

# SOFTWARE EDUCACIONAL: O DIFÍCIL COMEÇO

TEIXEIRA, Adriano Canabarro<sup>1</sup>  
BRANDÃO, Edemilson Jorge Ramos<sup>2</sup>

## Resumo

O presente artigo tem por objetivo discutir questões referentes ao papel do software educacional no processo ensino aprendizagem, bem como apresentar mitos criados em relação ao seu ciclo de vida, desde o desenvolvimento até a sua aplicação pedagógica. A idéia do professor como produtor de software é explorada no sentido de apresentar uma alternativa viável e eficiente na construção de soluções informatizadas para o dia-a-dia em sala de aula.

**Palavras-chave:** Software educacional, formação docente.

## Introdução

Quando nos referimos ao uso do computador em Educação, a idéia de que as novas tecnologias vêm substituir o professor já não possui a força de outrora e, mesmo onde tal mito se faz ainda presente, justifica-se pela falta de informação e de esclarecimento sobre o assunto.

Hoje, toda e qualquer tecnologia que possui potencialidades e características de comunicação e manipulação de informações, parece adequar-se perfeitamente as atividades ligadas à Educação, na medida em que o ato de ensinar/aprender consiste, sobretudo, em uma relação de comunicação por excelência.

Marisa Lucena, ao analisar fatores determinantes da utilização do computador como ferramenta educacional, em Diretrizes para a Capacitação do Professor na Área de Tecnologia Educacional, conclui que o processo de informatização da sociedade brasileira é irreversível e que se a escola também não se informatizar, correrá o risco de não ser mais compreendida pelas novas gerações.

A utilização do computador em Educação só faz sentido na medida em que os professores o conceberem como uma ferramenta de auxílio as suas atividades didático-pedagógicas, como instrumento de planejamento e realização de projetos interdisciplinares, como elemento que motiva e ao mesmo tempo desafia o surgimento de novas práticas pedagógicas, tornando o processo ensino-aprendizagem uma atividade inovadora, dinâmica, participativa e interativa.

---

<sup>1</sup> Professor, MSc., Universidade de Passo Fundo, teixeira@upf.br, <https://vitoria.upf.tche.br/~teixeira>

<sup>2</sup> Professor, Dr., Universidade de Passo Fundo, brandao@upf.br, <https://vitoria.upf.tche.br/~brandao>

## **Produção *ad-hoc* de software educacional**

Dentre as várias formas de se desenvolver ambientes computacionais que favoreçam o processo ensino-aprendizagem, destacamos o software educacional, que segundo Lucena (1992) é todo aquele programa que possa ser usado para algum objetivo educacional, pedagogicamente defensável, por professores e alunos, qualquer que seja a natureza ou finalidade para o qual tenha sido criado.

Entretanto, continua Lucena, para que um software seja utilizado com finalidade educacional ou em atividades curriculares, é necessário que sua qualidade de interface e pertinência pedagógica sejam previamente avaliadas de modo a atender às áreas de aplicação a que se destina e, principalmente, satisfazer as necessidades dos usuários.

Talvez as afirmações de Lucena pudessem hoje ecoar de forma mais eloqüente na medida em que os professores passassem de meros utilizadores de softwares educacionais, ou simples avaliadores, para idealizadores e produtores destes produtos, "desenvolvendo-os" de maneira a atender às necessidades e realidades específicas de seus alunos. Afinal, quem melhor do que o professor, com sua vivência diária com os alunos, conhecedor de suas realidades e anseios, poderia desenvolver um produto eficiente e pedagogicamente correto?

Em outras ocasiões, provavelmente, esta idéia soasse como um projeto insano e incoseqüente. Afinal, afirmam técnicos, e uma grande maioria de professores, que construir software educacional é uma tarefa árdua, penosa e economicamente inviável para profissionais que não são da área da informática. Na verdade, com os avanços da tecnologia de software, a proliferação de linguagens de programação orientada ao objeto, as linguagens de autoria de tipo "arrastar e soltar", ferramentas WYSIWYG, engana-se quem pensa que o professor não pode ser agente ativo na construção de softwares educacionais.

