Estudo de Usabilidade do Microsoft Visual C++

ANTONIO ESCAÑO SCURI
FLÁVIO SZENBERG
CLARISSE SIECKENIUS DE SOUZA

Departamento de Informática, PUC-Rio
Rua Marquês de São Vicente, 255, 22453-900, Rio de Janeiro, RJ, Brazil
{scuri, szenberg, clarisse}@inf.puc-rio.br

PUC-RioInf.MCC44/98 November, 1998

Abstract: This work tries to identify the major usability problems related with Microsoft Visual C++, regarding to begginers, intermediate and advanced users. We organize a test that follows a script, common to all user classes. The test was recorded with Microsoft Camcorder in Windows 95, but some important user actions were manual registered. At the end of the test, every user answered a questionnare, to show us the aplications facilities and dificulties. The results are presented in tables and in a descriptive conclusion.

Keywords: user interace, usability test, Visual C++, subjects, scenes, observation methods

Resumo: Este trabalho procura levantar as dificuldades que os programadores, iniciantes, intermediários e avançados, encontram quando usam uma ferramenta de desenvolvimento sofisticada, como é o caso do Microsoft Visual C++. Foram realizados testes onde era apresentado o mesmo roteiro para cada um dos sujeitos, independente de sua classificação. Estes testes foram analisados com ajuda de um gravador de telas, chamado Camcorder, e cada "manifestação" importante observadas dos sujetos eram registradas. No final de cada teste, cada sujeito respondeu um questionário, com objetivo de conhecermos a facilidade do uso da ferramenta. Os resultados são apresentados em formas de tabelas, seguido de uma conclusão descritiva.

Palavras-chave: interface, teste de usabilidade, Visual C++, sujeitos, cenários e métodos de observações

1 Introdução

O **Microsoft Visual C++** (VC) é um ambiente integrado de desenvolvimento (chamado de IDE) que permite o gerenciamento de projetos de programas e bibliotecas, incluindo seus respectivos makefiles. Além disso, o VC permite ainda disparar a compilação e linquedição dos projetos, como também possui ferramentas de depuração integradas ao ambiente. Atualmente o ambiente IDE do **Visual C++** se chama **Developer Studio** e é compartilhado pelo **Visual C** e **Visual J++**, e é configurável para usar o **Visual Teste** e o **Visual Source Safe**, assim como outros utilitários, por exemplo o **Bounds Checker**.

As tarefas mais freqüentes realizadas pelos usuário estão relacionadas com o processo de desenvolvimento do seu projeto, tais como compilação, linquedição e depuração. Nesses pontos o VC oferece recursos bastante eficazes e funcionais.

Mas quando se trata da criação e manutenção dos próprios projetos, existem diversas deficiências identificadas por vários usuários do TeCGraf.

Gostaríamos de mostrar como evoluiu e como funciona esta parte do VC, inclusive ao longo das versões 2, 4 e 5, assim como compará-lo, neste quesito, a outros IDEs conhecidos, tais como o Borland C++ e o Watcom C++.

Em seguida, realizaremos uma experiência com grupos de usuários propondo que estes criem e modifiquem projetos seguindo determinadas tarefas por nós especificadas.

2 Descrição da Experiência

2.1 Natureza

A natureza da experiência é exploratória. Gostaríamos de verificar quais são os caminhos encontrados pelos usuários para realizar as tarefas pré-determinadas.

2.2 Dados a priori

Realizando um levantamento dos mecanismos disponíveis para manipular projetos no VC identificamos os seguintes:

a) A criação de um novo projeto é feita apenas de uma maneira, através da seqüência de menus **File/New**. O usuário precisa então fornecer o tipo, o diretório e o nome do projeto.

   Um dos problemas mais comuns que ocorre neste diálogo é que o VC supõe que você quer criar um diretório abaixo do diretório especificado com o nome do projeto. Assim, como várias pessoas colocam o diretório primeiro e depois o nome do projeto, o VC acrescenta o nome do projeto ao diretório especificado e se o usuário perceber isso ele deverá voltar ao campo de diretório para corrigi-lo.

