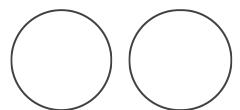


BMAD Method – Como multiagentes de IA podem otimizar o desenvolvimento de software

Adilson Filho

Tech Manager | Senior Analyst | Tech Leader | Mobile Apps | Frontend | Fullstack | Digital Banking | Azure | Microsoft Azure Certified | IA Generativa | Squad Leader no Bradesco



October 22, 2025

Antes de entrarmos no BMAD é interessante termos alguns conceitos em mente. De início precisamos entender um pouco do que seria a engenharia de contexto.

Engenharia de Contexto

Engenharia de Contexto é a prática de estruturar e fornecer à IA generativa todas as informações necessárias para que ela possa executar tarefas com precisão, consistência e relevância. Diferente da simples criação de prompts, essa abordagem considera todo o ecossistema informacional que envolve um modelo de linguagem (LLM), como:

- Histórico de interações
- Dados externos
- Ferramentas disponíveis
- Instruções de sistema

“A Engenharia de Contexto é a arte de preencher a janela de contexto com a informação exata necessária para o próximo passo.” — Data Science Academy

É dar as ferramentas necessárias para o agente saber o que fazer antes de executar a tarefa. Costumo dizer que é turbinar a IA com todo conhecimento necessário para que ela possa executar os prompts com a menor taxa de falha ou alucinação. É como montar o palco ideal para uma big band: cada músico (ou agente) entra no momento certo, com a partitura certa, em perfeita harmonia com os demais. O resultado? Uma execução precisa, fluida e cheia de propósito e no caso da IA, uma resposta contextualizada, útil e alinhada ao objetivo.

Componentes do Contexto

Cada elemento abaixo contribui para que a IA comprehenda melhor a tarefa e gere respostas mais úteis:

Para mais entendimento sobre a engenharia de contexto você pode consultar a minha postagem sobre o assunto através do link: https://www.linkedin.com/posts/adilson-andrade_um-pouco-sobre-engenharia-de-contexto-ia-activity-7386200714836062208-JSvt?utm_source=share&utm_medium=member_desktop&rcm=ACoAAAzGCDkBXRHYN53nlzDyylpSvKykk6RRcU

Greenfield e Brownfield

Para utilizarmos o BMAD precisamos também entender duas terminologias fundamentais que são o Greenfield e Brownfield e que têm origem na indústria de construção civil e engenharia, mas foram amplamente adotados em tecnologia da informação, especialmente em desenvolvimento de software e gestão de projetos.

- **Greenfield:** Refere-se a projetos iniciados do zero, em um “campo verde” (literalmente), sem restrições de infraestrutura existente.
- **Brownfield:** Refere-se a projetos realizados em áreas já desenvolvidas, onde há estruturas existentes que precisam ser adaptadas, reformadas ou integradas.

BMAD Method: A Revolução da Engenharia de Software com Agentes Inteligentes

Imagine ter uma equipe de especialistas em IA trabalhando com você um Product Manager, um Arquiteto, um QA, um Desenvolvedor todos virtuais, todos prontos para colaborar em tempo real. Isso não é ficção científica. É o **BMAD Method**, ou **Breakthrough Method of Agile AI-Driven Development**, uma abordagem que está transformando a forma como desenvolvemos software e aplicamos IA em projetos complexos.

O que é o BMAD?

O BMAD é um **framework universal de desenvolvimento orientado por agentes de IA**, que combina os princípios ágeis com o poder da inteligência artificial. Ele foi criado para resolver dois grandes problemas em projetos com IA:

- Inconsistência no planejamento
- Perda de contexto durante o desenvolvimento

Ao invés de depender de prompts soltos e improvisados (o famoso “vibe coding”), o BMAD propõe uma estrutura clara, com papéis bem definidos e documentos consistentes desde o início.

Para que serve?

O BMAD serve para:

- Planejar e executar projetos com IA de forma estruturada
- Automatizar tarefas repetitivas com agentes especializados
- Gerar documentação técnica completa (PRD, arquitetura, epics, histórias)
- Evitar perda de contexto entre etapas do projeto
- Facilitar a colaboração entre humanos e agentes de IA

Ele pode ser aplicado em **software development, escrita criativa, estratégia de negócios, educação, bem-estar pessoal**, entre outros.

