

A importância do desenho de arquitetura de software

Como o desenho de arquitetura de software ajuda a explicar e justificar ideias de solução para projetos de TI

Maringá – PR, julho de 2021 – Ao optar por uma carreira técnica, é possível notar que algumas necessidades sempre se repetem. Explicar e justificar ideias de solução para projetos de TI é uma delas. Felizmente existem algumas alternativas simples, porém eficazes, para solucionar esse problema.

Dentre os motivos para apresentar um desenho de arquitetura de software, o principal talvez seja o de convencer seus líderes e pares que a sua estratégia funciona. Manter uma documentação viva da arquitetura e representar visualmente a sua solução, também são motivos importantes.

Como fazer isso de forma simples e eficaz?

Para começar, é preciso criar uma representação gráfica simples que contenha seus componentes e a conexão entre eles, deixando claro as características de coesão e acoplamento (engenharia de software), com o cuidado de manter o desenho simples para não atrapalhar o entendimento. Crie nomes expressivos para todos os elementos do desenho e, quando necessário, utilize legendas para facilitar a compreensão.

Apenas com essas dicas, já é possível obter um bom resultado com um desenho de alto valor capaz de ajudar a guiar os desenvolvedores. Mas não é só isso. Existem algumas soluções simples e cheias de valor para representações de arquitetura. É o caso do c4model, uma alternativa enxuta que permite navegar entre vários níveis de conhecimento (do negócio ao técnico), garantindo que qualquer perfil do seu time entenda a solução.

C4MODEL

O modelo c4 apresenta 4 níveis de “detalhamento” da solução, permitindo que o desenvolvedor crie uma apresentação eficiente em todos os níveis. No nível 1 (system contexto) é possível mostrar como a sua solução interage com parceiros, ferramentas e serviços externos.

Já o nível 2 (container) se propõe a expandir a sua solução, mostrando os maiores módulos existentes e suas comunicações, enquanto o nível 3 (componente) detalha cada componente existente na solução, já definido a nível de projeto, bem como alguns detalhes de implementação.

O 4º e último nível (code) se destina a mostrar como cada componente (a nível de código) deve ser implementado. Vale ressaltar que em alguns casos, sobretudo

quando o time já possui experiência e uma boa base sobre como implementar a proposta da solução, o último nível pode não ser necessário.

Ferramentas Úteis

Para desenhos e soluções mais simples é possível usar qualquer ferramenta de desenho de propósito geral. A maioria delas permite que você desenvolva a sua própria biblioteca de componentes para reutilização em outros projetos. A ferramenta mais indicada para esses casos é o DrawIO.

Para quem deseja permanecer no básico, mas queira manter um registro mais elaborado dos desenhos, o PlantUML permite criar vários diagramas suportados pela UML, com uma simples linguagem própria. Ou seja: você escreve o código e a ferramenta “desenha” para você. É possível até criar uma versão com git, por exemplo.

Para um cenário mais avançado, faz sentido olhar para uma ferramenta mais estruturada que permita fazer uma melhor gestão e padronização do desenho. Nesse caso é preciso cuidado para não escolher alternativas complexas que permitem fazer muita coisa, quando na maioria das vezes, não há necessidade.

Uma boa indicação é o Structuriz, uma ferramenta específica pensada para quem usa c4model, que também trabalha com a ideia de Diagrams as code. Para ir ainda mais longe e trabalhar com ferramentas de modelagem, o ArchiMate é uma ótima ferramenta, porém não tão intuitiva e fácil de usar.

Para Nilson Junior, arquiteto de soluções do Grupo DB1, [formado por empresas de tecnologia brasileiras, com sede em Maringá - PR - Brasil e bases operacionais no Brasil, Argentina e Estados Unidos](#), e desenvolvedor há 10 anos, “mais importante do que qual ferramenta usar, o desenvolvedor deve garantir que os desenhos tenham um padrão e/ou sigam uma notação proposta. Isso ajudará a explicar, justificar e documentar soluções técnicas e de arquitetura para sua equipe, stakeholders e lideranças”.

Para mais informações sobre a importância do desenho de arquitetura de software, acesse <https://bit.ly/guia-arquitetura-imprensa>.

Sobre o Grupo DB1:

O Grupo DB1 é formado por [empresas de tecnologia brasileiras, com sede em Maringá - PR - Brasil e bases operacionais no Brasil \(3\), Argentina \(1\) e Estados Unidos \(1\)](#). Tem 20 anos de experiência no mercado de tecnologia da informação e oferece software e serviços de desenvolvimento para empresas (B2B). Sempre com posicionamento de produtos e serviços “premium” e de alta qualidade as empresas do grupo são líderes de mercado como ANYMARKET, Consignet, DB1 Global Software, DUCZ, Koncili, EIVE, Predize e o robô Tinbot. O grupo tem conquistado inúmeros prêmios de gestão e inovação e figura há 11 anos consecutivos entre as melhores empresas para se trabalhar pela GPTW (Great Place to Work). **O Grupo DB1 também foi reconhecido pelo Financial Times, um dos mais renomados**

jornais britânicos, [como uma das 500 empresas que mais crescem nas Américas](#). Com o lema “Sua luz move o universo DB1”, e como parte do [Código de Cultura do grupo](#), todos na organização trabalham com o propósito de transformar o futuro e impactar a vida das pessoas e organizações por meio das soluções criadas pelo grupo.

Mais informações: <https://www.db1group.com/>

Atendimento à imprensa:

Neris Comunicação

Alessandra Neris (11) 99104-4938 – alessandra@neris-comunicacao.com.br

Daniel Neris (13) 98881-9596 – daniel@neris-comunicacao.com.br

Ana Maria C. Garcia – Diretora de Marketing – (44) 3033-6300 - marketing@db1.com.br