

UNIVERSIDADE LUTERANA DO BRASIL

ULBRA – *CAMPUS* GUAÍBA

CURSO DE SISTEMAS DE INFORMAÇÃO



PROPOSTA DE TCC
TRABALHO DE CONCLUSÃO DE CURSO

ALEXANDRE COUTINHO EVANGELISTA

LUIZ GUSTAVO GALVES MAHLMANN
Professor Orientador

Guaíba, março de 2008.

DADOS DE IDENTIFICAÇÃO

1. DADOS DO ALUNO

Nome: Alexandre Coutinho Evangelista

Endereço: Rua Cel. Evaristo Lopes dos Santos, 52 Lot. do Engenho – Guaíba / RS

E-mail: alex.evangelista@gmail.com

Fones: 51 – 99576149 / 34915574

2. DADOS DO PROFESSOR ORIENTADOR

Nome: Luiz Gustavo Galves Mahlmann

E-mail: mahlmann@gmail.com

SUMÁRIO

1	DEFINIÇÃO DO TEMA	4
1.1	Título do TCC	4
1.2	Tema	4
1.3	Delimitação do Tema	4
2	MOTIVAÇÃO	4
3	HIPÓTESES DE SOLUÇÃO	5
4	OBJETIVOS	6
5	FUNDAMENTAÇÃO TEÓRICA	6
6	SOLUÇÃO PROPOSTA	10
7	CRONOGRAMA	11
8	REFERÊNCIAS BIBLIOGRÁFICAS	12

1 DEFINIÇÃO DO TEMA

1.1 Título do TCC

Verificar a portabilidade de aplicações .NET quando executadas em ambientes diversos (Windows, Web e Mobile).

1.2 Tema

O tema proposto envolve a verificação da portabilidade que uma aplicação pode ter quando for executada em mais de um ambiente (Windows, Web e Mobile), utilizando a mesma plataforma de desenvolvimento, abordando as respectivas similaridades e diferenças entre os mesmos.

1.3 Delimitação do Tema

O escopo do trabalho envolve a verificação da portabilidade de um único sistema, utilizando mais de um ambiente para execução, suas características, similaridades, diferenças, compatibilidades e incompatibilidades.

No ambiente Desktop, será utilizada a plataforma Windows, assim como no Web, já no Móvel, será utilizada a plataforma Windows Mobile [YAN07].

Será feito um comparativo a respeito da evolução dos frameworks [THA01] existentes na plataforma .NET [THA01], analisando as respectivas características, atualizações e diferenças até a versão 3.5, disponível no ambiente de desenvolvimento Microsoft Visual Studio 2008 [TRO07].

Será desenvolvida uma aplicação para demonstrar os estudos realizados. Tanto a aplicação quanto os estudos serão baseados na linguagem de programação C#.NET [TRO07].

2 MOTIVAÇÃO

Com a constante evolução tecnológica em que vivemos, onde, semanalmente, novidades não param de surgir, novos sistemas vêm facilitando nosso dia-a-dia, por isso precisamos ficar sempre atentos ao que está acontecendo e o que está para acontecer.

Existem aplicações em diversas áreas de atuação muito bem consolidadas no mercado, os dispositivos móveis já não são mais uma novidade e sim uma realidade que vêm tomando uma dimensão cada vez maior. Sua infinidade de recursos faz com que aplicações possam ser usadas em qualquer local, sem que seja necessário o usuário estar em casa ou no seu trabalho.

Minha maior motivação neste trabalho é poder unir os pontos fortes que cada ambiente proporciona em uma única aplicação. Desse modo, será possível desenvolver uma aplicação e adaptar a sua adequação a outros ambientes, propiciando uma economia de esforço considerável para o desenvolvedor.

A aplicação não ficará dependente a um único ambiente, fazendo com que o usuário final tenha maior mobilidade na execução de suas tarefas, além de um ganho de produtividade, agilidade, controle e confiabilidade.

Além disso, espero que este trabalho possa contribuir para que novas soluções nesse âmbito possam surgir, agregando ambientes e diferentes plataformas para que não nos tornemos dependentes a um determinado hardware e tecnologia.

3 HIPÓTESES DE SOLUÇÃO

O presente estudo precisa avaliar as seguintes hipóteses para chegar a uma solução adequada:

- Hipótese 1: A aplicação desenvolvida é portátil entre os ambientes Windows, Web e Mobile, utilizando o mesmo framework, pois as classes disponíveis nas duas variações de framework são as mesmas;
- Hipótese 2: A aplicação desenvolvida é portátil para os ambientes Windows, Web e Mobile, utilizando o .NET Framework [THA01] além do Compact Framework [YAN07], pois algumas classes de determinado framework não podem ser referenciadas para a utilização em outro;
- Hipótese 3: É necessário que seja desenvolvida uma aplicação para cada ambiente, pois algumas classes de determinado framework não podem ser referenciadas para a utilização em outro, além disso, existe a incompatibilidade de componentes.

