

PROJETO: TaskFlow

Metodologia: BMAD – Breakthrough
Method for Agentic Agile AI-Driven
Development

Introdução

- O BMAD-METHOD é uma metodologia ágil orientada por agentes de IA, projetada para integrar planejamento e desenvolvimento com preservação de contexto.
- Neste PROJETO, aplicaremos o BMAD no desenvolvimento do aplicativo **TaskFlow**, um sistema de gestão de tarefas colaborativas.

Objetivos do Projeto

- Desenvolver um MVP (Minimum Viable Product).
- Implementar funcionalidades de criação, atribuição e acompanhamento de tarefas.
- Reduzir retrabalho e aumentar eficiência utilizando agentes de IA para planejamento e QA.
- Integrar documentação, arquitetura e histórias de usuário em um fluxo contínuo.

Funcionalidades Principais (MVP – versão inicial)

1. Autenticação e Perfis

- Cadastro e login de usuários (email/senha, autenticação JWT).
- Perfis: administrador, gerente e colaborador.
- Recuperação de senha via e-mail.

2. Gestão de Tarefas

- Criar, editar, excluir e atribuir tarefas.
- Campos: título, descrição, responsável, status, prioridade, data de entrega.
- Visualização em lista e em quadro (kanban simples).
- Filtro e busca por status, responsável e prioridade.

3. Gestão de Projetos

- Criar projetos e associar tarefas a eles.
- Definir nome, descrição e equipe envolvida.
- Painel com resumo do progresso (percentual concluído).

Funcionalidades Principais (MVP – versão inicial)

4. Colaboração

- Comentários em tarefas.
- Notificações básicas (ex: “Tarefa atribuída a você”).
- Histórico de alterações (log de atividade por tarefa).

5. Dashboard e Relatórios

- Painel principal com contagem de tarefas por status (A Fazer / Em Progresso / Concluído).
- Relatório por usuário e por projeto.
- Exportação simples (CSV ou PDF).

6. Configurações e Acesso

- Gestão de usuários (somente para administrador).
- Permissões básicas por papel.
- Configurações de idioma e tema (opcional).

Equipe e Papéis BMAD

- **Product Manager** (Agente PM): Define escopo, roadmap e PRD.
- **Analyst Agent**: Pesquisa de mercado e levantamento de requisitos.
- **Architect Agent**: Cria arquitetura técnica e padrões de código.
- **UX Agent**: Elabora protótipos e fluxos de experiência do usuário.
- **Dev Agent**: Implementa código conforme histórias de usuário.
- **QA Agent**: Realiza testes automatizados e validações.
- **Scrum Master** (SM Agent): Coordena sprints e monitoramento de progresso.

Fase 1 – Planejamento com Agentes

1. O Analyst Agent realiza análise de mercado e define personas.
2. O PM Agent elabora o PRD (Product Requirements Document).
3. O Architect Agent define arquitetura Java + Spring Boot + PostgreSQL.
4. O UX Agent cria wireframes e fluxos de navegação.
5. Todos os artefatos são salvos em formato Markdown no repositório Git.

Fase 2 – Execução e Desenvolvimento

1. O SM Agent converte PRD e arquitetura em histórias de usuário completas.
2. Cada história contém: objetivo, contexto, padrões, critérios de aceitação.
3. O Dev Agent implementa módulos conforme backlog priorizado.
4. O QA Agent executa testes automatizados e manuais.
5. Revisões e retrospectivas ao final de cada sprint.

Artefatos Gerados no Processo

- /docs/prd.md – Documento de Requisitos de Produto
- /docs/architecture.md – Arquitetura e Diagrama UML
- /docs/ux-spec.md – Protótipo e fluxos de tela
- /stories/*.yaml – Histórias de usuário com contexto completo
- /tests/*.md – Casos de teste e validações
- /backlog/*.csv – Registro de progresso por sprint

Fluxo de Trabalho BMAD

- 1. Planejamento
 - 2. Geração de Artefatos
 - 3. Transição para Execução
 - 4. Desenvolvimento Contextualizado
 - 5. QA Automatizado
 - 6. Entrega e Retrospectiva.
- Cada etapa mantém o contexto do projeto para evitar perda de informação entre papéis.

ENTREGA

- Quando?
 - SEXTA-FEIRA (07/11/2025) as 14hs.
- Como?
 - Artefatos criados durante a construção do projeto: Slides Powerpoint, arquivos .MD, código do projeto, código do projeto, link do GitHUB.
- Onde?
 - On-line.
- De que forma?
 - 10 Minutos para cada equipe.
 - Usar Slides, filmes, whatever.
 - As equipes são auto-organizadas.
- .