   Quanto ao tipo do projeto é preciso conhecer a priori o tipo desejado. Não é intuitivo para o usuário que não está acostumado com os tipos disponíveis, escolher o tipo correto.

Um fato interessante é que se o diretório não existe o VC vai automaticamente criá-lo, portanto se o usuário errar no campo do diretório terá que remove-lo posteriormente.

b) No VC5 a inclusão de novos arquivos no projeto é feita pela seqüência de menus **Project/Add to Project/Files**, ou abrindo-se o arquivo no editor e chamando-se o menu de contexto (botão direito do mouse) do editor e executando a seqüência de menus **Insert File Into Project/<nome do projeto>**.

   Aqui o problema mais comum é o fato de que para acrescentar múltiplos arquivos que estão em diretórios diferentes é preciso chamar o diálogo, uma vez para cada diretório.

   Um outro problema deste diálogo é a seleção de filtros, que muitas vezes não é a esperada pelo usuário fazendo com que este se confunda ao procurar pelos arquivos de interesse.

c) No VC5 a modificação da configuração do projeto é feita pela seqüência de menus **Project/Settings**.

   Este é o diálogo mais complexo do VC. Possui inúmeros Tabs e dentro de um mesmo tab ainda existem várias sub-tabs, chamados de categorias, que são visualizados modificando-se uma caixa de seleção.

   Além do mais, as modificações na configuração podem ser feitas para as versões de release, debug ou ambas, ou até mesmo em outras versões.

   É vital, para a geração correta do projeto, que os arquivos de inclusão e de bibliotecas necessários sejam encontrados. A localização dos arquivos pode ser feita de duas maneiras.

   Na seqüência de menus **Tools/Options** e ativando o Tab de **Directories**, pode-se colocar diretórios tanto de inclusão, como de localização de bibliotecas, mas estas configurações irão afetar a geração de todos os projetos daquele usuário. No Windows 95 isso é mais crítico, pois se os Perfis de Usuário não estiverem ativos, todos os usuários daquele computador serão afetados. Mas a utilização dessa opção é mais interessante quando se trafega com o projeto de um ambiente de trabalho para outro, onde mudam exatamente esses diretórios.

   No diálogo da configuração de cada projeto, já mencionado, pode-se especificar diretórios para localização de arquivos de inclusão no Tab de **C/C++** na categoria de **Preprocessor** em **Adicional Include Directories**.

   No caso das bibliotecas pode-se acrescentá-las a própria lista de arquivos do projeto, ficando-se sujeito ao problema de que cada biblioteca em geral estará em um diretório diferente e será necessário chamar o diálogo de inclusão várias vezes. A outra forma é incluir os nomes de cada biblioteca no diálogo de configuração do projeto no tab de **Link** na categoria de **Input**, especificando os diretórios em **Adicional Library Path** neste mesmo tab.

OBS: Estamos chamando Settings de Configuração, e Configuration de Versões.

É preciso mencionar que nos pareceu ser muito difícil formular uma hipótese concreta, sem ser muito específico, correndo o risco de tornar o teste tendencioso ou irrelevante.

2.3 Design da Experiência

2.3.1 Metas

Explicitar os prós e contras dos mecanismos de criação e manutenção de projetos e avaliar sua evolução ao longo das diversas versões. Neste caso, também queremos saber o quanto que cada usuário pode aproveitar de sua experiência com as versões anteriores.

2.3.2 Seleção de Sujeitos

Todos os usuários devem possuir experiência no uso de compiladores. A seleção de sujeitos para a experiência será feita dividindo-se em 3 grupos:

- Usuários "avançados" – já usam o VC.
- Usuários "intermediários" – usam IDEs que não o VC.
- Usuários "iniciantes" – não usam IDEs para desenvolvimento.

Para que os dados levantados tenham validade estatística, vamos procurar pelo menos 3 usuários para cada um dos grupos.

