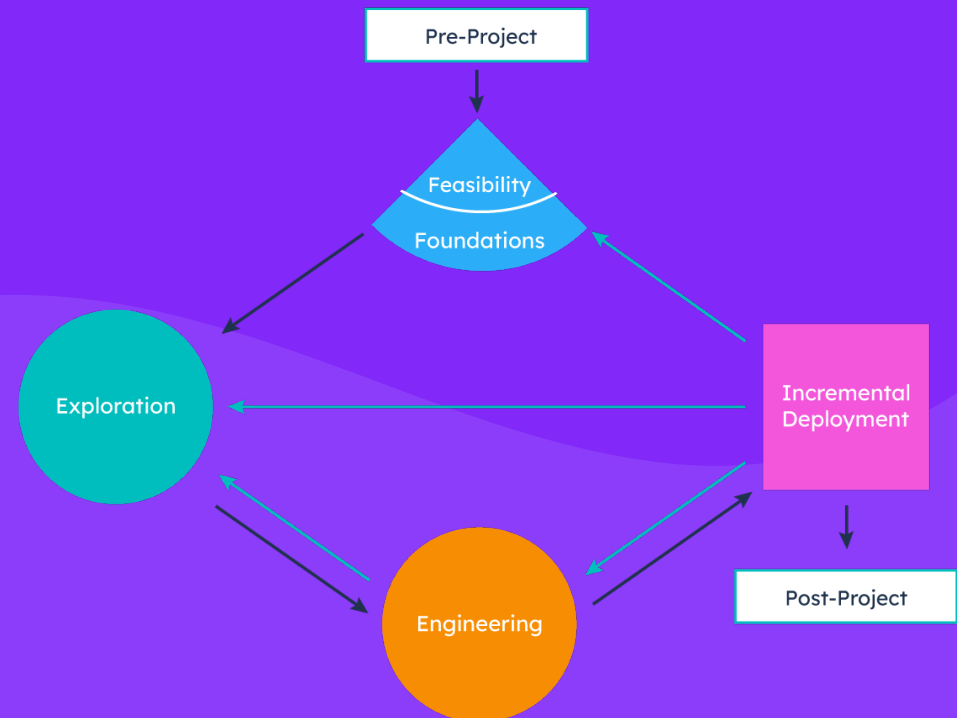
Implantação regular

As características completadas são frequentemente entregues ao cliente, garantindo um fluxo contínuo de valor e permitindo feedback rápido.

Metodologia Dynamic Systems Development Method (DSDM)

O Dynamic Systems Development Method (DSDM) é uma metodologia ágil de gerenciamento de projetos e desenvolvimento de software que foca na entrega de soluções em tempo hábil, mantendo a qualidade e aderindo a um orçamento pré-definido. Originada no início dos anos 90, o DSDM foi uma das primeiras metodologias a promover uma abordagem ágil ao desenvolvimento de software, oferecendo uma alternativa aos métodos tradicionais, mais lentos e menos flexíveis.

Project Phases in DSDM Atern



PRINCÍPIOS DO DSDM

Envolvimento ativo do usuário

A participação constante do usuário ao longo da execução do projeto.

Fitness for Business Purpose

O critério principal para aceitação de entregáveis é que eles devem satisfazer as necessidades de negócio atuais.

Equipes empoderadas

As equipes devem ter o poder de tomar decisões importantes durante o projeto.

Desenvolvimento iterativo e incremental

O desenvolvimento ocorre em iterações, permitindo a incorporação de mudanças e melhorias.

Entrega frequente

Prioriza a entrega frequente de produtos para garantir progresso contínuo.

Reversibilidade

As mudanças podem ser revertidas se necessário para garantir alinhamento com os objetivos do projeto.

Requisitos fixos em um nível alto

Os requisitos são estabelecidos em alto nível no início do projeto, mas detalhados de forma iterativa.

