

Carnê-Leão – Uma Experiência de Desenvolvimento com Componentes e APIs

FÁBIO DAROS DE FREITAS

SECRETARIA DA RECEITA FEDERAL
DELEGACIA DA RECEITA FEDERAL EM VITÓRIA E.S.
CEMICRO-CENTRO DE MICROINFORMÁTICA

Rua Pietrângelo de Biase, 56 sala 308 – Centro
CEP 29010-190
Vitória - ES
ffreitas@receita.fazenda.gov.br

Resumo

Este trabalho relata aspectos do desenvolvimento de uma aplicação de ampla distribuição que, por essa razão, requer um alto padrão de qualidade – o programa Carnê-Leão da Secretaria da Receita Federal. Para a implementação foi utilizado um modelo híbrido, conjugando *Rapid Application Development* (RAD) com componentes e *Application Programming Interfaces* (APIs) proprietárias dotadas de importantes mecanismos de depuração. Esse modelo de implementação reduziu a duração dos ciclos de correção de erros, potencializando os resultados do modelo de desenvolvimento, fortemente orientado a testes. O padrão de qualidade alcançado, com cerca de uma dezena de erros reportados em 170.000 instalações estimadas, atendeu plenamente os requisitos de qualidade estabelecidos.

Abstract

This work presents some development issues of an application with a broad distribution that, for this reason, demands a high product quality – The Secretaria da Receita Federal's Carnê-Leão. In order to implement this application, a hybrid model has been used, putting together *Rapid Application Development* (RAD) with components and custom *Application Programming Interfaces* (APIs) with important debugging facilities. This implementation model has reduced the duration of the debugging cycles, resulting on a highly test-oriented development model. The quality level achieved, with a dozen errors reported for an estimated installation base of 170.000 copies, fulfilled the required quality requisites.

PALAVRAS-CHAVE: Desenvolvimento, Arquitetura de Software, Teste.

1 Introdução

No ano de 1991, a Secretaria da Receita Federal (SRF) optou pela utilização dos Programas Geradores de Declaração (PGD) como um modelo alternativo para a entrega de declarações do Imposto de Renda das Pessoas Físicas (IRPF). Este modelo consiste no preenchimento das declarações através de um software com interface gráfica de usuário (GUI) amigável, desenhada para conduzir o contribuinte na tarefa de preenchimento. Este software realiza todos os cálculos e testes de consistência determinados pela legislação vigente, esclarece dúvidas de utilização e legislação através da ajuda ao usuário (*help on-line*), e disponibiliza outras facilidades tais como impressão e cópia de segurança (*backup*). A saída gerada pelo PGD pode ser entregue em meio magnético ou, principalmente nos últimos anos, transmitida diretamente para os computadores da SRF através da Internet.

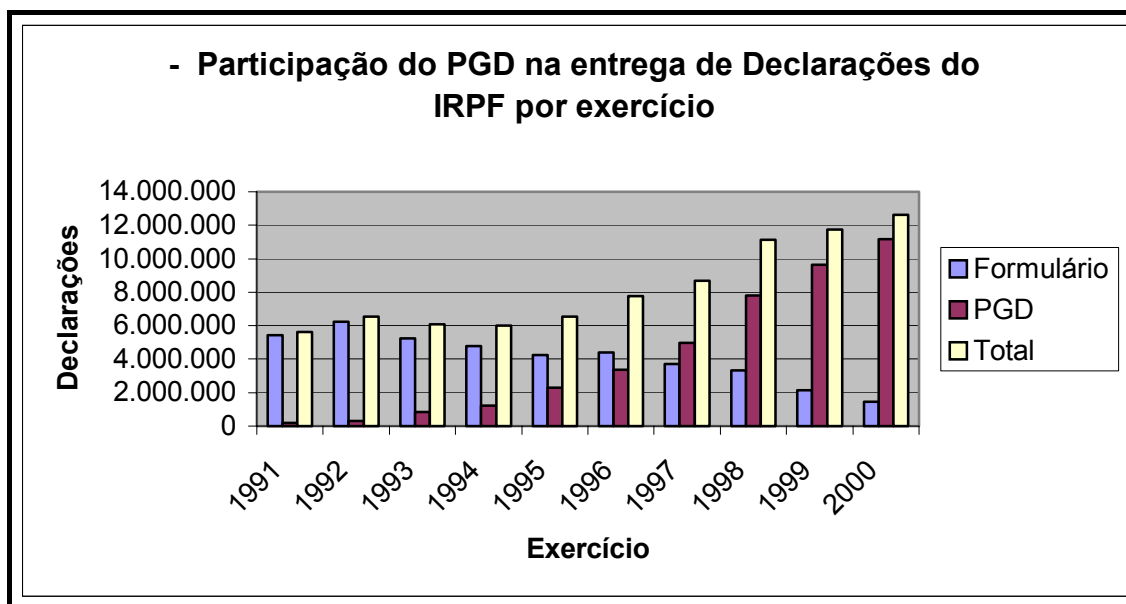
O grande sucesso deste modelo de entrega de declarações levou o PGD da Declaração de Ajuste Anual do IRPF a se tornar o principal meio de entrega de declarações a partir do ano de 1997. A Tabela 1 e o Figura 1 mostram a evolução da adoção do PGD do IRPF até o ano de 2000. Entre as principais vantagens deste modelo estão a enorme economia com os custos de transcrição e impressão de formulários e a grande agilidade no processamento das declarações.

