

Sistemas de Informação
Disciplina: Gestão de Sistemas Digitais na WEB
Professor: José Maurício S. Pinheiro

AULA 1 – Introdução a Análise e Projeto de Sistemas

O número de sistemas e aplicativos para a Internet tem crescido muito nos últimos anos causando impacto no desenvolvimento dos sistemas computacionais. A Internet possibilita novos meios e importantes modos de comunicação comercial ou não, diversos do tradicional, os quais se refletem também no projeto de desenvolvimento de sistemas Web. Por esses motivos, hoje se faz necessário o uso de uma metodologia para a construção destes sistemas. Os aplicativos para uso na Internet são diferentes de outras categorias de software e têm características exclusivas: são dirigidos a conteúdo, estão em constante evolução, têm curto prazo de desenvolvimento, dentre outras.

1. Introdução

A Internet foi concebida inicialmente para compartilhar informações científicas entre universidades e cientistas. O conteúdo era estático e apenas textual, não havia imagens, sons, animações ou conteúdo gerado dinamicamente para cada usuário, a interação era limitada, a navegabilidade era fácil, alto desempenho era desejável, mas não essencial, os sites eram desenvolvidos por apenas uma pessoa ou um pequeno grupo. A Figura 1 mostra as evoluções que passa a Internet.



Figura 1 - Evolução da Internet

Escopo e complexidade foram aumentando, pequenos serviços foram cedendo espaço para grandes aplicativos, e também aumentou a complexidade dos projetos e as dificuldades de desenvolvimento, manutenção e gerenciamento. Tudo feito com pouca disciplina, sem preocupação com técnicas e métodos padronizados ou maneiras de controlar a qualidade. Segundo Ginige e Murugesan, especialistas e pesquisadores da área de computação, 84% dos sistemas entregues não atendem às necessidades do cliente; 79% dos projetos são entregues com atrasos e 63% têm custo maior que o orçamento previsto. Mais de 50% dos sistemas prontos são de baixa qualidade e faltam funcionalidades necessárias.

Como resultado, desenvolvedores e usuários começaram a se preocupar com a maneira como sistemas Web complexos estão sendo criados, bem como com seus níveis de desempenho, qualidade e integridade. E aí surge a Engenharia para a Web.

“Num projeto de software você sempre deve começar levando em consideração os dados - a base para todos os demais elementos do projeto. Após estabelecida a base, a arquitetura tem de ser extraída. Só então deve realizar outras tarefas de projeto.” (PRESSMAN, 2011).

Tanto a Engenharia de Software tradicional quanto a Engenharia de Software para WEB preocupam-se em facilitar o desenvolvimento e manutenção de produtos de software complexos, aplicando para isso métodos, técnicas, ferramentas e modelos específicos. Todas essas metas e técnicas utilizadas visam gerenciar a qualidade de produção, prazos e pessoas envolvidas no projeto.

Entretanto, ainda que as duas engenharias tenham o mesmo objetivo e que a Engenharia de Software para o desenvolvimento WEB utilize alguns princípios da Engenharia Tradicional, não é possível utilizar as mesmas técnicas da Engenharia de Software Tradicional na Engenharia de Software para o desenvolvimento de aplicativos para WEB.

Se por um lado uma aplicação convencional possui um público muitas vezes definido de usuários, que pode ser controlado, o desenvolvimento para WEB, além de diversificado, o número de usuários pode ser muito grande, às vezes sendo impossível saber exatamente o perfil de todos os usuários que irão interagir com o sistema a ser construído.

1.1. Ciclo de Vida

O desenvolvimento e avaliação de Software tradicional normalmente não são totalmente adequados o desenvolvimento de aplicativos na WEB. Esses sistemas devem ser desenvolvidos com enfoque forte no usuário, ter um ciclo de vida rápido e uma grande preocupação com a acessibilidade. Portanto, uma das principais diferenças entre as duas Engenharias é o ciclo de vida do produto a ser desenvolvido (Fig. 2).