Esta idéia não é o único entrave associado à produção "caseira" de softwares educacionais. Outras pressuposições são feitas como, por exemplo:

- Altos investimentos em hardware;
- Altos custos com a formação de profissionais;
- Necessidade de muito tempo para a produção;
- Softwares de autoria muito caros;
- Linguagens de programação de difícil manipulação para professores.

Esta é a hora de esclarecer estas conclusões prematuras. Não há necessidade de se construir uma estrutura com grandes recursos tecnológicos e laboratórios de última geração, tampouco é preciso formar profissionais de alta competência tecnológica e titulação para a construção de software. Outra idéia que deve ser esquecida é a de que para cada hora de aula são necessárias 1000 horas de programação ou que os softwares de autoria e as linguagens de programação são as únicas ferramentas capazes de realizar tarefas de construção de um software educacional.

É preciso que se desenvolvam mecanismos que torne a produção de softwares nas escolas uma atividade economicamente possível, adequada à formação profissional dos professores e coerente com a infra-estrutura tecnológica da escola e, principalmente passível de realização em tempo hábil.

Foi pensando exatamente nisto que foram realizadas várias experiências no laboratório de informática da FAED, as quais resultaram na proposta de construção de software educacional utilizando tão somente os recursos de multimídia presentes em aplicativos de utilização maciça para ambiente windows, neste caso específico o Microsoft Power Point.

Desta iniciativa vários projetos foram iniciados e devidamente concluídos com êxito por alunos do curso de Pedagogia da faculdade.

Desta forma pode-se afirmar categoricamente, que podemos substituir todas as dificuldades fantasiosas citadas e discutidas acima por uma única, concreta e determinante para a realização ou não de um software educacional por professores: A CRIATIVIDADE.

### **Utilizando power-point na produção de software educacional**

O programa Microsoft Power Point, em sua concepção inicial, foi projetado para a produção de slides e apresentações multimídia, entretanto, devido à própria natureza de seus recursos básicos e da interface gráfica que o hospeda, é possível propor projetos que envolvam outros tipos de aplicações como, por exemplo, *banners*, *pôsters*, *folders*, murais eletrônicos, softwares educacionais, etc.

Os recursos oferecidos pelo programa, muitos dos quais representados na barra de ferramentas através de atalhos na forma de ícones, seguem a mesma lógica e representações simbólicas dos demais aplicativos da família Office da Microsoft. Por serem de fácil manipulação e possuírem ricos módulos de apoio ao usuário, inclusive um sofisticado Assistente de Autoconteúdo que auxilia na construção e formatação dos projetos, os procedimentos para a construção das telas de apresentação do projeto do software educacional serão suprimidos deste artigo.

Uma vez realizadas todas as telas de apresentação utilizando os recursos tradicionais de inserção (novos slides, caixa de texto, *wordart*, figuras, formas, som, vídeo, etc.), de formatação (objetos, fontes, segundo plano, etc.) e de animação (conforme procedimentos descritos no manual do programa), configurar o projeto realizado de modo a ser exibido não mais como uma simples apresentação, mas como um sofisticado software educacional multimídia que utiliza botões de ação, navegação baseada em *hyperlinks*, sons, efeitos de tela, etc.

Durante as etapas de construção das telas (slides) que darão corpo ao projeto de um software educacional alguns comentários são importantes para aqueles que, pela primeira vez, se aventurarão nesta área:

É importante definir *a priori* um detalhado roteiro didático-pedagógico com todas as informações e objetos a serem utilizados, os esquemas de navegação previstos, os níveis de interatividade, as ações previstas para cada botão, etc.

Um bom roteiro inclui também um minucioso *story-board* com esquemas de utilização da tela (*layouts*) e definição de espaço a serem preenchidos pelos objetos na proporção e localização mais ou menos definitiva na tela.

