**FABIO DELA BRUNA**

**MÁRCIO OZÓRIO TEIXEIRA**

**MODELO DE PROCESSO DE SOFTWARE: RAD (RAPID APPLICATION DEVELOPMENT)**

**Tubarão, 2009.**

**1 CONCEITO**

O termo RAD (Rapid Application Development ou Desenvolvimento Rápido de Aplicação), registrado por James Martin em 1991, é uma metodologia de processo de desenvolvimento de software, diferente das técnicas habituais, constituída por uma série de técnicas de desenvolvimento e aplicativos que possibilitam as organizações a realizarem o desenvolvimento de programas de forma rápida (entre 60 e 90 dias), reduzindo assim, os custos de desenvolvimento e mantendo a qualidade.

* 1. VANTAGENS
* Poupança de tempo, dinheiro e esforço.
* Grande proximidade entre os requisitos do usuário e as especificações do sistema.
* Apropriado em situações de rápida alteração das condições de negócio e no aproveitamento de oportunidades estratégicas.
* Alterações rápidas ao design a pedido dos usuários.
* Concentra-se nos elementos essenciais segundo o ponto de vista do usuário.
* Sentimento de posse do usuário.
  1. DESVANTAGENS
* A qualidade geral do software pode ser inferior (ex. falta de atenção a mecanismos de controle interno e validação).
* Falta de alinhamento entre o software e a realidade do negócio devido à falta de informação sobre aspectos importantes associados aos processos.
* Não verificação dos standards de programação (consistência dos nomes, falta de documentação).
* Dificuldades na reutilização dos módulos em sistemas futuros.
* Falta de escalabilidade do sistema, face a requisitos de novos usuários.
* Administração de sistemas pouco cuidada (falta de integração com o modelo de dados da empresa, falta de sistemas de recuperação de falhas).

**2 TÉCNICAS ENVOLVIDAS**

Os sistemas computacionais usados desde a década de 70 são baseados na técnica estruturada de desenvolvimento de software, tornando o processo de desenvolvimento muito mais lento, fazendo com que muitas vezes após a conclusão do software pela empresa de desenvolvimento ele se torne obsoleto, pois já existem outros requisitos a serem preenchidos pelos desenvolvedores. Desta forma, atualmente são utilizadas as ferramentas RAD, a qual são utilizadas ferramentas poderosas de desenvolvimento para desenvolver rapidamente sistemas de qualidade. Essas novas ferramentas são mais intuitivas e visuais, possibilitando ao projetista de software que ele desenvolva o software de forma “visual”, desenvolvendo a interface com o usuário, além do mais geram o código numa linguagem orientada a objetos, permitindo o reuso de componentes.

Entretanto, com a finalidade de proporcionar maior "adaptabilidade" possível a qualquer necessidade de programação, essas ferramentas tendem a incluir um amplo número de componentes, recursos, objetos, propriedades, e vários outros elementos, oferecendo assim uma ampla gama de opções aos programadores do sistema.

**3 FERRAMENTAS**

As ferramentas computacionais são elementos fundamentais, precisando possuir capacidades grandiosas no que se refere aos aspectos da Gestão de Projetos e à construção de Diagramas, não sendo de desfavorecer as habilidades em termos de Prototipagem e geração de Código. As ferramentas CASE integrado, como camada de interface com o usuário, serviço de integração, funções de controle de acesso, pretendem desenvolver uma plataforma na qual os desenvolvedores e os usuários tenham a capacidade de trabalhar em conjunto.

As ferramentas especializadas empregadas no RAD são destinadas a suportar o desenvolvimento visual, a criação de falsos protótipos para simulação, a criação de protótipos de trabalho, múltiplas linguagens, a calendarização da equipe, o trabalho e a colaboração de equipe, a utilização de componentes reutilizáveis, a utilização de APIs standard, ou o controle de versões.

Exemplo de ferramentas RAD utilizadas hoje em dia: e-Gen Developer, Netbeans, Borland Delphi, Delphi for Php, Magic eDeveloper e dentre outras.

**4 METODOLOGIAS**

As tarefas a serem realizadas para desenvolver o software de acordo com o processo de desenvolvimento RAD são os seguintes, conforme abaixo detalhados:



* O **Planejamento de Requisitos** reúne informações das fases de planejamento e análise tradicionais. Nesta fase, as informações entre as funções de negócio são modeladas, por meio de analistas de negócios e usuários com bons conhecimentos que definem os requisitos do software. Após a identificação dos subsistemas específicos para desenvolvimento, os analistas de negócios, os usuários e os desenvolvedores realizam uma reunião de planejamento de requisitos para alcançar um acordo em conjunto.
* Durante o **Design do Utilizador**, os usuários finais e os profissionais de sistemas de informação participam de sessões para criar protótipos, obtendo desta forma os requisitos do sistema. Os protótipos são o apoio do design físico do sistema em desenvolvimento. Neste processo não existem especificações baseadas em papel e o usuário apresenta o seu acordo imediato. Contudo, o período entre o fim do design e a entrega do software ao usuário pode demorar apenas três meses.
* Na fase de **Construção**, os profissionais de sistemas de informação que criaram o design, geram o código com o auxílio das ferramentas case. Os usuários finais participam do mesmo modo, validando todas as funções do software de acordo com o andamento do desenvolvimento da aplicação. Para softwares mais simples a fase de Design pode ser tratada com a fase de Construção, encurtando o desenvolvimento.
* Na fase de **Entrega** o software é entregue aos usuários finais. Como o processo do RAD procede tão rapidamente, esta fase deve ser idealizada nas fases antecedentes, pois abrange várias atividades habituais associadas à implementação (testes, formação dos usuários, mudanças organizacionais, suportar o funcionamento em paralelo dos sistemas antigos e dos novos), mas de uma forma bastante acelerada. Desta forma o RAD pretende produzir em apenas a metade do tempo que levaria da forma tradicional.

**REFERÊNCIAS**

DEBONI, José Eduardo Zindel. **CASE e RAD:** concorrência ou colaboração. Disponível em: < http://www.voxxel.com.br/pages/pdf/caserad.pdf >. Acesso em: 28 mar. 2009.

DIZ, Ricardo. **Rapid Application Development**. Disponível em: <http://alunosisgb.home.sapo.pt/conteudos/cgsi/requisitos\_si/rapid\_application\_development.pdf>. Acesso em: 28 mar. 2009.

SANTOS, Mário. **Rapid Application Development.** Disponível em: < http://paginas.ispgaya.pt/~msantos/es\_artigos\_tecnicos\_1/33\_RAD.pdf >. Acesso em: 28 mar. 2009.

SOMMERVILLE, Ian. **Engenharia de Software**. 8 ed. São Paulo: Addison-Wesley, 2007. 552 p.