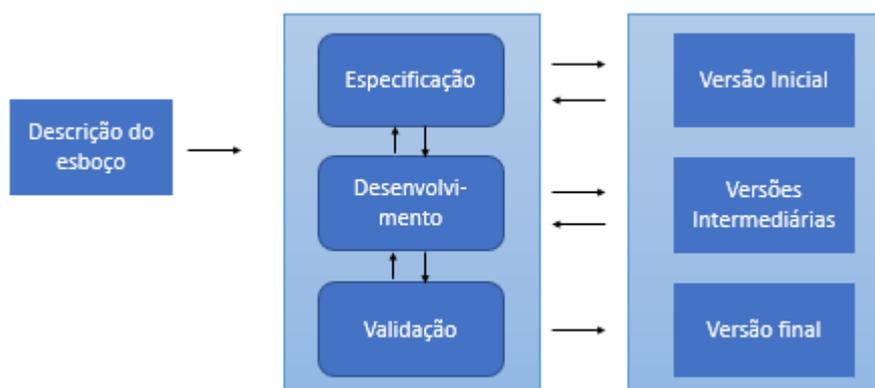


Figura 2 - Ciclo de vida do software

A evolução do Software tradicional é muitas vezes menor que a evolução de um aplicativo para WEB, devido as tecnologias e informações estarem em constante evolução. O que influi realmente no ciclo de vida de uma aplicação para WEB é seu

desenvolvimento ser uma mistura de arte e programação, entre marketing e computação, entre relações internas e externas além da multidisciplinaridade envolvida: várias pessoas com diferentes conhecimentos necessitam trabalhar juntas (designers, Webmasters, programadores, publicitários etc.) fazendo com que o seu desenvolvimento se torne um desafio.

2. Protocolo HTTP

A base da comunicação entre os diversos dispositivos que se conectam pela Internet é feita de uma maneira padronizada. A esta padronização dá-se o nome de protocolo. Os protocolos de comunicação descrevem a maneira como o contato e o envio de pacotes é feito pela rede. Apesar de existirem diversos protocolos de comunicação utilizados na internet, no nível de aplicação o mais comum é o protocolo HTTP (Hyper-Text Transfer Protocol). O protocolo HTTP pertence a camada 7 do Modelo de Referência OSI (RM-OSI), conforme ilustra a Figura 3.

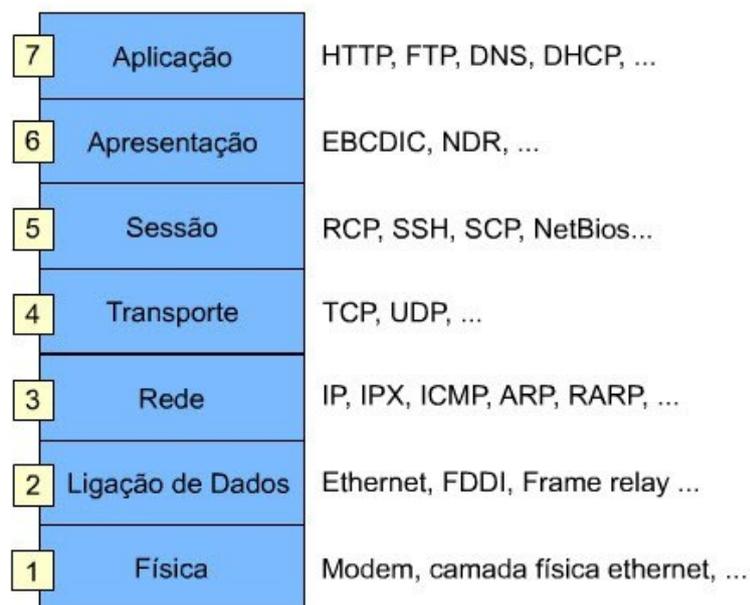


Figura 3 - Protocolo HTTP no Modelo OSI

O significado do HTTP, em português, é: protocolo de transferência de hipertexto. Neste caso, uma padronização para a transferência de hipertexto via rede. Os hipertextos são textos que possuem mais recursos como imagens, sons, formatação específica, hiperlinks (utilizados para a navegação entre diferentes hipertextos), entre outros. Um exemplo de arquivo de hipertexto que é comumente transmitido via internet é o arquivo com a extensão .html (ou .htm). A linguagem HTML (Hyper-Text Markup Language) é utilizada para o desenvolvimento de páginas de internet (arquivos de hipertexto) que são transmitidos via Internet sempre que um site desenvolvido com esta tecnologia é acessado.