4 OBJETIVOS

Atualmente, desenvolver uma aplicação para o ambiente Windows com diversas funcionalidades e disponibilizar um módulo desta aplicação para que seja executada em outro ambiente, não é nenhuma novidade. Porém, essa situação deve ocasionar um retrabalho considerável para o desenvolvedor, inviabilizando a reutilização de código por se tratar de ambientes e até mesmo com linguagens de programação diferentes. Com isso, dificilmente o profissional que desenvolveu a aplicação Windows será o mesmo que desenvolverá o módulo para Web ou para um dispositivo móvel.

Diante dessas circunstâncias, o objetivo deste trabalho é realizar um estudo que verifique a portabilidade de uma aplicação, para que a mesma possa ser executada em mais de um ambiente, facilitando a reutilização de código além de anular o retrabalho, pois a aplicação será portátil para que seja executada em outros ambientes.

Especificamente, a pesquisa irá focar a eficiência e eficácia do emprego de uma linguagem de programação em uma única aplicação, mas que essa seja executada independente da plataforma escolhida, seja ela no próprio desktop, através de um navegador de internet ou até mesmo de um dispositivo móvel.

5 FUNDAMENTAÇÃO TEÓRICA

- Plataforma .NET

Plataforma de desenvolvimento que possui uma série de tecnologias em conjunto. A integração delas faz com que o desenvolvimento fique facilitado, pois esta mesma plataforma pode ser usada para ambiente desktop, web e mobile.

O objetivo de utilizar esta plataforma é proporcionar a integração de diversas tecnologias em uma única solução. Ela possui vantagens significativas como a independência de linguagem de programação, ou seja, independente da linguagem escolhida para o desenvolvimento. Sendo esta uma linguagem suportada pela plataforma, o resultado será o mesmo, além disso, possui um único ambiente de desenvolvimento. Outro aspecto muito importante é a interoperabilidade entre tecnologias, que é a capacidade de uma aplicação se comunicar com outra através de padrões abertos como, por exemplo, o XML [SHO02].

- .NET Framework

Um framework é uma estrutura pré-definida que contém um conjunto de componentes de software oferecendo uma estrutura padrão para o desenvolvimento de uma aplicação.

O .NET Framework oferece suporte a mais de 20 linguagens de programação, como já foi mencionado acima, independente da linguagem escolhida, sendo ela suportada por este framework o resultado da aplicação será o mesmo.

- Componentes do .NET Framework

Grande parte dos recursos disponibilizados por este framework se dá através de dois componentes, o Common Language Runtime (CLR) [THA01] e o Class Library [THA01].

O Common Language Runtime é responsável por fornecer o ambiente de execução, além dos serviços para a construção de aplicações. Antigamente, quando um programa era compilado, era o sistema operacional que gerava esses recursos, tornando o programa dependente ao sistema operacional, caso o programa tivesse que ser utilizado em outra plataforma era necessário recriá-lo.

No .NET Framework, quando um programa é compilado, é gerado um código intermediário conhecido por MSIL (Microsoft Intermediate Language) [THA01], à medida que o programa for executado, o CLR compila o código MSIL para o código de máquina, desse modo, o sistema operacional executa este código.

Sem utilizar os recursos diretamente do sistema operacional o CLR faz com que o programa desenvolvido possa ser executado em outras plataformas, bastando que o .NET Framework esteja disponível no sistema operacional.

Outro componente importante dentro do .NET Framework é o Class Library. Como o próprio nome diz, é a biblioteca de classes, o responsável por fornecer funcionalidades fundamentais para o desenvolvimento, como entrada e saída de dados, interface com o usuário, além de funcionalidades mais avançadas como conexão de banco de dados.

- UML

UML é a linguagem de modelagem unificada (Unified Modelling Language) [PIL05]. Através de seus diagramas, ela facilita a especificação, visualização e documentação de modelos de sistemas orientados a objeto. Seus diagramas são divididos em duas categorias: Diagramas Estruturais e Diagramas de Comportamento.

São usados diagramas estruturais para capturar a organização física dos elementos dentro do seu sistema, como um objeto se relaciona a outro. Os diagramas estruturais são:

Diagrama de Classe: Usa classes e interfaces para capturar detalhes sobre as entidades que compõem seu sistema e as relações estáticas entre eles.

Diagrama de Componentes: Mostra a organização e dependências que envolvem a implementação de um sistema.

Diagrama de Objetos: Usa a mesma sintaxe do diagrama de classe, mostrando um conjunto de objetos e seus respectivos relacionamentos em um determinado ponto no tempo.