Como funciona?

O BMAD se baseia em duas inovações principais:

1. Agentic Planning

Agentes especializados (como o Analista, PM, Arquiteto, SM) trabalham juntos para criar:

- PRDs (Product Requirement Documents)
- Arquiteturas de sistema
- Epics e histórias de usuário

Esses agentes usam técnicas avançadas de elicitação e engenharia de prompt para garantir que tudo esteja alinhado e consistente.

2. Context-Engineered Development

Depois do planejamento, os documentos são transformados em **pacotes de contexto autossuficientes**. Cada história de desenvolvimento contém:

- Requisitos funcionais e não funcionais
- Diretrizes arquiteturais
- Critérios de aceitação
- Notas para desenvolvedores e QA

Isso elimina a necessidade de ficar “caçando” informações em documentos dispersos. O agente desenvolvedor abre uma história e já sabe **o que fazer, como fazer e por que fazer**.

Fundamentos e Conceitos-Chave

- Agentes Especializados: Cada agente tem um papel claro (PM, QA, Dev, etc.)
- Human-in-the-loop: O humano participa ativamente, validando decisões
- Documentos com contexto embutido: Nada de perder informações entre etapas
- Expansão modular: Pode ser adaptado para diferentes domínios com "expansion packs"
- Integração com IDEs e Web UIs: Funciona com ferramentas como Cursor, Claude, Gemini, VS Code

Processos e Workflow

O BMAD segue um fluxo em duas fases:

Fase 1: Planejamento Estratégico

- Análise de projeto com o agente Analista
- Geração de PRD e arquitetura
- Validação com o PO

Fase 2: Desenvolvimento Contextualizado

- Sharding dos documentos em epics e histórias
- Execução por agentes de desenvolvimento e QA
- Testes automatizados e validação contínua

Aplicações Práticas

- Projetos Greenfield: Desenvolvimento do zero com planejamento completo
- Projetos Brownfield: Adição de funcionalidades em sistemas existentes
- Startups e MVPs: Agilidade e estrutura desde o início
- Educação e pesquisa: Aplicação em projetos acadêmicos e experimentais
- Empresas de software: Escalabilidade e padronização de processos

Por que o BMAD é Diferente?

Enquanto outras abordagens tentam adaptar IA ao desenvolvimento tradicional, o BMAD **reimagina o processo inteiro**. Ele não é apenas uma ferramenta, mas uma **metodologia completa**, com agentes que pensam, planejam e executam como uma equipe ágil real.

É como sair de um carro manual para um carro autônomo você ainda está no controle, mas com muito mais eficiência e segurança.

Hands ON - BMAD

Antes de iniciar qualquer passo prático, é essencial conhecer o repositório oficial do BMAD. Nele você encontrará todas as versões disponíveis, documentação, exemplos e arquivos necessários para começar a utilizar o framework.

Acesse o repositório através do link abaixo:

 <https://github.com/bmad-code-org/BMAD-METHOD>

Versões

- v4.x – Versão estável atual, disponível via npm
- v5 – Pulada propositalmente. A equipe decidiu não lançar a v5 e avançar diretamente para a v6
- v6-alpha – Disponível agora para testes iniciais

Sobre a v6-alpha

A versão v6-alpha é uma reescrita completa do BMAD-METHOD, com mudanças significativas na arquitetura. Ela está voltada para early adopters (usuários que gostam de testar novidades antes da versão final) e vem com os seguintes alertas:

- Pode conter mudanças quebráveis (breaking changes)
- Está sujeita a atualizações diárias e instabilidade
- Algumas funcionalidades estão incompletas ou são experimentais

Sem mais delongas, vamos ao exemplo prático:

BMAD Hands-on: Exemplo Prático

1. Instalação via npx (opção completa)

Instale Node.js (v20 ou superior)

```
npx bmad-method install  
npx bmad-method status  
npx bmad-method list
```

2. Uso direto do arquivo team-fullstack.txt (mais rápido)

- Acesse o repositório: <https://github.com/bmad-code-org/BMAD-METHOD>
- Vá para a pasta: dist/teams/
- Faça o download do arquivo team-fullstack.txt
- Crie uma pasta para o projeto em algum diretório de sua preferência
- Na sua **IDE** (VS Code com extensão de chat, Cursor ou Windsurf):
- Abra o **Chat**, anexe o arquivo team-fullstack.txt como contexto e coloque a orientação recomendada pela comunidade:

"Your critical operating instructions are attached, do not break character as directed. All conversation must be in brazilian portuguese because I don't speak english."