O protocolo HTTP trabalha com um modelo de requisição e resposta e utiliza-se de outros protocolos de comunicação: O TCP e o IP. A junção dos protocolos TCP e IP são a base da comunicação na internet e a Figura 4, mostra um diagrama da interação entre os protocolos citados.

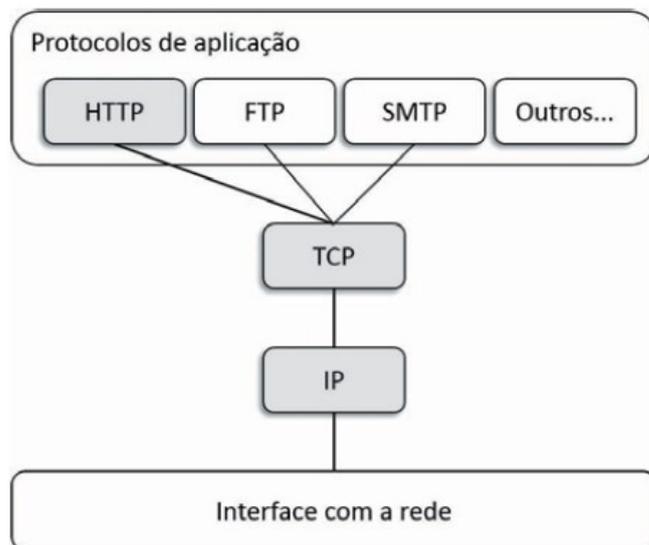


Figura 4 - Diagrama de interação entre os protocolos HTTP, TCP e IP

O protocolo HTTP trabalha com um modelo de requisição e resposta. Para melhor exemplificar este modelo, um diagrama é exibido na Figura 5. Este diagrama indica a interação entre o cliente e o servidor por meio de requisições utilizando o protocolo HTTP.



Figura 5 - Modelo requisição-resposta do protocolo HTTP

As requisições e respostas enviadas utilizando o protocolo HTTP podem funcionar de diferentes formas, como por exemplo:

- Requisição de exibição de uma página web;
- Envio de informações de formulário;
- Envio de dados por meio da URL;
- Outros.

Um exemplo clássico de requisição e resposta utilizando-se o protocolo HTTP é o acesso à uma página web. Quando o cliente (usuário de um navegador de internet) acessa o endereço de uma página (que está armazenada em um servidor na internet), uma requisição é enviada a este servidor indicando qual página deve ser exibida. O servidor, como resposta envia a página HTML para que seja exibida no corpo do navegador.

Pelo lado do servidor, a programação das rotinas que são executadas em um sistema web é uma tarefa que exige conhecimento de diversas tecnologias e métodos de programação, assim como conhecimento de arquiteturas de desenvolvimento, interação com bancos de dados e engenharia de software. As rotinas mais importantes e vitais de um sistema devem seguir uma arquitetura de

desenvolvimento que garanta manutenibilidade, organização, desempenho e produtividade e aliado às tecnologias.

3. Engenharia WEB

Pode-se dizer que a importância de se construir um sistema está relacionada a um sistema que atenderá às expectativas do cliente. Deve ser projetado para ser um sistema confiável, usual, adaptável, apresentando essas e outras características no sistema finalizado. É no projeto de sistema que os profissionais envolvidos, bem como o cliente podem perceber necessidades não contempladas durante a extração de requisito com o profissional responsável, podendo essas serem implantadas em tempo hábil, com menor desgaste dos profissionais e menor custo. Segundo Pressman, sistemas e aplicativos da Web são caracterizados por disponibilizar grande quantidade de conteúdo e funcionalidade para grande população de usuários. A Engenharia para a Web é, portanto, o processo utilizado para criar aplicativos Web de alta qualidade.

