

Relatório Final de Bolsa de IC

APRIMORAMENTO DE UMA APLICAÇÃO DISTRIBUÍDA PARA COMPARTILHAMENTO DE APRESENTAÇÕES POWERPOINT EM COMPUTADORES PORTÁTEIS

Aluno: Gabriel de Oliveira Barbosa Leite/ Artur Otávio Fernandes Maligo
Orientador: Markus Endler

Introdução

O iPH (Interactive Presenter for Handhelds) foi uma aplicação distribuída, desenvolvida na PUC-Rio, para compartilhamento e co-edição de transparências eletrônicas em dispositivos portáteis do tipo Notebook/TabletPC e PDA, que dispunham de uma interface de rede sem fio IEEE 802.11. Houve suporte para contribuições do tipo "ink" e texto, que podiam ser submetidas ao dispositivo mestre e compartilhadas pelos demais participantes.

O iPH foi útil para reuniões ou apresentações baseadas em transparências eletrônicas do tipo PowerPoint, onde todos os participantes podiam criar e compartilhar contribuições (desenhos, anotações, textos) em cima de algumas transparências.

O iPH foi inspirado no aplicativo Classroom Presenter, que por sua vez foi baseado no framework para colaboração ConferenceXP. No entanto, como o ConferenceXP não executa no .Net Compact Framework, desenvolveu-se na LAC/PUC-Rio uma versão modificada, o Compact Conference XP (CCXP), que funcionou como middleware para comunicação e sincronização multi-ponto, e que executou tanto em Windows XP como em Windows Mobile.

Objetivos

O projeto teve como objetivo aprimorar, complementar e otimizar o iPH, que atualmente se encontrava na forma de um protótipo funcional (tanto a versão Tablet como a versão Handheld). Além disso, o aplicativo era sensível ao contexto, através do seu acoplamento com os serviços de contexto e localização do middleware MoCA, também desenvolvido na PUC-Rio. Esse acoplamento podia ainda ser expandido através da criação de novas interfaces gráficas e de novas funções.

Metodologia

Existiam basicamente dois métodos de programação: a programação procedimental e a programação orientada a objetos. O iPH foi escrito usando C#, que é uma linguagem de programação orientada a objetos. Primeiramente, então, para que se pudesse entender o aplicativo, foi estudada a programação orientada a objetos.

Após o estudo deste tipo de programação, foi estudada especificamente a linguagem C#, na qual o programa foi escrito. O iPH foi desenvolvido na plataforma do programa Visual Studio, então foi preciso estudar esta plataforma também.

O iPH é um programa extenso, sendo que sua solução consiste de vários projetos. Assim, sua estrutura interna, contendo suas classes, métodos e hierarquias, foi estudada, usando-se o próprio programa Visual Studio como ferramenta para entender o código.

Além do estudo do programa, foram realizados também testes de usabilidade do iPH em sala de aula. Essas aulas especiais usaram slides de apresentações feitos especificamente para

serem usados com o iPH, de forma que os alunos pudessem expor suas opiniões e enviá-las ao professor. Assim, vários aspectos do programa puderam ser testados na prática.

Os principais aprimoramentos implementados

- A função SaveDeckSet() foi implementada em "InteractivePresentationForm.cs", sendo que ela é análoga a SaveDeck(), do "Deck.cs", só que salva tanto o MainDeck quanto o ContributionDeck.
- Implementação da função LoadDeckSet(), em "InteractivePresentationForm.cs". A função abre os dois decks e os acrescenta na apresentação.
- Implementação da função AppendDeckAtContributionDeck(), em "InteractivePresentationForm.cs".
- Modificação da função SaveDeckSet(), acrescentando duas constantes bool. Elas são usadas para testar se os Decks existem, pois se não existirem não precisam ser salvos.
- As funções LoadDeckSet e SaveDeckSet foram alteradas quanto à forma de testar se o Deck existe ou não, em particular, inclusão dos métodos save e load do ContributionsManager.
- Acrescentou-se à função LoadDeckSet um teste para saber se já havia alguma apresentação aberta. Se houver, primeiro ela é fechada, chamando-se a função ClearPresentation, e depois passa-se para o restante da função LoadDeckSet.
- Na MainForm, incluiu-se um botão Save e alterou-se o antigo para SaveAs.
- Alterou-se a função SaveDeckSet, pois basta criar um stream e um data para salvar os 3 objetos, pois a cada vez realoca-se uma nova memória para ele.
- Modificou-se o teste inicial de BroadcastDeck, para não enviar o deck caso não tenha slides, ou seja nulo, ou o participante esteja desconectado.
- Criou-se o método ClearSlideContributions(Guid slideGuid), na InteractivePresentationForm.cs e um método de mesmo nome dentro do ContributionsManager.cs. Os métodos servem para apagar todas as contribuições de um determinado slide, cuja Guid é passada como parâmetro.

Conclusões

Uma modificação importante feita no iPH foi a implementação das funções de Save e Load (funções de persistência). Essas funções foram implementadas no iPH tanto na sua versão para Windows XP quanto para Windows Mobile.

Outra mudança feita foi uma alteração na interface gráfica do programa, que consiste basicamente na tela principal e seus menus. Essa mudança na interface foi feita visando melhor entendimento, de forma que um usuário do programa não tenha dificuldades em usar as ferramentas disponíveis pelo iPH.

Os testes que foram feitos em sala de aula serviram para diagnosticar alguns problemas existentes com o programa e trouxeram também várias idéias de como ele poderia ser aprimorado. O estudo do iPH e seu código também geraram algumas idéias de futuras implementações.

Referências

1 - DEITEL, H. M. and DEITEL, P. J. **C# How to Program**. 1.ed. Upper Saddle River, NJ: Prentice Hall, 2001. 1568p.

2 - DEITEL, H. M. and DEITEL, P. J. **Java How to Program**. 4.ed. Upper Saddle River, NJ: Prentice Hall, 2001. 1546p.