2.3.3 Cenários

Para as experiências, serão passados aos sujeitos as seguintes tarefas:

- Criação de um novo projeto
- Inclusão de arquivos no projeto
- Alteração da configuração do projeto

Queremos observar a utilização dos diálogos para devidos fins e também usos de manipulação direta, como no caso de poder incluir arquivos em projetos quando possível.

2.3.4 Métodos de Observação e Registro de Dados

Antes de iniciarmos o teste, o usuário será informado que "o teste é sobre o programa e não sobre sua habilidade em usá-lo". O teste terá um tempo finito a ser determinado pelos avaliadores.

Será então apresentado por escrito um roteiro a ser percorrido pelo sujeito, que refletirá os cenários já mencionados.

As observações serão feitas utilizando-se um programa de gravação de telas chamado Microsoft Camcorder, mas também haverá a presença de ambos os avaliadores, embora sem a sua interferência.

Após os testes os sujeitos são submetidos a um questionário para obtermos um grau de usabilidade para o VC com propósito de compará-lo nos mesmos termos com o teste de outros usuários.

2.3.5 Roteiro

O roteiro a seguir será cumprido em um tempo máximo de **15 minutos**, caso o tempo se exceda o teste será interrompido. Contudo não deve-se haver uma preocupação com a completude do teste neste tempo, cada usuário deve realizá-lo no tempo que lhe for adequado. Caso não se saiba como completar uma determinada tarefa, esta pode permanecer incompleta.

O diretório **y:\teste** já existe e contém o subdiretório **y:\teste\src** que contém os arquivos fontes necessários ao projeto. Os diretórios que contém as bibliotecas e as mesmas também já existem e serão fornecidos ao longo do roteiro.

1) Criar um projeto com nome TESTE do tipo **Win32 Application**. Os arquivos que definem o projeto devem estar no diretório **y:\teste\mak**.
2) Incluir os seguintes arquivos no projeto:
   y:\teste\src\teste.c
   y:\teste\src\util.c
3) Este projeto, além das bibliotecas de sistema necessita das seguintes bibliotecas: IUP e CD. Os arquivos binários das bibliotecas, a serem incluídos no projeto, estão em:
   w:\iup\lib\vc4\iup.lib
   w:\cd\lib\vc4\cd.lib
4) Modifique a configuração (settings) para os arquivos de inclusão das bibliotecas CD e IUP serem encontrados. Esses arquivos ficam localizados em:
   w:\iup\inc
   w:\cd\inc
5) Modifique a configuração (settings) do projeto para que os arquivos de objeto (intermediate files) sejam gerados no diretório **c:\teste\obj** e os arquivos executáveis (output files) no diretório **c:\teste\bin**.
6) A IUP precisa ainda que você modifique a configuração (settings) do projeto de ambas as versões **Debug** e **Release** para incluir a biblioteca de sistema COMCTL32.LIB.
7) Gere e Execute a versão de **Debug**.
8) Modifique a configuração do projeto para incluir a definição do símbolo "**NO_MENU**" apenas para a versão de **Release**.
9) Gere e Execute a versão de **Release**.
10) Por fim, abra no editor do VC o arquivo **y:\teste\src\teste.c**.

2.3.6 Questionário

O questionário será aplicado verbalmente, para não dar mais trabalho ao usuário, já que este teste é apenas um experimento informal. Além do mais isso deve acelerar o final do teste.

1. Qual sua experiência em compilação e linquedição de aplicações em C?
2. É usuário de outras IDEs? Quais?
3. Encontrou dificuldades em criar o projeto? Quais?
4. Aproveitou experiências de outras IDEs para criar o projeto? Em que pontos?
5. Encontrou dificuldades em incluir arquivos ao projeto? Quais?
6. Aproveitou conhecimentos de outras IDEs para incluir arquivos? Em que pontos?
7. Encontrou dificuldade em alterar configurações do projeto? Quais?
8. Qual é a sua opinião sobre os diálogos surgidos ao longo da criação, inserção de arquivos e alteração da configuração?
9. Gostaria de mencionar algum outro ponto problemático da interface do VC?