Teste integrado ao ciclo de vida

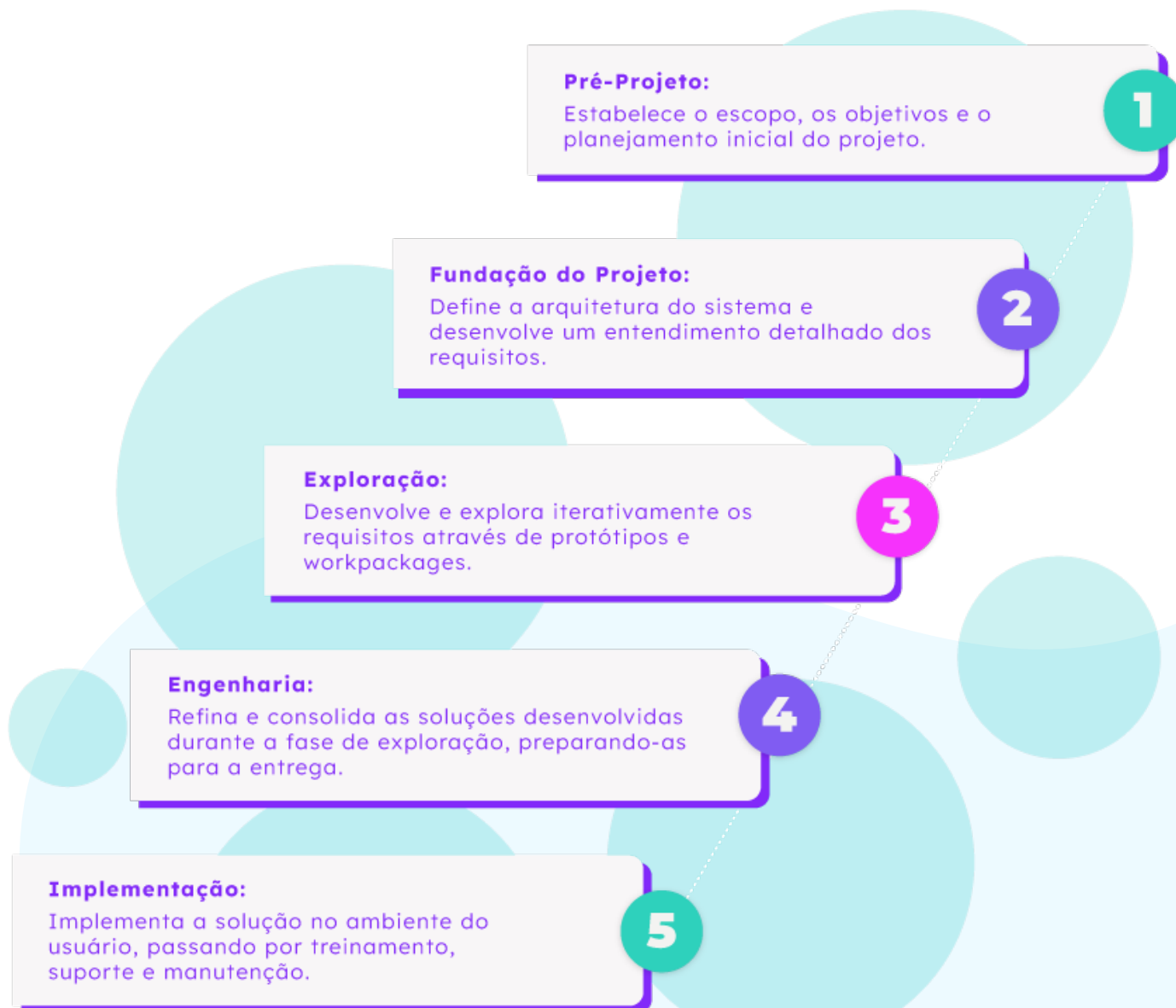
Testes são realizados ao longo de todo o projeto para garantir a qualidade.

Colaboração e cooperação

Todos os stakeholders devem trabalhar juntos de forma colaborativa para o sucesso do projeto.