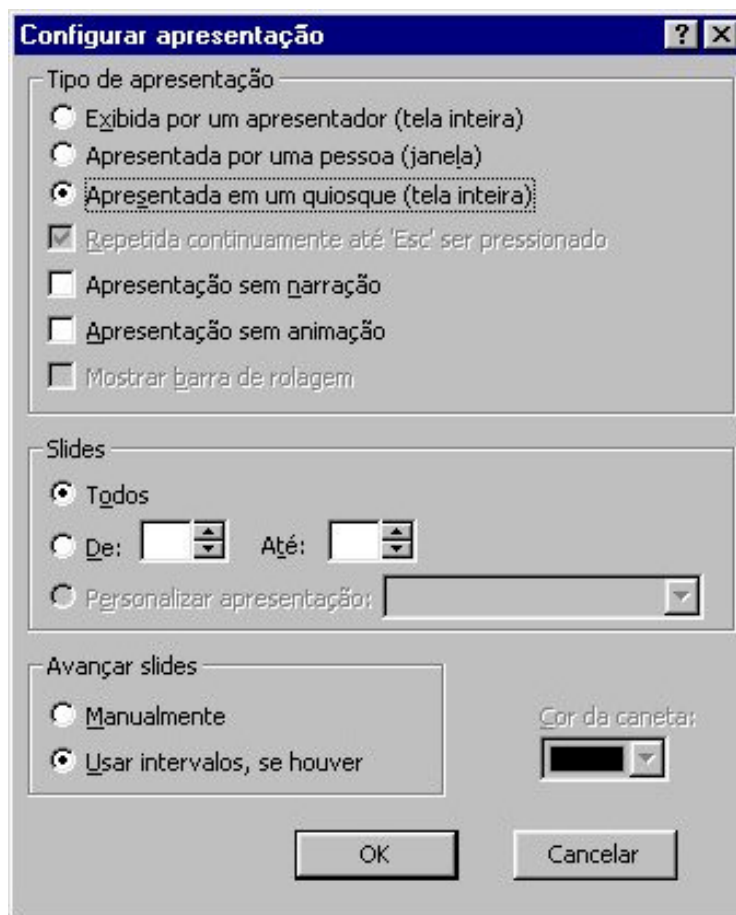
Todos os objetos a serem inseridos, sobretudo as imagens, devem constituir um repertório à parte, selecionado seja através de livrarias de *cliparts*, *download* na Internet, seja através de processos de digitalização.

Vale ressaltar que durante todo o processo de construção das telas deve respeitar critérios de conformidade ao usuário para qual o projeto se destina, através da utilização equilibrada de textos, imagens, sons, vídeos, etc.

Para que o resultado final do trabalho de construção das telas que farão parte do projeto de software educacional seja satisfatório, é sempre bom lembrar que uma avaliação feita por outros especialistas pode evidenciar, inclusive, falhas na composição formal do projeto ou na estética dos elementos que compõem as telas. Neste caso, a utilização frequente de softwares educacionais pode auxiliar nas definições de padrões universalmente adotados na composição das telas, na distribuição das barras de navegação, na formatação do *help*, na construção da janela de confirmação de saída, etc.

### Configurar a apresentação

Clique no menu **Apresentações** e selecione a opção **Configurar apresentação** (*acompanhe as opções da tela abaixo*).



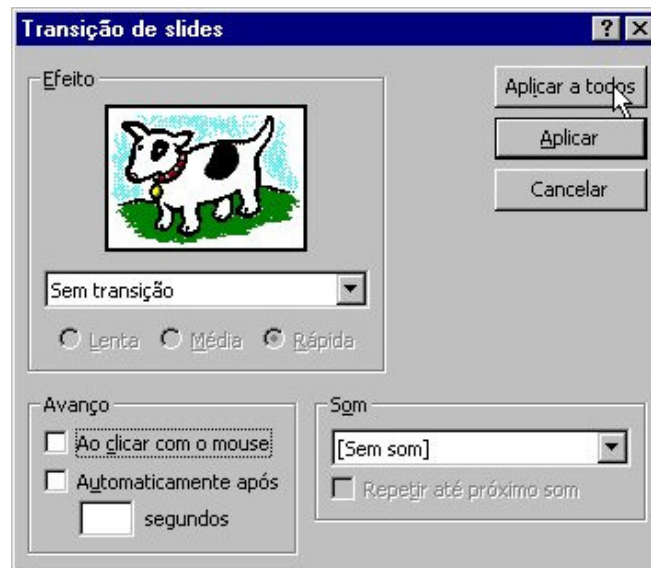
Aqui a opção mais adequada em se tratando de uma produção de software é **Apresentada em um quiosque (tela inteira)**.

Feitas estas alterações, o projeto será apresentado sempre em tela cheia e a única maneira de finalizar o programa será pressionando a tecla 'Esc'.