Tabela 1 - Participação do PGD na entrega de Declarações do IRPF por exercício

Ano/Meio	Formulário	PGD	Total	% PGD
1991	5.416.681	193.311	5.609.992	3,4%
1992	6.232.229	315.040	6.547.269	4,8%
1993	5.246.760	842.113	6.088.873	13,8%
1994	4.799.487	1.224.815	6.024.302	20,3%
1995	4.250.878	2.279.766	6.530.644	35,0%
1996	4.403.887	3.363.028	7.766.915	43,3%
1997	3.723.304	4.964.418	8.687.722	57,1%
1998	3.336.229	7.799.266	11.135.495	70,04%
1999	2.136.798	9.624.192	11.760.990	81,83%
2000	1.461.996	11.162.865	12.624.861	88,42%

Fonte: Serviço Federal de Processamento de Dados – SERPRO (posição em 09/04/2001)

Figura 1 - Participação do PGD na entrega de Declarações do IRPF por exercício



Fonte: Serviço Federal de Processamento de Dados – SERPRO (posição em 09/04/2001)

A Declaração de Ajuste Anual do IRPF é composta pelas Fichas comuns a todos os contribuintes e Anexos de atividade específica. Devido a complexidade de alguns destes Anexos, a partir de 1998 o PGD IRPF iniciou um processo de modularização, com o objetivo de transformá-los em programas auxiliares independentes. Uma vantagem direta oferecida por esta abordagem foi a possibilidade da disponibilização de novos recursos e funcionalidades do domínio dos Anexos.

Este artigo relata aspectos do projeto e implementação do programa Carnê-Leão, concebido a partir do Anexo Carnê-Leão da Declaração de Ajuste Anual do IRPF a partir do ano de 1998, bem como alguns aspectos do modelo de desenvolvimento adotado pela SRF para o PGD IRPF e seus programas auxiliares.

Após esta introdução, os objetivos do programa Carnê-Leão e o modelo de desenvolvimento utilizado são apresentados na seção 2. A seção 3 apresenta aspectos da arquitetura de software utilizada e suas motivações. A seção 4 relata os resultados percebidos durante o ciclo de desenvolvimento e após a distribuição do produto, e a seção 5 finaliza este trabalho apresentando nossas conclusões.

2 O Programa Carnê-Leão

Os contribuintes sujeitos à apuração mensal do imposto de renda das pessoas físicas através do Carnê-Leão são aquelas pessoas físicas residentes no Brasil e que recebem rendimentos oriundos de outra pessoa física ou do exterior, tais como profissionais autônomos e locatários de imóveis. Dentre estes contribuintes, os profissionais liberais são obrigados a manter, ao término do exercício, um livro Caixa com o resultado da sua atividade profissional.

