Sistemas Multimídia

Tópicos de Aula - Autoria multimídia

1. Ferramentas para desenvolvimento de multimídia

1.1 Aplicativos fechados

- Para os profissionais da área de aplicação => Aplicações fechadas => Requerem menor curva de aprendizado

* Interfaces amigáveis e “simples”
* Usuário utiliza diversos recursos de interface: caixas de diálogo, mensagens de ajuda, menus, botões intuitivos

- Tendência de se ter uma “linguagem de programação” nas aplicações para descrição de ações (algo parecido com as Macros)

1.2 Linguagens de programação

- Maior flexibilidade para criação de aplicações

- Exige conhecimento das técnicas de programação, estruturas de dados, ambientes de programação ...

- Dependendo da aplicação, programação em baixo nível, garantia de QoS, operações críticas (semáforos, threads ...)

- Linguagens de vovô: Cobol, Fortran, Lisp, Pascal ... (praticamente sem recursos multimídia

- Linguagens mais utilizadas: C, C++, Java, Python

- Linguagem C:

* Rápida
* Nem sempre tão simples de se manusear
* Utilizada na maioria dos sistemas operacionais
* Programação estruturada
* Facilita uso de recursos de hardware

- C++:

* Orientada a objetos
* Principal linguagem de programação em SOs e aplicativos que exigem maior desempenho
* Biblioteca com MUITOS recursos multimídia: Qt
* API bem desenvolvida

- Java:

* Baseadas em elementos de C++
* Criada para ser portável
* Interpretada (Máquina Virtual)
* Desempenho aquém do necessário para aplicações críticas
* Máquina virtual desenvolvida em C/C++
* Muito utilizada na internet
* API bem desenvolvida

- Python:

* “Recente”
* Emergindo no cenário acadêmico
* Interpretada como Java
* Melhor desempenho
* Facilidade de importação de bibliotecas multimídia

1.3 Ferramentas de autoria

- Utilização de ferramenta puramente visuais => Trabalhos não profissionais

- Utilização somente de linguagens de programação => Dificuldade + Curva de aprendizado demorada

- Meio campo: Ferramentas de autoria

- Parte da interface é desenvolvida através de manipulação direta

- Requisições do usuário programadas através de ações

- **Obs:** Não há uma distinção absoluta entre as ferramentas de autoria e os ambientes de programação.

2. Autoria de títulos

2.1 Autoria de títulos lineares

- Caracterização: Ferramentas de autoria de títulos lineares => Permitem, geralmente, apresentações com avanços manuais (palestras) ou avanços automáticos (temporizados)

- Pode ser conveniente gerar um vídeo das apresentações automáticas => Exemplo: apresentação de artigos publicitários e propagandas

- Exemplo de ferramenta de autoria de títulos lineares: PowerPoint, Impress (BROffice)

2.2 Autoria de títulos hipermídia

- Caracterização: Ferramentas para autoria de títulos hipermídia => Recursos multimídia + Recursos de navegação não-linear

- Controle de Navegação:

* Capacidade de percorrer o título de forma não seqüencial
* Podem tomar a forma de botões, palavras sensíveis ou pontos sensíveis (em imagens)
* Resultado do controle de navegação pode ser um salto para outra página, o surgimento de uma janela temporária ...

- Aspectos requeridos de uma autoria hipermídia:

* Estilo de autoria: linguagem de autoria pode ser visual ou textual
* Modo de operação: modo de autoria ou modo de consulta (somente leitura)
* Intercâmbio de dados: capacidade de importação de todos os padrões de texto, imagem, desenhos, sons e animações

- Hipermídia HTML:

* HTML é um tipo de linguagem de marcação (Mark-up Language)
* Essencialmente é um texto que contém palavras-chaves indicadores de formato tipográfico: <b>, <i> ...
* Não é de propriedade particular
* Deriva da SGML (standard generalized Mark-up language), linguagem padronizada para intercâmbio de dados tipográficos multiplataforma
* Inclui padrão de referência para outros arquivos HTML => Por isso não é considerada como uma linguagem de marcação de documentos lineares
* O arquivo referenciado pode ter qualquer formato multimídia suportado
* Referências podem apontar para arquivos locais, o que permite a utilização de hiperdocumentos locais
* Internet: Utilização da **URL** (universal resource locator) para localizar os arquivos em um servidor distante
* WWW permite a interligação de documentos entre computadores de diferentes ambientes operacionais
* Consulta aos hiperdocumentos feita através de um navegador (browser)

- Padrões para Hipermídia:

* Ainda não existem formatos padronizados de hipermídia genéria
* Padrão bastante utilizado: SMIL, definido pelo W3C como linguagem de programação apropriada para transmissão de conteúdo multimídia de fluxo contínuo

3. Autoria de aplicativos

3.1 Aplicativos com interface multimídia:

- Se distinguem dos demais aplicativos para autoria devido:

* Construção visual da interface de usuário: Criação de controles de usuário através de ferramentas de desenho e formulários
* Mecanismos para inclusão de material multimídia na interface
* Possibilidade de programação textual em linguagem de programação simples

4. Autoria de sítios

4.1 Visão geral:

- Do ponto de vista de ambiente operacional:

* Sítio é visto como uma coleção de arquivos organizados em uma estrutura de pastas
* Contém páginas hipertexto e materiais vinculados à estas páginas

- Possíveis materiais que constituem o sítio:

* Páginas hipertexto
* Material gráfico e de multimídia
* Outros tipos de documentos distribuídos através do sítio
* Arquivos de trabalho usados pelo software servidor do sítio
* Softwares ativos (ActiveX, JavaScript, Ajax ...)

- Onde podem estar localizados

* Na WWW, quando o sítio é acessível através da Internet
* Na **intranet**, quando só for acessível através da rede interna de uma organização

4.2 Sítios estáticos:

- Envolve as seguintes tarefas:

* Autoria de cada página hipertexto
* Autoria do material gráfico e de multimídia incluído nas páginas
* Autoria da estrutura de hiperligações
* Publicação em um servidor
* Testes de visualização através de navegadores conhecidos

- Exemplos de aplicativos de autoria: FrontPage, Dreamweaver, Nvu, FreshHTML

4.3 Sítios dinâmicos:

- O que são:

* São sítios em que uma ou mais páginas contém código ativo, ou seja, programas que são ativados por ações do usuário.
* Principais tipos de código ativo: scripts (pequenos programas com código fonte embutido no HTML) e componentes (arquivos de código binário, invocados quando necessário)
* O código ativo pode ser executado no cliente (navegador é o responsável pela invocação dentro do computador do usuário final) ou no servidor (executados pelo servidor Web, produzindo dinamicamente as páginas HTML que serão enviadas para o cliente)

- Exemplos de códigos ativos:

* VBScript: subconjunto do Visual Basi, reconhecido pelas ferramentas Microsoft
* JavaScript: linguagem de script associada a linguagem Java
* Dynamic HTML (ou DHTML): extensão de HTML baseada em objetos que torna mais flexível a manipulação dos elementos da página através de outras linguagens de script

- Destaque para utilização de códigos PHP e CGI para consultas em BD e elaboração dinâmica de resultados em formulários