2.3.7 Dados Coletados

Foram recrutadas 13 pessoas, dentre as quais 5 classificamos como Iniciantes, 4 como Intermediários e 4 como Avançados.

Todos os testes foram realizados na mesma máquina, esta estando em as condições normais de uso. Não foram reportados problemas nesse sentido.

3 Interpretação dos Dados

Nós analisamos os dados coletados usando parâmetros gerais e parâmetros específicos do teste.

3.1 Parâmetros Gerais

Comportamento e Atitude: **Calmo, Concentrado, Interferência, Apressado, Nervoso**
Consulta ao Help: **Nenhuma, Poucas, Muita**
Tempo: **Suficiente, Insuficiente**
Percentual da Tarefa Completada: **%**
Uso do Computador: **Menu Bar, Tool Bars, Context Menu, Teclas de Atalho**

3.2 Parâmetros Específicos

Localização da Criação de Projeto: **Difícil, Razoável, Direta**
Dados da Criação Corretos: **Nome, Tipo, Diretório**
Localização da Inclusão de Arquivos: **Difícil, Razoável, Direta**
Adição da Bibliotecas como Arquivos: **Sim, Não**
Uso do Diálogo de Inclusão: **Uso de Filtros, *.lib, *.***
Localização dos Settings: **Difícil, Razoável, Direta**
Preocupação com Release, Debug e All Configurations: **Sim, Não, Eventual**
Localização dos Intermediate e Output Files Dir: **Difícil, Razoável, Direta**
Localização do Adic. Include Path: **Difícil, Razoável, Direta**
Localização do Object/Library Modules: **Difícil, Razoável, Direta**
Geração das versões de Release e Debug: **Difícil, Razoável, Direta**

3.3 Interpretação do Dados

Comportamento e Atitude	**Calmo**	**Concentrado**	**Interferência**	**Apressado**	**Nervoso**
1 - Avançado	x	x			
2 - Intermediário	x	x			
3 - Avançado	x			x	
4 - Iniciante	iniciou	x	x		terminou
5 - Iniciante	x	x			
6 - Iniciante	iniciou	x	x		terminou
7 - Intermediário	x	x		x	
8 - Intermediário	x	x			
9 - Intermediário	x	x			
10 - Iniciante	x	x			
11 - Avançado	x	x			
12 - Avançado				x	x
13 - Iniciante	iniciou	x	x		terminou

Consulta ao Help	**Nenhuma**	**Poucas**	**Muita**
1	x		
2	x		
3	x		
4	x		
5	x		
6	x		
7	x		
8	x		
9	x		
10	x		
11	x		
12	x		
13	x		
Percentual Total:	100		

Tempo	**Suficiente**	**Insuficiente**
1		x
2		x
3	x	
4		x
5		x
6		x
7	x	
8		x
9	x	
10		x
11	x	
12		x
13		x
Percentual Total:	30	70

Percentual da Tarefa Completada	**%**
1	90
2	70
3	100
4	30
5	40
6	40
7	100
8	40
9	100
10	40
11	100
12	50
13	20
Percentual Médio:	63

Uso do Computador	**Menu Bar**	**Tool Bars**	**Context Menu**	**Teclas de Atalho**
1	x	x		
2	x	x		
3	x	x	x	x
4	x			
5	x			
6	x			
7	x	x		
8	x			
9	x			
10	x			
11	x	x	x	
12	x			
13	x			
Percentual Total:	100	38	15	8

Localização da Criação de Projeto	**Difícil**	**Razoável**	**Direta**
1	x		
2		x	
3			x
4	x		
5		x	
6	x		
7		x	
8		x	
9	x		
10		x	
11			x
12			x
13			x
Percentual Total:	31	38	31

Dados da Criação Corretos	**Nome**	**Tipo**	**Diretório**
1	x	x	
2	x	x	
3	x	x	x
4	x	x	
5	x	x	
6	x	x	
7	x	x	
8	x	x	
9	x	x	
10	x	x	
11	x	x	
12	x	x	
13	x	x	
Percentual Total:	100	100	8

