

UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

**Desenvolvimento de software de precificação e
contratação de câmbio pronto em banco de varejo**

Autor: Fernando Lahoz Pardo Reiz

Orientador: Profa. Dra. Liliane Ventura

São Carlos

2017

Fernando Lahoz Pardo Reiz

**Desenvolvimento de software de precificação e
contratação de câmbio pronto em banco de varejo**

Trabalho de conclusão de Curso
apresentado à Escola de
Engenharia de São Carlos, da
Universidade de São Paulo.

Curso de Engenharia Elétrica

Orientador: Profa. Dra. Liliane
Ventura

São Carlos

2017

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

L363d Lahoz Pardo Reiz, Fernando
Desenvolvimento de software de precificação e
contratação de câmbio pronto em banco de varejo /
Fernando Lahoz Pardo Reiz; orientadora Liliane Ventura
Schiabel. São Carlos, 2017.

Monografia (Graduação em Engenharia Elétrica com
ênfase em Eletrônica) -- Escola de Engenharia de São
Carlos da Universidade de São Paulo, 2017.

1. operações de câmbio. 2. desenvolvimento de
software de contratação. 3. banco de varejo. 4. mercado
internacional. 5. programação em visual basic. I.
Título.

FOLHA DE APROVAÇÃO

Nome: Fernando Lahoz Pardo Reiz

Título: “Desenvolvimento de software de precificação e contratação de câmbio pronto em banco de varejo”

Trabalho de Conclusão de Curso defendido e aprovado
em 19 / 06 / 2017,

com NOTA 8,0 (oito, zero), pela Comissão Julgadora:

*Profa. Associada Liliane Ventura Schiabel - Orientadora -
SEL/EESC/USP*

Prof. Associado Ivan Nunes da Silva - SEL/EESC/USP

Prof. Associado Adilson Gonzaga - SEL/EESC/USP

Coordenador da CoC-Engenharia Elétrica - EESC/USP:
Prof. Associado José Carlos de Melo Vieira Júnior

Agradecimentos

Gostaria de agradecer em primeiro lugar às duas pessoas que são e serão responsáveis por todos os bons acontecimentos em minha vida: meus pais. Aos meus pais Carlos e Patrícia pelo infinito amor e apoio em todas as decisões que tomei, por mais difíceis que fossem. Sem eles nada seria possível; à minha companheira Giovana, que assim como meus pais me apoiou e me deu forças para continuar seguindo em frente, mesmo com todas as dificuldades encontradas; aos meus irmãos Karla, Julio e Juliana que também estiveram ao meu lado e sempre me motivaram a nunca desistir. Enfim, dedico este trabalho a todos aqueles que de alguma forma contribuíram para a minha formação ou que simplesmente desejaram o meu sucesso e bem-estar.

Agradeço especialmente à professora e amiga Liliane, que me apoiou incondicionalmente por mais de dois anos de graduação. Seus conselhos foram essenciais para a minha formação profissional e pessoal. Com ela aprendi o verdadeiro significado da palavra professor e, mesmo não conseguindo me levar para a área acadêmica, me mostrou a beleza da formação e da transmissão do conhecimento.

Não posso deixar de agradecer aos amigos Felipe e Julio pelo apoio, conhecimentos que dividimos e momentos felizes que tivemos durante a graduação. Tudo fica mais fácil quando os amigos verdadeiros estão presentes.

À minha ex-gestora Suely Teshima e aos meus amigos e colegas de trabalho Amanda, Mariana e Marcello Wohnrath o meu muito obrigado. A confiança que depositaram em mim viabilizou a confecção deste trabalho.

Fernando Lahoz Pardo Reiz

*“O investimento em conhecimento
é aquele que traz maiores retornos. ”*

Benjamin Franklin

Resumo

As operações de câmbio no Brasil são alvos de intensa regulação por parte do Banco Central do Brasil (Bacen), que é o órgão máximo responsável por executar as políticas monetárias e pela fiscalização do cumprimento das mesmas no país. Devido à política cambial atualmente adotada pelo Bacen, todas as operações de câmbio, sejam de entrada ou saída de reservas estrangeiras, são monitoradas através de um canal eletrônico de comunicação entre os bancos e instituições que são autorizadas a operar com câmbio e o próprio Banco Central. Dessa forma, o Bacen recebe diariamente, bases de dados com todas as operações de câmbio realizadas no país e, conseqüentemente, consegue controlar e até mesmo manipular o preço das divisas estrangeiras frente à moeda nacional. Com o expressivo crescimento do número de operações de câmbio realizadas diariamente, cria-se a necessidade do desenvolvimento de sistemas que sejam capazes de processar as transações dentro de uma janela de 24 horas, visto que os bancos enviam os relatórios ao Bacen através do SISBACEN diariamente. Além disso, com a grande capilaridade dos bancos de varejo, torna-se indispensável o desenvolvimento de sistemas que consigam transmitir e centralizar informações das mais diversas geografias do território nacional. Este trabalho consiste no desenvolvimento de um software, programado em Visual Basic, que seja capaz de facilitar o fluxo de informações entre as diversas agências de um banco de varejo e sua sede, e tornar mais rápida e acessível a contratação de câmbio por pessoas físicas em todo o país, sem fugir das especificidades e formalidades requeridas pelos órgãos reguladores.

Palavras-Chave: Operações de câmbio, desenvolvimento de software de contratação, banco de varejo, mercado internacional, programação em Visual Basic.

Abstract

Forex transactions in Brazil are target of intense regulation by the Brazilian Central Bank, which is the utmost responsible organ for executing the monetary policies and for assuring that these regulations are fulfilled. Due to the current monetary policy adopted by the country, Bacen monitors every forex transaction through an electronic channel of data exchange with the institutions that are authorized to operate in the forex market. By doing so the Central Bank can also control and monitor the exchange rate, receiving daily databases with all the forex transactions booked in the country. With the substantial increase in the number of daily forex transactions, arises the necessity of developing systems that are capable of processing these transactions in a 24-hour window, considering that the banks have to provide Bacen with daily reports, through its system - SISBACEN. Furthermore, the wide capillarity of retail banks and the large territorial extension of the country, demands the development of systems that can establish an optimal flow of information nationwide. Finally yet importantly, this work aims to develop a software based on Visual Basic that is capable of promoting a faster and accurate information flow between the offices of retail banks. Making the forex hiring process accessible without losing the formalities and specificities required by governmental authorities.

Keywords: Forex trasactions, Forex booking software development, retail banks, international markets, Visual Basic programming.

Lista de Figuras

Figura 1 - Balança comercial brasileira de 1950-2014.....	23
Figura 2 - Evolução das Exportações Mundiais - 1950-2014.	24
Figura 3 - Etapas de desenvolvimento do modelo em cascata. Fonte: (PRESSMAN, 2011)	29
Figura 4 – Probabilidade de ocorrência de falhas ao longo do tempo de desenvolvimento de software. Fonte: (PRESSMAN, 2011)	39
Figura 5- Delimitação da comunicação entre o software e as demais entidades envolvidas no sistema.....	41
Figura 6- Tela de abertura do programa.....	42
Figura 7 - Terceira tela da ferramenta, responsável por armazenar os dados do usuário.....	42
Figura 8 – Quarta tela da ferramenta, responsável por armazenar os dados do usuário.	43
Figura 9 - Primeira trava de verificação.....	44
Figura 10 - Segunda trava de verificação.....	44
Figura 11 - Terceira trava de verificação.	45
Figura 12 - Quarta trava de verificação.....	45
Figura 13 - Quinta trava de verificação.....	45
Figura 14 - Sexta trava de verificação.....	45
Figura 15 - Quinta tela: enquadramento da natureza da operação.	47
Figura 16 - Sexta tela: enquadramento de cláusulas especiais para incidência do Imposto de Renda.....	48
Figura 17 - Sétima tela: preenchimento com os dados da operação a ser contratada.....	49
Figura 18 - Trava de verificação de moeda.....	50
Figura 19 - Canal bancário para envio de remessas ao exterior.....	50
Figura 20 - Canal bancário para recebimento de remessas do exterior.....	51
Figura 21 - Tela de confirmação e impressão do contrato de câmbio gerado.	52
Figura 22 - Tela de encerramento do programa.	52

Figura 23 - Diagrama de tela I – Tela de abertura.....	53
Figura 24 - Diagrama de Tela II - Preenchimento de dados do funcionário.	54
Figura 25 - Diagrama de Tela III - Preenchimento de dados do cliente.....	55
Figura 26 - Diagrama de Tela IV - Enquadramento da natureza da operação.	56
Figura 27- Diagrama de tela V – Cláusulas contratuais.....	57
Figura 28 - Diagrama de tela VI - Dados da operação.....	58
Figura 29 - Diagrama de tela VII - Dados do canal bancário.....	59
Figura 30 –Diagrama de tela VIII – Geração do contrato de câmbio.	59
Figura 31 - Comparação do número de erros derivados do contrato de câmbio entre o processo antigo e o processo implementado.	64
Figura 32 – Comparação dos erros gerados pelo mal preenchimento do contrato de câmbio. ...	65
Figura 33 – Comparação do total de erros gerados pelos dois processos diferentes.....	65
Figura 34 - Primeira página do molde de contrato de câmbio disponibilizado pelo Bacen no anexo I à circular 3.691 de 16 de dezembro de 2013.	68
Figura 35 – Segunda página do molde de contrato de câmbio disponibilizado pelo Bacen no anexo I à circular 3.691 de 16 de dezembro de 2013.	69
Figura 36 - Primeira página do contrato de câmbio gerado pelo programa.	70
Figura 37- Segunda página do contrato de câmbio gerado pelo programa.	71

Lista de abreviaturas e siglas

BACEN – BANCO CENTRAL DO BRASIL

SISBACEN – SISTEMA DO BANCO CENTRAL

FEBRABAN – FEDERAÇÃO BRASILEIRA DE BANCOS

FOREX – FOREIGN EXCHANGE

CFMNU – CONFERÊNCIA FINANCEIRA E MONTÁRIA DAS NAÇÕES UNIDAS

FMI – FUNDO MONETÁRIO INTERNACIONAL

*BIRD – BANCO INTERNACIONAL DA RECONSTRUÇÃO E DO
DESENVOLVIMENTO*

USD – DÓLARES AMERICANOS

ECU – EUROS

VBA – VISUAL BASIC FOR APPLICATIONS

Sumário

Capítulo 1 – Introdução	20
1.1 O mercado cambial brasileiro	20
1.2 Objetivos	24
1.3 Justificativa	24
Capítulo 2 – Desenvolvimento do Software	27
2.1 Análise de Sistemas	27
2.1.1 Introdução ao projeto	29
2.1.2 Identificação da necessidade	30
2.1.3 Estudo da viabilidade	33
2.1.3.1 Viabilidade operacional.....	34
2.1.3.2 Viabilidade político-econômica	36
2.1.3.3 Viabilidade técnica-legal.....	39
2.1.4 Diagrama de arquiteturas.....	40
2.2 Análise de requisitos	41
2.2.1 Requisitos funcionais.....	41
2.2.1.1 Requisito funcional I – Abertura.....	41
2.2.1.2 Requisito funcional II – Preenchimento de dados do funcionário	42
2.2.1.3 Requisito funcional III – Preenchimento de dados do cliente.....	43
2.2.1.4 Requisito funcional IV – Enquadramento da natureza da operação .	46
2.2.1.5 Requisito funcional V – Cláusulas contratuais	47
2.2.1.6 Requisito funcional VI – Dados da operação	49
2.2.1.7 Requisito funcional VII – Canal bancário.....	50
2.2.1.8 Requisito funcional VIII – Contrato de câmbio	51
2.2.2 Diagrama de fluxo de dados	53
2.2.2.1 Diagrama Tela I – Abertura e Informações gerais	53
2.2.2.2 Diagrama Tela II – Preenchimento de dados do funcionário.....	54
2.2.2.3 Diagrama Tela III – Preenchimento de dados do cliente	54
2.2.2.4 Diagrama Tela IV – Enquadramento da natureza da operação	55
2.2.2.5 Diagrama Tela V – Cláusulas contratuais.....	56
2.2.2.6 Diagrama Tela VI – Dados da operação	57
2.2.2.7 Diagrama Tela VII – Canal bancário	58
2.2.2.8 Diagrama Tela VIII – Contrato de câmbio.....	59

Capítulo 3 – Testes e Manutenção.....	60
Capítulo 4 – Análise dos resultados obtidos.....	62
Capítulo 5 – Conclusão	66
Referências Bibliográficas.....	67
Apêndice.....	68
Anexo A – Molde de contrato de câmbio fornecido pelo Bacen.....	68
Anexo B – Exemplo de contrato de câmbio gerado pelo programa	70
Anexo C – Código	72
7.1.1 Página inicial	72
7.1.2 Formulário de dados do funcionário.....	72
7.1.3 Formulário de dados do cliente	74
7.1.4 Formulário de natureza da operação.....	78
7.1.5 Formulário de cláusulas de IR	81
7.1.6 Formulário de dados da operação	82
7.1.7 Formulário de Canal Bancário – Compra.....	89
7.1.8 Formulário de Canal Bancário – Venda	90
7.1.9 Formulário de geração do contrato	92
7.1.10 Formulário de encerramento do programa	98
7.1.11 Módulo externo de declaração das variáveis globais	99
7.1.12 Módulo de acesso às cotações online	101
7.1.13 Módulo de cálculo das taxas.....	104
7.1.14 Módulo de conversão de numeral em texto.....	115

Capítulo 1 – Introdução

1.1 O mercado cambial brasileiro

O objetivo deste trabalho é desenvolver um sistema simples e robusto que facilite a contratação de câmbio pronto por pessoas físicas. No entanto, para chegar ao patamar em que uma pessoa que não tem nenhum conhecimento de mercado possa converter determinada quantia de sua moeda local em uma outra moeda de sua escolha, é necessário compreender os aspectos históricos que levaram à criação do sistema utilizado atualmente e também entender os fenômenos que culminaram na necessidade de os países trocarem moedas entre si.

Uma operação de câmbio, é uma negociação de diferentes moedas entre pessoas interessadas em movimentá-las (FORTUNA, 2005). Consiste, basicamente, em trocar determinada quantia de uma moeda por outra quantia de uma moeda diferente. As quantias equivalentes e resultantes da troca comercial dependerão, de forma simplista, da oferta e da demanda por cada uma delas.

“Câmbio: operação financeira que consiste em vender, trocar ou comprar valores em moedas de outros países ou papéis que representem moedas de outros países. ”

(SANDRONI, 2008)

De posse da definição correta do termo câmbio, pode-se inferir que até meados do final da segunda guerra mundial, não era efetuada nenhuma operação de câmbio entre os países. Isso significa que os países não trocavam mercadorias entre si? Não necessariamente. Até o ano de 1944, a “moeda” utilizada para trocas comerciais entre os

países era a onça de ouro, ou seja, apesar de os países possuírem suas moedas locais e uma economia em que todas as trocas comerciais eram precificadas por tais moedas, ao realizar transações com o exterior, as mercadorias a serem trocadas eram precificadas em onças de ouro.

No ano de 1944, na cidade de Bretton Woods em New Hampshire, ocorreu um evento que reuniu as principais lideranças dos países mais desenvolvidos à época: Conferência Financeira e Monetária das Nações Unidas (CFMNU), que mais tarde ficou conhecido simplesmente por acordo de Bretton Woods. O principal objetivo do evento era definir um sistema financeiro e monetário internacional que permitisse que trocas comerciais entre diferentes países fossem realizadas e precificadas de forma unificada (MOFFITT, 1984). Com os efeitos devastadores da segunda guerra mundial, muitos países, como a própria Alemanha, estavam assolados e precisavam de financiamento externo para financiar a reconstrução do país e reerguer suas economias. Para viabilizar o sistema, os dirigentes das potências da época definiram diversas políticas que viriam a reger as primeiras transações cambiais de que se tem notícia.

O sistema em si basicamente estabelecia um limite máximo em que alguma moeda poderia flutuar em relação a outra sem que o governo tomasse alguma atitude para reverter esse cenário, ou seja, foi definido que a partir do preço determinado à época, os preços poderiam oscilar livremente até 1% em relação ao dólar, que foi a moeda escolhida como base de troca e padrão das reservas mundiais (SCAMMELL, 1975). Caso esse valor fosse ultrapassado, o governo responsável pela moeda deveria tomar alguma atitude (aumentando ou diminuindo a liquidez da moeda) para reestabelecer o preço da moeda dentro o patamar previsto. Outra medida que mais tarde viria a contribuir para a falência do próprio sistema, era que os demais países poderiam, se quisessem, trocar livremente os dólares americanos por ouro com um preço fixado em 35 dólares por cada onça de

ouro. Além das políticas supracitadas, o acordo foi responsável pela criação do Fundo Monetário Internacional (FMI) e do Banco Internacional da Reconstrução e Desenvolvimento (BIRD), órgão que ficaria responsável por garantir liquidez e viabilizar os empréstimos entre os países com destino à reconstrução e recuperação dos danos infligidos pela segunda guerra mundial.

Com o começo da guerra do Vietnam, o governo americano teve que aumentar significativamente seus gastos e, ao invés de aumentar os impostos para aumentar a arrecadação de dinheiro, o governo começou a emitir mais papel-moeda. Além disso, começou a ser implementada a política do “American Dream” ou sonho americano, que consistia em prover à todas as famílias americanas um alto padrão de consumo. Para manter os dois tipos de gastos concomitantemente sem elevar os impostos e exonerar a população, o governo continuou emitindo mais e mais moeda, elevando a taxa de inflação a patamares bem elevados. No entanto, pelas regras do acordo de Bretton Woods, a taxa de câmbio poderia oscilar em no máximo 1% para mais ou para menos. Dessa forma, para os americanos que estavam ganhando cada vez mais dólares e estavam inseridos em um mercado com elevada inflação, começou a ficar mais barato importar bens de consumo de outros países. Com o súbito aumento do número de importações e manutenção das exportações nos mesmos patamares, o saldo da balança comercial americana começou a ruir, causando impactos negativos na economia e gerando uma crise financeira.

Com a moeda americana cada vez mais desvalorizada, tornou-se impossível para os demais países manterem a taxa de câmbio travada com oscilação máxima de 1%. A primeira moeda a romper com o acordo de Bretton Woods e a oscilar livremente foi o marco alemão, seguido por outras moedas como libra esterlina, franco suíço e etc. Além de permitir que a taxa de câmbio perante o dólar oscilasse livremente, esses países fizeram com que os Estados Unidos cumprissem outro aspecto pactuado no acordo, o de converter

35 dólares americanos em uma onça de ouro. Como a moeda americana estava muito desvalorizada, o impacto dessa conversão foi terrível para os Estados Unidos e em um cenário de ruína de sua economia, o presidente Richard Nixon se viu forçado a quebrar o acordo de Bretton Woods e cancelar a conversibilidade de dólar em ouro a uma taxa fixa (ANDREWS, 2006).

No Brasil, para o escopo deste trabalho, será analisado o período pós plano-real (1994 em diante). Quando os Estados Unidos, durante o governo do presidente Richard Nixon, no início da década de 70, deixam de adotar o sistema de conversibilidade entre o dólar americano e a onça do ouro, inicia-se um processo de livre conversibilidade entre as moedas dos países mais desenvolvidos. No entanto, para os países em desenvolvimento, a adesão as trocas comerciais entre moedas se deu apenas mais tarde, em meados da década de 80. (BRUM; ZILIO, 2013)

As figuras 1 e 2, retiradas do site do Ministério da Indústria, Comercio Exterior e Serviços (MDIC), ilustram a evolução do comércio exterior brasileiro e mundial, respectivamente, durante o período compreendido entre os anos 1950 e 2015.

