



Universidade Presbiteriana
Mackenzie

RAD – Rapid Application Development

Desenvolvimento Rápido de Aplicação

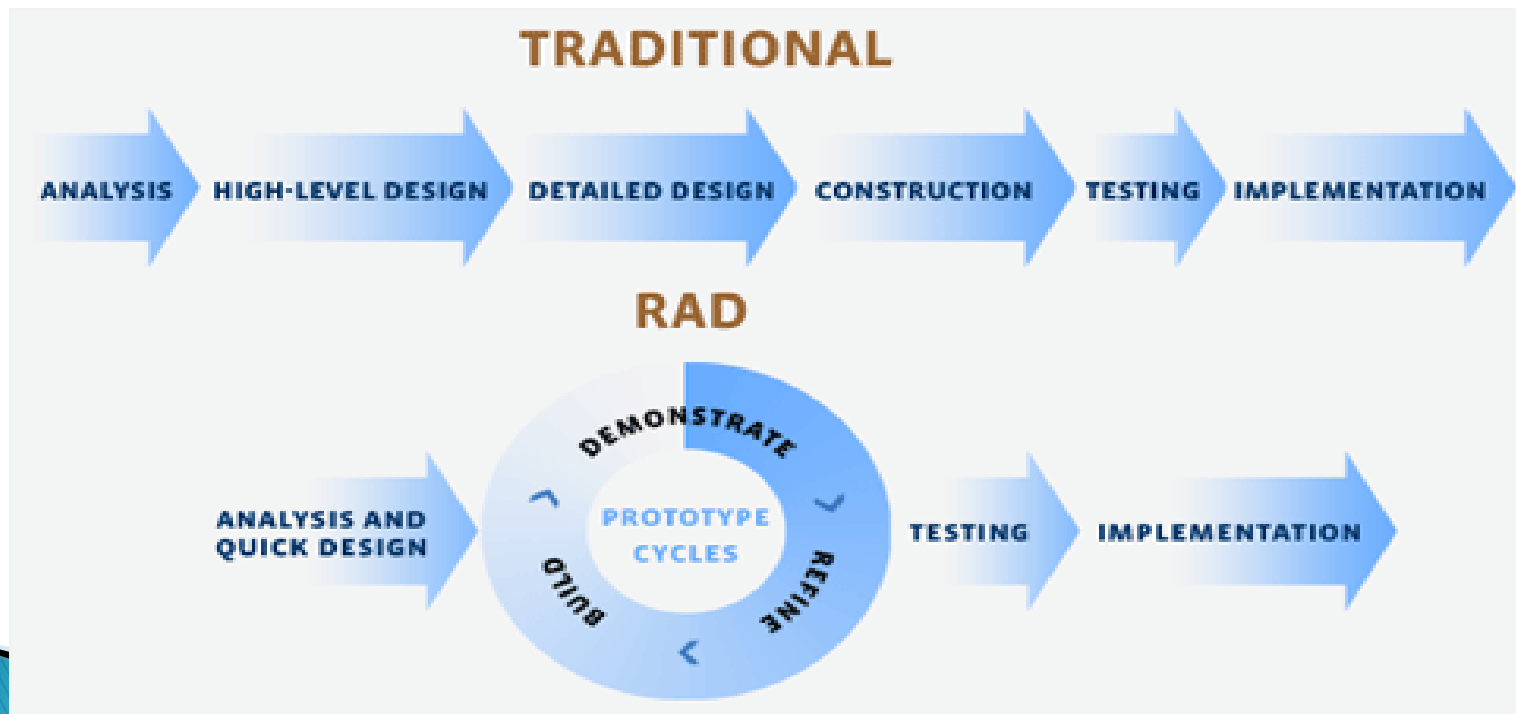
Brunna Leite

Rafaela Trevizan

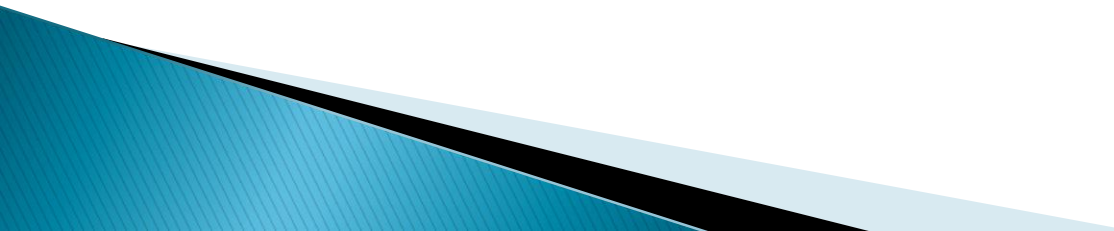
Renan Gonçalves

Samanta de Masi

- ▶ RAD é um modelo de processo de desenvolvimento de software iterativo e incremental que enfatiza um ciclo de desenvolvimento extremamente curto (entre 60 e 90 dias).



- ▶ A quantidade de ferramentas livres disponíveis para o desenvolvimento RAD aumentou nos últimos anos e mesmo assim uma parcela dos desenvolvedores tem se mostrado cética em relação a maturidade e funcionalidades disponíveis nestas ferramentas devido ao fato de elas não estarem presentes, por padrão, na maioria das distribuições mais utilizadas. Além disso, elas não contam com o suporte de nenhum grande distribuidor e, ainda que seja bem suportada pela comunidade, este acaba sendo um empecilho para alguns desenvolvedores.