Localização da Inclusão de Arquivos	**Difícil**	**Razoável**	**Direta**
1			x
2		x	
3			x
4	x		
5		x	
6		x	
7			x
8			x
9			x
10	x		
11			x
12		x	
13		x	
Percentual Total:	15	38	47

Adição da Bibliotecas como Arquivos	**Sim**	**Não**
1	x	
2		x
3		x
4	x	
5	x	
6	x	
7	x	
8	x	
9	x	
10	x	
11	x	
12	x	
13		
Percentual Total:	83	17

Uso do Diálogo de Inclusão	**Uso de Filtros**	***.lib, *.***
1		x
2	x	
3	x	
4		x
5		x
6		x
7		x
8		x
9	x	
10		x
11	x	
12		x
13		
Percentual Total:	33	67

Localização dos Settings	**Difícil**	**Razoável**	**Direta**
1			x
2	x		
3			x
4	x		
5	x		
6			x
7			x
8		x	
9			x
10	x		
11			x
12			x
13			
Percentual Total:	33	8	59

Preocupação com Release, Debug e All Configurations	**Sim**	**Não**	**Eventual**
1			x
2		x	
3	x		
4		x	
5		x	
6	x		
7			x
8		x	
9		x	
10		x	
11	x		
12	x		
13			
Percentual Total:	33	50	17

Localização dos Intermediate e Output Files Dir	**Difícil**	**Razoável**	**Direta**
1	x		
2		x	
3			x
4			
5	x		
6	x		
7			x
8	x		
9		x	
10			
11		x	
12	x		
13			
Percentual Total:	50	30	20

Localização do Adic. Include Path	**Difícil**	**Razoável**	**Direta**
1		x	
2	x		
3			x
4			
5	x		
6	x		
7		x	
8	x		
9		x	
10			
11		x	
12	x		
13			
Percentual Total:	50	40	10

Localização do Object/Library Modules	**Difícil**	**Razoável**	**Direta**
1		x	
2		x	
3			x
4			
5			
6			
7			x
8		x	
9		x	
10			
11			x
12			x
13			
Percentual Total:		50	50

Geração das versões de Release, Debug	**Difícil**	**Razoável**	**Direta**
1			x
2		x	
3			x
4			
5			
6			
7			x
8			
9			x
10			
11			x
12	x		
13			
Percentual Total:	14	14	72

3.4 Análise da Interpretação

O comportamento dos usuários durante o texto foi tranqüilo, embora quando não conseguiam realizar alguma tarefa ficavam constrangidos. Mesmo não conseguindo fazer, praticamente nenhum usuário consultou o Help.

Para usuários Intermediários e Avançados, o tempo para realizar o teste foi suficiente e em geral completaram 90% do teste. Os usuário iniciantes, tiveram muita dificuldade e na maioria das vezes não completou o teste no tempo determinado, além de precisarem de algumas dicas em pontos cruciais para que o teste pudesse prosseguir.

A grande maioria dos usuários utilizou os menus, poucos usuários usaram as Tool Bars e alguns usuários intermediários e avançados usaram os menus de contexto do projeto. Somente um usuário usou teclas de atalho durante o teste.

Poucos usuários se confundiram na criação do projeto. Os que o fizeram foi porque deduziram erradamente que esta opção estaria no menu Project e não no menu File. Isso mostra que o excesso de menus, mesmo que desabilitados gera confusões por parte do usuário.

No diálogo de criação de projeto o problema mais comum é devido a uma sofisticação do diálogo que acrescenta o nome do projeto ao nome do diretório do projeto, quando o usuário entra com o nome do projeto. Também neste mesmo ponto, o botão de Browse permitia selecionar apenas um diretório existente e não um diretório novo, isso gerou problemas para alguns usuários que não queria digitar o nome do diretório e sim selecionar o diretório base e no mesmo diálogo acrescentar o path "mak". Uma curiosidade sobre a criação do projeto foi que uns poucos usuários se preocuparam em colocar o nome do projeto em maiúsculo, pois assim estava no roteiro, embora isso fosse irrelevante.