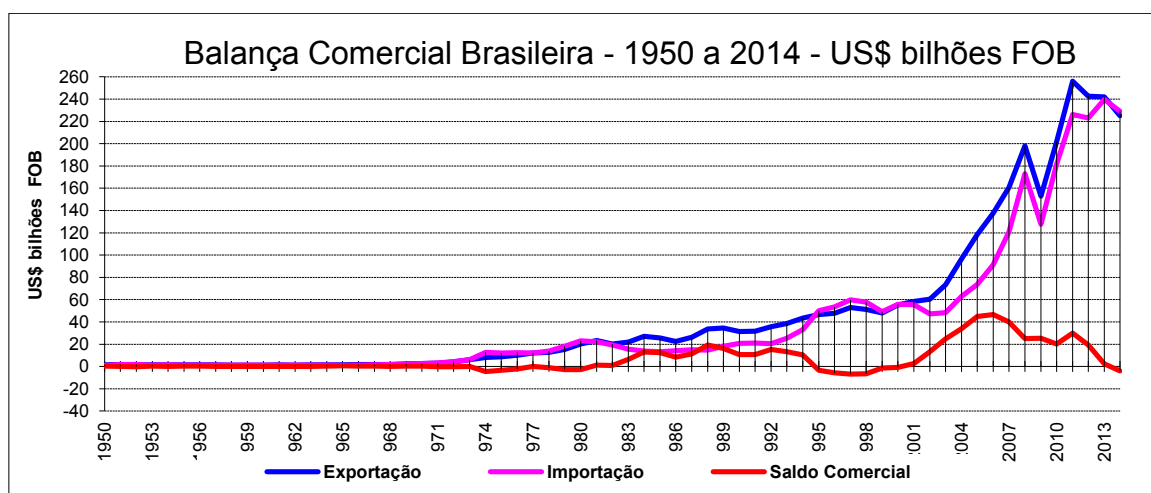


Figura 1 - Balança comercial brasileira de 1950-2014.¹

¹ Fonte: <http://www.mdic.gov.br/comercio-exterior/estatisticas-de-comercio-exterior/outras-estatisticas-de-comercio-exterior>, acessado em 21/02/2017

Nelas pode-se observar o rápido desenvolvimento das trocas comerciais a partir da década de 70 (na figura 2) e a partir da década de 80 (na figura 1).

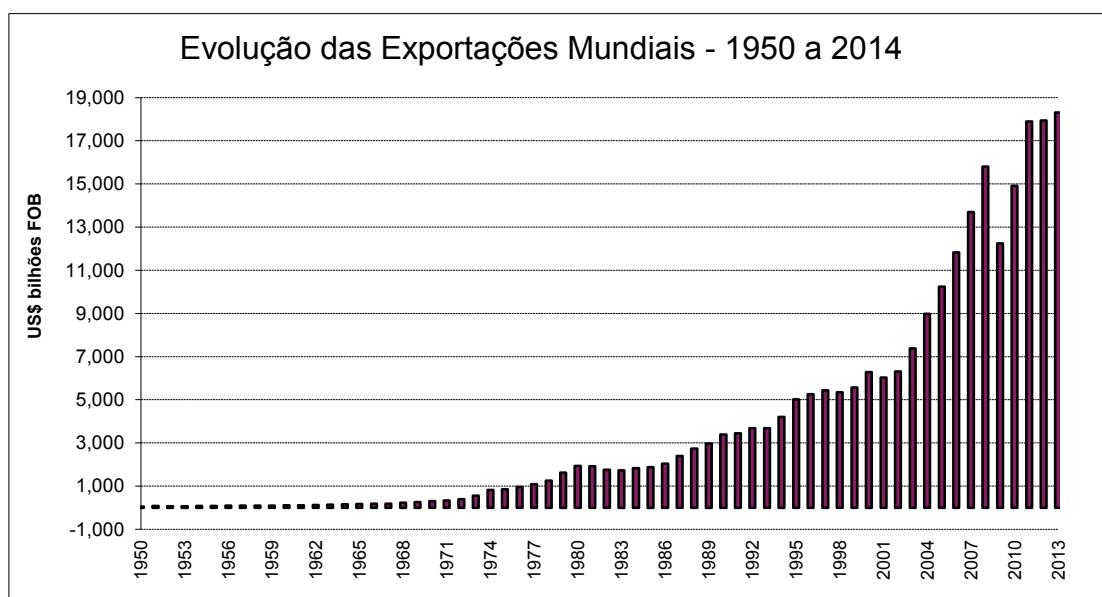


Figura 2 - Evolução das Exportações Mundiais - 1950-2014.²

1.2 Objetivos

O objetivo deste trabalho é desenvolver um software que simplifique a operacionalização das transações de câmbio para pessoas físicas. Especificamente, buscou-se desenvolver uma maneira simplificada de se gerar o contrato de câmbio, instrumento que formaliza as operações de câmbio celebradas entre as instituições financeiras e seus clientes.

1.3 Justificativa

A principal razão para o início da confecção deste trabalho, foi a observação e o levantamento do elevado número de erros gerados pelo mal preenchimento do contrato

² Fonte: <http://www.mdic.gov.br/comercio-externo/estatisticas-de-comercio-externo/outras-estatisticas-de-comercio-externo>, acessado em 21/02/2017

de câmbio. Conforme será descrito nos capítulos seguintes, previamente ao início deste trabalho, os contratos de câmbio eram preenchidos de forma manual pelos gerentes pessoa física das agências espalhadas pelo país. O modelo utilizado anteriormente está exemplificado no anexo C deste documento. A tabela 1 apresenta a quantidade de erros gerados durante o preenchimento do contrato de câmbio através do processo manual então adotado. O período de amostragem utilizado foi de um mês (junho 2016).

Tabela 1 - Principais erros encontrados em um contrato de câmbio - junho 2016.

Principais pendências/erros encontrados no processo antigo:	Número de ocorrências:	%
Utilização de versão antiga do formulário	292	16%
Motivo/Natureza errado (campo 8.2)	191	10%
Inversão do campo Compra/Venda (campo 1)	158	8%
Enquadramento errado da operação (IOF errado) (campo 7.5)	148	8%
Ausência da cláusula sobre o Imposto de Renda (campo 10.1)	87	5%
Preenchimento errado do IOF (Exportação de serviços)	85	5%
Canal bancário divergente (campo 8.4)	62	3%
Ausência da rubrica (8.3)	61	3%
Taxa de câmbio preenchida incorreta (campo 7.2)	53	3%
Valor da moeda estrangeira/reais divergente da escrita por extenso	42	2%
Contrato faltando páginas	41	2%
Sem assinaturas/abono	31	2%
VET preenchido errado (Campo 7.7)	19	1%
Total de pendências no formulário de câmbio:	1270	67%
Outros tipos de pendências além do contrato de câmbio:	613	33%
TOTAL DE PENDÊNCIAS:	1883	100%

Observando a tabela 1, pode-se inferir que para o período de amostragem (22 dias úteis), a quantidade de erros gerados pelo mal preenchimento do contrato de câmbio corresponde a quase 70% do total de pendências geradas pelas operações de câmbio contratadas por pessoas físicas. Além de não gerar qualquer receita para a instituição financeira, essa operação tem alto custo operacional e acaba gerando risco de imagem para a instituição, visto que o cliente não terá sua operação concretizada.

A partir deste contexto iniciou-se um estudo para a implementação de um novo sistema que mitigasse, pelo menos parte dos erros mencionados na tabela 1.

A melhor alternativa encontrada foi o desenvolvimento de um software que auxiliasse o gerente durante a contratação da operação, tornando o processo mais eficaz. Diante das opções analisadas, optou-se, inicialmente, pelo desenvolvimento do programa em uma plataforma que fosse intuitiva e de uso comum por todo banco: o Excel da Microsoft. Como já é um programa utilizado diariamente pelos funcionários, foram evitados os gastos com treinamento dos usuários e com aquisição de novas licenças para utilização do programa. O custo mais baixo foi o fator de maior impacto para a escolha da linguagem de programação e ambiente de desenvolvimento a serem utilizados para o desenvolvimento do software.

Capítulo 2 – Desenvolvimento do Software

2.1 Análise de Sistemas

Logo após a definição da ferramenta que seria utilizada para o desenvolvimento do software, buscou-se adotar uma “filosofia” de desenvolvimento, também conhecida como paradigma, que mais se adequasse ao ambiente de trabalho e com os recursos disponíveis. Dentre os paradigmas conhecidos, foram analisados os prós e os contras dos três principais modelos de processos descritivos mais utilizados pelos desenvolvedores: cascata, de prototipagem e incremental.

“Originalmente, modelos de processo prescritivo foram propostos para trazer ordem ao caos existente na área de desenvolvimento de software.” (PRESSMAN, 2005). Os modelos prescritivos, como o próprio nome já diz, são um conjunto de métodos que proporcionam ao desenvolvedor construir um software de forma organizada e eficiente. Portanto, a escolha do modelo prescritivo, dentre os vários existentes, que mais se adequa ao projeto a ser desenvolvido, é parte essencial para o sucesso do mesmo.

- **Modelo em cascata:** Neste modelo, o desenvolvedor possui bem definidas todas as etapas do processo. Sua principal característica é que para o início da etapa seguinte, deve-se obrigatoriamente terminar a etapa anterior. Este processo torna o desenvolvimento mais estático, ou seja, é menos suscetível a mudanças nas etapas anteriores do processo. No entanto, por ser simples e eficiente é o modelo mais utilizado atualmente.
- **Modelo de prototipagem:** A principal característica deste modelo é a velocidade em que versões intermediárias ou protótipos do projeto final são desenvolvidas e publicadas. As etapas não são bem definidas e sequenciais como no modelo em cascata,

mas sim cíclicas. Ao terminar o protótipo inicial, o desenvolvedor e os responsáveis por homologar e testar o software avaliam o que pode ser melhorado para a versão seguinte e refazem todo o projeto novamente. É bastante utilizado quando não se sabe ao certo todas as funções e processos que o software deve possuir. Por esta razão, são lançadas diversas versões intermediárias até que se atinja um resultado satisfatório com o produto final.

- **Modelo incremental:** Este modelo é basicamente uma combinação entre os modelos em cascata e de prototipagem. Quando opta por utilizá-lo, o desenvolvedor já conhece de antemão o núcleo ou as principais funções que o software deverá ter. Dessa forma a parte inicial de análise e descrição inicial do projeto são bastante comuns ao modelo em cascata. No entanto, após o desenvolvimento do núcleo central de funções, são lançadas algumas versões intermediárias com melhorias ou funcionalidades incrementais que são observadas após o lançamento das versões iniciais do projeto.

Para o desenvolvimento da ferramenta de contratação de câmbio por pessoas físicas, optou-se por utilizar o modelo incremental. Este modelo se adequou ao software a ser desenvolvido porque, inicialmente, o desenvolvedor já conhecia alguns dos requisitos e funcionalidades mínimos que o software deveria possuir, visto que todo o desenvolvimento foi focado em mitigar os erros existentes no processo anterior. No entanto, a medida em que o software foi testado e homologado por outras áreas, surgiram sugestões de melhorias que foram incorporadas às versões posteriores do software. Outra característica do modelo incremental que tornou sua escolha a mais viável para o desenvolvimento do projeto foi a modularidade. Essa característica permite que algumas funcionalidades adicionais sejam programadas e testadas de maneira independente do núcleo principal do programa, o que facilita o desenvolvimento dessas funcionalidades à

medida em que a necessidade for identificada. A figura 3 ilustra as principais etapas do modelo incremental e uma breve descrição de cada uma delas.

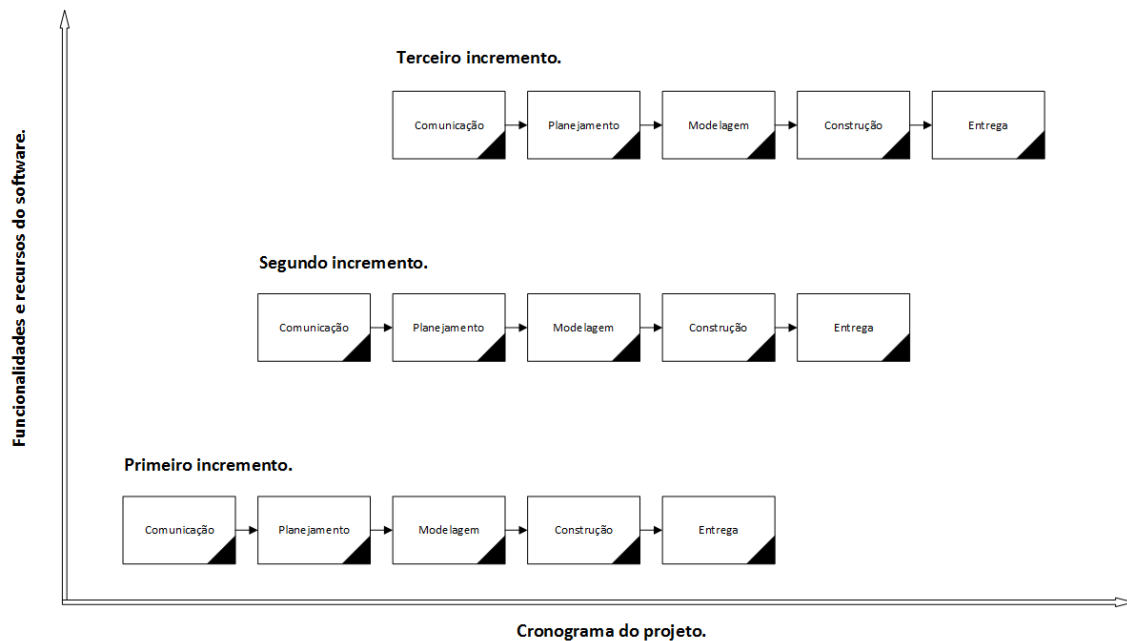


Figura 3 - Etapas de desenvolvimento do modelo em cascata. Fonte: (PRESSMAN, 2011)

2.1.1 Introdução ao projeto

Em concordância com as diretrizes da circular 3.691, as operações de câmbio com volume superior a US\$ 10.000 (dez mil dólares dos Estados Unidos) ou o seu equivalente em outras moedas, devem ser formalizadas por um contrato de câmbio segundo o molde exposto no anexo I à circular em questão. Este molde também se encontra disponível no anexo A deste trabalho.

“Art. 55.

V - Operações de compra e de venda de moeda estrangeira de até US\$10.000,00 (dez mil dólares dos Estados Unidos) ou do seu equivalente em outras moedas.

Parágrafo único.

Nas operações de compra e de venda de moeda estrangeira com clientes nas quais não houver formalização do contrato de câmbio, é obrigatória a entrega ou a disponibilização ao cliente, de forma imediata e sem ônus, de comprovante para cada operação realizada, contendo pelo menos a identificação das partes e a indicação da moeda estrangeira, do fato natureza da operação, da taxa de câmbio, dos valores em moeda estrangeira e em moeda nacional, bem como do VET. ”

Embora as operações de valor inferior a US\$10.000 tenham a formalização da operação por contrato de câmbio dispensada, as instituições financeiras são obrigadas a fornecer aos clientes, ao menos um comprovante com os principais detalhes da operação, como VET (valor efetivo total), valor da operação em moeda nacional e moeda estrangeira, taxa de câmbio, contrapartes envolvidas na transação e etc. Dessa forma, muitos bancos adotaram um contrato de câmbio padrão para todas as operações de câmbio, sejam elas inferiores ou superiores ao limite estipulado de dez mil dólares.

As pessoas físicas que desejarem efetuar ou receber pagamentos em outras moedas devem comparecer até sua agência de origem e solicitar a contratação da operação. O gerente ou assistente habilitado deverá fazer o preenchimento manual do contrato de câmbio e solicitar que o cliente o assine. Dessa forma, o objetivo do presente trabalho é facilitar o preenchimento do contrato e diminuir os erros que impedem que as operações sejam efetivadas.

2.1.2 Identificação da necessidade

O primeiro passo para o início do desenvolvimento de um software é a identificação da necessidade do mesmo. Em um projeto que envolve a melhoria de um

processo já existente, faz-se necessária a identificação das falhas no processo anterior e o levantamento das possíveis soluções para os problemas identificados.

Atualmente, o processo de contratação de câmbio pronto por pessoas físicas nos grandes bancos de varejo pode ser dividido em duas diferentes etapas. A primeira etapa ocorre diretamente nas agências e a segunda ocorre na sede operacional do banco. O termo front office é utilizado para designar as áreas que possuem contato direto com os clientes e são responsáveis pela geração de novos negócios. Já o back office, é responsável pelo processamento de todas as operações do banco. A seguir estão detalhadas as atribuições de cada uma das áreas (especificamente para o produto câmbio):

Etapa do Front office para contratação de câmbio por pessoa física:

- I. Inicialmente o cliente deve comparecer até a sua agência de origem e manifestar a intenção pela contratação do produto.
- II. O funcionário do banco deverá identificar a necessidade do cliente e classificar a operação em uma das mais de duzentas naturezas disponibilizadas pelo próprio Bacen.
- III. Após o enquadramento da operação na natureza adequada, o funcionário deverá solicitar ao cliente os documentos que respaldam a operação caso o volume seja superior a US\$3.000 ou valor equivalente em outras moedas. Cabe a cada banco estipular a documentação que deve ser apresentada para respaldar tais operações³.

³ Embora seja de responsabilidade da própria instituição financeira determinar quais documentos são necessários para respaldar cada operação de câmbio, os bancos tomam decisões conjuntas através das reuniões da FEBRABAN.

- IV. Com os documentos em boa ordem, o funcionário deverá preencher o contrato de câmbio manualmente com todos os detalhes da operação e solicitar a assinatura do cliente.
- V. Após o abono da assinatura, o funcionário deve encaminhar o contrato e os documentos que respaldam a operação para a área de processamento.

Etapas do Back office para contratação de câmbio por pessoa física:

- I. Todas as operações de câmbio geradas por todas as agências do país e pelas mesas de câmbio são recepcionadas pelos analistas do Back office. Os funcionários são divididos de acordo com os tipos de operação de câmbio que irão analisar (Exportação, Importação ou Financeiras).
- II. Cada analista é responsável por avaliar e confrontar o contrato de câmbio com os documentos recebidos. Se os documentos apresentados forem suficientes para respaldar a operação pretendida e o contrato de câmbio estiver preenchido corretamente, a operação é validada.
- III. Após ser legitimada, a operação é registrada no SISBACEN e o crédito ou débito, assim como as demais tarifas da operação, são efetuados na conta corrente do cliente.

Pode-se observar que o preenchimento correto do contrato de câmbio é parte crucial para a efetivação e legitimação da operação junto ao Banco Central. No entanto, para o correto preenchimento do contrato, faz-se necessário que o funcionário que receba a solicitação do cliente seja um especialista em câmbio, devido ao elevado número de naturezas disponibilizadas pelo Bacen (mais de duzentas) e à especificidade da documentação demandada para cada operação.

Diante de um cenário de ineficiência do processo atualmente utilizado, optou-se pelo desenvolvimento de um software que auxilie o operador de câmbio e facilite o preenchimento do contrato de câmbio. Com a implementação do novo sistema, espera-se que a quantidade de erros apresentados na tabela 1 seja reduzida.

2.1.3 Estudo da viabilidade

Todo projeto de engenharia, independente do ramo a qual pertença, está sujeito a limitações operacionais, econômicas, técnicas e até mesmo políticas. Antes de iniciar qualquer projeto de engenharia de software em especial, é essencial que se faça uma análise detalhada de todos os cenários que envolvam a viabilidade do projeto, visto que, geralmente, são projetos de custo elevado, tanto financeiro quanto técnico e operacional. O mal dimensionamento de algum quesito ou de alguma função do software que impacte diretamente na viabilidade do produto final pode gerar um prejuízo da ordem de milhões de reais.

Pode-se observar que a história do desenvolvimento de software é marcada por alguns eventos catastróficos que geraram prejuízos milionários para as companhias detentoras de tais programas. Um exemplo que foi amplamente divulgado nas mídias, foi o caso de uma operadora do mercado financeiro, especializada em negociar ações com instituições financeiras (*Institutional stock-trading*), ou seja, era responsável por comprar e vender ações para instituições financeiras como bancos, corretoras, gestoras de fundos e etc. Em Agosto de 2012, a KCG (Knight Capital Group) era responsável por cerca de 20% de todas as negociações registradas nas principais bolsas de valores do mercado americano e internacional, a NYSE (New York Stock Exchange) e a NASDAQ (*National Association of Securities Dealers Automated Quotation*). Devido a um erro em um dos diversos algoritmos de negociação automática utilizados pela companhia, algumas ações

foram automaticamente compradas a preços superiores ao valor de mercado das mesmas. Durante 30 minutos, foi acumulado um prejuízo de quase 500 milhões de dólares, o que equivalia a quase quatro vezes o resultado líquido da companhia no ano anterior. Após o ocorrido, as ações da própria companhia caíram por quase 70% e o prejuízo total causado por um erro de software atingiu quase 1 bilhão de dólares. Fonte: <http://money.cnn.com/2012/08/09/technology/knight-expensive-computer-bug>.

Para evitar situações como a citada acima, cada etapa do desenvolvimento de software deve ser precisa e bem calculada, buscando prever e solucionar qualquer movimento que possa levar ao colapso do programa ou até mesmo fim da empresa que o utiliza.

O estudo da viabilidade de um projeto, pode ser dividida em três: viabilidade operacional, político econômica e técnica-legal.