## Configurando a transição de slides

Ao selecionar a opção **Transição de slides** no menu **Apresentações**, todas as opções devem ser desabilitadas (*conforme tela a seguir*). Feito isto, estas configurações devem ser aplicadas a todos os slides clicando no botão **Aplicar a todos**.

Configurada desta forma a apresentação, a única maneira de finalizar o programa será através da tecla 'Esc'.



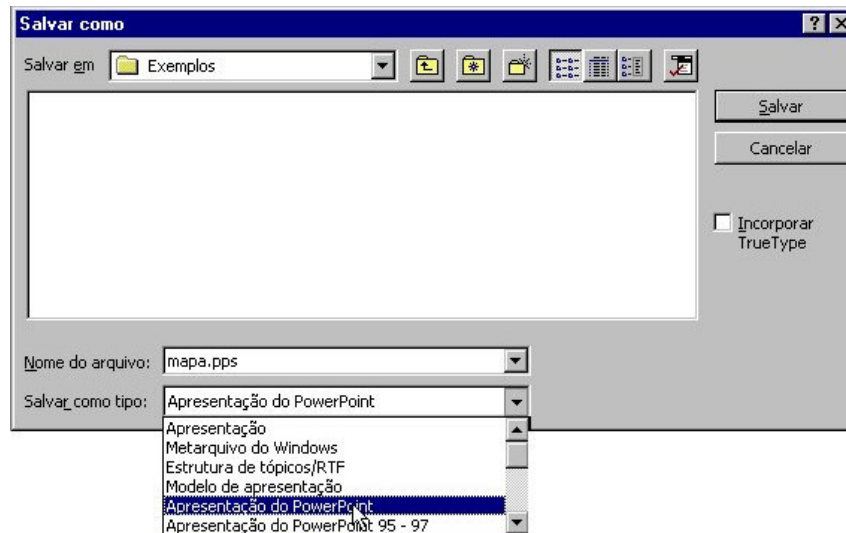
As únicas exceções de transição de slide sem avanço com o clicar do mouse e sem avanço automático serão para as telas de abertura e telas que exigem obrigatoriamente transição temporizada. Nesses casos, o procedimento a ser utilizado começa por selecionar os slides não *standard* e determinar um intervalo de tempo na opção de avanço automático suficiente para a leitura das informações apresentadas ou para uma eventual narração ou efeito sonoro.

Selecione novamente a opção **Transição de slides** no menu **Apresentações**. Nela, a única opção que deve ser habilitada é a que se refere ao avanço de slides **automaticamente após... segundos**. No exemplo abaixo a transição da tela de abertura para a próxima tela ocorrerá Automaticamente após 7 segundos.

Essa configuração de avanço só deverá ser aplicada nas telas que fogem ao padrão de exibição, ou seja, que não necessitam de interatividade por parte do usuário para avançar. Assim sendo, utilizar a opção **Aplicar** e não **Aplicar a todos**.



## Salvando o projeto



Uma vez concluído o é possível finalizá-lo de modo que o arquivo gerado rode em abrir totalmente o programa PowerPoint. Na verdade, uma vez salvo o projeto como a extensão.PPS e não.PPT o arquivo é executado lançando somente a aplicação View do PowerPoint o que, para o usuário final, pode dar a entender que se trata de um verdadeiro módulo *run-time* (que roda independente da aplicação geradora).

Uma vez que o projeto foi salvo no formato.PPS, o arquivo gerado torna-se independente do programa PowerPoint. Toda navegação, transição, ou efeitos sobre objetos devem ser produzidos de forma automática ou através de cliques em objetos definidos como botões de ação.

## Considerações finais

Embora se reconheça que a produção de software educacional de qualidade deve ser uma ação multidisciplinar, na medida em que muitas competências estão envolvidas neste processo, foi possível observar que, devidamente orientados e motivados, este pode ser apropriado por professores e utilizado com êxito na construção de pequenas

aplicações destinadas a situações/problemas corriqueiros em suas salas de aula, não sendo necessários grandes investimentos de tempo ou recursos financeiros.

### **Bibliografia**

LUCENA, M. *A Gente é uma Pesquisa: Desenvolvimento Cooperativo da Escrita Apoiado pelo Computador*; Dissertação de Mestrado; Departamento de Educação, PUC-Rio; Rio de Janeiro: 1992.