AS FASES DO DSDM



Checklist: como implementar uma metodologia ágil?

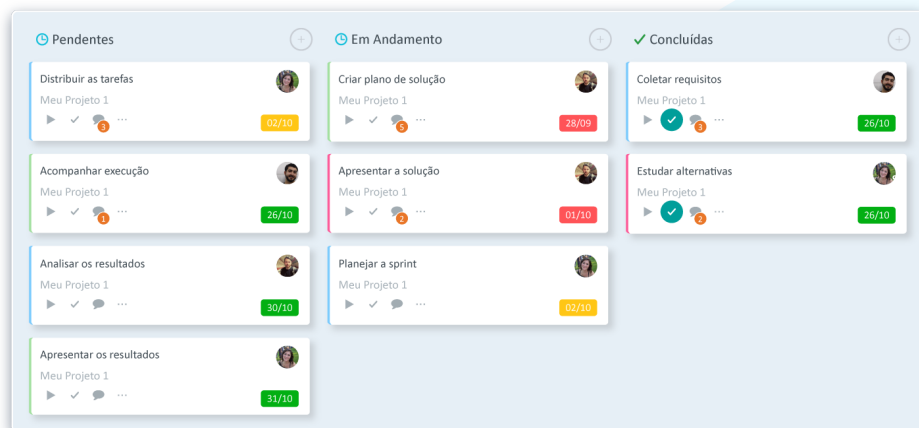
- ☐ Estude os valores e princípios do manifesto ágil
- ☐ Avalie e selecione a metodologia ágil mais adequada para as necessidades do projeto
- ☐ Familiarize-se com as práticas e ferramentas da metodologia escolhida
- ☐ Realize workshops ou sessões de treinamento sobre agilidade para a equipe
- ☐ Reforce a cultura de colaboração, transparência e aprendizado contínuo
- ☐ Escolha as ferramentas que suportem a metodologia escolhida
- ☐ Revise e defina processos claros de execução das atividades
- ☐ Realize as reuniões diárias de acompanhar o progresso e identificar impedimentos
- ☐ Realize revisões ao final de cada sprint ou ciclo para avaliar o trabalho entregue
- ☐ Apresente os resultados e colete feedback dos stakeholders
- ☐ Conduza retrospectivas para identificar o que funcionou bem e o que pode ser melhorado
- ☐ Ajuste processos e práticas com base no feedback e aprendizados
- ☐ Mantenha-se atualizado com as melhores práticas e tendências em gerenciamento ágil



FERRAMENTAS ESSENCIAIS PARA GESTÃO DE PROJETOS ÁGEIS

Kanban

Ideal para acompanhar a execução e visualizar o trabalho em progresso.



Backlog do produto

Lista de tarefas a serem executadas nos próximos ciclos.