2.1.3.1 Viabilidade operacional

A viabilidade operacional analisa todos os requisitos funcionais do software, de forma a garantir que a necessidade do consumidor final seja atendida em todas as funções desenvolvidas. Diversos aspectos como: performance do sistema, confiabilidade nas informações a serem transmitidas, capacidade técnica dos operadores do sistema, segurança nos dados armazenados, capacidade de se fazer manutenção no software e etc., devem ser verificadas antes mesmo do início do projeto propriamente dito. Cada hora investida no desenvolvimento de software em si é bastante dispendiosa para as empresas. Portanto, antes de se iniciar o desenvolvimento de qualquer projeto de software, deve-se certificar de que todos os requisitos mínimos para operacionalização do mesmo sejam atendidos.

Para o projeto em questão, desenvolvimento de ferramenta que auxilie durante o preenchimento de um contrato de câmbio, os seguintes pontos foram determinantes para a continuidade do desenvolvimento:

a. User-friendliness: por se tratar de um projeto cujo intuito é facilitar um processo já estabelecido em uma instituição financeira, este quesito é essencial para a viabilidade do projeto. Se a utilização do programa for mais dispendiosa, tanto em tempo quanto em capacidade técnica, do que o preenchimento manual de um contrato de câmbio, o prosseguimento do projeto já seria inviabilizado. Uma das práticas utilizadas para que este quesito fosse atendido, foi a utilização de formulários simples, com poucas opções e com botões de ajuda espalhados por todo o software.

b. Comunicação com os sistemas já utilizados: em instituições financeiras, faz-se o uso concomitante de diversos softwares. Dessa forma a intercomunicação, ou seja, a capacidade de trocar informações com os softwares já utilizados pelo banco também é um fator determinante para a viabilidade de todo o projeto. Para o preenchimento dos contratos de câmbio, o software deveria ser capaz de extrair informações de uma página de internet do sistema interno do banco (intranet) e de se comunicar ou extrair resultados tabelados em softwares como o próprio Microsoft Excel.

c. Capacidade de manutenção: por se tratar de um software que tem como objetivo inicial uma solução provisória para determinado processo de uma instituição financeira, optou-se pela utilização de uma linguagem que já era de comum utilização aos demais membros da equipe. Além disso, optou-se pelo desenvolvimento modularizado do projeto, ou seja, dividiu-se o programa em diversos subprogramas que se comunicam, tornando possível a modificação de qualquer trecho de código, sem que outros módulos fossem comprometidos. A adoção das duas práticas supracitadas contribuiu para que o software fosse de fácil manutenção.

d. Confiabilidade e segurança das informações fornecidas: Dentro de uma instituição financeira, segurança é um dos quesitos mais importantes para o desenvolvimento de qualquer software. Portanto, qualquer sistema que esteja diretamente conectado à internet, deve passar por rotinas exaustivas de proteção e segurança da informação. Essas rotinas são dispendiosas e os testes de segurança também demandam bastante tempo durante o desenvolvimento do projeto. Como o cronograma do projeto era apertado e não foram designadas equipes de programadores para o desenvolvimento do mesmo, optou-se por um sistema de comunicação exclusivo com a intranet da instituição, que já possui diversas restrições e equipes de segurança responsáveis pelo monitoramento de todas as informações que estão disponíveis em sua rede interna.

2.1.3.2 Viabilidade político-econômica

Além da viabilidade que envolve a construção do software em si, existem um outro quesito que é tão importante quanto: a viabilidade político-econômica. Assim como qualquer outro tipo de trabalho, o desenvolvimento de software envolve gastos e muitas vezes, a aprovação destes gastos está sujeita à aprovação de um gestor ou até mesmo de um departamento financeiro inteiro (para projetos grandes que alterem diversos processos dentro da instituição). Por esta razão, o suporte e aprovação do responsável ou da área responsável é essencial para que o projeto saia do rascunho.

O primeiro passo para o desenvolvimento deste projeto foi a identificação de um processo ineficaz dentro da própria instituição. Assim que identificado, foi iniciado um processo de *brainstorming* para a proposição de diversas ideias que fossem capazes de suprir os erros encontrados no processo. O segundo passo foi a realização de uma apresentação para os *stakeholders*, ou tomadores de decisão, para apresentar as possíveis soluções encontradas. Neste ponto, a capacidade política para convencer os gestores da

real necessidade da implementação do projeto foi essencial para o início da implementação do mesmo.

Concomitante à viabilidade política, veio a viabilidade econômica, pois os gestores das áreas afetadas, só aprovaram o projeto após a verificação dos custos envolvidos e do potencial retorno após a implementação do mesmo. Nesta etapa foram destacados na Tabela 1, todos os tipos de erros que poderiam ser corrigidos com a aprovação do projeto. A tabela 2 destaca em amarelo os erros que, serão sanados com o implemento da nova ferramenta. Caso a ferramenta consiga corrigir todos os erros destacados, o total de erros gerados por preenchimento do contrato de câmbio poderá diminuir em mais de 50%, impactando positivamente no resultado operacional da instituição financeira.

Tabela 2 - Destaque para os tipos de erros que serão corrigidos pelo novo processo.

Principais pendências/erros encontrados no processo antigo:	Número de ocorrências:	%
Utilização de versão antiga do formulário	292	16%
Motivo/Natureza errado (campo 8.2)	191	10%
Inversão do campo Compra/Venda (campo 1)	158	8%
Enquadramento errado da operação (IOF errado) (campo 7.5)	148	8%
Ausência da cláusula sobre o Imposto de Renda (campo 10.1)	87	5%
Preenchimento errado do IOF (Exportação de serviços)	85	5%
Canal bancário divergente (campo 8.4)	62	3%
Ausência da rubrica (8.3)	61	3%
Taxa de câmbio preenchida incorreta (campo 7.2)	53	3%
Valor da moeda estrangeira/reais divergente da escrita por extenso	42	2%
Contrato faltando páginas	41	2%
Sem assinaturas/abono	31	2%
VET preenchido errado (Campo 7.7)	19	1%
Total de pendências no formulário de câmbio:	1270	67%
Outros tipos de pendências além do contrato de câmbio:	613	33%
TOTAL DE PENDÊNCIAS:	1883	100%

A análise econômica consiste na avaliação de todo tipo de custo envolvido com o projeto do software e à manutenção do mesmo. Dessa forma, pode-se estabelecer, a priori, custos relacionados à aquisição de equipamentos, à iniciação, ao projeto em si e custos permanentes.

Os custos relacionados à iniciação são aqueles necessários para o início da utilização do software, custo de instalação de equipamentos de comunicação como linhas telefônicas ou linhas de dados e custo de pessoal responsável por manter o sistema operante. Já os custos relacionados ao projeto, envolvem o preço dos aplicativos comprados, as modificações do software para adaptações locais e o treinamento de pessoas capazes de oferecer suporte ao cliente. Para o projeto em questão, não houveram custos com compra de equipamentos e aquisição de softwares, visto que foram utilizados os mesmos recursos já disponíveis no ambiente de trabalho. Como custo de iniciação, houve apenas o custo das horas do funcionário exclusivamente dedicado ao projeto.

Os custos permanentes são aqueles relacionados à manutenção do sistema como um todo, ao aluguel de linhas e estruturas e a depreciação do hardware envolvido no projeto. Como o projeto abrange apenas software, não foram estipulados custos relacionados à manutenção de hardware e nem depreciação do mesmo. Pode-se elencar como custo permanente do projeto, o treinamento constante dos funcionários que irão utilizar o programa e o eventual tempo gasto com a manutenção do mesmo, visto que os softwares, de maneira geral, tendem a apresentar um número maior de erros durante a fase de desenvolvimento e após um determinado ciclo de vida. A figura 4 ilustra a probabilidade de ocorrência de erros ao longo do ciclo de vida dos softwares em geral. Por ser um software de simples utilização, para a capacitação dos funcionários que irão utilizar a ferramenta, optou-se pela disponibilização de um vídeo explicativo e uma

apresentação com o passo a passo contendo o detalhamento de cada um dos recursos do programa na própria página interna da instituição financeira.

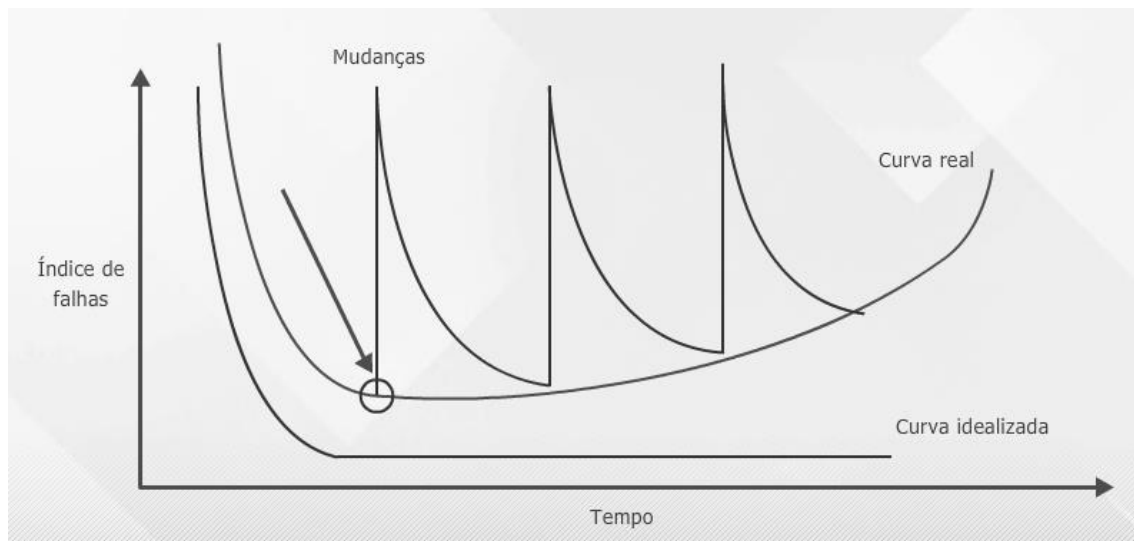


Figura 4 – Probabilidade de ocorrência de falhas ao longo do tempo de desenvolvimento de software. Fonte: (PRESSMAN, 2011)

2.1.3.3 Viabilidade técnica-legal

A viabilidade técnica consiste na análise de todos os requisitos técnicos que viabilizam ou impossibilitam a execução do projeto. Para o projeto em questão, primeiramente foi analisado o cronograma do projeto, por se tratar de uma solução provisória e não desenvolvimento completo de um novo sistema, o cronograma do projeto foi bastante reduzido e todos os processos tiveram que ser adequados para o prazo de 90 dias. Para cumprir com o prazo solicitado, foram adotados como ambiente de desenvolvimento e linguagem o Microsoft Excel e o Visual Basic, respectivamente.

Por ser um software já consagrado e utilizado em larga escala nas instituições financeiras, o Excel já possui a vantagem de ser conhecido pela maioria dos colaboradores, o que contribui para a viabilização técnica do projeto e reduz os gastos com treinamento dos usuários. Além disso, as interfaces simplistas do programa são ideais para o desenvolvimento de projetos específicos para determinada aplicação, neste

caso o desenvolvimento de uma ferramenta que auxilie durante o preenchimento de um contrato de câmbio.

A viabilidade legal consiste no estudo da legislação que rege os tópicos envolvidos no projeto. Como o produto câmbio é regulamentado e fiscalizado pelo Bacen, todo o software teve de seguir as circulares 3.689, 3.690, 3.691 e demais circulares que aditam estas três, dadas como base para a legislação de câmbio.

Um dos principais requisitos funcionais do programa é fornecer ao operador as principais naturezas de câmbio contratadas por pessoas físicas. São apresentadas ao operador as 20 principais naturezas, ordenadas por maior ocorrência, seguidas pela alíquota do IOF (Imposto sobre Operações Financeiras) incidente sobre cada operação. Para apresentar ao operador a alíquota correta do IOF sobre cada uma das naturezas fornecidas pelo Bacen, foi consultado o decreto 6.306 / 2007 que dispõe das regras para tributação do IOF sobre cada uma das operações financeiras e a instrução normativa IN 1.645 e da lei 13.315 de maio de 2016, que dispõe de algumas regras especiais para a incidência do imposto de renda sobre operações de câmbio.

2.1.4 Diagrama de arquiteturas

O diagrama de arquiteturas é a primeira representação gráfica do projeto do software. Nele estão representadas as entidades que participarão do funcionamento do mesmo e também as delimitações ou fronteiras entre as trocas de informações com o sistema. Conforme a ilustração da figura 5, o software em si troca informações com apenas três entidades: o usuário, a rede de computadores interna do banco (através de uma query) e com o visualizador de PDF já instalado no computador.

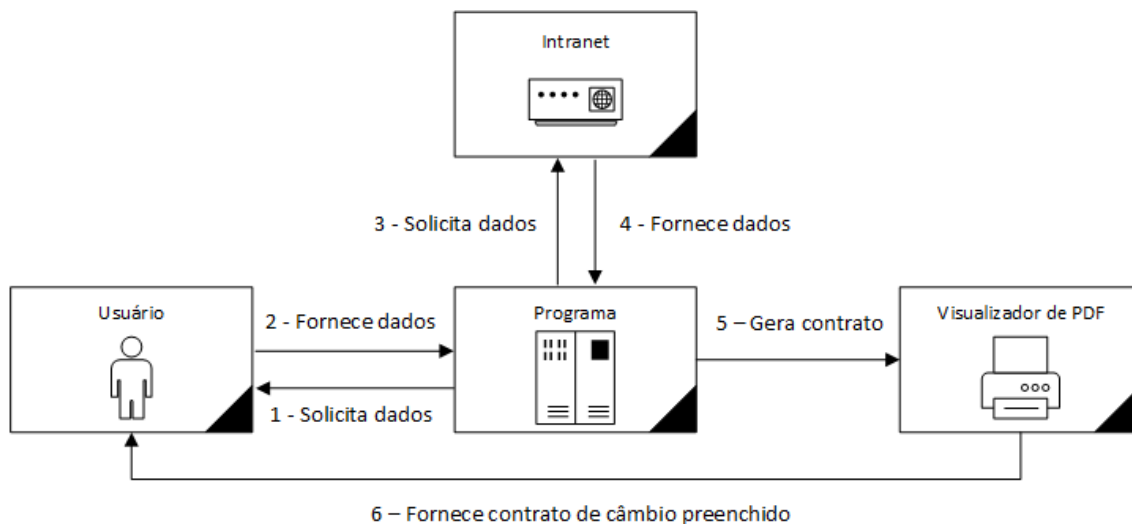


Figura 5- Delimitação da comunicação entre o software e as demais entidades envolvidas no sistema.

2.2 Análise de requisitos

Esta etapa do projeto visa detalhar os requisitos necessários para o funcionamento do programa que auxilia a contratação de câmbio por pessoas físicas. Nele estão contidos apontamentos sobre o escopo do programa em questão, expondo suas características e funcionalidades, uma relação de siglas presentes neste documento e as referências que forneceram o embasamento necessário para o desenvolvimento do software.

2.2.1 Requisitos funcionais

2.2.1.1 Requisito funcional I – Abertura

Ao executar o programa, será apresentada ao usuário a capa do programa com um único botão: “Iniciar cotação”. A única função deste requisito é informar ao funcionário que ainda não conhece a ferramenta o objetivo da mesma. A figura 6 apresenta a tela inicial do programa.



Figura 6- Tela de abertura do programa.

2.2.1.2 Requisito funcional II – Preenchimento de dados do funcionário

Nesta etapa é que se iniciará o preenchimento dos dados propriamente ditos. Primeiramente, a ferramenta já retira diretamente da máquina do usuário dois dados: nome e e-mail. Esta ferramenta se dá através da integração da linguagem VBA com os demais programas que compõe o Microsoft Office, neste caso o próprio Outlook. No terceiro campo o usuário deverá preencher com seu telefone para contato e então, no último campo, deverá selecionar se está operando diretamente de uma agência ou da central de atendimento de câmbio.

Figura 7 - Terceira tela da ferramenta, responsável por armazenar os dados do usuário.

O telefone do usuário é importante para que o mesmo seja facilmente localizado pelo Back Office em casos de erros durante o preenchimento do formulário.

2.2.1.3 Requisito funcional III – Preenchimento de dados do cliente

O quarto requisito funcional é um dos mais importantes da ferramenta. É aqui que o funcionário deverá preencher todos os dados da contraparte que deseja contratar o serviço. Este requisito funcional possui algumas travas de segurança adicionais, que são ilustrados pelas figuras de 8 a 13.

Formulário 0795

Dados do Cliente
Preencha todos os campos obrigatórios. Se tiver alguma dúvida, clique no botão "Ajuda".

NOME: ENDEREÇO:

CPF / PASSAPORTE: CIDADE:

TELEFONE CELULAR: UF/PAÍS:

E-MAIL:

SEGMENTO: ☐ Seg 1 ☐ Seg 2 ☐ Seg 3 ☐ Seg 4 ☐ FOPA 1 ☐ FOPA 2 ☐ FOPA 3

AGÊNCIA: CONTA CORRENTE:

Cliente está habilitado para operações de câmbio ?

v20161128

Figura 8 – Quarta tela da ferramenta, responsável por armazenar os dados do usuário.

- A primeira trava, ao ser acionando o botão “Próximo”, verifica se todos os campos anteriores foram preenchidos. Caso falte algum campo, a ferramenta retornará uma mensagem de erro ao usuário. (Figura 8).
- A segunda trava, se certifica de que o funcionário verificou se o seu cliente está apto para operar com câmbio através do conheça seu cliente. O conheça seu cliente, é uma das maneiras de se certificar de que a contraparte envolvida na

operação de câmbio não tenha práticas ilícitas como: ocultação, lavagem de dinheiro ou evasão de divisas. (Figura 9).

- Além de verificar se todos os campos foram digitados corretamente, a ferramenta também se certifica de que um dos segmentos do cliente foram selecionados. Um erro bastante comum em versões anteriores dos formulários era o envio de contratos com o segmento do cliente não preenchido. (Figura 10).
- A quarta trava, está no campo de CPF do cliente. Caso o atendente tente preencher com mais de 11 dígitos, a ferramenta irá acusar um erro com uma mensagem de número inválido de caracteres. (Figura 11).
- O campo telefone celular, garante que o funcionário esteja digitando apenas números, caso contrário, a seguinte mensagem de erro será exibida. (Figura 12).
- O campo e-mail, por sua vez, garante que o caractere “@” foi inserido no campo, evitando que o funcionário se confunda com outros dados ao preencher este campo. (Figura 13).

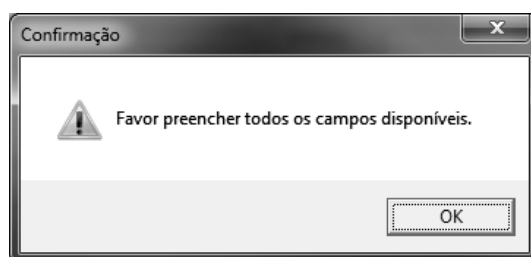


Figura 9 - Primeira trava de verificação.

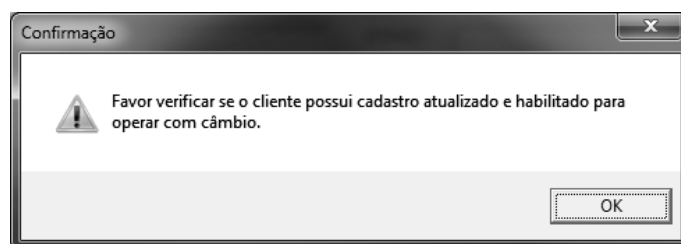


Figura 10 - Segunda trava de verificação.

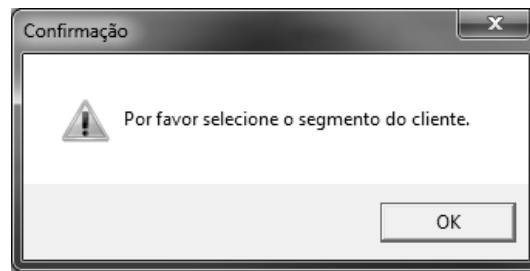


Figura 11 - Terceira trava de verificação.

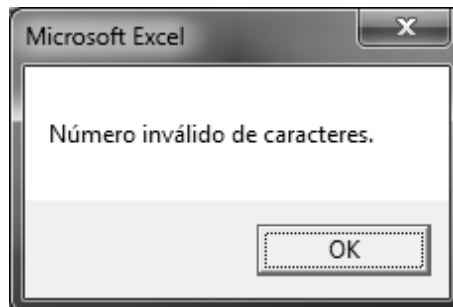


Figura 12 - Quarta trava de verificação.