Isso ativa os agentes BMAD.

Digite /help.

A partir daqui fica a seu critério a escolha do agente, dada a necessidade do projeto em questão. Entretanto, afim de termos melhores resultados e diminuir a chance de falhas ou alucinações, sugiro realizar o ciclo de desenvolvimento por completo, seguindo desde a concepção do Brief até o desenvolvimento da aplicação.

Para o nosso exemplo eu vou pular a etapa do analyst e vou direto para o PM construir o PRD. Ainda no seu chat digite /pm e entre no perfil do product manager

Caso esteja lidando com um projeto **brownfield**, utilize o comando *create-doc brownfield-prd para gerar o documento PRD com base nas especificidades desse tipo de projeto.

Neste exemplo, estamos desenvolvendo um projeto **greenfield**, ou seja, começando do zero. Por isso, utilizaremos apenas o comando create-doc prd, fornecendo como prompt a ideia central do projeto: a criação de uma **landing page de portfólio**.

Para orientar o agente na elaboração do documento, incluí no prompt um exemplo de landing page, que servirá como referência para o desenvolvimento da estrutura e dos elementos desejados.

Pronto! O documento PRD foi criado e salvo na pasta docs.

Após a geração do PRD, você pode seguir por diferentes caminhos, dependendo da etapa que deseja abordar:

- **Revisar e Refinar:** Precisa ajustar alguma seção do PRD? Faça os ajustes necessários para garantir clareza e alinhamento.
- **Design Técnico:** Use o comando /architect para gerar uma arquitetura detalhada do projeto.
- **UX/UI Design:** Com o comando /ux-expert, você pode criar wireframes e mockups para validar a experiência visual.
- **Implementação:** Fragmentar o PRD e iniciar o desenvolvimento com base nas histórias de usuário.

No meu caso, optei por seguir direto para a **implementação**. Como próximo passo, você pode alternar o agente para o perfil de **Scrum Master (SM)**, que é o caminho recomendado para construir as histórias de usuário com mais precisão.

Neste exemplo, vamos passar pelo SM para entender como ele estrutura as histórias e prepara o terreno para o desenvolvimento.

 **Ação necessária:** Digite /sm para alternar para o perfil de Scrum Master e iniciar a criação da primeira *user story*.

Neste momento, nosso agente **Scrum Master (SM)** está analisando o PRD e convertendo os requisitos em **User Stories**, facilitando a organização e o planejamento do desenvolvimento.

Após a criação das histórias, você pode validá-las e seguir para a próxima etapa: **alternar para o perfil de desenvolvedor** utilizando o comando /dev.

Ao executar /dev, você muda de agente e inicia a **implementação do projeto**, com base nas histórias estruturadas pelo SM.

Ao percorrermos todas as etapas, da concepção do PRD à execução, chegamos ao desenvolvimento e, por fim, ao resultado final: a **landing page de portfólio** construída com base no prompt inicial.

 Confira abaixo o print da página gerada, que reflete a estrutura e os elementos definidos durante o processo de desenvolvimento

Conclusão

O **BMAD Method** representa uma verdadeira evolução na forma como desenvolvemos software com o apoio da inteligência artificial. Ao estruturar o processo com **agentes especializados e documentos contextualizados**, ele reduz falhas, acelera entregas e promove uma colaboração mais inteligente e eficiente entre humanos e IA.

Um exemplo prático dessa agilidade foi a criação da minha própria **landing page profissional**, desenvolvida com o apoio do BMAD em apenas algumas horas. Desde o planejamento até a implementação, cada etapa foi conduzida por agentes que compreenderam o contexto, geraram os documentos certos e facilitaram a execução com precisão e fluidez.

Esse tipo de aplicação mostra que o BMAD vai muito além da teoria: é uma **ferramenta concreta** para quem busca **eficiência, escalabilidade e inovação** em projetos digitais.

Se você trabalha com tecnologia, vale muito a pena conhecer e experimentar essa abordagem. O futuro do desenvolvimento já começou e ele é colaborativo, inteligente e mais ágil do que nunca.