A Engenharia para Web não é igual à Engenharia de Software tradicional, mas compartilha muitos conceitos e princípios fundamentais, com ênfase nas mesmas técnicas de gerenciamento e atividades. Há pequenas diferenças na maneira como essas atividades são conduzidas, mas a filosofia que dita uma abordagem disciplinada para o desenvolvimento de um sistema de computador é a mesma.

Ao mesmo tempo em que adota muitos princípios da Engenharia de Software, a Engenharia para a Web incorpora novas abordagens, metodologias, ferramentas, técnicas e normas para atender os requisitos exclusivos dos sistemas para a Web. O desenvolvimento de sistemas para a Web é significativamente diferente do desenvolvimento de software tradicional e apresenta vários desafios adicionais. Também é um erro achar que o desenvolvimento de aplicativos para Web é apenas a criação de páginas utilizando HTML, FrontPage ou Dreamweaver. E da mesma forma é equivocado achar que desenvolvimento para a Web envolve apenas a manipulação de diversas mídias e criação de conteúdo. O desenvolvimento para a Web é uma mistura de publicações impressas e desenvolvimento de software, entre marketing e computação, entre comunicações internas e relações externas, e entre arte e tecnologia.

A construção de um sistema para a Web necessita do conhecimento de pessoas de diferentes áreas. Como resultado, a Engenharia para a Web é multidisciplinar, e dela participam áreas como: análise de sistemas e projetos; engenharia de software; engenharia de hipermídia e hipertexto; engenharia de requisitos; interação humano-computador; desenvolvimento de interface de usuário; engenharia de informação; indexação e recuperação de informações; teste; modelagem e simulação; gerenciamento de projetos; e projeto gráfico e apresentação (Fig. 6).

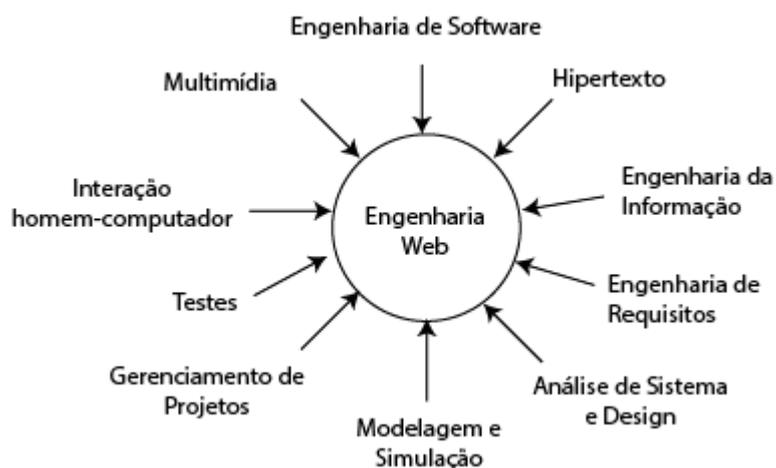


Figura 6 - Engenharia WEB é multidisciplinar

Sistemas e aplicações baseados na Web (WebApps) produzem uma complexa matriz de conteúdo e funcionalidade para ampla população de usuários finais. A engenharia da Web (WebE) é o processo usado para criar WebApps de alta qualidade. A WebE não é um clone perfeito da engenharia de software, mas toma emprestados muitos dos conceitos e princípios fundamentais da engenharia de software. Além disso, o processo WebE enfatiza atividades técnicas e de gestão similares. Há diferenças sutis no modo pelo qual essas atividades são conduzidas, mas a filosofia dominante determina uma abordagem disciplinada para o desenvolvimento de um sistema baseado em computador.

Para Pressman (2011), o objetivo da atividade de projetar é gerar um modelo ou representação que apresente solidez, comodidade e deleite. Para tanto, temos de praticar a diversificação e, depois, a convergência.