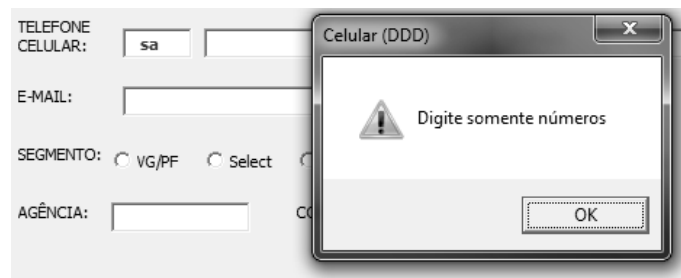


Figura 13 - Quinta trava de verificação.

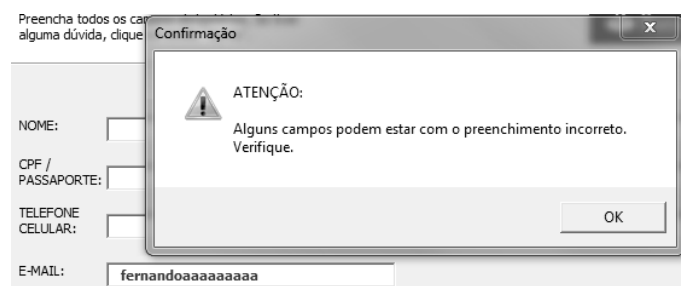


Figura 14 - Sexta trava de verificação.

Por fim, o botão “consulta FOPA”, exibe uma tela com as empresas que possuem parceria em folha de pagamento com o banco e, portanto, possuem tarifas e taxas

diferenciadas na contratação de operações de câmbio. Após ter preenchido todos os dados corretamente, o usuário é levado à próxima tela de navegação.

2.2.1.4 Requisito funcional IV – Enquadramento da natureza da operação

A tela de enquadramento de natureza da operação é responsável por auxiliar o funcionário a pré-classificar a transação do cliente em uma das mais de 200 naturezas de câmbio disponibilizadas pelo Bacen. Como a ferramenta se destina somente à contratação de câmbio por pessoas físicas, várias dessas naturezas podem ser imediatamente excluídas da lista, pois algumas se destinam exclusivamente a pessoas jurídicas e outras às instituições financeiras.

Inicialmente, de maneira simplista, o funcionário deve selecionar o tipo de contrato de câmbio: envio ou recebimento. O excesso de simplicidade neste quesito foi necessário porque tanto os clientes quanto os funcionários, ainda ficam incertos ao preencher apenas compra ou venda, visto que não se sabe exatamente se a perspectiva adotada é a visão do banco ou a visão do cliente. Ou seja, se o banco vende moeda estrangeira, o cliente compra e vice-versa.

Ao clicar na seta de natureza da operação, é exibida uma *drop-down list* com as principais naturezas utilizadas por pessoas físicas por ordem de ocorrência, ou seja, aquelas que são mais utilizadas serão as primeiras a serem mostradas. Caso o funcionário sinta dificuldades ou não consiga exprimir a situação do cliente em uma das naturezas disponibilizadas, o mesmo pode selecionar “Outros...”. Ao selecionar esta opção, o quadro abaixo que até então estava inativo irá se habilitar e o funcionário poderá descrever brevemente a situação solicitada pelo cliente.

Formulário 0795

Natureza da Operação
Preencha todos os campos obrigatórios. Se tiver alguma dúvida, clique no botão "Ajuda".

TIPO DE CONTRATO DE CÂMBIO:

☒ Cliente está recebendo dinheiro do exterior. ☐ Cliente está enviando dinheiro para o exterior.

NATUREZA DA OPERAÇÃO:

Possui dúvidas com o enquadramento da natureza? Acesse a ferramenta: [Acessar](#)

Caso tenha selecionado : Outros, favor especificar o motivo da operação:

v20161128 [Ajuda](#) [Anterior](#) [Próximo](#)

Figura 15 - Quinta tela: enquadramento da natureza da operação.

2.2.1.5 Requisito funcional V – Cláusulas contratuais

Para as remessas financeiras (excluindo as que envolvem importação de mercadorias) pode haver a incidência do imposto de renda e a alíquota varia com a natureza da operação. Com a edição recente da instrução normativa IN 1.645 e da lei 13.315 de maio de 2016, algumas naturezas receberam condições especiais de tributação.

Remessas destinadas às despesas médico-hospitalares, para qualquer tratamento de saúde no exterior do declarante ou de alguns de seus dependentes declarados, remessas destinadas à manutenção de dependentes declarados no exterior e remessas com fins educacionais, científicos ou culturais recebem isenção da tributação do imposto de renda. Sobre as remessas destinadas à cobertura de gastos pessoas no exterior de pessoas físicas residentes no Brasil em viagens de turismo, negócios, serviço, treinamentos ou missões oficiais, incidirá o imposto de renda à alíquota de 6% até o limite global de R\$20.000,00 por mês por CPF. Ao excedente deste valor incidirá o imposto de renda à alíquota de 20%.

Desta forma, caso a operação solicitada pelo cliente seja enquadrada em uma natureza que possa incorrer em qualquer benefício ou condição de tributação diferenciada, será exibida a tela que contém as cláusulas contratuais diferenciadas. Este requisito funcional busca mitigar o erro: “Cláusula sobre o Imposto de Renda (Campo 10.1)” descrito nas tabelas 1 e 2. Durante o preenchimento manual, o funcionário se confundia com algumas dessas cláusulas. Com a implementação do requisito, a tela de cláusulas contratuais será disponibilizada ao usuário apenas quando a natureza da operação incorrer de alguma cláusula diferenciada, evitando confusões para as demais naturezas cambiais que não são sensibilizadas.

Figura 16 - Sexta tela: enquadramento de cláusulas especiais para incidência do Imposto de Renda.

2.2.1.6 Requisito funcional VI – Dados da operação

O sétimo requisito funcional, em conjunto com o quarto requisito, formam o conjunto de informações essenciais que deverão estar contidos no contrato de câmbio. Portanto, o correto preenchimento destes campos é essencial para o preenchimento correto do contrato.

Na primeira *drop-down list*, o funcionário deverá selecionar uma das 13 moedas que o banco disponibiliza para pessoas físicas. Logo abaixo deverá preencher o valor da operação em moeda estrangeira e, conforme exigência da circular 3.691 do Bacen, deverá escrever por escrito o valor digitado anteriormente.

A imagem mostra uma janela de software intitulada "Formulário 0795". O formulário é dividido em seções. A primeira seção, "Dados da Operação", contém instruções: "Preencha todos os campos obrigatórios. Se tiver alguma dúvida, clique no botão 'Ajuda'". Abaixo disso, há campos para "MOEDA" (menu suspenso), "VALOR" (campo de texto), "VALOR M/E (POR EXTENSO)" (campo de texto grande), "TAXA DE CÂMBIO (R\$)" (campo de texto com um botão "Obter taxa" ao lado) e "FORMA DE ENTREGA" (menu suspenso). A segunda seção, "Em R\$", contém campos para "VALOR EQUIV. (R\$)", "VALOR EQUIV. (R\$) (POR EXTENSO)", "VALOR DO IOF (R\$)", "TARIFA (R\$)" e "VALOR EFETIVO TOTAL (VET)". Na base da janela, há o número de versão "v20161128" e três botões: "Ajuda", "Anterior" e "Próximo".

Figura 17 - Sétima tela: preenchimento com os dados da operação a ser contratada.

Quando o funcionário clicar em obter taxa da intranet, o sistema irá, automaticamente, através de uma *query*, buscar na rede interna do banco, as taxas publicadas para cada moeda e para o intervalo de volume da operação selecionada. O banco fornece taxas mais favoráveis aos clientes que desejam transacionar volumes mais elevados de moeda estrangeira. Neste ponto, foi adaptada uma pequena trava de segurança para garantir que o funcionário selecionou alguma moeda antes de clicar em “obter taxa da intranet”. Caso o mesmo não o faça, será exibida a seguinte mensagem de erro:



Figura 18 - Trava de verificação de moeda.

2.2.1.7 Requisito funcional VII – Canal bancário

Figura 19 - Canal bancário para envio de remessas ao exterior.

O oitavo requisito funcional possui duas variantes. Caso o cliente esteja fazendo um pagamento para o exterior, será exibida a tela da figura 17. Caso a operação seja um recebimento do exterior, será apresentada ao funcionário uma versão simplificada da tela, ilustrada pela figura 18, com menos informações a respeito do canal bancário.

.: Formulário 0795 :.

Canal bancário no exterior
Preencha todos os campos obrigatórios. Se tiver alguma dúvida, clique no botão "Ajuda".

NOME DO BENEFICIÁRIO:

BANCO: CIDADE/PAÍS:

AG/CONTA DO BENEFICIÁRIO (OU IBAN):

CÓDIGO SWIFT DO BANCO NO EXTERIOR:

INFORMAÇÕES ADICIONAIS (CASO NECESSÁRIO):

v20161128

Ajuda Anterior Próximo

Figura 20 - Canal bancário para recebimento de remessas do exterior.

Nesta tela, assim como em algumas telas anteriores, a única trava de segurança é dada por uma mensagem de erro que será apresentada caso o funcionário deixe um dos seguintes campos em branco: nome do beneficiário, banco, cidade/país, agência/conta do beneficiário ou código *swift* do banco no exterior.

2.2.1.8 Requisito funcional VIII – Contrato de câmbio

O último requisito funcional da ferramenta é a criação do contrato de câmbio, contendo todas as informações preenchidas nos passos anteriores. Assim que o funcionário selecionar o botão “Gerar Contrato”, ilustrado pela figura 19, será gerado o

contrato de câmbio em formato PDF (*Portable Document Format*), que já virá com todas as informações preenchidas em etapas anteriores.

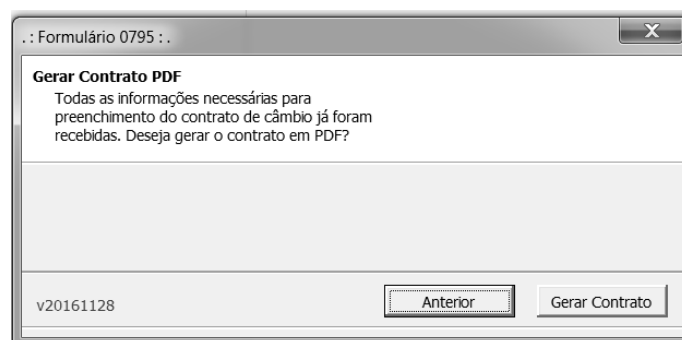


Figura 21 - Tela de confirmação e impressão do contrato de câmbio gerado.

Por fim, mesmo após já ter gerado o contrato de câmbio, o funcionário ainda pode verificar o documento apresentado e conferir se todos os dados foram inseridos corretamente. Caso note que faltou alguma informação ou caso houver algum erro no contrato, o mesmo pode descartar o arquivo gerado e selecionar a opção “Anterior” na última tela gerada pela ferramenta. Após voltar nas telas anteriores e corrigir as informações, o funcionário irá imprimir o novo contrato gerado e selecionar a opção “Encerrar”, finalizando o processo e o programa. A última tela apresentada pela ferramenta é ilustrada na figura 20. O anexo B ilustra um exemplo de contrato gerado pela ferramenta.

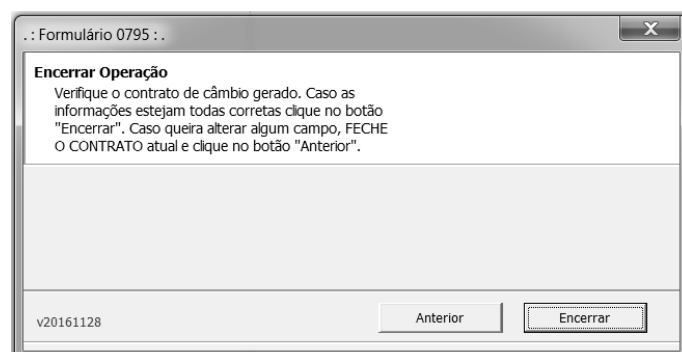


Figura 22 - Tela de encerramento do programa.

2.2.2 Diagrama de fluxo de dados

O diagrama de fluxo de dados descreve o algoritmo desenvolvido para as soluções apresentadas pelo software. Nele são apresentados os fluxogramas que definem a lógica da programação.

2.2.2.1 Diagrama Tela I – Abertura e Informações gerais

Como o paradigma utilizado permite modularidade ao código, a programação foi feita toda em módulos. Dessa forma, em cada formulário foi desenvolvido o código com as funções específicas para aquele módulo. A única interligação entre os formulários são as variáveis globais, que foram declaradas em um ambiente comum a todos os formulários, em um módulo separado.

Na tela de abertura, assim que o usuário inicia o programa, executa-se uma pequena verificação: caso as macros do Excel do usuário estejam habilitadas, é apresentada a tela de cotação; caso contrário, é apresentado um breve tutorial, ensinando o usuário a habilitar as macros para poder utilizar a ferramenta.

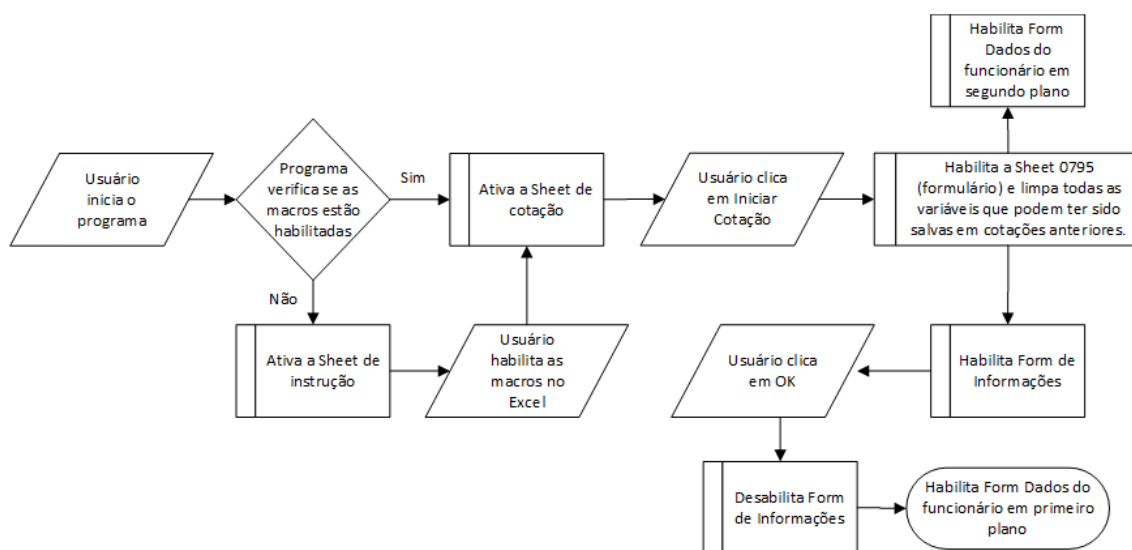


Figura 23 - Diagrama de tela I – Tela de abertura.

2.2.2.2 Diagrama Tela II – Preenchimento de dados do funcionário

A terceira tela é responsável pela aquisição e armazenamento das variáveis correspondentes aos dados do funcionário que está utilizando o programa. Dados como nome do funcionário e e-mail são retirados automaticamente do usuário do Windows e do Outlook, através de funções de interligação entre os programas. Já a variável telefone é preenchida manualmente pelo usuário. Assim que preenchido, uma função varre todos os caracteres do telefone digitado e certifica de que todos os caracteres digitados são números.

Caso todas as variáveis sejam preenchidas corretamente, o formulário armazena todas as variáveis preenchidas e conduz o usuário à próxima tela.

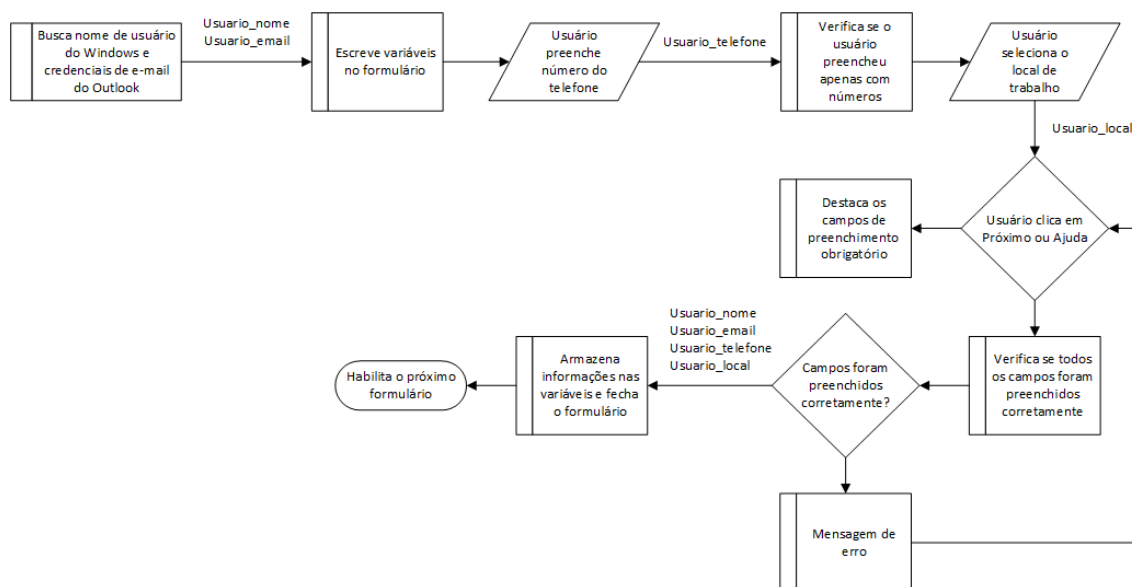


Figura 24 - Diagrama de Tela II - Preenchimento de dados do funcionário.

2.2.2.3 Diagrama Tela III – Preenchimento de dados do cliente

A quarta tela é o ambiente responsável por adquirir os dados do cliente que deseja contratar a operação. De forma semelhante ao preenchimento do telefone do funcionário,

o telefone do cliente também é verificado para que o programa se certifique que foram digitados apenas números. O campo CPF também é verificado através da função *length*, que verifica se o total de números digitados perfaz os 11 dígitos padrão da célula de identificação de pessoas físicas. O botão consulta FOPA abre um link de uma página interna do banco que contém todas as parcerias de folha de pagamentos com empresas. Algumas empresas possuem condições de tarifas e taxas diferenciadas devido à aquisição conjunta da folha de pagamentos de seus funcionários. Quando o usuário selecionar o botão próximo, o módulo irá verificar se todas as variáveis foram preenchidas adequadamente e então conduzirá o usuário à próxima tela.

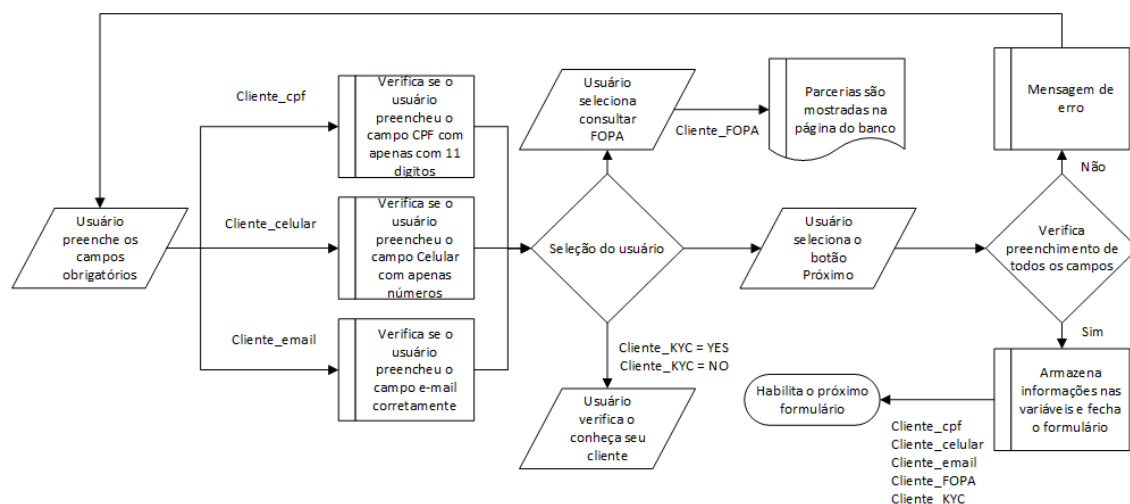


Figura 25 - Diagrama de Tela III - Preenchimento de dados do cliente.