O programa Carnê-Leão foi concebido para ser utilizado pelos contribuintes durante todo o exercício, realizando os cálculos da apuração mensal e a escrituração eletrônica do livro Caixa. Quando do preenchimento da Declaração de Ajuste Anual do IRPF no ano subsequente, o programa exporta os dados

consolidados do exercício para o PGD IRPF, onde é apurado o imposto a pagar ou a restituição a ser recebida.

Nos primeiros anos foram produzidos cerca de 120.000 disquetes/ano para distribuição nas unidades da SRF, porém a partir de 2000 a SRF passou a distribuir todos os programas somente em CD-ROM. Adicionalmente, são realizados cerca de 35.000 *downloads*/ano no site da SRF (www.receita.fazenda.gov.br).

Modelo de Desenvolvimento Utilizado

O modelo de desenvolvimento utilizado para o Carnê-Leão, e demais PGDs e programas auxiliares do IRPF, é baseado no modelo cascata [1] com etapas formais de especificação, desenvolvimento, teste e homologação. As equipes são mapeadas pelas etapas do modelo e formadas exclusivamente por técnicos da SRF. A Tabela 2 mostra os quantitativos e perfis de cada equipe.

Tabela 2 - Distribuição das Equipes de Desenvolvimento do Carnê-Leão

Equipe	Quantitativo	Perfil
Especificação e Gerência	3	Técnico tributário
Desenvolvimento	3	Análise e programação
Testes e Homologação	5 a 20	Técnico tributário de técnico em informática

Devido à natureza legal do programa, a sua disponibilização no início do exercício é um fator determinante para a sua utilização. Todas alterações tardias da legislação, entre outros fatores, têm que ser incorporadas a tempo pelo programa. A ampla distribuição e a dificuldade de suporte ao usuário, imposta principalmente pela diversidade de instalações e equipamentos, impõem um elevado nível de qualidade para uma aplicação desta natureza.

O Carnê-Leão pode ser classificado como um *sistema-E* [2]. Assim, para acomodar os requisitos acima, os quais são determinantes para o sucesso da aplicação [3], foi adotado um modelo de desenvolvimento com uma forte ênfase nos testes e com uma etapa formal de homologação.

No modelo de desenvolvimento adotado, a partir do primeiro protótipo do programa, elaborado no início da etapa de desenvolvimento com objetivo de validar os critérios da GUI, foi criado o documento *Ata de Teste*. Este documento é constituído basicamente pelos erros de qualquer natureza encontrados durante os testes (*bug-list*) e pelos refinamentos da especificação e melhorias funcionais sugeridos pelas equipes (*wish-list*). Uma vez criada uma entrada na Ata de Teste, esta somente é retirada após a sua resolução. A Ata de Teste foi o principal mecanismo de acompanhamento (*tracking*) do projeto, sendo controlada exclusivamente pelo gerente de projeto.

3 Arquitetura e Implementação

A forte ênfase em testes do processo de desenvolvimento, o grande número de ambientes de usuário e a diversidade de plataformas alvo especificadas (Windows 3.X, 95, 98 e NT) e desejadas (UNIX e Macintosh) motivaram a adoção de uma arquitetura de aplicação que facilitasse os testes.

O estilo de arquitetura de aplicação escolhido foi o de sistema em camadas [4]. Foram especificadas três camadas funcionais básicas: Apresentação, que trata toda a interação reativa com o usuário, tal como navegação, entrada de dados, *help*

e relatórios; Regras de Negócio, que implementa as validações, cálculos e consistências relativas à apuração do tributo; e Persistência, que implementa todas as funcionalidades de armazenamento e recuperação, integridade dos dados, cópia de segurança (*backup*) e a comunicação com o PGD IRPF. A Figura 2 mostra o diagrama da arquitetura do Carnê-Leão.

