

DISTINGUIR QUANDO APLICAR E QUANDO NÃO APLICAR RAD

CONCEITOS

A metodologia RAD tem por objetivo fazer a entrega dos sistemas em menos tempo e com menos erros do que os métodos tradicionais de desenvolvimento. No entanto, para implementar esta metodologia, as empresas precisam satisfazer algumas condições, que são (BERGER & BEYNON-DAVIES, 2009):



Pessoas

Profissionais qualificados e com rápida adaptação, além de trabalhar bem em equipe.



Gerenciamento

Equipes com poder de decisão para evitar perda de tempo, o que é comum nos modelos tradicionais.



Uso de ferramentas computacionais (CASE)

Programas que facilitem criação de diagramas e interface com usuário, componentes reutilizáveis (APIs, frameworks e templates, por exemplo) e de fácil manutenção.

A aplicação da RAD gera sistemas cujas telas e demais componentes são padronizados, devido ao uso de ferramentas com bibliotecas e templates reutilizáveis. No entanto, algumas características, como desempenho do sistema e análise de risco, são menos tratadas, pois são atividades que demandam tempo em qualquer projeto. Portanto, **a RAD é mais adequada para softwares de baixa complexidade.**

Nesta parte do texto, é natural associar a RAD a uma metodologia ágil, inclusive alguns autores fazem essa associação, pois, de fato, há muitas semelhanças entre as duas metodologias e, portanto, as vantagens e desvantagens de ambas seriam idênticas. Entretanto, existem algumas diferenças entre elas. Na RAD, há uma limitação de trabalhar com várias equipes, enquanto no desenvolvimento ágil isso ocorre de modo normal. Outro ponto a ser considerado é o cumprimento dos prazos. Na RAD, o comprometimento é com a rapidez e qualidade das entregas nas iterações, na expectativa, é claro, de

que isso se reflita para o projeto como um todo. No caso dos métodos ágeis, existem prazos a cumprir do ponto de vista global do projeto.

Atenção

A RAD nem sempre é adequada para ser aplicada a um projeto, como será discutido mais adiante. Existem casos, inclusive, em que métodos tradicionais são mais pertinentes. Trata-se de uma metodologia que funciona muito bem sob certas circunstâncias e disponibilidade de recursos e que, em outros casos, não é recomendada, como veremos.

METODOLOGIA RAD – VANTAGENS E DESVANTAGENS

O modelo RAD tem por objetivo a entrega rápida, pois o tempo total de desenvolvimento é reduzido devido à reutilização dos componentes e ao desenvolvimento paralelo. Para que funcione bem, a RAD precisa de profissionais qualificados e que o cliente também se comprometa a colaborar a fim de que os protótipos evoluam para o sistema desejado no prazo determinado. **Caso não haja esse compromisso de ambos os lados, a metodologia poderá falhar.**



Apesar da rapidez e qualidade da entrega serem as principais vantagens das RAD, também há desvantagens em relação à escalabilidade dos projetos e à demanda por recursos. Sobre a rapidez da entrega, isso é obtido através do uso de ferramentas que auxiliam a conversão de requisitos em código, permitindo, assim, que os desenvolvedores e usuários possam interagir através de protótipos que já são funcionais. A respeito da melhoria da qualidade, com a RAD, será tão melhor quanto um software entregue que atende às necessidades dos usuários, onde são necessárias poucas intervenções para correção de erros, implicando, assim, em baixos custos de manutenção.

Portanto, a aplicação da RAD tem muitos benefícios para o desenvolvimento do projeto, em especial por integrar as equipes de software e seus clientes. Os times de desenvolvimento têm um aumento da sua produtividade, uma vez que rapidez e agilidade são prioridades. Isso melhora os resultados do projeto e torna possível que as entregas ocorram nos prazos estimados. Agora, clique e

saiba algumas das principais vantagens do uso de um método RAD (Berger & Beynon-Davies, 2009):

PRINCIPAIS VANTAGENS:

INTEGRAÇÃO ANTECIPADA DO SISTEMA E REDUÇÃO DE RISCOS

A entrega rápida de protótipos que funcionam por parte das equipes da RAD viabiliza que as empresas façam revisões das funcionalidades do projeto desde o início do ciclo de vida do software. Todos os aspectos que se referem ao sistema podem ser reavaliados, devido às iterações frequentes e pelos comentários e sugestões dos usuários, permitindo a análise do progresso dos trabalhos de forma mensurável, que determina se os cronogramas e orçamentos estão no caminho certo. A integração com outros sistemas e serviços é feita no final de um ciclo de vida de desenvolvimento, como ocorre nas metodologias tradicionais. Os testes são realizados durante cada iteração, de modo que as partes interessadas identifiquem e analisem os erros e tratem as vulnerabilidades, resolvendo-os rapidamente, para que não impactem no progresso do desenvolvimento. Redução no risco significa redução nos custos.