2.2.2.4 Diagrama Tela IV – Enquadramento da natureza da operação

Inicialmente, a tela de enquadramento de natureza da operação, através de um ComboBox, fornece ao usuário algumas das principais naturezas cambiais disponibilizadas pelo Bacen.

Através de funções do tipo *hide* e *show*, o programa habilitará o quadro de informações adicionais caso a natureza escolhida seja “outros...”. Já o botão de acesso à

ferramenta de auxílio, habilita uma planilha escondida e converte a área utilizada em um arquivo PDF que é disponibilizado ao usuário que selecionar este botão. Antes de finalizar a exibição do arquivo em PDF, o programa ainda emite um aviso alertando para a necessidade da instalação de algum programa visualizador de arquivos neste formato.

O programa só permitirá que o usuário avance para a tela seguinte caso as variáveis tipo de contrato de câmbio (compra ou venda) e natureza da operação estejam preenchidas.

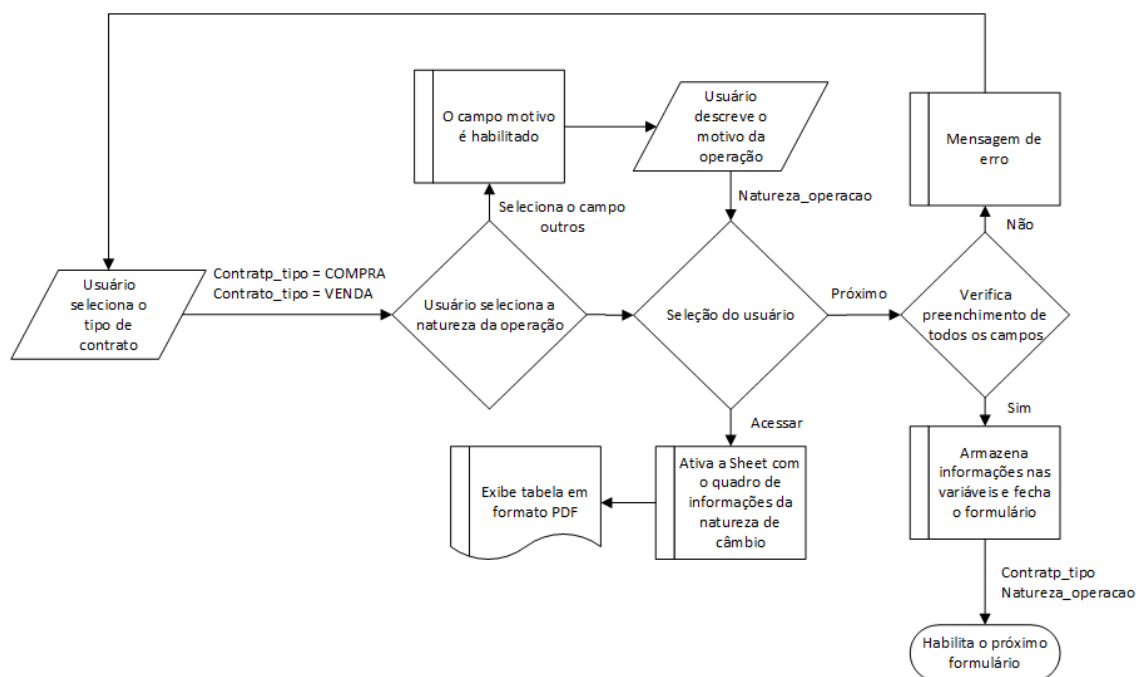


Figura 26 - Diagrama de Tela IV - Enquadramento da natureza da operação.

2.2.2.5 Diagrama Tela V – Cláusulas contratuais

Esta etapa consiste em um dos módulos mais simples do programa. Antes de inicia-lo o programa verifica, através de estruturas do tipo *if-else*, se a natureza escolhida pelo usuário possui cláusulas diferenciadas para a tributação do imposto de renda. Em caso positivo, o usuário é apresentado às cláusulas e o programa se certifica de que o mesmo irá selecionar uma das disponíveis antes de proceder para o próximo módulo.

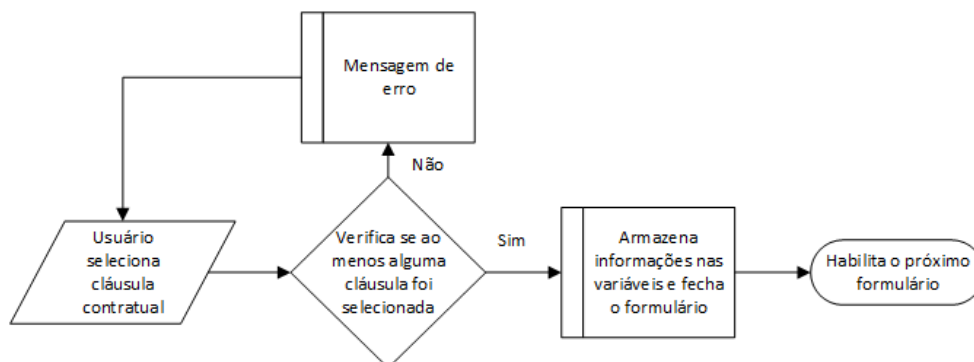


Figura 27- Diagrama de tela V – Cláusulas contratuais.

2.2.2.6 Diagrama Tela VI – Dados da operação

Inicialmente o usuário deverá selecionar a moeda utilizada para a operação de câmbio. As moedas disponíveis serão apresentadas através de um ComboBox e, assim que o usuário digitar o valor da operação, uma função de formatação irá converter o valor digitado para o padrão de contabilidade, com pontos e duas casas após a vírgula.

Quando o usuário clicar em obter taxas, um módulo separado do formulário em questão será ativado. Este módulo executará a query que fará a leitura de todas as taxas publicadas na rede interna do banco e as escreverá em forma de tabela em uma das planilhas invisíveis ao usuário. Assim que construídas, as tabelas serão formatadas para conter apenas as informações necessárias. Dependendo do valor, do segmento do cliente (funcionário ou cliente padrão) e da moeda selecionada, o programa buscará nas tabelas a taxa de câmbio que será utilizada para a operação celebrada e a escreverá no campo taxa com formatação de quatro casas após a vírgula.

Com estes dados já selecionados, o programa calculará e apresentará ao usuário as variáveis: valor equivalente em reais, valor equivalente em reais escrito por extenso⁴, valor do imposto sobre operação financeira (IOF) e valor efetivo total (VET), que corresponde à uma taxa equivalente à soma de: taxa de câmbio, tarifa cobrada pelo contrato e IOF que incidirá sobre a operação, divididos pelo valor da operação em moeda estrangeira.

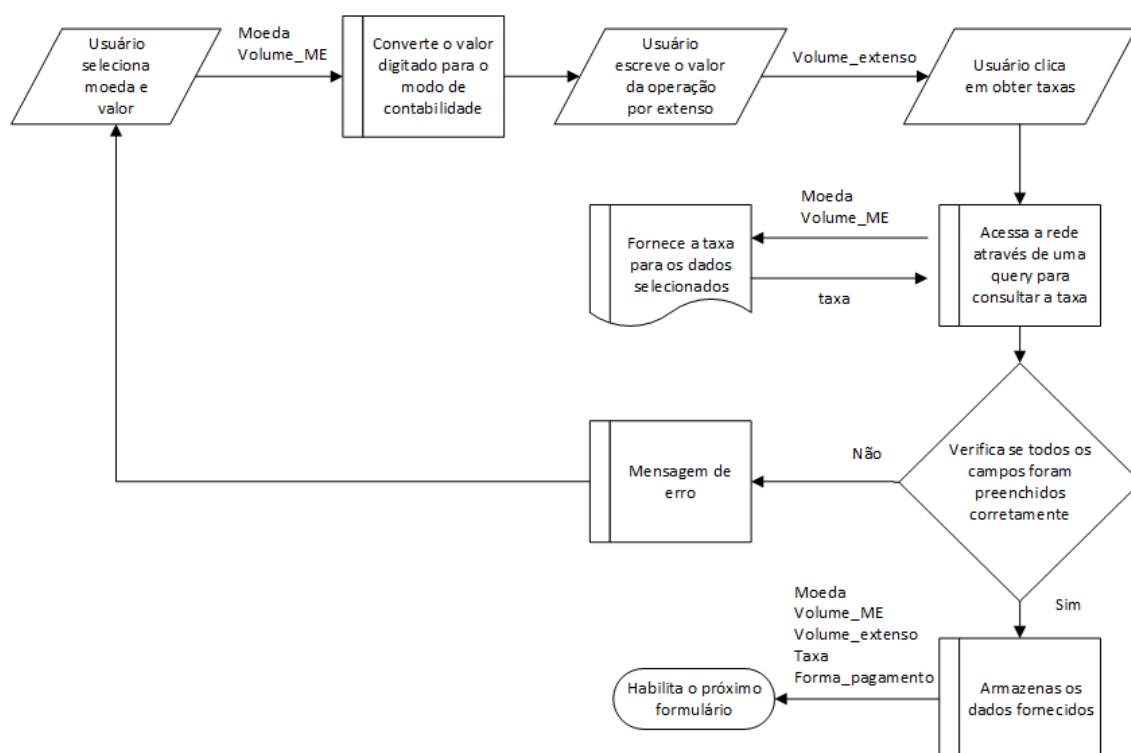


Figura 28 - Diagrama de tela VI - Dados da operação.

2.2.2.7 Diagrama Tela VII – Canal bancário

Esta etapa é responsável por coletar as informações do canal bancário ao qual o cliente está pagando ou do qual está recebendo a ordem de pagamento em moeda

⁴ Foi criado um módulo exclusivo para executar a função responsável por converter o número digitado em uma *string* que corresponda ao número escrito por extenso. Este código também consta no anexo D deste trabalho.

estrangeira. O módulo apenas armazena as variáveis digitadas nos campos exibidos nas variáveis correspondentes.

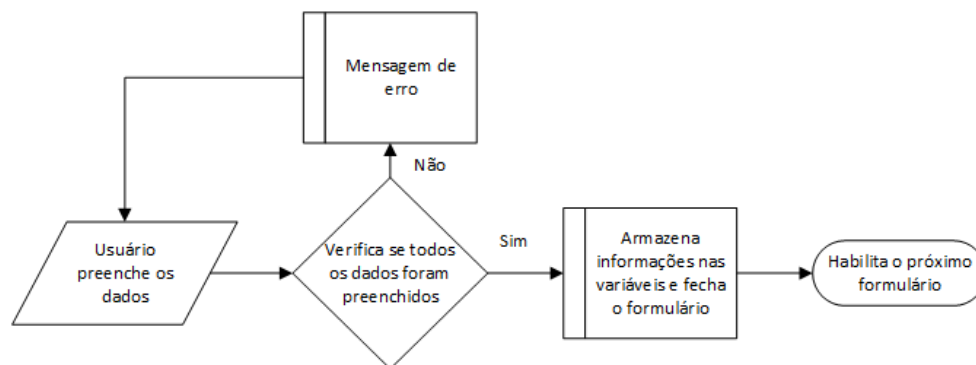


Figura 29 - Diagrama de tela VII - Dados do canal bancário.

2.2.2.8 Diagrama Tela VIII – Contrato de câmbio

A última etapa do programa é responsável por escrever todos os dados armazenados nas variáveis globais nos campos determinados do contrato de câmbio. Inicialmente será habilitada a planilha oculta que contém o molde do contrato. Após ser habilitada, todas as variáveis armazenadas serão escritas nos respectivos campos. Por último, através da função “ExportAsFixedFormat”, o programa irá converter toda a planilha que contém o contrato para o formato PDF e preparará o ambiente para o encerramento do programa, liberando todas as instâncias que ainda estiverem alocadas na memória do programa.

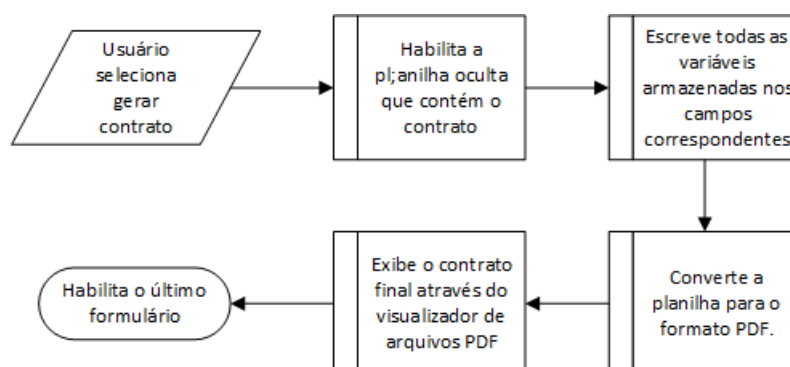


Figura 30 –Diagrama de tela VIII – Geração do contrato de câmbio.

Capítulo 3 – Testes e Manutenção

A etapa de testes e manutenção é essencial para a finalização do projeto, visto que muitos erros não previstos podem surgir durante a utilização do mesmo. Como o número de sistemas englobados pelo projeto é pequeno, foram efetuados apenas três tipos de teste para validação e homologação do programa.

Inicialmente utilizou-se o teste de assertividade do programa com o auxílio da área de produtos de câmbio do banco. Neste tipo de teste, todos os requisitos funcionais do projeto foram utilizados com todas as combinações possíveis de variáveis e foi verificado se algum dos requisitos apresentou qualquer tipo de divergência do padrão desejado. Os analistas da área de produtos ajudaram a levantar todos os tipos de erros que surgiram ao longo deste teste.

Em seguida, foram selecionadas algumas agências para realizarem os testes de campo, ou testes em pleno funcionamento. Este tipo de teste consiste em várias simulações completas de utilização do programa. Todo o processo foi acompanhado por um representante da área de produtos de câmbio do banco e por um profissional responsável pela área de tecnologia da informação. Além de assegurar que o programa funcionará nos ambientes em que será utilizado, este teste ajudou a mensurar o grau de dificuldade que os usuários finais sentiram ao deixar o processo antigo e aderir a um processo mais automatizado.

Finalmente, após a disponibilização de treinamento para toda a rede de agências que iriam utilizar o programa, foi feito um teste de stress. Neste tipo de teste, que simula o pleno funcionamento do sistema, todos os usuários habilitados utilizaram o programa simultaneamente. Desta forma pode-se analisar se não havia sobrecarga em nenhuma das

etapas do programa e se todos os contratos de câmbio eram enviados com êxito ao Back Office.

Para este tipo de sistema, que apresenta uma solução provisória para determinado problema, não foram desenvolvidas rotinas de manutenção preventiva. Todas as manutenções que ocorreram ao longo dos 6 meses em que o software funcionou plenamente, foram feitas apenas com o surgimento da necessidade da mesma.

A única grande alteração foi responsável por adequar o sistema às mudanças feitas na legislação que versam sobre as operações de câmbio. Como a nova legislação alterava algumas das naturezas cambiais disponibilizadas pelo Bacen, o projeto também teve que ser readequado.

Capítulo 4 – Análise dos resultados obtidos

Para analisar a eficiência dos resultados obtidos com o produto acabado, foram comparados os relatórios de pendências das operações de dois períodos de tempo distintos, um em que o produto ainda não havia sido implementado – junho de 2016 - e outro com a implementação do novo produto – novembro de 2016. Foram feitas as seguintes análises:

1. Comparação do percentual das operações de câmbio de pessoas físicas que eram pendenciadas ao todo nos dois períodos.
2. Comparação dos tipos de erros que surgiram em cada um dos processos, de acordo com os tópicos indicados nas tabelas
3. Comparação do percentual de perda de clientes, ou seja, aqueles que tentavam efetuar a operação uma ou duas vezes, mas que, devido a erros durante o processo, não tinham suas operações concretizadas. Para efeitos de análise, foram considerados clientes perdidos aqueles que não tentaram operar novamente com o banco após 30 dias da tentativa inicial.

A tabela 3 apresenta a análise dos erros obtidos durante o mês de novembro de 2016. Ao comparar as tabelas 1 e 3, pode-se observar que a quantidade de erros por cada requisito apontado diminuiu sensivelmente. O gráfico da figura 31 apresenta a comparação dos erros obtidos durante os dois períodos utilizando-se os dois processos diferentes.

Tabela 3 - Principais erros encontrados em um contrato de câmbio - novembro 2016.

Principais pendências/erros encontrados no processo novo:	Número de ocorrências:	%
Utilização de versão antiga do formulário	0	0%
Motivo/Natureza errado (campo 8.2)	72	13%
Inversão do campo Compra/Venda (campo 1)	95	17%
Enquadramento errado da operação (IOF errado) (campo 7.5)	120	21%
Ausência da cláusula sobre o Imposto de Renda (campo 10.1)	0	0%
Preenchimento errado do IOF (Exportação de serviços)	63	11%
Canal bancário divergente (campo 8.4)	74	13%
Ausência da rubrica (8.3)	0	0%
Taxa de câmbio preenchida incorreta (campo 7.2)	12	2%
Valor da moeda estrangeira/reais divergente da escrita por extenso	58	10%
Contrato faltando páginas	12	2%
Sem assinaturas/abono	55	10%
VET preenchido errado (Campo 7.7)	0	0%
Total de pendências no formulário de câmbio:	561	52%
Outros tipos de pendências além do contrato de câmbio:	527	48%
TOTAL DE PENDÊNCIAS:	1088	100%

Ao analisar o gráfico da figura 31, podem ser feitas as seguintes inferências:

- Com a implementação da função que verifica se o formulário utilizado corresponde ao mais recente, foi totalmente mitigado o erro causado pela utilização de formulários antigos.
- O erro causado pelo enquadramento equivocado da natureza da operação não foi totalmente mitigado, mas teve uma queda de 63%. Dessa forma pode-se comprovar que a ferramenta de auxílio embutida no programa colaborou para a instrução do funcionário e para a diminuição deste tipo de erro.
- O erro causado pela inversão dos campos compra e venda também teve uma redução significativa com a explicação mais simplista dos dois tipos de operação. Obteve-se uma queda de 40%.

- A ausência da cláusula para as operações que possuem condições especiais de tributação também foi totalmente mitigada pelo sistema, visto que o mesmo não permite que o usuário avance sem selecionar ao menos uma delas.
- O erro originado pela ausência da rubrica também foi eliminado, visto que o contrato foi alterado e redesenhado em conjunto com o departamento jurídico e tributário do banco. O novo modelo dispensa a rubrica nas cláusulas de imposto de renda e contém apenas duas páginas, com todo o conteúdo condensado, contribuindo também para a diminuição do erro causado pelo envio incompleto dos formulários.
- O cálculo do VET também foi outro erro totalmente mitigado, visto que o sistema calcula as tarifas de contrato de câmbio automaticamente. A segmentação errada do cliente e, consecutivamente, o cálculo errado da tarifa de contrato de câmbio era o principal motivo de erro no cálculo do VET.

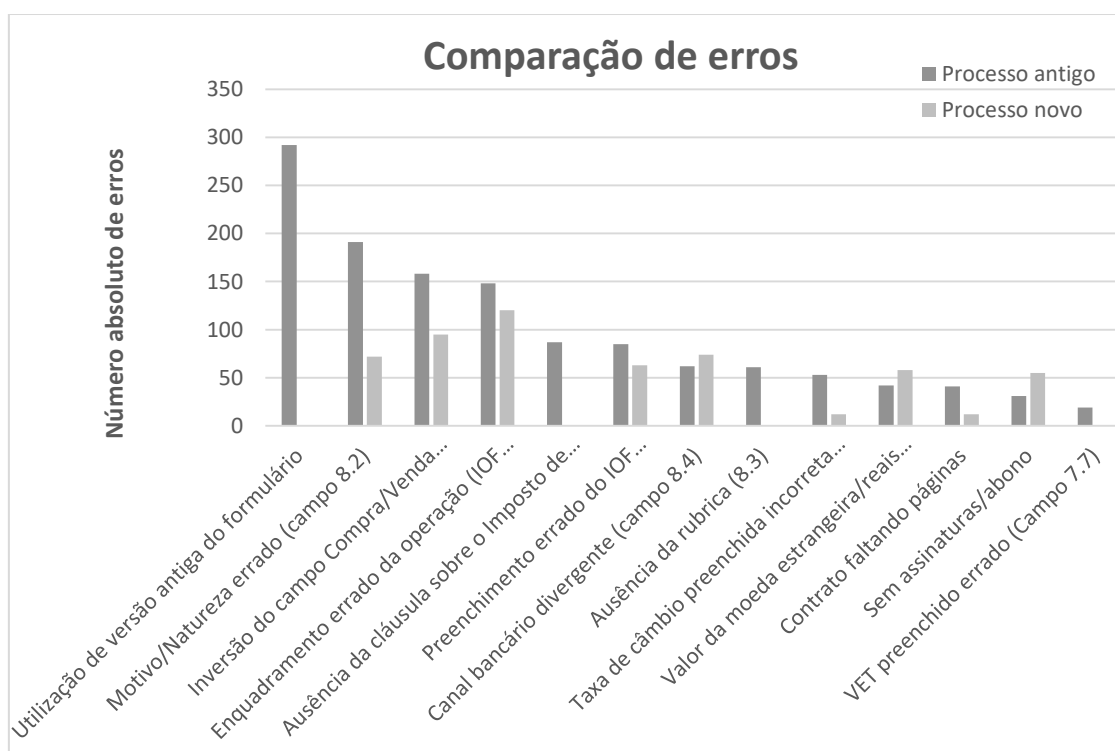


Figura 31 - Comparação do número de erros derivados do contrato de câmbio entre o processo antigo e o processo implementado.