Além dos princípios gerais de projeto, Hooker (1996, apud PRESSMAN, 2006) enumera sete princípios gerais da engenharia de software que se aplicam também ao projeto de software. São eles:

- Um sistema de software existe para fornecer valor aos clientes e usuários.
- Todas as decisões, inclusive as de projeto, devem ser tomadas tendo isso em mente.
- Todo projeto de software deve ser tão simples quanto possível sem, no entanto, descartar características de qualidade importantes em nome da simplicidade.
- O comprometimento com a visão arquitetural do sistema é essencial para o sucesso do projeto de software.
- Os modelos elaborados na fase de projeto serão usados posteriormente por desenvolvedores responsáveis pela implementação, testes e manutenção do sistema. Assim, esses modelos devem ser claros, não ambíguos e fáceis de entender.
- Um sistema com um longo tempo de vida tem mais valor. Contudo, para ter vida longa, um sistema deve ser projetado para estar pronto para acomodar mudanças.
- A reutilização pode ajudar a poupar tempo e esforço, bem como aumentar a qualidade do sistema em desenvolvimento. Para conseguir um bom nível de reutilização, é necessário planejar o reuso com antecedência. Na fase de projeto, padrões arquitetônicos e padrões de projeto detalhado (design patterns) são bastante maduros e documentados.

4. Características dos aplicativos WEB

Segundo Pressman, as seguintes características podem ser encontradas na grande maioria dos aplicativos para a Web:

- **Rede Intensiva:** Aplicativos para a Web utilizam recursos de rede por natureza. Ele está em uma rede e deve atender as necessidades de diversas comunidades de clientes. Ele pode estar disponível na Internet (permitindo comunicação com o mundo todo), em uma Intranet (implementando comunicação em uma organização) ou ainda em uma Extranet (comunicação Inter redes).
- **Dirigido a Conteúdo:** Em muitos casos, a função primária do aplicativo para a Web é usar hipermídia para apresentar textos, gráficos, e vídeo para os usuários.
- **Evolução contínua:** Ao contrário dos aplicativos convencionais que evoluem através de uma série de versões planejadas e lançadas em determinados intervalos de tempo, os aplicativos para a Web evoluem continuamente.

Pressman também ressalta as principais diferenças entre desenvolver um aplicativo para a Web e desenvolver um software tradicional:

- **Imediatismo:** o tempo que um site completo precisa ficar pronto pode ser apenas alguns poucos dias ou semanas. Desenvolvedores devem, portanto, utilizar métodos de planejamento, análise, projeto, implementação e teste que estejam adaptados para estes cronogramas necessários no desenvolvimento para a Web.
- **Segurança:** aplicativos para a Web estão disponíveis via rede, é difícil ou até mesmo impossível limitar a população de usuários que irão acessar o aplicativo. Para poder proteger o conteúdo e fornecer os métodos seguros de transmissão de dados é preciso implementar medidas rígidas de segurança no aplicativo e na infraestrutura do mesmo.
- **Estética:** Boa parte do apelo dos aplicativos para a Web é o seu visual. Quando um aplicativo é projetado para vender produtos ou ideias, estética pode ser tão importante para o sucesso quanto o projeto técnico.

Estas características gerais se aplicam a todos os aplicativos para Web, mas com diferentes graus de influência. Podemos categorizar os aplicativos Web da seguinte forma:

- **Informacional** - Conteúdo apenas para leitura é fornecido com navegação simples e links;
- **Download** - Um usuário faz o download de informações dos servidores apropriados;
- **Personalizável** - O usuário personaliza o conteúdo para suas necessidades específicas;
- **Interação** - Comunicação entre uma comunidade de usuários ocorre em salas de bate-papo, fóruns ou mensagens instantâneas;
- **Entrada de Usuário** - Entradas baseadas em formulários são os mecanismos primários para a comunicação necessária;

- **Orientado a transações** - O usuário faz um pedido que é atendido pelo aplicativo;
- **Orientado a serviços** - O aplicativo fornece um serviço para o usuário;
- **Portal** - O aplicativo direciona o usuário para outros conteúdos ou serviços fora do domínio do portal do aplicativo;
- **Acesso a Banco de Dados** - O usuário faz uma consulta em um banco de dados e extrai informações;
- **Data warehousing** - O usuário consulta uma coleção de grandes bancos de dados e extrai informações.