Atividades

Adicionar Atividade

Adicionar

Ações

Lista de Tarefas

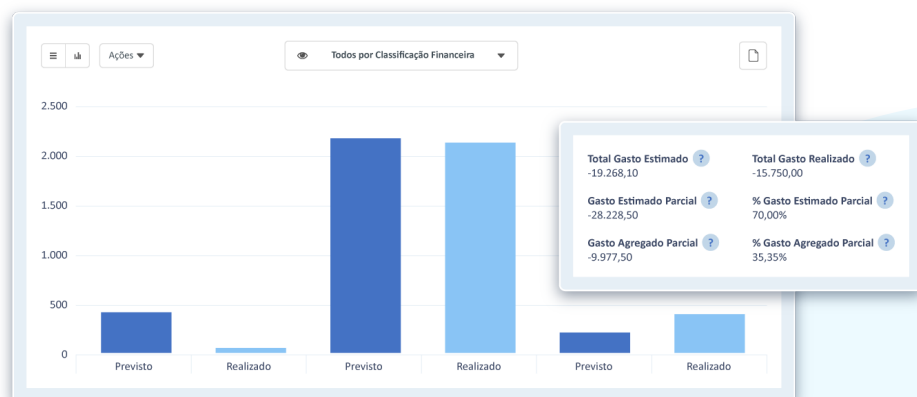
	Título	Responsável	Tipo de atividade	Produto	Precisa para quando?	Término Estimado	Término Real	Esforço Estimado	Esforço Real	Situação
☐	Distribuir as tarefas	Julia	Marketing	Artia	02/10	02/10		6,00	0,00	Não iniciado
☐	Acompanhar execução	Marcos	Marketing	Artia	26/10	26/10		3,00	0,00	Não iniciado
☐	Analisar os resultados	Lucas	Marketing	Artia	30/10	30/10		10,00	0,00	Não iniciado
☐	Apresentar os resultados	Julia	Marketing	Artia	31/10	31/10		2,00	0,00	Não iniciado
☐	Criar plano de solução	Lucas	Marketing	Artia	28/09	28/09		5,00	4,00	Em andamento
☐	Apresentar a solução	Lucas	Marketing	Artia	01/10	01/10		3,00	2,00	Em andamento
☐	Planejar a sprint	Julia	Marketing	Artia	02/10	02/10		1,00	1,00	Em andamento
☐	Coletar requisitos	Marcos	Marketing	Artia	26/10	26/10	20/10	6,00	7,00	Concluído
☐	Estudar alternativas	Julia	Marketing	Artia	26/10	26/10	24/10	10,00	9,00	Concluído
Total								46,00	23,00	

Exibindo 09 de 09 item(s)

Ações

Controle de custos

Possibilidade de estimar, registrar e monitorar as despesas.



Gestão de recursos

Visibilidade da equipe disponível e mitigação da sobrecarga ou ociosidade.

Julia Distribuir as tarefas Marketing/Empresa Empresa	Jorge Criar plano de solução Marketing/Empresa Empresa
Claudia Acompanhar execução Marketing/Empresa/03 - Hipóteses Empresa	Daniela Planejar a sprint Marketing/Empresa/03 - Hipóteses Empresa
Pedro Analisar os resultados Marketing/Empresa/03 - Hipóteses Empresa	André Estudar alternativas Marketing/Empresa/03 - Hipóteses Empresa



Comunicação

Fóruns, chat e repositório de arquivos para colaboração e compartilhamento de informação.

Respostas Histórico

Pedro
Posts: 6
Postado em Quarta, 17 de Maio, 10:37h
Como o PMO pode melhorar a gestão de projetos?

Julia
Posts: 6
Postado em Quarta, 17 de Maio, 11:10h
O PMO fornecerá uma estrutura centralizada e processos padronizados para melhorar a eficiência e qualidade dos projetos.

Daniela
Posts: 6
Postado em Quarta, 17 de Maio, 14:43h
Qual é o cronograma estimado para a implantação do PMO?

Julia
Posts: 6
Postado em Quarta, 17 de Maio, 15:55h
A fase de planejamento deve levar cerca de 4 semanas, e a implementação, de 6 a 8 semanas.

Sua resposta:

☒ Enviar-me um e-mail para cada resposta nova.

Postar

Apontamento de horas

Registro e análise do esforço estimado e realizado pela equipe para executar as atividades.

00 - Apontamentos

	seg 09/04	ter 10/04	qua 11/04	qui 12/04	sex 13/04
Pendente					
☐ Coletar requisitos	00:45	03:00	02:30	00:30	-
☐ Estudar alternativas	01:00	02:30	-	03:00	-
☐ Criar plano de solução	00:30	04:00	-	02:30	03:30
☐ Acompanhar execução	00:30	01:45	04:00	-	02:00
☐ Apresentar a solução	02:00	-	01:00	03:30	00:30



QUAIS SÃO AS PRINCIPAIS COMPETÊNCIAS DE UM GESTOR DE PROJETOS ÁGEIS?