Além das correções citadas acima, alguns outros erros, mesmo que independentes do sistema, como preenchimento do canal bancário, envio incompleto do formulário e o preenchimento do valor em moeda estrangeira por extenso, acabaram diminuindo ou mantendo nos mesmos patamares observados anteriormente. O gráfico da figura 32 ilustra que com o novo procedimento, os erros causados devido ao mal preenchimento do contrato de câmbio passaram a representar uma menor fatia do total de erros gerados pelo processo. Além disso, conforme ilustrado pela figura 33, o total de erros reduziu 43%, de 1088 para 1883 durante o período analisado.

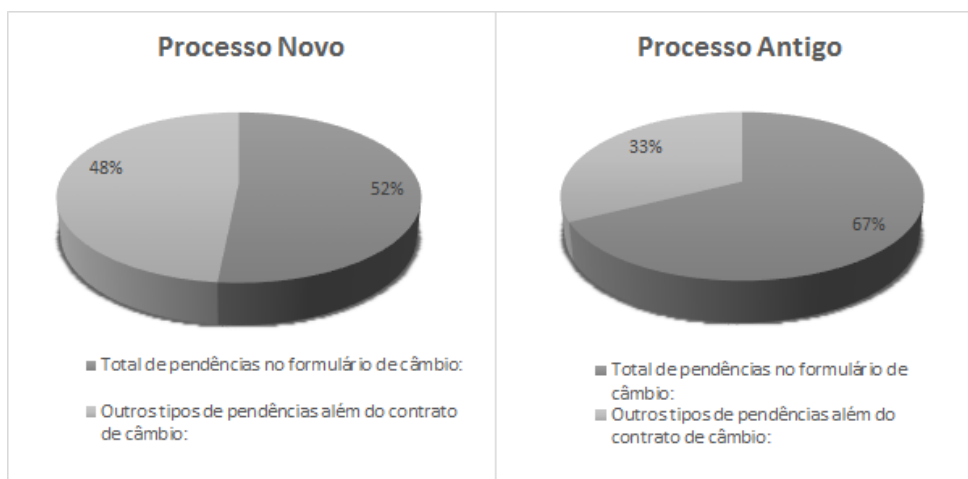


Figura 32 – Comparação dos erros gerados pelo mal preenchimento do contrato de câmbio.

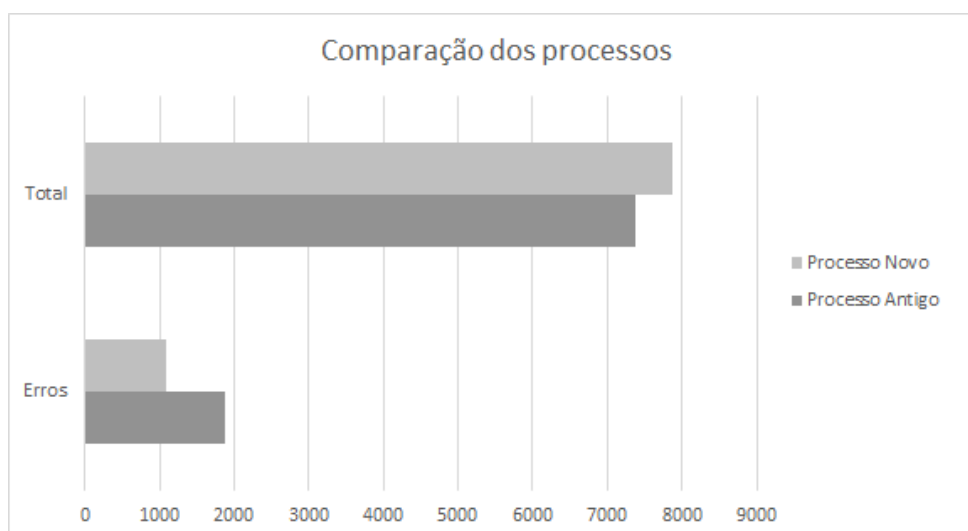


Figura 33 – Comparação do total de erros gerados pelos dois processos diferentes.

Capítulo 5 – Conclusão

O desenvolvimento do projeto foi concluído com sucesso no ambiente de trabalho e as estatísticas puderam comprovar que o software melhorou a qualidade de um processo responsável por alto volume de receita dentro da instituição financeira. Os números apontados na sessão anterior comprovam que os erros gerados pelo mal preenchimento do contrato de câmbio foram substancialmente reduzidos. Basicamente, obteve-se uma redução de 56% no número de pendências geradas pelo contrato de câmbio. Além disso, o número total de pendências no processo como um todo foi reduzido em 43% mesmo para um aumento de 7% no número total de operações para os dois períodos comparados.

É meritório ressaltar a importância de se seguir todas as etapas prescritas no processo de engenharia de software, desde a definição do escopo do projeto até a conclusão do mesmo. Apesar de ser um programa simples, com pouca integração com outros sistemas e totalmente desenvolvido apenas por um programador apenas, o prosseguimento pelas etapas iniciais como: estudo das viabilidades, definição do escopo e a própria escolha do paradigma a ser utilizada foram essenciais para o cumprimento do prazo de dois meses estipulado para a conclusão do trabalho.

Toda a evolução do projeto só foi possível graças às disciplinas de programação e desenvolvimento de software já estudadas anteriormente. Embora o ambiente das instituições financeiras demande um tipo de conhecimento bastante específico e que, na maioria das vezes, não é coberto pelos cursos de engenharia em geral, as demais qualidades exigidas dos alunos durante o curso de graduação fazem com que este profissional seja de grande importância em praticamente todos os segmentos de empresas.

Referências Bibliográficas

ANDREWS, M. D. **The Bretton Woods Agreement as an Invitation to Struggle**. In: Workshop: Bretton Woods Revisited, Scripps College. Claremont, February 2006, Working Paper 06-1.

(Banco Central do Brasil. **Circular 3.689**, 2013)

(Banco Central do Brasil. **Circular 3.690**, 2013)

(Banco Central do Brasil. **Circular 3.691**, 2013)

(BRASIL. **Decreto-Lei 6.306**, 2007)

(BRASIL. **Lei 13.315**, 2016)

(BRASIL. **Instrução normativa 1.645**, 2016)

BRUM, A. L.; ZILIO, M. **Aspectos da evolução do câmbio no Brasil: 1990-2011**. Perspectiva, Erechim. V37, n.138, p.69-80, junho/2013

FORTUNA, E. **Mercado Financeiro: Produtos e Serviços** 16º edição - Rio de Janeiro: Qualitymark Ed., 2005. p.378-379.

MOFFIT, M. **O dinheiro do mundo: de Bretton Woods à beira da insolvência**. Rio de Janeiro: Paz e Terra, 1984. p.13-39.

PRESSMAN, R. S. **Engenharia de software: uma abordagem profissional**. 7º edição – Porto Alegre: AMGH, 2011.

SANDRONI, P. **Dicionário de economia do século XXI**. 4º edição - Rio de Janeiro: Record, 2008. p.111.

SCAMMELL, W. M. **International monetary policy: Bretton Woods and after**. New York: The Macmillan Press LTD, 1975.

Apêndice

Anexo A – Molde de contrato de câmbio fornecido pelo Bacen



BANCO CENTRAL DO BRASIL

ANEXO I À CIRCULAR Nº 3.691, DE 16 DE DEZEMBRO DE 2013
Modelo de contrato de câmbio celebrado com clientes

<i>Contrato de câmbio</i>		
Tipo do contrato de câmbio [] compra [] venda		Número do contrato de câmbio
Evento [] contratação [] cancelamento [] alteração		Data
As partes a seguir denominadas, instituição autorizada a operar no mercado de câmbio e cliente, cpntratam a presente operação de câmbio nas condições aqui estipuladas e declaram que a mesma subordina-se às normas, condições e exigências legais e regulamentares aplicáveis à matéria.		
Instituição autorizada a operar no mercado de câmbio		
Nome		CNPJ
Endereço		
Cidade		UF
Cliente		
Nome		CPF/CNPJ/Ident. do estrangeiro
Endereço		
Cidade		UF/País
Instituição intermediadora*		
Nome		CNPJ
Dados da operação		
Cód. da moeda estrangeira	Valor em moeda estrangeira ()	
Taxa cambial	Valor em moeda nacional R\$ ()	
Valor Efetivo Total (VET)*	Descrição da forma de entrega da moeda estrangeira	Liquidação até
Código da natureza	Descrição da natureza do fato	
Pagador ou recebedor no exterior*		
País do pagador ou do recebedor no exterior*		Código da relação de vínculo entre o cliente e o pagador/recebedor no exterior*
Percentual de adiantamento sobre o contrato de câmbio*		RDE*
Outras especificações		

Figura 34 - Primeira página do molde de contrato de câmbio disponibilizado pelo Bacen no anexo I à circular 3.691 de 16 de dezembro de 2013.



Cláusulas contratuais

Instruções de recebimento/pagamento

O cliente declara ter pleno conhecimento do texto constante do respectivo contrato de câmbio, do art. 23 da Lei nº 4.131, de 3 de setembro de 1962, e em especial dos seus §§ 2º e 3º, transcritos neste documento, bem como da Circular nº 3.691, de 16 de dezembro de 2013, que regem a presente operação.

Art. 23, §§ 2º e 3º, da Lei nº 4.131, de 1962, com a redação dada pelo art. 72 da Lei nº 9.069, de 29 de junho de 1995:

“§ 2º Constitui infração imputável ao estabelecimento bancário, ao corretor e ao cliente, punível com multa de 50 (cinquenta) a 300% (trezentos por cento) do valor da operação para cada um dos infratores, a declaração de falsa identidade no formulário que, em número de vias e segundo o modelo determinado pelo Banco Central do Brasil, será exigido em cada operação, assinado pelo cliente e visado pelo estabelecimento bancário e pelo corretor que nela intervierem.

§ 3º Constitui infração, de responsabilidade exclusiva do cliente, punível com multa de 5 (cinco) a 100% (cem por cento) do valor da operação, a declaração de informações falsas no formulário a que se refere o § 2º.”

Assinaturas

Instituição autorizada a
operar no mercado de
câmbio

Cliente

Instituição intermediadora*

Figura 35 – Segunda página do molde de contrato de câmbio disponibilizado pelo Bacen no anexo I à circular 3.691 de 16 de dezembro de 2013.

Anexo B – Exemplo de contrato de câmbio gerado pelo programa

Comex

Origem: Agência					
Nome de quem está preenchendo o formulário:	Telefone de contato:				
<input type="text"/>	<input type="text"/>				
E-mail:	Número de contrato:	Data:			
<input type="text"/>	<input type="text"/>	<input type="text"/>			
Tipo de contrato de câmbio: () Compra () Venda Evento: () Contratação					
1. As partes a seguir denominadas, "Instituição autorizada a operar no mercado de câmbio", que passa a ser designada BANCO, e o CLIENTE, contratam a presente operação de câmbio nas condições aqui estipuladas e declaram que a mesma subordina-se às normas, condições e exigências legais e regulamentares aplicáveis à matéria.					
2. Banco					
Nome/Razão Social:	CNPJ:				
<input type="text"/>					
Endereço:					
<input type="text"/>					
3. Cliente					
Nome:	CPF ou Id do estrangeiro:	Celular:			
<input type="text"/>	<input type="text"/>	<input type="text"/>			
Segmento:					
() Seg 1 () Seg 2 () Seg 3 () Seg 4 () Seg 5 () Seg 6 () Seg 7					
Endereço:					
<input type="text"/>		Cidade: <input type="text"/>			
UF/País:					
<input type="text"/>	Agência: <input type="text"/>	Conta: <input type="text"/>			
E-mail do cliente: <input type="text"/>					
4. Dados da Operação:					
Código da Moeda Estrangeira:	<input type="text"/>	Valor ME: <input type="text"/>			
Valor da moeda estrangeira (por extenso):					
<input type="text"/>					
Taxa:	Valor da Operação:	Tarifa:	Alíquota IOF:	Valor IOF:	VET:
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Valor da moeda nacional (por extenso):					
<input type="text"/>					
Forma de entrega de ME: <input type="text"/>			Data Liquidação: <input type="text"/>		
Motivo do recebimento/envio de moeda estrangeira:					
<input type="text"/>					
Vínculo entre pagador e receptor:			<input type="text" value="30-Pagador/Recebedor não é Empresa"/>		
5. Canal Bancário:					
Nome do Recebedor: <input type="text"/>					
Banco no exterior: <input type="text"/>			Cidade/País: <input type="text"/>		
Conta corrente no exterior ou IBAN: <input type="text"/>					
Código SWIFT: <input type="text"/>					

0795-S v.30 (12/2016)

Figura 36 - Primeira página do contrato de câmbio gerado pelo programa.

Outras informações sobre o canal bancário:

6. Cláusulas Contratuais

O cliente declara ter pleno conhecimento do texto constante do respectivo contrato de câmbio, do art. 23 da Lei nº 4.131, de 3 de setembro de 1962, e em especial dos seus §§ 2º e 3º, transcritos neste documento, bem como da Circular nº 3.691, de 16 de dezembro de 2013, que regem a presente operação.

Art. 23, §§ 2º e 3º, da Lei nº 4.131, de 1962, com a redação dada pelo art. 72 da Lei nº 9.069, de 29 de junho de 1995:

“§ 2º Constitui infração imputável ao estabelecimento bancário, ao corretor e ao cliente, punível com multa de 50 (cinquenta) a 300% (trezentos por cento) do valor da operação para cada um dos infratores, a declaração de falsa identidade no formulário que, em número de vias e segundo o modelo determinado pelo Banco Central do Brasil, será exigido em cada operação, assinado pelo cliente e visado pelo estabelecimento bancário e pelo corretor que nela intervierem.

§ 3º Constitui infração, de responsabilidade exclusiva do cliente, punível com multa de 5 (cinco) a 100% (cem por cento) do valor da operação, a declaração de informações falsas no formulário a que se refere o § 2º.”

O CLIENTE/COMPRADOR declara para todos os fins de direito que cumpre a legislação vigente, especialmente as normas sobre imposto de renda, a Lei 13.315/2016 e demais normativos que possam, no futuro, serem editados, nos pagamentos e créditos que efetuar a beneficiários no exterior, e que:

- () a remessa é destinada a cobertura de **despesas médico-hospitalares** para tratamento de saúde, no exterior do próprio declarante ou de seus dependentes, nos termos de sua Declaração de Ajuste Anual.
- () a presente remessa tem como fim a **manutenção de seu dependente** no exterior, nos termos de sua Declaração de Ajuste Anual e, que tal valor não representa rendimento auferido pelo favorecido.
- () a presente remessa é destinada ao exterior para **fins educacionais, científicos ou culturais**, bem como o pagamento de taxas escolares, taxas de inscrição em congressos, conclave, seminários ou semelhantes e taxas de exames de proficiência.
- () A remessa é destinada à cobertura de gastos pessoais, no exterior, de pessoas físicas residentes no País (**manutenção de residentes/estudantes, quando estes não são dependentes do remetente para fins de IR**), em viagens de turismo, negócios, serviço, treinamento ou missões oficiais, e que observa o limite global de R\$ 20.000,00, por mês. Ainda, caso a remessa seja destinada a paraíso fiscal, cumpre com os requisitos e limites estabelecidos pelo art. 2º, § 4º da Instrução Normativa nº 1.645/2016, art. 60, da Lei nº 12.249/ 2010 e Lei nº 13.315/2016 e que o beneficiário da remessa atende aos requisitos previstos no artigo 26 da Lei 12.249, de 11 de junho de 2010.
- () a remessa não se enquadra em nenhuma das cláusulas anteriores e, portanto, não possui nenhum benefício fiscal.

O declarante afirma que cumpre as obrigações legais e tributárias ante às exigências dos normativos acima indicados, por fim, responsabilizando-se e assumindo integralmente, qualquer cobrança relativa ao referido tributo e eventuais encargos. Está ciente de que a falsidade na prestação destas informações o sujeitará às penalidades previstas na legislação criminal e tributária, relativas à falsidade ideológica (art. 299 do Código Penal) e ao crime contra a ordem tributária (art. 1º da Lei nº 8.137, de 27 de dezembro de 1966). Adicionalmente, declara que conhece e cumpre cumulativamente os requisitos definidos no O beneficiário da remessa atende aos requisitos previstos no artigo 2º, § 4º da Instrução Normativa nº 1645/2016, a saber:

- i - A identificação do efetivo beneficiário ou entidade no exterior, destinatário dessas importâncias.
- ii - A comprovação da capacidade operacional da pessoa física ou entidade no exterior de realizar a operação; e dos bens e direitos ou da utilização de serviço.
- iii - A comprovação documental do pagamento do preço respectivo e do recebimento dos bens e direitos ou da utilização de serviço.

O fechamento de cada operação de câmbio está condicionado à correta apresentação dos documentos pelo cliente e sua validação pela Área de Câmbio.

O cliente está ciente da possível incidência do Imposto de Renda em razão da operação ora contratada, concordando desde já com o seu recolhimento, bem como autoriza expressamente o respectivo débito em sua conta corrente para recolhimento do tributo.

Assinatura do cliente
(abono obrigatório)

Assinatura do Gerente da Agência
(com carimbo) ou Autenticação Mecânica

0795-S v.30 (12/2016)

Figura 37- Segunda página do contrato de câmbio gerado pelo programa.