ADAPTABILIDADE E COMPARTIMENTAÇÃO DOS COMPONENTES DO SISTEMA

Durante o desenvolvimento, é mais fácil fazer modificações no software, ou seja, enquanto o projeto não é concluído, o software é maleável. É claro que alterações devem ser realizadas com cuidado, pois, por menor que possam parecer, podem afetar todo o sistema. De modo geral, os desenvolvedores podem usar essa flexibilidade para criar protótipos ao longo do processo. Dada essa característica iterativa, designers e desenvolvedores são incentivados a criar componentes funcionais e independentes que sejam reutilizáveis.

VERSÕES ITERATIVAS E MENOR TEMPO DE COLOCAÇÃO NO MERCADO

O uso de ferramentas que suportam o desenvolvimento fortalece as equipes de desenvolvimento a entregar protótipos prontos, ou seja, utilizáveis, para produção mais rápida do que o desenvolvimento feito pelas metodologias tradicionais. As equipes devem ser pequenas, pois a ideia é que o trabalho “repetitivo” seja feito por essas ferramentas, pois, assim, ocorre o aumento da produtividade. As iterações frequentes incentivam a quebra das tarefas, o que é conhecido como “granularização”. Essas tarefas são atribuídas aos membros da equipe, conforme a especialidade e experiência de cada um.

FEEDBACK CONSTANTE DO USUÁRIO

Trata-se de uma das principais características da RAD. A eficiência e a qualidade do projeto aumentam com a comunicação regular e o feedback constante dos usuários finais. A estrutura iterativa e o acesso aos componentes de UI / UX de um sistema aumentam ainda mais a importância do feedback dos usuários. O fato de os desenvolvedores terem a oportunidade de apresentar para os usuários os protótipos que construíram faz com que fiquem mais confiantes de que estão no caminho certo para satisfazer o cliente quando o produto final é entregue.

Uma das principais características da metodologia RAD é a entrega de protótipos funcionais. Exatamente por isso, a escalabilidade dos projetos é reduzida, pois a aplicação da metodologia sem adaptações inviabiliza a interação com o usuário para um sistema complexo. Este assunto será tratado um pouco mais adiante neste texto. A limitação do tempo de desenvolvimento das iterações é uma característica muito importante para fazer as entregas rápidas, porém é um limitador para a implementação de recursos mais avançados. Portanto, embora a RAD tenha diversos benefícios, há situações para as quais a metodologia não é adequada. Por exemplo, a RAD não funciona bem em projetos onde o risco técnico é alto. A seguir, clique e saiba sobre algumas das desvantagens da RAD (BERGER & BEYNON-DAVIES, 2009):

PRINCIPAIS DESVANTAGENS:

NECESSIDADE DE EQUIPES TECNICAMENTE MUITO QUALIFICADAS

A RAD depende fortemente das habilidades de modelagem – de dados e processos –, e, além disso, é necessário que os profissionais tenham rápida adaptação, pois, devido à natureza da metodologia, é possível haver mudanças significativas ao longo das iterações. Portanto, uma equipe bem qualificada é essencial para identificar e entregar os requisitos de negócios.

FOCO EXIGENTE NA INTERFACE

Os clientes fazem a avaliação da qualidade de uma solução com base no que eles podem interagir em um protótipo. A RAD é caracterizada por iterações nas quais são implementadas novas funcionalidades – incrementos – e, ao final delas, são entregues protótipos. Os clientes podem avaliar os progressos a cada novo lançamento. Devido à rapidez com que os protótipos são implementados, às vezes, os desenvolvedores não aplicam as melhores práticas no projeto, preocupando-se, essencialmente, com o que deve ser entregue para satisfazer as necessidades imediatas do usuário. Isso pode

gerar dívidas técnicas, o que pode implicar em problemas quando for entregar a versão final do sistema.

REQUER ALTO NÍVEL DE COMPROMETIMENTO DE TODAS AS PARTES INTERESSADAS

Nos métodos de desenvolvimento tradicionais, os requisitos funcionais são definidos no início do projeto e não são revisitados posteriormente; portanto, clientes e equipes de desenvolvimento não interagem ao longo do projeto. Já na RAD, o entendimento do projeto evolui à medida que é feito; logo, a colaboração entre as partes interessadas é essencial. Caso isso não ocorra, pode haver um comprometimento muito sério da qualidade do projeto.

REQUER SISTEMAS MODULARES E É DIFÍCIL PARA PROJETOS DE GRANDE ESCALA

A natureza da RAD em fazer entregas rápidas ao longo das iterações demanda foco nas partes essenciais do sistema. Desse modo, outros elementos importantes, como questões de segurança, desempenho e tratamento de erros, por exemplo, podem ficar comprometidos. Outro ponto que deve ser considerado é que, para fazer entregas “rápidas” de protótipo, a RAD demanda ferramentas que são padronizadas, ou seja, o desenvolvedor tem pouca flexibilidade no desenvolvimento. Para projetos de baixa complexidade, essa é uma característica muito interessante, mas, para projetos de grande escala, em que há muitas outras tecnologias envolvidas, a RAD não é adequada.