Alguns passos podem ser observados para a construção de um aplicativo para a Web de sucesso:

- Entender o funcionamento geral e operacional do ambiente dos sistemas;
- Identificar e especificar requisitos técnicos e não-técnicos;
- Desenvolver uma arquitetura apropriada;
- Tratar satisfatoriamente as limitações não-técnicas;
- Identificar subprojetos ou subprocessos para implementar a arquitetura;
- Desenvolver e implementar os subprojetos;
- Incorporar mecanismos apropriados e efetivos para gerenciar a evolução e as manutenções.

E alguns problemas que normalmente ocorrem:

- Inconsistência das informações;
- Manutenção normalmente utiliza muitos recursos e tempo;
- Falta de escalabilidade;
- Necessidade de um “visual” comum;
- No início, requisitos são muito vagos;
- Os requisitos irão mudar consideravelmente durante o desenvolvimento e especialmente quando o aplicativo já estiver em uso;
- A tecnologia também vem mudando muito rapidamente.

5. Projeto de Sistemas WEB

As técnicas utilizadas na produção de Software tradicional aplicando Engenharia de Software são mais estáveis, isto decorre do fato que o desenvolvimento deste tipo de software ocorre a muito mais tempo, além disso, existem excelentes ferramentas disponíveis para análise, projeto, implementação e testes. Já no desenvolvimento WEB, as tecnologias estão em constante evolução, embora existam diversas ferramentas ao nível de implementação, existem poucas para análise, projetos e outras atividades para o desenvolvimento.

Por esses motivos, o projeto de sistemas Web necessita ser observado a partir de princípios da engenharia de software. Trata-se de atividade complexa e formada por componentes inter-relacionados, pensados e desenvolvidos, geralmente, por equipes de especialistas, de profissionais que possuem conhecimentos sobre o assunto e familiarizados com técnicas e ferramentas que proporcionam melhor aplicação da metodologia escolhida para a construção do projeto.

Um projeto de sistemas Web deve apresentar características que funcionarão de maneira eficaz com a construção dos sistemas. Propriedades como funcionalidade,

eficiência, robustez, confiabilidade, portabilidade, facilidade para a utilização, entre outras características.

Segundo Pressman (2011), a atividade de projeto de software engloba um conjunto de princípios, conceitos e práticas que levam ao desenvolvimento de um sistema ou produto com alta qualidade. Os princípios de projeto estabelecem uma filosofia que prevalece sobre as atitudes e ações do desenvolvimento, orientando as atividades para realizar o projeto.

O projeto de software é um processo iterativo através do qual os requisitos são traduzidos em uma “planta” para construir o software. Inicialmente, a planta representa uma visão holística do software. O projeto é representado em um alto nível de abstração - um nível que pode ser associado diretamente ao projeto específico do sistema e aos requisitos mais detalhados de dados, funcionalidade e comportamento.

5.1. Modelo de Processo

À medida que os aplicativos para a Web evoluem de estáticos para dinâmicos é necessário aplicar um gerenciamento sólido e os princípios de engenharia passam a ter fundamental importância.

A construção de projeto de sistemas Web apresenta peculiaridades refletindo as necessidades a serem atendidas pelos sistemas, particularmente os aspectos da arquitetura e da usabilidade do projeto para sistema Web que envolve múltiplas especificidades e particularidades da plataforma. É necessário, portanto, desenvolver um modelo de desenvolvimento que atenda tais requisitos de forma eficiente. Pressman propõe o seguinte modelo apresentado na Figura 7:

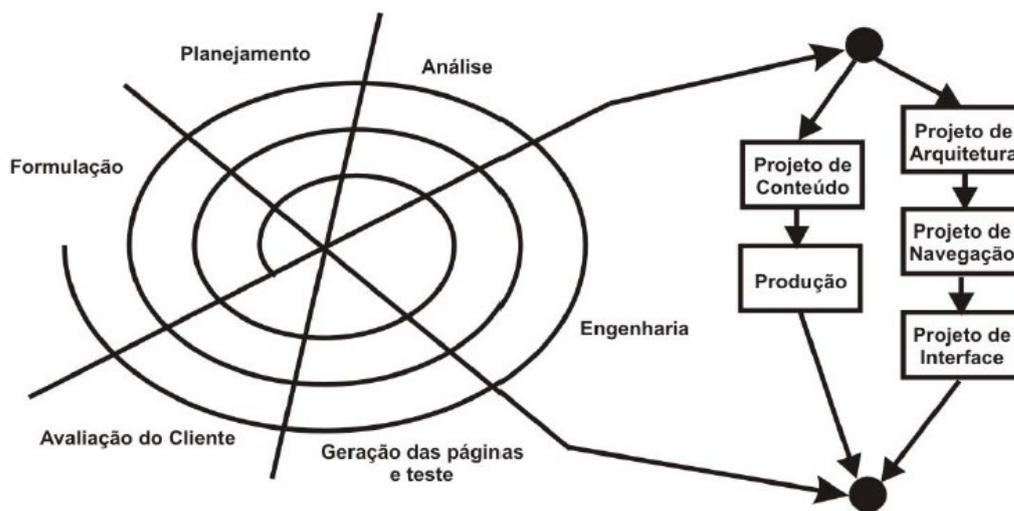


Figura 7 - Modelo para desenvolvimento de sistemas WEB

O processo começa com a formulação – uma atividade que identifica as metas e objetivos do aplicativo e determina um escopo para o primeiro incremento. O planejamento estima os custos do projeto, avalia riscos associados com o desenvolvimento, elabora um cronograma bem definido para o primeiro incremento, e um menos definido para os demais. Na Análise são estabelecidos os requisitos técnicos e identificados os itens de conteúdo que serão incorporados. Requisitos para projeto gráfico (estética) também são definidos.

A atividade de Engenharia incorpora duas atividades paralelas ilustradas no lado direito da figura. Projeto de Conteúdo e Produção são tarefas desenvolvidas pelos

membros “não-técnicos” da equipe. O objetivo destas tarefas é projetar, produzir e/ou obter todos os textos, gráficos, conteúdo de áudio e vídeo que serão integrados ao aplicativo. Ao mesmo tempo, uns conjuntos de tarefas técnicas de projeto são conduzidos.

Geração das Páginas é uma atividade de construção que faz uso pesado de ferramentas de automatização. O conteúdo definido na atividade de engenharia é fundido com os projetos de arquitetura, navegação e interface para produzir páginas em HTML, XML e outras linguagens orientadas a processo (como Java). Integração com middleware (CORBA, DCOM ou JavaBeans) também é feita durante esta atividade.

A atividade de Teste tenta descobrir erros em applets, scripts e formulários; e ajuda a garantir que o aplicativo irá funcionar corretamente em diferentes ambientes (com diferentes browsers).

Cada incremento produzido como parte do processo é revisto durante a Avaliação do Cliente. Neste ponto mudanças são pedidas (extensões de escopo ocorrem). Estas mudanças serão integradas ao sistema no próximo ciclo do processo incremental.

6. Avaliação da qualidade de um Sistema WEB

Os usuários da Internet possuem pontos de vista diferentes sobre o que constitui um bom sistema. Muitos apreciam gráficos atraentes, enquanto outros textos simples, boa navegação, muita informação, ou ainda informações resumidas. Mas quando se fala em alta qualidade, as seguintes características definidas por Olsina (1999) e detalhadas por ROCHA (2001) são indispensáveis por qualquer usuário:

- **Usabilidade:** entendimento geral do site, feedback e help on-line, facilidade de localização das informações, ausência de erros na navegação, disponibilidade de atalhos;
- **Funcionabilidade:** recursos de busca e recuperação, características de navegação e navegador e características relacionadas ao domínio da aplicação;
- **Confiabilidade:** processar corretamente todos os links, recuperar erros, validar, recuperar dados de entrada do usuário, tolerância às falhas, segurança;
- **Eficiência:** ser eficiente no tempo de resposta, rapidez na geração de páginas e de gráficos;
- **Manutenibilidade:** ter facilidade na correção, adaptação e extensibilidade