Anexo C – Código

7.1.1 Página inicial

```
Private Sub Workbook_BeforeClose(Cancel As Boolean)
    ActiveWorkbook.Saved = True
End Sub
```

```
Private Sub Workbook_Open()

    On Error Resume Next
    Sheet1.Visible = xlSheetVisible
    Sheet7.Visible = xlSheetHidden
    Application.Cursor = xlDefault

End Sub
```

7.1.2 Formulário de dados do funcionário

```
Sub CommandButton1_Click()

    KYC = 0
    Nome_Funcionario = TextBox1.Text
    Email_Funcionario = TextBox2.Text
    Telefone_Funcionario = TextBox3.Text

    If OptionButton1 = True Then
        origem_op = "Agência"
    ElseIf OptionButton2 = True Then
        origem_op = "Central de Atendimento"
    End If

    If OptionButton1 = False And OptionButton2 = False Then
        MsgBox "Favor preencher todos os campos.", vbExclamation, "Dados Funcionário"
    Else
        If Nome_Funcionario = "" Or Email_Funcionario = "" Or Telefone_Funcionario = "" Then
            MsgBox ("Favor preencher todos os campos.")
        Else
            Dados_Func.Hide
            With Dados_Clie
                .StartupPosition = 0
                .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
                .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
                .Show
            End With
        End If
    End If
End Sub

Private Sub CommandButton2_Click()
    On Error Resume Next

    lblHelp01.Visible = Not lblHelp01.Visible
    lblHelp02.Visible = Not lblHelp02.Visible
    lblHelp03.Visible = Not lblHelp03.Visible
    lblHelp04.Visible = Not lblHelp04.Visible
```



```

    lblHelp05.Visible = Not lblHelp05.Visible

    If lblHelp01.Visible Then
        Call OpenHelpFile
    End If
End Sub

Private Sub TextBox3_Change()
    If TextBox3.Value = vbNullString Then
        Exit Sub
    Else
        If Not IsNumeric(Replace(TextBox3.Value, " ", "")) Then
            MsgBox "Digite somente números", vbExclamation, "TELEFONE"
            TextBox3.ForeColor = RGB(255, 0, 0)
        Else
            TextBox3.ForeColor = RGB(0, 0, 0)
        End If
    End If
End Sub

Private Sub UserForm_Initialize()
    On Error GoTo err
    lblSysVersion.Caption = SYSTEM_VERSION
    Dim str As String
    str = Environ("username")
    Dim olApp As Object
    Set olApp = CreateObject("Outlook.Application")
    Dim olNameSpace As Object
    Set olNameSpace = olApp.GetNamespace("MAPI")
    Dim olRecipient As Object
    Set olRecipient = olNameSpace.CreateRecipient(str)
    Dim oEU As Object

    olRecipient.Resolve
    If olRecipient.Resolved Then
        Select Case olRecipient.AddressEntry.AddressEntryUserType
            Case 0, 5
                Set oEU = olRecipient.AddressEntry.GetExchangeUser
                If Not (oEU Is Nothing) Then
                    Me.TextBox1.Value = oEU
                    Me.TextBox2.Value = oEU.PrimarySmtpAddress
                End If
            End Select
        End If

        TextBox3.SetFocus

    ext:
        taxa_manual = False
        Set oEU = Nothing
        Set olApp = Nothing
        Set olNameSpace = Nothing
        Set olRecipient = Nothing
        Exit Sub

    err:
        GoTo ext
End Sub

Private Sub Dados_Func_Activate()

    Me.StartupPosition = 0
    Me.Top = Application.Top + 125
    Me.Left = Application.Left + Application.Width - Me.Width - 125

End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)

```

```

Dim formulario As UserForm

For Each formulario In UserForms
    Unload formulario
Next formulario

End Sub

```

7.1.3 Formulário de dados do cliente

```

Private Sub CommandButton5_Click()

    Dados_Clie.Hide

    With Dados_Func
        .StartPosition = 0
        .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
        .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
        .Show
    End With

End Sub

Private Sub CommandButton6_Click()
    On Error Resume Next

    lblHelp01.Visible = Not lblHelp01.Visible
    lblHelp02.Visible = Not lblHelp02.Visible
    lblHelp03.Visible = Not lblHelp03.Visible
    lblHelp04.Visible = Not lblHelp04.Visible
    lblHelp05.Visible = Not lblHelp05.Visible
    lblHelp06.Visible = Not lblHelp06.Visible
    lblHelp07.Visible = Not lblHelp07.Visible
    lblHelp08.Visible = Not lblHelp08.Visible
    lblHelp09.Visible = Not lblHelp09.Visible
    lblHelp10.Visible = Not lblHelp10.Visible
    lblHelp11.Visible = Not lblHelp11.Visible
    lblHelp12.Visible = Not lblHelp12.Visible
    lblHelp13.Visible = Not lblHelp13.Visible

    If lblHelp01.Visible Then
        Call OpenHelpFile
    End If
End Sub

Private Sub OptionButton1_Click()

End Sub

Private Sub OptionButton4_Click()
    MsgBox "É considerado diplomata, qualquer pessoa física que esteja representando o seu país em uma missão diplomática em outro país. Ex: Cônsul e embaixadores.", vbInformation, "Dados do Cliente"
End Sub

Private Sub TextBox2_Change()
    If TextBox2.TextLength > 11 Then
        MsgBox "Número inválido de caracteres."
        TextBox2.Text = Left(TextBox2.Text, TextBox2.TextLength - 1)
    End If
End Sub

Private Sub CommandButton1_Click()

```

On Error GoTo ERRO:

Call

NavigateToURL("http://intranet.local.santander.br.corp/santander/Abas/Auxiliar/select/produtos/cambio/fer
ramentas/ferramentas/Acordos_cambio_FOPA")

MsgBox "ATENÇÃO " & vbCrLf & vbCrLf & "Verifique se um link foi carregado." & vbCrLf & _
"(Formato internet explorer)", vbInformation, ". : Formulário 0795 : ."

Exit Sub

ERRO:

Application.ScreenUpdating = False
Sheets("FOPA").Visible = True
Sheets("FOPA").Activate
ActiveSheet.PageSetup.PrintArea = "\$A\$1:\$G\$37"
ActiveSheet.PageSetup.Orientation = xlPortrait
pdfName = "Naturezas Cambiais"
fileSaveName = "Naturezas Cambiais" & ".pdf"
ThisWorkbook.Sheets("FOPA").ExportAsFixedFormat Type:=xlTypePDF, FileName:= _
fileSaveName _
, Quality:=xlQualityStandard, IncludeDocProperties:=True, IgnorePrintAreas _
:=False, OpenAfterPublish:=True
Sheets("FOPA").Visible = False
Application.ScreenUpdating = True

End Sub

Sub CommandButton2_Click() 'Botao SIM

KYC = 2

Label15.Visible = True
Label15.Caption = "RESPOSTA: SIM"

End Sub

Sub CommandButton3_Click() 'Botao Nao

KYC = 1

Label15.Visible = True
Label15.Caption = "RESPOSTA: NÃO"

End Sub

Sub CommandButton4_Click() 'Botao Confirma

If _
TextBox1.ForeColor = RGB(255, 0, 0) Or _
TextBox4.ForeColor = RGB(255, 0, 0) Or _
TextBox2.ForeColor = RGB(255, 0, 0) Or _
TextBox5.ForeColor = RGB(255, 0, 0) Or _
TextBoxDDD.ForeColor = RGB(255, 0, 0) Or _
TextBox3.ForeColor = RGB(255, 0, 0) Or _
TextBox6.ForeColor = RGB(255, 0, 0) Or _
TextBox9.ForeColor = RGB(255, 0, 0) Or _
TextBox7.ForeColor = RGB(255, 0, 0) Or _
TextBox8.ForeColor = RGB(255, 0, 0) Then

MsgBox "ATENÇÃO: " & vbCrLf & vbCrLf & "Alguns campos podem estar com o preenchimento
incorreto. Verifique.", vbExclamation, "Confirmação"

Exit Sub

End If

If KYC = 0 Then

```

    MsgBox "Favor verificar se o cliente possui cadastro atualizado e habilitado para operar com
    câmbio.", vbExclamation, "Confirmação"

    ElseIf KYC = 1 Then
        MsgBox "Cliente não está habilitado a fazer este tipo de operação, favor regularizar o cadastro.",
        vbExclamation, "Confirmação"

    Else

        Nome_Cliente = TextBox1.Text
        CPF_Cliente = TextBox2.Text
        Telefone_Cliente = TextBoxDDD.Text & " " & TextBox3.Text
        Endereco_Cliente = TextBox4.Text
        Cidade_Cliente = TextBox5.Text
        Pais_Cliente = TextBox6.Text
        Agencia_Cliente = TextBox7.Text
        Conta_Cliente = TextBox8.Text
        email_cliente = TextBox9.Text

        If OptionButton1 = True Then
            segmento_cliente = "Seg1"
            chave_confirmacao = True

        ElseIf OptionButton2 = True Then
            segmento_cliente = "Seg2"
            chave_confirmacao = True

        ElseIf OptionButton3 = True Then
            segmento_cliente = "Seg3"
            chave_confirmacao = True

        ElseIf OptionButton4 = True Then
            segmento_cliente = "Seg4"
            chave_confirmacao = True

        ElseIf OptionButton5 = True Then
            segmento_cliente = "Seg5"
            chave_confirmacao = True

        ElseIf OptionButton6 = True Then
            segmento_cliente = "Seg6"
            chave_confirmacao = True

        ElseIf OptionButton7 = True Then
            segmento_cliente = "Seg7"
            chave_confirmacao = True

        Else
            MsgBox "Por favor selecione o segmento do cliente.", vbExclamation, "Confirmação"

        End If

        If Nome_Cliente = "" Or CPF_Cliente = "" Or Telefone_Cliente = "" Or Endereco_Cliente = "" Or
        Cidade_Cliente = "" Or Pais_Cliente = "" Or Agencia_Cliente = "" Or Conta_Cliente = "" Then
            MsgBox "Favor preencher todos os campos disponíveis.", vbExclamation, "Confirmação"

        ElseIf chave_confirmacao = True Then
            Dados_Clie.Hide
            With Natureza_Operacao
                .StartPosition = 0
                .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
                .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
                .Show
            End With

        End If
    
```

```

End If

End Sub

'CELULAR
Private Sub TextBox3_Change()
    If TextBox3.Value = vbNullString Then
        Exit Sub
    Else
        If Not IsNumeric(TextBox3.Value) Then
            MsgBox "Digite somente números", vbExclamation, "Celular"
            TextBox3.ForeColor = RGB(255, 0, 0)
        Else
            TextBox3.ForeColor = RGB(0, 0, 0)
        End If
    End If
End Sub

'AGENCIA
Private Sub TextBox7_Change()
    If TextBox7.Value = vbNullString Then
        Exit Sub
    Else
        If Not IsNumeric(TextBox7.Value) Then
            MsgBox "Digite somente números", vbExclamation, "Agência"
            TextBox7.ForeColor = RGB(255, 0, 0)
        Else
            TextBox7.ForeColor = RGB(0, 0, 0)
        End If
    End If
End Sub

'CONTA CORRENTE
Private Sub TextBox8_Change()
    If TextBox8.Value = vbNullString Then
        Exit Sub
    Else
        If Not IsNumeric(TextBox8.Value) Then
            MsgBox "Digite somente números", vbExclamation, "Conta Corrente"
            TextBox8.ForeColor = RGB(255, 0, 0)
        Else
            TextBox8.ForeColor = RGB(0, 0, 0)
        End If
    End If
End Sub

'E-MAIL
Private Sub TextBox9_AfterUpdate()
    If TextBox9.Value = vbNullString Then
        Exit Sub
    Else
        If Not InStr(TextBox9.Value, "@") > 0 Then
            MsgBox "Digite um e-mail válido", vbExclamation, "E-mail"
            TextBox9.ForeColor = RGB(255, 0, 0)
        Else
            TextBox9.ForeColor = RGB(0, 0, 0)
        End If
    End If
End Sub

'CELULAR (DDD)
Private Sub TextBoxDDD_Change()
    If TextBoxDDD.Value = vbNullString Then
        Exit Sub
    Else
        If Not IsNumeric(TextBoxDDD.Value) Then
            MsgBox "Digite somente números", vbExclamation, "Celular (DDD)"
        End If
    End If
End Sub

```

```

        TextBoxDDD.ForeColor = RGB(255, 0, 0)
    Else
        TextBoxDDD.ForeColor = RGB(0, 0, 0)
    End If
End If
End Sub

Private Sub UserForm_Activate()
    lblSysVersion.Caption = SYSTEM_VERSION

    If KYC = 0 Then
        Label15.Visible = False
    End If
End Sub

Private Sub UserForm_Initialize()
    lblSysVersion.Caption = SYSTEM_VERSION

    If KYC = 0 Then
        Label15.Visible = False
    End If
End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    Dim formulario As UserForm

    For Each formulario In UserForms
        Unload formulario
    Next formulario

End Sub

```

7.1.4 Formulário de natureza da operação

```

Private Sub ComboBox1_Change()
    On Error Resume Next

    If ComboBox1.Value = "Outros..." Then
        TextBox1.Locked = False
        TextBox1.Enabled = True
        TextBox1.BackColor = &H80000005
        Label7.ForeColor = RGB(255, 0, 0)
        TextBox1.SetFocus
    Else
        TextBox1.Value = ""
        TextBox1.Locked = True
        TextBox1.Enabled = False
        TextBox1.BackColor = &H8000000F
        Label7.ForeColor = RGB(0, 0, 0)
    End If
End Sub

Private Sub CommandButton1_Click()
    On Error GoTo err
    Dim ulin As Integer
    ulin = Sheet2.Cells(Rows.Count, "a").End(xlUp).Row
    Application.ScreenUpdating = False
    Sheets("Configuracoes").Visible = True

    Sheets("configuracoes").Activate
    ActiveSheet.PageSetup.PrintArea = "$A$1:$E$" & ulin
    ActiveSheet.PageSetup.Orientation = xlLandscape

```

```

pdfName = "Naturezas Cambiais"
fileSaveName = "Naturezas Cambiais" & ".pdf"
ThisWorkbook.Sheets("Configuracoes").ExportAsFixedFormat Type:=xlTypePDF, FileName:= _
    fileSaveName _
    , Quality:=xlQualityStandard, IncludeDocProperties:=True, IgnorePrintAreas _
    :=False, OpenAfterPublish:=True

MsgBox "ATENÇÃO " & vbCrLf & vbCrLf & "Verifique se um documento foi carregado." & vbCrLf & _
    "(Formato PDF)", vbInformation, ". : Formulário 0795 : ."

Sheets("Configuracoes").Visible = False

ext:
    Application.ScreenUpdating = True
    Exit Sub
err:
    GoTo ext
End Sub

Private Sub CommandButton2_Click() 'Botao Confirma
    Dim item As Variant
    natureza_da_operacao = ComboBox1.Value
    clausulas_ir = "0"

    If OptionButton1 = False And OptionButton2 = False Then
        MsgBox "Favor preencher todos os campos.", vbExclamation, "Confirmação"
        Exit Sub
    End If

    If natureza_da_operacao = "" Then
        MsgBox "Favor preencher todos os campos.", vbExclamation, "Confirmação"
        Exit Sub
    End If

    If OptionButton1 = True Then
        tipo_operacao = "Compra"

        If natureza_da_operacao = "Outros..." Then
            natureza_da_operacao = TextBox1.Value
            If natureza_da_operacao = "" Then
                MsgBox "Favor preencher o motivo da operação.", vbExclamation, "Confirmação"
                Exit Sub
            Else
                Natureza_Operacao.Hide
                With Dados_Operacao
                    .StartupPosition = 0
                    .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
                    .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
                    .Show
                End With
            End If
        Else
            Natureza_Operacao.Hide
            With Dados_Operacao
                .StartupPosition = 0
                .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
                .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
                .Show
            End With
        End If

        ElseIf OptionButton2 = True Then
            tipo_operacao = "Venda"

            exhibir_clausula = "N"
            For Each item In Sheet8.Range("PARAM_NATUREZA")
                If item.Value = natureza_da_operacao Then

```

```

        exibir_clausula = Replace(item.Offset(0, 3), "", "N")
    Exit For
End If
Next item

If natureza_da_operacao = "Outros..." Then
    natureza_da_operacao = TextBox1.Value
    If natureza_da_operacao = "" Then
        MsgBox "Favor preencher o motivo da operação.", vbExclamation, "Confirmação"
    Exit Sub
    Else
        Natureza_Operacao.Hide
        With Dados_Operacao
            .StartPosition = 0
            .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
            .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
            .Show
        End With
    End If
Else
    If exibir_clausula = "S" Then
        Natureza_Operacao.Hide
        With Clausulas
            .StartPosition = 0
            .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
            .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
            .Show
        End With
    Else
        Natureza_Operacao.Hide
        With Dados_Operacao
            .StartPosition = 0
            .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
            .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
            .Show
        End With
    End If
End If
End Sub

Private Sub CommandButton3_Click() 'Botao voltar

    Natureza_Operacao.Hide
    With Dados_Clie
        .StartPosition = 0
        .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
        .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
        .Show
    End With

End Sub

Private Sub CommandButton4_Click()
    On Error Resume Next

    lblHelp01.Visible = Not lblHelp01.Visible
    lblHelp02.Visible = Not lblHelp02.Visible
    lblHelp03.Visible = Not lblHelp03.Visible

    If lblHelp01.Visible Then
        Call OpenHelpFile
    End If
End Sub

Private Sub UserForm_Activate()
    lblSysVersion.Caption = SYSTEM_VERSION

```



```

End Sub

Private Sub UserForm_Initialize()
    lblSysVersion.Caption = SYSTEM_VERSION

    Call ComboBox1_Change
End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)

    Dim formulario As UserForm

    For Each formulario In UserForms
        Unload formulario
    Next formulario

End Sub

```

7.1.5 Formulário de cláusulas de IR

```

Private Sub CommandButton2_Click()

    If OptionButton1 = True Then
        clausulas_ir = "1"

    ElseIf OptionButton2 = True Then
        clausulas_ir = "2"

    ElseIf OptionButton3 = True Then
        clausulas_ir = "3"

    ElseIf OptionButton4 = True Then
        clausulas_ir = "4"

    End If

    If OptionButton1 = False And OptionButton2 = False And OptionButton3 = False And OptionButton4 =
False Then
        MsgBox "Favor selecionar uma das cláusulas.", vbExclamation, "Cláusulas"
    Else

        Clausulas.Hide
        'Call teste
        With Dados_Operacao
            .StartPosition = 0
            .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
            .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
            .Show
        End With
    End If
End Sub

Private Sub UserForm_Activate()
    lblSysVersion.Caption = SYSTEM_VERSION
End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    Dim formulario As UserForm

    For Each formulario In UserForms
        Unload formulario
    Next formulario

End Sub

```

7.1.6 Formulário de dados da operação

Dim chaveconfirmacao As Boolean

Private Sub Refresh()

Dim item As Variant, col As Integer

col = IIf(tipo_operacao = "Compra", 1, 2)

forma_entrega = ComboBox2.Value

If tipo_operacao = "Venda" And InStr(forma_entrega, "Traveller Cheque") > 0 Then

forma_entrega = vbNullString

ComboBox2.Value = forma_entrega

End If

If segmento_cliente <> vbNullString Then

If InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0 Then

If segmento_cliente = "VG/PF" Or segmento_cliente = "Select" Or _

segmento_cliente = "Diplomata" Or segmento_cliente = "FOPA Prata" Or _

segmento_cliente = "FOPA Bronze" Then

valor_tarifa = 20

Label10.Caption = Format(valor_tarifa, " #,##0.00; (#,##0.00); ""-"";_(@_)"

ElseIf segmento_cliente = "Funcionário" Or segmento_cliente = "FOPA Ouro" Then

valor_tarifa = 0

Label10.Caption = Format(valor_tarifa, " #,##0.00; (#,##0.00); ""-"";_(@_)"

End If

Else

For Each item In Sheet8.Range("PARAM_SEGMENTO")

If item.Value = segmento_cliente Then

valor_tarifa = item.Offset(0, col)

Label10.Caption = Format(valor_tarifa, " #,##0.00; (#,##0.00); ""-"";_(@_)"

Exit For

End If

Next item

End If

End If

If natureza_da_operacao <> vbNullString Then

For Each item In Sheet8.Range("PARAM_NATUREZA")

If item.Value = natureza_da_operacao Then

If segmento_cliente = "Diplomata" Then

```

        aliquota_iof = 0
    Exit For
Else
    aliquota_iof = item.Offset(0, col)
    Exit For
End If
End If
Next item
End If
If natureza_da_operacao <> vbNullString Then
    For Each item In Sheet8.Range("PARAM_FORMA_ENTREGA")
        If item.Value = forma_entrega Then

            If segmento_cliente = "Diplomata" And InStr(forma_entrega, "Espécie") > 0 Then
                aliquota_iof = 0
                Exit For
            Else
                aliquota_iof_forma_entrega = item.Offset(0, col)
                Exit For
            End If
        End If
    Next item
End If
If aliquota_iof_forma_entrega <> vbNullString And aliquota_iof <> vbNullString Then
    If aliquota_iof_forma_entrega > aliquota_iof Then
        If segmento_cliente = "Diplomata" And InStr(forma_entrega, "Espécie") > 0 Then
            aliquota_iof = aliquota_iof
        Else
            aliquota_iof = aliquota_iof_forma_entrega
        End If
    End If
End If
End If
codigo_me = ComboBox1.Value
valor_me = ""
If TextBox3.Value <> vbNullString Then
    If IsNumeric(TextBox3.Value) Then

```

```

        valor_me = CDbI(TextBox3.Value)
    End If
End If
If ComboBox1.Value = vbNullString Then
    MsgBox "Selecione a Moeda.", vbExclamation, "Obter Taxa"
    Exit Sub
End If
If TextBox3.Value = vbNullString Then
    'MsgBox "Digite o valor em moeda estrangeira (VALOR M/E).", vbExclamation, "Obter Taxa"
    Exit Sub
End If
If taxa_manual = True Then
    taxa = TextBox7.Value
Else
    taxa = ""
    Call pega_taxa
    If taxa <> vbNullString Then
        TextBox7.Text = taxa
        TextBox7.Locked = False
    End If
End If
Label9.Caption = ""
If taxa <> vbNullString And valor_me <> vbNullString Then
    valor_operacao = taxa * valor_me
    Label9.Caption = Format(valor_operacao, " #,###0.00; (#,###0.00); ""-"";_(@_)")
End If
valor_iof = 0
If aliquota_iof <> vbNullString And valor_operacao <> vbNullString Then
    valor_iof = aliquota_iof * valor_operacao
    Label13.Caption = valor_iof
End If
Label16.Caption = ""
If valor_operacao <> vbNullString And valor_tarifa <> vbNullString And aliquota_iof <> vbNullString And
valor_me <> vbNullString Then
    If tipo_operacao = "Compra" Then
        vet = (valor_operacao - valor_tarifa - valor_iof) / valor_me
        Label16.Caption = Format(vet, " #,###0.0000; (#,###0.0000); ""-"";_(@_)")
    End If
End If