O primeiro ponto que gerou dúvidas importantes foi a inclusão de arquivos no projeto. Vários usuários iniciantes não conseguiam descobrir que a mesma era feita em Project / Add to Project / Files. Alguns tentavam selecionar File / New, outros selecionavam Project / Add to Project / New, embora estivesse escrito no roteiro que esses arquivos já existiam. Alguns usuários não perceberam que poderiam inserir os dois arquivos .c simultaneamente no projeto, simplesmente chamavam o diálogo repetidas vezes para cada arquivo. Um usuário tentou realizar drag&drop, mas percebeu que o arquivo, ao invés de ir para o projeto, ia para o editor.

A grande maioria dos usuários inseriu as bibliotecas IUP e CD no projeto como se fossem arquivos fontes. Isso é plenamente aceitável e é muito provável que tenha sido induzido pelo roteiro. Mas no diálogo de inserção de arquivos o filtro default faz com que os arquivos de bibliotecas fiquem escondidos. Neste caso alguns usuários usavam o controle de filtros para colocar o filtro certo, outros simplesmente digitavam "*.lib" para que os arquivos de bibliotecas aparecessem.

Uma vez encontrado o diálogo de inclusão de arquivos no projeto, o diálogo de Settings é facilmente encontrado, mas esse é o diálogo mais complicado do teste, talvez até de todo o VC.

O primeiro ponto problemático neste diálogo é que você pode modificar os Settings da versão de Release, da Debug ou de ambas. Alguns usuários notavam isso, outros notavam algumas vezes e outras ignoravam, e outros nem notavam.

Como o primeiro tab deste diálogo contém a localização dos arquivos de intermediários e de saída, esses são facilmente localizáveis. Mas quando se trata da localização do caminho para arquivos de inclusão a coisa complica. Estes ficam no Tab de **C/C++**, e dentro deste Tab deve-se selecionar a categoria de Pre-Processor, para enfim chegar ao lugar certo. O problema que acontece é devido a seleção da categoria ser feita por uma Combo Box que está distribuída no Tab como um controle qualquer, parecendo ser uma outra opção daquele Tab e não uma categoria (É claro que o label onde diz category não chama a atenção o suficiente). A localização do campo para bibliotecas adicionais segue o mesmo problema.

Chegando até esse ponto, gerar a versão de Debug é simples, já que é a configuração ativa. Mudar a configuração ativa para a Release é que gerou dúvidas para alguns usuários, chegando ao extremo de não conseguirem gerar a versão de Release.

4 Conclusão do Estudo

Durante os primeiros testes tivemos que fazer pequenas modificações no roteiro para deixar mais claro as tarefas a serem realizadas. Uma dessas modificações foi quanto a nomenclatura, nós passamos a incluir, entre parênteses, o termo em inglês usado pelo VC. Isso simplificou muito o processo, deixando os usuários mais a vontade.

As perguntas do questionário foram feitas bastante informalmente, deixando o usuário livre para comentar o que ele quisesse.

A idéia de gerar uma aplicação com interface, que mostrasse um resultado, foi de dar para o usuário um pouco da sensação de que ele fez alguma, recompensando-o.

O Microsoft Camcorder se mostrou muito efetivo para registro dos testes. Os arquivos gerados foram da ordem de 5 Mb, para os 15 minutos de cada teste, usando-se uma resolução de tela de 800x600 com 256 cores. Sua única limitação foi o fato de não poder ser executado no Windows NT, funcionando apenas no Windows 95.

Infelizmente, pelo tempo disponível de cada usuário para realizar o teste, não pudemos realizar um dos cenários propostos que mostraria a problemática relacionada com a evolução do VC. Neste cenário os usuários iriam realizar as mesmas tarefas propostas, nos VCs 2.0, 4.0 e 5.0. Existe uma grande mudança da versão 2 para a versão 4, e algumas mudanças nos menus da versão 4 para a versão 5, que seriam muito interessantes serem analisadas no contexto de aproveitamento da experiência do usuário.