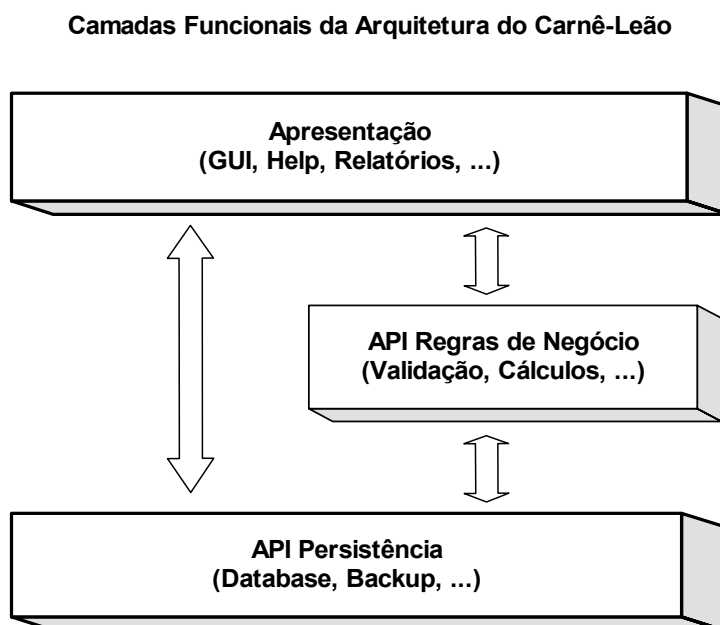


Figura 2 – Arquitetura da aplicação

Implementação

A partição da aplicação objetivou o tratamento independente da implementação das camadas funcionais. Diante das tendências atuais de desenvolvimento baseado em componentes, e toda a calorosa discussão que cerca este tema [5, 6], optamos pelo pragmatismo de um modelo híbrido de desenvolvimento, conjugando *Rapid Application Development* (RAD) e componentes com *Application Programming Interface* (APIs) proprietárias. Desta forma, pode-se ter liberdade de escolha de técnicas e ambientes de desenvolvimento adequados às características funcionais de cada camada, apesar dos custos de integração intrínsecos a esta abordagem [7].

Através do uso de interfaces bem definidas, foi possível tornar as camadas opacas e esconder suas implementações umas das outras. Assim, cada camada pôde ser testada separadamente, e o teste e a depuração da aplicação foi bastante facilitado.

A camada de Apresentação foi implementada com a ferramenta RAD Borland Delphi e seus componentes nativos. Adicionalmente, foram utilizados componentes comerciais e *freeware* para GUI e geração relatórios. Estas escolhas foram norteadas principalmente pela facilidade de prototipação e produtividade dos ambientes RAD.

A camada da Regras de Negócio consiste basicamente em um conjunto de funções invocadas pelas outras camadas. Desta forma, optamos por implementá-la através de uma *Dynamic Link Library* (DLL) escrita em C com o compilador Borland

C++ 4.5. Uma motivação adicional para esta escolha foi a possibilidade da sua distribuição como um componente de software contendo algumas regras de negócio da SRF, tais como validação de CPF e CNPJ (antigo CGC), para utilização pela indústria.

A qualidade da camada de Persistência é de especial importância para a aplicação. Como, ao contrário do PGD IRPF, o Carnê-Leão é utilizado durante todo o exercício, falhas na preservação dos dados podem resultar em prejuízos significativos para o contribuinte, abalando a sua confiança no programa e prejudicando o sucesso da aplicação. Desta forma, optamos por adquirir uma biblioteca comercial de banco de dados, de reconhecida qualidade, com código-fonte em C, e adequá-la às nossas necessidades, gerando assim uma API de persistência no domínio da aplicação na forma de uma DLL utilizada pelas outras camadas da aplicação.

Estas escolhas nos permitiram dotar o código-fonte da aplicação de facilidades de depuração, tais como *log* de execução, *tracking* de erros, bem como elevar a robustez da aplicação, através de mecanismos de *crash-recovery* disponibilizados pela biblioteca de banco de dados. O controle e disponibilidade do código-fonte das partes críticas da aplicação nos permitiu realizar inspeções de código e *walkthroughs* [8], onde foram descobertos dezenas de erros durante a etapa final de testes e homologação, melhorando significativamente qualidade final do produto.

4 Resultados Alcançados

A primeira versão do Carnê-Leão (Carnê-Leão 1998) para a plataforma Windows foi concluída em abril de 1998 com 113.138 linhas de código (LOC) e 1988 pontos de função ajustados (FP) [1]. A partir deste ano, será elaborada uma versão anual do programa em função das alterações da legislação, melhorias e correções de erros.