- ▶ Outro foco para se utilizar no desenvolvimento RAD é a utilização de frameworks, onde esses estão disponíveis para desenvolvimento em linguagens como C e C++ sendo as mais utilizadas em ambientes baseados em software livre, embora estas linguagens não sejam tão produtivas para o desenvolvimento de aplicações rápidas.
 - ▶ O número de fases do processo varia de acordo com os autores.
- 


Segundo Kerr

O processo se divide em 5 fases:


- ▶ **Modelagem de Negócio**
- ▶ O fluxo de informações entre as funções de negócio é modelado de modo a responder às seguintes questões: – Que informação direciona o processo de negócio? – Que informação é gerada? – Quem a gera? – Para onde vai a informação? – Quem a processa? Na modelagem de negócio são levantados os processos suportados pelo sistema.
- ▶ **Modelagem dos dados**
- ▶ A modelagem de dados responde a um conjunto de questões específicas que são relevantes a qualquer aplicação. O fluxo de informação definido na fase de modelagem de negócio refinado e de forma a extrair os principais objetos de dados a serem processados pelo sistema, qual a composição de cada um dos objetos de dados, onde costumam ficar, qual a relação entre eles e quais as relações entre os objetos e os processos que os transformam.
- ▶ **Modelagem do Processo**
- ▶ Os objetos de dados definidos na modelagem de dados são transformados para conseguir o fluxo necessário para implementar uma função do negócio. Descrições do processamento são criadas para adicionar, modificar, descartar ou recuperar um objeto de dados.
- ▶ **Geração da Aplicação**
- ▶ O RAD considera o uso de técnicas de quarta geração, trabalha com a reutilização de componentes de programa existentes quando possível, ou cria componentes reusáveis. São usadas ferramentas automatizadas para facilitar a construção do software.
- ▶ Ex: Clarion, Delphi, Visual Basic, Asp.net, etc.
- ▶ **Teste e Modificação**
- ▶ Como o processo do RAD enfatiza o reuso, muitos componentes já estão testados, isso reduz o tempo total de teste. Todavia os novos componentes devem ser testados e todas as interfaces devem ser exaustivamente exercitadas.

Segundo Stephen E. Cross Diretor do SEI (Software Engineering Institute),


Ele divide em 6 passos:

- ▶ Projeto e análise baseado no cenário
 - ▶ Projeto e análise de Arquitetura
 - ▶ Especificação de Componentes com o máximo de reuso
 - ▶ Desenvolvimento rápido dos módulos remanescentes
 - ▶ Testes freqüentes com o usuário final
 - ▶ Campo com ferramentas de suporte para permitir a evolução
- 

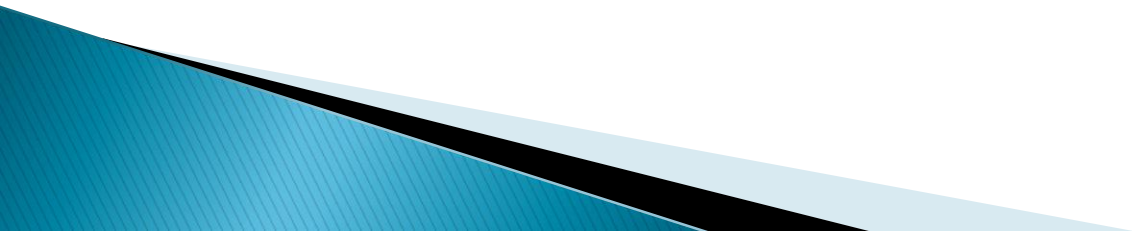
Vantagens

- ▶ Permite o desenvolvimento rápido e/ou a prototipagem de aplicações;
 - ▶ Enfatiza um ciclo de desenvolvimento extremamente curto (entre 60 e 90 dias);
 - ▶ Criação e reutilização de componentes;
 - ▶ Maior flexibilidade (desenvolvedores podem reprojetar praticamente a vontade)
 - ▶ Grande redução de codificação manual
 - ▶ Envolvimento maior do usuário
 - ▶ Provável custo reduzido
- 

Desvantagens

- ▶ Se uma aplicação não puder ser modularizada de modo que cada função principal seja completada em menos de 3 meses, não é aconselhável o uso do RAD
 - ▶ Não é possível usar em projetos grandes, pois necessitaria uma grande equipe, deixando o custo muito elevado
 - ▶ O envolvimento com o usuário tem que ser ativo
 - ▶ Não é aconselhável quando os riscos técnicos são altos
 - ▶ Mais difícil de acompanhar o projeto
 - ▶ Requisitos podem não se encaixar (conflitos entre desenvolvedores e clientes)
 - ▶ Sucessos anteriores são difíceis de se reproduzir
- 

O RAD deve ser evitado quando:

- ▶ A aplicação precisa interagir com outros programas
 - ▶ Performance é essencial
 - ▶ A distribuição do produto será em grande escala
 - ▶ Jogos de computador
- 

FIM