Diagrama de Implantação: Mostra como seu sistema é executado e associado ao hardware. É usado para expor como os componentes são configurados em tempo de execução.

Diagramas de comportamento focam o comportamento de elementos em um sistema. É possível usar diagramas de comportamento para capturar exigências, operações e a alteração de estado dos elementos. Os diagramas de comportamento são:

Diagrama de Atividades: Captura o fluxo de um comportamento ou atividade, para o próximo. É semelhante a um fluxograma clássico, entretanto, é mais expressivo.

Diagrama de Caso de Uso: Capturam as exigências funcionais do sistema. Mostram os atores que vão interagir com o sistema e os casos de uso com os respectivos relacionamentos.

Diagrama de Estados: Enfatiza o fluxo de controle de um estado para o outro. Ele contém estados simples, compostos e transições, incluindo eventos e ações.

Diagramas de Interação: São usados para mostrar a interação formada pelos objetos e seus relacionamentos, incluindo as mensagens trocadas entre os mesmos. Os tipos de diagramas de interação são os de Seqüência e os de Colaboração. Eles são equivalentes, podendo ser um convertido em outro, entretanto, não é uma regra que eles visualizem exatamente as mesmas informações.

- Tecnologias

O ambiente de desenvolvimento adotado será o Microsoft Visual Studio 2008 e o desenvolvimento da aplicação será feito na linguagem de programação C#.NET.

Em relação ao banco de dados, será utilizado o Microsoft SQL Server 2005 [DEW06]. Para os ambientes Web e Mobile, será desenvolvido um Web Service [SHO02], que fará o elo desses ambientes com o banco de dados.

- Ambientes e Plataformas

Abaixo seguem os ambientes e suas respectivas plataformas que irão atender a aplicação.

- Windows
 - Microsoft Windows XP e Microsoft Windows Vista;
- Web
 - Internet Explorer 6 ou superior e Mozilla Firefox;
- Mobile
 - Smartphone e PocketPC;
 - Windows Mobile 5 ou superior.

6 SOLUÇÃO PROPOSTA

Após a realização dos estudos na plataforma .NET o próximo passo no TCC I será definir uma metodologia de desenvolvimento para a plataforma .NET que contemple os diversos aspectos existentes entre os ambientes (Windows, Web e Mobile), procurando identificar o que pode ser objeto de programação compartilhada ou não.

A partir da definição dessa metodologia, será desenvolvido um modelo para validar e verificar o funcionamento destes três ambientes. Além de conhecer o nível de portabilidade existente, aspectos como reutilização de métodos, classes e interfaces serão consideradas, até mesmo a maneira de como os três ambientes se integrarão referentes ao acesso a dados.

O modelo desenvolvido no TCC I, servirá de base para que no TCC II seja desenvolvida uma aplicação para o Departamento de Criminalística do Instituto-Geral de Perícias do Estado, com a finalidade de demonstrar os resultados obtidos ao longo da pesquisa.

A proposta é desenvolver uma única aplicação para o Departamento, sendo executada em ambientes Windows, Web e Mobile, com total integração entre eles. A aplicação consiste no controle das atividades dos peritos, podendo, as mesmas estarem em andamento, finalizadas e ainda a serem realizadas.

7 CRONOGRAMA

Abaixo segue o cronograma para as atividades do Trabalho de Conclusão do Curso I.

Atividade	Março				Abril				Maió				Junho			
Reunião Inicial																
Elaboração da Proposta																
Entrega da Proposta																
Revisão Bibliográfica																
Elaboração do Relatório Final																
Elaboração do Artigo																
Apresentação do Seminário de Andamento – TCC1																
Desenvolvimento de Protótipo																
Testes do Protótipo																
Entrega do Relatório Final																

8 REFERÊNCIAS BIBLIOGRÁFICAS

[DEW06] – DEWSON, Robin. **Beginning SQL Server 2005 for Developers**. [1-59059-588-2], 2006. 536p.

[PIL05] – PILONE, Dan. PITMAN, Neil. **UML 2.0 in a Nutshell**. [0-596-00795-7], 2005. 234p.

[THA01] – THAI, Thuan. LAM, Hoang. **.NET Framework Essentials**. [0-596-00165-7], 2001. 320p.

[TRO07] - TROELSEN, Andrew. **Pro C# 2008 and the .NET 3.5 Platform, 4th Edition**. [978-1-59059-884-9], 2007. 1370p.

[SHO02] – SHORT, Scott. **Building XML Web Services for the .NET Platform**, [0-735-61406-7], 2002. 426p.

[YAN07] – YANG, Baijian. ZHENG, Pei. NI, Lionel. **Professional Microsoft Smartphone Programming**. [0-471-76293-5], 2007. 494p.