Planejamento e organização Competência para planejar sprints, releases e adaptar o planejamento com base no feedback e nas mudanças das necessidades do projeto.	Gerenciamento de riscos Capacidade de antecipar, identificar e mitigar riscos que possam afetar o cronograma, o orçamento ou a qualidade do projeto.	Controle e monitoramento Gerar insights e promover melhorias a partir de ferramentas de análise e assegurar a correção de desvios.	Comunicação eficaz Clareza e objetividade para comunicar objetivos, expectativas e status, além de negociar e influenciar.	Gerenciamento de stakeholders Identificar e gerenciar as expectativas das partes interessadas, garantindo sua satisfação e engajamento.
Adaptabilidade e flexibilidade Habilidade para se adaptar rapidamente a mudanças no projeto ou na organização e flexibilidade para ajustar planos de projeto conforme necessário.	Foco no cliente Orientação para compreender profundamente as necessidades do cliente e garantir que o produto atenda ou exceda essas expectativas.	Resolução de problemas Capacidade de rastrear e resolver problemas de forma criativa quando surgem desafios inesperados.	Liderança Motivar e orientar a equipe de forma orientada aos objetivos do projeto, além de resolver conflitos.	Conhecimento técnico da metodologia Compreensão das práticas, ferramentas e técnicas de gestão de projetos ágeis.



Atualize seus conhecimentos:

GUIA AVANÇADO DA GESTÃO DE PROJETOS

Quando pensamos em evolução e melhoria de processos, geralmente associamos isso ao uso de novas ferramentas e tecnologias avançadas, não é?

Mas, você sabia que na nova versão do PMBOK são as **habilidades de liderança, comunicação e colaboração** que ganham destaque?

O **PMBOK 7** trouxe uma grande reformulação do que consideramos como gestão de projetos. Por isso, reunimos todas as **atualizações e tendências** que o guia apresenta para te ajudar a se manter atualizado neste universo!





Blog

Acesse artigos sobre o mercado, estudos de caso e tendências de gerenciamento de projetos.

Acessar



Central de conteúdos

Acesse guias, templates e planilhas para facilitar as rotinas da gestão de projetos.

Acessar





SOBRE O ARTIA

Tudo que você precisa para gerenciar projetos, tarefas e equipes em uma plataforma única.

O Artia MULTIPLICA SEU PODER DE GESTÃO.

Elimine imprevistos, atrasos e prejuízos, fazendo projeções realistas com base no histórico e aumente sua capacidade de entregar projetos com sucesso.

Visualize seu portfólio completo com status, monitore a alocação e produtividade da sua equipe pelo dashboard, em tempo real, e entregue mais projetos do que nunca!

Com o artia você decide ter mais ou menos controle sem perder a visão do todo, a partir de um pacote funcional avançado e robusto.



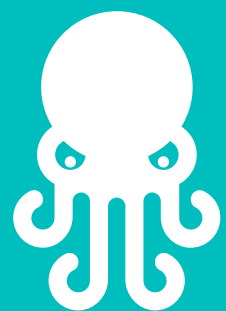


ROBERTO GIL ESPINHA
CEO do Artia

SOBRE O AUTOR

Roberto Gil Espinha é um profissional com mais de 20 anos de experiência em projetos, com ênfase na área de Finanças e Tecnologia da Informação. Bacharel em Administração de Empresas, com especializações em Gestão Empresarial, pela FGV-RJ, e em Engenharia de Software, pela PUC-PR, possui certificações PMP, PMI-ACP, ITIL Foundation, CSM e CSP.

Trabalhou como executivo na Datasul, atualmente conhecida como Totvs, e, hoje, é CEO do Artia, uma ferramenta inovadora voltada para a Gestão de Projetos. Além disso, atua como consultor em empresas para a estruturação de seus processos e metodologias de gestão de projetos, infraestrutura de TI e adoção de boas práticas de engenharia de software.



artia

transforme planos em conquistas