Apesar da falta de estatísticas sobre o número preciso de instalações, estimamos em 85.000 instalações por ano, totalizando 170.000 instalações os dois anos de existência do Carnê-Leão. Neste período, foram reportados cerca de uma dezena de erros reclamações pelos usuários. Este resultado é atribuído principalmente ao processo de desenvolvimento utilizado, à adequação da arquitetura de aplicação escolhida e à atenção despendida nas decisões de projeto [9], especialmente nos critérios do projeto da interface de usuário [10] e nos mecanismos de depuração implementados.

5 Conclusões

Uma aplicação de natureza legal e de ampla distribuição requer altos padrões de qualidade, demandando um modelo de desenvolvimento fortemente orientado a testes e que contemple uma etapa formal de homologação.

Através de uma arquitetura de aplicação baseada em camadas funcionais, na implementação do programa Carnê-Leão foi adotado um modelo híbrido de desenvolvimento, conjugando RAD e componentes com APIs proprietárias. Esta escolha permitiu a implantação e utilização com sucesso de importantes mecanismos de depuração para as partes críticas da aplicação, resultando ciclos de correção de erros bastante curtos. O padrão de qualidade alcançado, com cerca de uma dezena de erros reportados em 170.000 instalações estimadas, atendeu plenamente aos requisitos estabelecidos para este projeto.

Agradecimentos

Este trabalho é resultado do esforço de inúmeras pessoas envolvidas no projeto Carnê-Leão, entre técnicos da SRF e colaboradores, e contou com total apoio institucional da SRF. Entretanto, não podemos deixar de agradecer a dedicação e empenho do coordenador do projeto, Cristóvão Barcelos da Nóbrega, e especialmente dos colegas da equipe de desenvolvimento Christian Daros de Freitas e Ricardo Caetano.

Referências

- [1] R. S. Pressman. *Software Engineering: A Practitioner's Approach*. McGraw-Hill International Editions, 1992.
- [2] M M. Lehman. *Programs, Life Cycles and the Laws of Software Evolution*. *Proc. IEEE*. IEEE, Piscataway, N.J., Vol. 68, N.9, Sept, 1980.
- [3] S. L. Pfleeger. *The Nature of System Change*. *IEEE Software*. Vol. 15, N.3, May, 1998.
- [4] M. Shaw and D. Garlan. *Software Architecture: Perspectives on na Emerging Discipline*. Prentice-Hall, Inc, 1996.
- [5] J. M. Voas. *COTS: The Economical Choice?*. *IEEE Software*. Vol. 15, N.2, March, 1998.
- [6] J. M. Voas. *The Challenges of Using COTS Software in Component-Based Development*. *IEEE Computer*. Vol. 31, N.6, June, 1998.
- [7] J. C. Zubeck. *Implementing Reuse with RAD Tool's Native Objects*. *IEEE Computer*. Vol. 30, N.10, October, 1997.
- [8] G. J. Myers. *The Art of Software Testing*. John Wiley & Sons, Inc, 1979.
- [9] C. Pescio. *When Past Solutions Cause Future Problems*. *IEEE Software*. Vol. 14, N.5, September, 1997.
- [10] B. Shneiderman. *Design the User Interface: Strategies for Efective Huma-Computer Interaction*. 3ed., Addison Wesley Longman, Inc., 1998.

Fábio Daros de Freitas

atua em diversas áreas da computação há mais de 10 anos, sendo atualmente analista de sistemas na Receita Federal e consultor independente. Ele recebeu Bsc. em Engenharia de Telecomunicações pela Universidade Federal Fluminense e Msc. em Informática pela Universidade Federal do Espírito Santo, onde atualmente está em Doutorado no Programa de Pós-Graduação em Engenharia Elétrica. Suas principais áreas de interesse são Sistemas Operacionais, Sistemas distribuídos, software multi-plataforma e aplicações de Redes Neurais no Mercado Financeiro. Freitas é membro do IEEE Computer Society e pode ser contactado em `freitas@computer.org` ou `ffreitas@receita.fazenda.gov.br`