COMPARAÇÃO DE RAD COM OUTRAS METODOLOGIAS

RAD

- A primeira e mais importante diferença entre a RAD e as demais metodologias de desenvolvimento é a concentração dos esforços pelo desenvolvimento rápido com entregas constantes de protótipos.
- Na RAD trabalha-se com uma única equipe de desenvolvimento, com poucos membros. A ideia é melhorar a qualidade da comunicação e a transferência rápida de informações.
- Na RAD, é fundamental envolver o usuário em todo o processo de desenvolvimento.

OUTRAS METODOLOGIAS

- Nos outros modelos, de modo geral, a preocupação principal é entregar um produto funcional para o cliente.
- Enquanto em outras metodologias trabalha-se com equipes maiores divididas em diferentes especializações, como, por exemplo, o modelo para equipe ágil,
- Nas metodologias tradicionais, usuários são envolvidos apenas no início e no final do ciclo de desenvolvimento do projeto.

Como foi apresentado no começo deste módulo, a RAD e as metodologias ágeis têm muitos pontos em comuns. De fato, a RAD é um predecessor das metodologias ágeis, embora essas últimas sejam mais abrangentes do que uma metodologia de desenvolvimento. Suas principais diferenças estão no estabelecimento de um cronograma de entregas, na importância dos comentários dos usuários para o projeto e no desenvolvimento focado em desenvolver protótipos do projeto – que é o caso da RAD – em detrimento do desenvolvimento de características do projeto – que é o caso das metodologias ágeis.

METODOLOGIA RAD – QUANDO APLICAR E QUANDO NÃO APLICAR

A metodologia RAD é adequada para determinados tipos de desenvolvimento, como, por exemplo, quando a interatividade do front-end dos sistemas é uma característica muito importante em detrimento da complexidade do back-end (BERGER & BEYNON-DAVIES, 2009).

Quando aplicar

A RAD não é adequada para as seguintes situações: desenvolvimento de sistemas críticos em tempo real; sistemas de infraestrutura muito grandes e quando os requisitos funcionais precisam ser especificados detalhadamente ainda no início do projeto. Para complementar, a aplicação da RAD não é adequada quando o sistema deve interagir com outros sistemas já existentes.

Quando não aplicar

A RAD é adequada para projetos de pequena escala com equipes também otimizadas de quatro a oito pessoas. No caso de projetos de grande escala, caso seja aplicada a RAD, é necessário que sejam divididos em projetos menores e mais facilmente gerenciáveis. Além disso, é preciso que haja o

envolvimento das partes interessadas, de modo a maximizar o entendimento do sistema. Para isso, as **equipes precisam ter poderes para tomar decisões**.

Nas oficinas de trabalho, em que ocorrem as reuniões entre usuários e desenvolvedores, são usadas ferramentas e técnicas de modelagem para confirmar e documentar o entendimento dos requisitos. Além disso, o desenvolvimento de protótipos auxilia de modo concreto a possibilidade de avaliar se o caminho escolhido para o desenvolvimento do sistema está correto através dos comentários dos usuários.

De forma resumida, a metodologia RAD é adequada desde que as seguintes condições sejam satisfeitas, clique para conhecê-las:

CONDIÇÕES PARA APLICAÇÃO

- Disponibilidade de profissionais experientes e comprometidos com um processo de desenvolvimento intensivo e contínuo.
- Comprometimento do cliente para participar efetivamente do desenvolvimento do projeto avaliando os protótipos e contribuindo com comentários que deem suporte para que os desenvolvedores avancem no desenvolvimento com aperfeiçoamentos e ajustes.
- Seu cliente está disposto a seguir cronogramas do projeto e um cronograma para conclusão do modelo? Todas as partes interessadas precisam estar presentes para aplicar efetivamente essa metodologia.
- É necessário que o sistema possa ser dividido em módulos.
- Também é necessário ter à disposição programas e infraestrutura adequada para aplicar a RAD.

Para a aplicação da RAD, o projeto, a estrutura tecnológica da organização e a própria cultura da empresa precisam estar adequados para que o desenvolvimento do sistema seja bem-sucedido.

A RAD funciona perfeitamente para sistemas que podem ser divididos em módulos. Em especial, esses são alguns exemplos para os quais a RAD se encaixa bem:

- Sistemas que podem ser modularizados.
- Aspectos de interatividade com o usuário UI/UX são muito importantes no projeto.
- Ter à disposição profissionais qualificados no uso de ferramentas adequadas ao desenvolvimento rápido; em especial, no uso de frameworks.
- Os clientes entendem a importância da interatividade com os desenvolvedores e têm a expectativa de receber protótipos ao longo do projeto.

- Desde o início do projeto, já é conhecido que haverá mudanças durante o processo de desenvolvimento.

Essas são algumas situações em que é esperado que a RAD funcionará bem. A RAD é uma metodologia de desenvolvimento com processos bem definidos, em que a colaboração entre usuários e desenvolvedores é fundamental para que os projetos tenham sucesso. A fim de que possa ser aplicada, algumas condições precisam ser satisfeitas. Além de possuir vantagens, a RAD possui desvantagens no sentido que não é adequada para sistemas complexos e de grande escala.