```

```

Elseif tipo_operacao = "Venda" Then
    vet = (valor_operacao + valor_tarifa + valor_iof) / valor_me
    Label16.Caption = Format(vet, " #,###0.0000; (#,###0.0000); ""-"";_(@_)")
End If
End If

vlrMeExtenso.Caption = ""

If Label9.Caption <> vbNullString Then
    If Len(Label9.Caption) > 2 Then
        vlrMeExtenso.Caption = UCase(Extenso_Valor(Label9.Caption))
        Extenso_MN = vlrMeExtenso.Caption
    End If
End If

End Sub

Private Sub ComboBox1_AfterUpdate() 'ComboBox de seleção da moeda
    Call Refresh
    TextBox6.Value = ""
    If InStr(ComboBox1.Value, "JPY") > 0 Then
        Label22.Visible = True
    Else
        Label22.Visible = False
    End If
End Sub

Private Sub ComboBox2_AfterUpdate()
    Call Refresh
End Sub

Private Sub CommandButton1_Click() 'Botao Voltar
    Dados_Operacao.Hide
    With Natureza_Operacao
        .StartPosition = 0
        .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
        .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
        .Show
    End With
End Sub

Private Sub CommandButton2_Click() 'Botao Confirma

```

```

forma_entrega = ComboBox2.Value

Extenso_ME = TextBox6.Value

Extenso_MN = vlrMeExtenso.Caption

If forma_entrega = "" Or Extenso_ME = "" Or codigo_me = "" Or valor_me = "" Or TextBox7.Value = ""
Then

    MsgBox "Favor preencher todos os campos.", vbExclamation, "Confirmação"

Else

    Dados_Operacao.Hide

    If tipo_operacao = "Compra" Then

        With Canal_Compra

            .StartPosition = 0

            .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)

            .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)

            .Show

        End With

    ElseIf tipo_operacao = "Venda" Then

        With Canal_Venda

            .StartPosition = 0

            .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)

            .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)

            .Show

        End With

    End If

End If

End Sub

Private Sub CommandButton3_Click() 'Botao Obter taxas

    taxa_manual = False

    Call Refresh

End Sub

Private Sub CommandButton4_Click()

    On Error Resume Next

    lblHelp01.Visible = Not lblHelp01.Visible

    lblHelp02.Visible = Not lblHelp02.Visible

    lblHelp03.Visible = Not lblHelp03.Visible

    lblHelp04.Visible = Not lblHelp04.Visible

    lblHelp05.Visible = Not lblHelp05.Visible

    lblHelp06.Visible = Not lblHelp06.Visible

```

```

If lblHelp01.Visible Then
    Call OpenHelpFile
End If
End Sub

Private Sub TextBox3_AfterUpdate() 'Valor operação
    If TextBox3.Value = vbNullString Then
        Exit Sub
    Else
        If Not IsNumeric(TextBox3.Value) Then
            MsgBox "Digite somente números", vbExclamation, "VALOR M/E"
            TextBox3.ForeColor = RGB(255, 0, 0)
        Else
            TextBox3.ForeColor = RGB(0, 0, 0)
        End If
    End If
    If InStr(ComboBox1.Value, "JPY") > 0 Then
        TextBox3.Value = Format(TextBox3.Value, "#,##0; (#,##0); ""-"";_(@_)")
    Else
        TextBox3.Value = Format(TextBox3.Value, "#,##0.00; (#,##0.00); ""-"";_(@_)")
    End If
    TextBox7.Locked = False
    Call Refresh
End Sub

Private Sub TextBox6_AfterUpdate()
    If TextBox6.Value = vbNullString Then
        Exit Sub
    Else
        If IsNumeric(TextBox6.Value) Then
            MsgBox "Escreva o valor em moeda estrangeira por extenso.", vbExclamation, "VALOR M/E (POR EXTENSO)"
            TextBox6.ForeColor = RGB(255, 0, 0)
        Else
            TextBox6.ForeColor = RGB(0, 0, 0)
        End If
    End If
End Sub

```

```

Private Sub TextBox7_AfterUpdate()
    If TextBox7.Value = vbNullString Then
        taxa_manual = False
        Exit Sub
    Else
        If Not IsNumeric(TextBox7.Value) Then
            taxa_manual = False
            MsgBox "Digite somente números", vbExclamation, "TAXA DE CÂMBIO"
            TextBox7.ForeColor = RGB(255, 0, 0)
        Else
            taxa_manual = True
            TextBox7.ForeColor = RGB(0, 0, 0)
        End If
    End If
    Call Refresh

    MsgBox "ATENÇÃO: " & vbCrLf & vbCrLf & "Só preencha a taxa manualmente se a mesma for cotada na mesa. " & _
    "Para operações de valor inferior a USD 100.000,00 favor utilizar a taxa da intranet." & vbCrLf & vbCrLf & _
    "Mesa de Câmbio: (11) 3012-5745", vbExclamation, "Obter Taxa"
End Sub

Private Sub TextBox7_KeyDown(ByVal KeyCode As MSForms.ReturnInteger, ByVal Shift As Integer)
    If TextBox7.Locked = True Then
        MsgBox "Favor preencher o campo VALOR M/E para liberar este campo.", vbExclamation, "Validação de parâmetros"
    End If
End Sub

Private Sub UserForm_Activate()
    lblSysVersion.Caption = SYSTEM_VERSION
    If tipo_operacao = "Venda" Then
        ComboBox2.RowSource = "Parametros!PARAM_FORMA_ENTREGA_ENVIO"
    Else
        ComboBox2.RowSource = "Parametros!PARAM_FORMA_ENTREGA"
    End If
    If Label16.Caption <> vbNullString Then
        Call Refresh
    End If
End Sub

```



```

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    Dim formulario As UserForm
    For Each formulario In UserForms
        Unload formulario
    Next formulario
End Sub

```

7.1.7 Formulário de Canal Bancário – Compra

```

Private Sub CommandButton1_Click()
    Me.Hide
    With Dados_Operacao
        .StartPosition = 0
        .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
        .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
        .Show
    End With
End Sub

Private Sub CommandButton2_Click()
    Dim resposta As Integer
    nome_ordenante = TextBox3.Value
    pais_ordenante = TextBox4.Value
    info_adicional_ordenante = TextBox5.Value
    If nome_ordenante = "" Or pais_ordenante = "" Or info_adicional_ordenante = "" Then
        MsgBox "Favor preencher todos os campos.", vbExclamation, "Confirmação"
    Else
        Me.Hide
        With Gerar_PDF
            .StartPosition = 0
            .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
            .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
            .Show
        End With
    End If
End Sub

Private Sub CommandButton3_Click()

```

```

On Error Resume Next

lblHelp01.Visible = Not lblHelp01.Visible
lblHelp02.Visible = Not lblHelp02.Visible
lblHelp03.Visible = Not lblHelp03.Visible
lblHelp04.Visible = Not lblHelp04.Visible

If lblHelp01.Visible Then
    Call OpenHelpFile
End If

End Sub

Private Sub CommandButton4_Click()
    On Error GoTo err

    Call
NavigateToURL("http://intranet.local1.santander.br.corp/sistemas/ordenspagamento/ASP/IEOP_index.asp", True)

    MsgBox "ATENÇÃO " & vbCrLf & vbCrLf & "Verifique se um link foi carregado." & vbCrLf & _
        "(Formato internet explorer)", vbInformation, ". : Formulário 0795 : ."

ext:

Exit Sub

err:

GoTo ext

End Sub

Private Sub Label7_Click()

End Sub

Private Sub TextBox4_Change()

End Sub

Private Sub UserForm_Activate()
    lblSysVersion.Caption = SYSTEM_VERSION
End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    Dim formulario As UserForm
    For Each formulario In UserForms
        Unload formulario
    Next formulario
End Sub

```

7.1.8 Formulário de Canal Bancário – Venda

```

Private Sub CommandButton1_Click()

```

```

Canal_Venda.Hide
With Dados_Operacao
.StartUpPosition = 0
.Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
.Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
.Show
End With
End Sub

Private Sub CommandButton2_Click()
    Dim resposta As Integer
    nome_beneficiario = TextBox3.Value
    banco_exterior = TextBox4.Value
    cidade_pais = TextBox6.Value
    iban = TextBox7.Value
    swift_exterior = TextBox8.Value
    info_adicional = TextBox5.Value

    If nome_beneficiario = "" Or banco_exterior = "" Or cidade_pais = "" Or iban = "" Or swift_exterior = ""
    Then
        MsgBox "Favor preencher todos os campos.", vbExclamation, "Confirmação"
    Else
        Canal_Venda.Hide
        With Gerar_PDF
            .StartUpPosition = 0
            .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
            .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
            .Show
        End With
    End If
End Sub

Private Sub CommandButton3_Click()
    On Error Resume Next
    lblHelp01.Visible = Not lblHelp01.Visible
    lblHelp02.Visible = Not lblHelp02.Visible
    lblHelp03.Visible = Not lblHelp03.Visible
    lblHelp04.Visible = Not lblHelp04.Visible
    lblHelp05.Visible = Not lblHelp05.Visible
    lblHelp06.Visible = Not lblHelp06.Visible

```

```

    If lblHelp01.Visible Then
        Call OpenHelpFile
    End If
End Sub

Private Sub UserForm_Activate()
    lblSysVersion.Caption = SYSTEM_VERSION
    If Trim(forma_entrega) = "50 - Traveller Check" Then
        If TextBox3.Value = vbNullString Then TextBox3.Value = "N/A"
        If TextBox4.Value = vbNullString Then TextBox4.Value = "N/A"
        If TextBox6.Value = vbNullString Then TextBox6.Value = "N/A"
        If TextBox7.Value = vbNullString Then TextBox7.Value = "N/A"
        If TextBox8.Value = vbNullString Then TextBox8.Value = "N/A"
        If TextBox5.Value = vbNullString Then TextBox5.Value = "N/A"
    End If
End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    Dim formulario As UserForm
    For Each formulario In UserForms
        Unload formulario
    Next formulario
End Sub

```

7.1.9 Formulário de geração do contrato

```

Private Sub CommandButton1_Click()
    Unload Me
    With Dados_Operacao
        .StartPosition = 0
        .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
        .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
        .Show
    End With
End Sub

Private Sub CommandButton2_Click()
    On Error Resume Next
    Application.ScreenUpdating = False

```

Sheets("0795").Visible = True

Sheets("0795").Unprotect

'Escrever todas as variáveis no contrato de câmbio

'DADOS FUNCIONARIO

Sheets("0795").Range("Form_origem_op").MergeArea.Value = origem_op 'Verificado

Sheets("0795").Range("Form_Nome_Funcionario").MergeArea.Value = Nome_Funcionario 'Verificado

*Sheets("0795").Range("Form_Telefone_Funcionario").MergeArea.Value = Telefone_Funcionario
'Verificado*

Sheets("0795").Range("Form_Email_Funcionario").MergeArea.Value = Email_Funcionario 'Verificado

If tipo_operacao = "Compra" Then

Sheets("0795").Range("Form_Compra").MergeArea.Value = "(X) Compra"

Sheets("0795").Range("Form_Venda").MergeArea.Value = "() Venda"

ElseIf tipo_operacao = "Venda" Then

Sheets("0795").Range("Form_Compra").MergeArea.Value = "() Compra"

Sheets("0795").Range("Form_Venda").MergeArea.Value = "(X) Venda"

End If

'DADOS CLIENTE

Sheets("0795").Range("Form_Nome_Cliente").MergeArea.Value = Nome_Cliente 'Verificado

Sheets("0795").Range("Form_CPF_Cliente").MergeArea.Value = CPF_Cliente 'Verificado

Sheets("0795").Range("Form_Telefone_Cliente").MergeArea.Value = Telefone_Cliente 'Verificado

Sheets("0795").Range("Form_Endereco_Cliente").MergeArea.Value = Endereco_Cliente 'Verificado

Sheets("0795").Range("Form_Cidade_Cliente").MergeArea.Value = Cidade_Cliente 'Verificado

Sheets("0795").Range("Form_Pais_Cliente").MergeArea.Value = Pais_Cliente 'Verificado

Sheets("0795").Range("Form_Agencia_Cliente").MergeArea.Value = Agencia_Cliente 'Verificado

Sheets("0795").Range("Form_Conta_Cliente").MergeArea.Value = Conta_Cliente 'Verificado

Sheets("0795").Range("Form_email_cliente").MergeArea.Value = email_cliente 'Verificado

'DADOS OPERACAO

Sheets("0795").Range("Form_codigo_me").MergeArea.Value = codigo_me 'Verificado

Sheets("0795").Range("Form_valor_me").MergeArea.Value = valor_me 'Verificado

Sheets("0795").Range("Form_forma_entrega").MergeArea.Value = forma_entrega 'Verificado

Sheets("0795").Range("Form_taxa").MergeArea.Value = taxa 'Verificado

Sheets("0795").Range("Form_valor_operacao").MergeArea.Value = valor_operacao 'Verificado

Sheets("0795").Range("Form_valor_tarifa").MergeArea.Value = valor_tarifa 'Verificado

Sheets("0795").Range("Form_vet").MergeArea.Value = vet 'Verificado

Sheets("0795").Range("Form_valor_iof").MergeArea.Value = valor_iof

Sheets("0795").Range("Form_Extenso_ME").MergeArea.Value = Extenso_ME 'Verificado

```

Sheets("0795").Range("Form_natureza_da_operacao").MergeArea.Value = natureza_da_operacao
'Verificado

Sheets("0795").Range("Form_Extenso_MN").MergeArea.Value = Extenso_MN      'Verificado

'DADOS CANAL BANCARIO

If tipo_operacao = "Venda" Then

    Sheets("0795").Range("Form_Canal_Nome").MergeArea.Value = "Nome do Beneficiário no exterior:"

    Sheets("0795").Range("Form_Canal_nome_beneficiario").MergeArea.Value = nome_beneficiario
'Verificado

    Sheets("0795").Range("Form_Canal_banco_exterior").MergeArea.Value = banco_exterior
'Verificado

    Sheets("0795").Range("Form_Canal_cidade_pais").MergeArea.Value = cidade_pais      'Verificado
    Sheets("0795").Range("Form_Canal_iban").MergeArea.Value = iban                  'Verificado
    Sheets("0795").Range("Form_Canal_swift_exterior").MergeArea.Value = swift_exterior  'Verificado
    Sheets("0795").Range("Form_Canal_info_adicional").MergeArea.Value = info_adicional  'Verificado

ElseIf tipo_operacao = "Compra" Then

    Sheets("0795").Range("Form_Canal_Nome").MergeArea.Value = "Nome do Recebedor:"

    Sheets("0795").Range("Form_Canal_nome_beneficiario").MergeArea.Value = nome_ordenante
'Verificado

    Sheets("0795").Range("Form_Canal_banco_exterior").MergeArea.Value = "N/A"          'Verificado
    Sheets("0795").Range("Form_Canal_cidade_pais").MergeArea.Value = pais_ordenante    'Verificado
    Sheets("0795").Range("Form_Canal_iban").MergeArea.Value = "N/A"                  'Verificado
    Sheets("0795").Range("Form_Canal_swift_exterior").MergeArea.Value = "N/A"          'Verificado
    Sheets("0795").Range("Form_Canal_info_adicional").MergeArea.Value = info_adicional_ordenante
'Verificado

End If

If segmento_cliente = "VG/PF" Then

    Sheets("0795").Range("Form_VGPF").MergeArea.Value = "( X ) VG/PF"
    Sheets("0795").Range("Form_Select").MergeArea.Value = "( ) Select"
    Sheets("0795").Range("Form_Funcionario").MergeArea.Value = "( ) Funcionário"
    Sheets("0795").Range("Form_Diplomata").MergeArea.Value = "( ) Diplomata"
    Sheets("0795").Range("Form_FOPA_Ouro").MergeArea.Value = "( ) FOPA Ouro"
    Sheets("0795").Range("Form_FOPA_Prata").MergeArea.Value = "( ) FOPA Prata"
    Sheets("0795").Range("Form_FOPA_Bronze").MergeArea.Value = "( ) FOPA Bronze"

ElseIf segmento_cliente = "Select" Then

    Sheets("0795").Range("Form_VGPF").MergeArea.Value = "( ) VG/PF"
    Sheets("0795").Range("Form_Select").MergeArea.Value = "( X ) Select"
    Sheets("0795").Range("Form_Funcionario").MergeArea.Value = "( ) Funcionário"
    Sheets("0795").Range("Form_Diplomata").MergeArea.Value = "( ) Diplomata"
    Sheets("0795").Range("Form_FOPA_Ouro").MergeArea.Value = "( ) FOPA Ouro"

```

```

Sheets("0795").Range("Form_FOPA_Prata").MergeArea.Value = "( ) FOPA Prata"
Sheets("0795").Range("Form_FOPA_Bronze").MergeArea.Value = "( ) FOPA Bronze"
Elseif segmento_cliente = "Funcionário" Then
    Sheets("0795").Range("Form_VGPF").MergeArea.Value = "( ) VG/PF"
    Sheets("0795").Range("Form_Select").MergeArea.Value = "( ) Select"
    Sheets("0795").Range("Form_Funcionario").MergeArea.Value = "( X ) Funcionário"
    Sheets("0795").Range("Form_Diplomata").MergeArea.Value = "( ) Diplomata"
    Sheets("0795").Range("Form_FOPA_Ouro").MergeArea.Value = "( ) FOPA Ouro"
    Sheets("0795").Range("Form_FOPA_Prata").MergeArea.Value = "( ) FOPA Prata"
    Sheets("0795").Range("Form_FOPA_Bronze").MergeArea.Value = "( ) FOPA Bronze"
Elseif segmento_cliente = "Diplomata" Then
    Sheets("0795").Range("Form_VGPF").MergeArea.Value = "( ) VG/PF"
    Sheets("0795").Range("Form_Select").MergeArea.Value = "( ) Select"
    Sheets("0795").Range("Form_Funcionario").MergeArea.Value = "( ) Funcionário"
    Sheets("0795").Range("Form_Diplomata").MergeArea.Value = "( X ) Diplomata"
    Sheets("0795").Range("Form_FOPA_Ouro").MergeArea.Value = "( ) FOPA Ouro"
    Sheets("0795").Range("Form_FOPA_Prata").MergeArea.Value = "( ) FOPA Prata"
    Sheets("0795").Range("Form_FOPA_Bronze").MergeArea.Value = "( ) FOPA Bronze"
Elseif segmento_cliente = "FOPA Ouro" Then
    Sheets("0795").Range("Form_VGPF").MergeArea.Value = "( ) VG/PF"
    Sheets("0795").Range("Form_Select").MergeArea.Value = "( ) Select"
    Sheets("0795").Range("Form_Funcionario").MergeArea.Value = "( ) Funcionário"
    Sheets("0795").Range("Form_Diplomata").MergeArea.Value = "( ) Diplomata"
    Sheets("0795").Range("Form_FOPA_Ouro").MergeArea.Value = "( X ) FOPA Ouro"
    Sheets("0795").Range("Form_FOPA_Prata").MergeArea.Value = "( ) FOPA Prata"
    Sheets("0795").Range("Form_FOPA_Bronze").MergeArea.Value = "( ) FOPA Bronze"
Elseif segmento_cliente = "FOPA Prata" Then
    Sheets("0795").Range("Form_VGPF").MergeArea.Value = "( ) VG/PF"
    Sheets("0795").Range("Form_Select").MergeArea.Value = "( ) Select"
    Sheets("0795").Range("Form_Funcionario").MergeArea.Value = "( ) Funcionário"
    Sheets("0795").Range("Form_Diplomata").MergeArea.Value = "( ) Diplomata"
    Sheets("0795").Range("Form_FOPA_Ouro").MergeArea.Value = "( ) FOPA Ouro"
    Sheets("0795").Range("Form_FOPA_Prata").MergeArea.Value = "( X ) FOPA Prata"
    Sheets("0795").Range("Form_FOPA_Bronze").MergeArea.Value = "( ) FOPA Bronze"
Elseif segmento_cliente = "FOPA Bronze" Then
    Sheets("0795").Range("Form_VGPF").MergeArea.Value = "( ) VG/PF"

```

```

    Sheets("0795").Range("Form_Select").MergeArea.Value = "( ) Select"
    Sheets("0795").Range("Form_Funcionario").MergeArea.Value = "( ) Funcionário"
    Sheets("0795").Range("Form_Diplomata").MergeArea.Value = "( ) Diplomata"
    Sheets("0795").Range("Form_FOPA_Ouro").MergeArea.Value = "( ) FOPA Ouro"
    Sheets("0795").Range("Form_FOPA_Prata").MergeArea.Value = "( ) FOPA Prata"
    Sheets("0795").Range("Form_FOPA_Bronze").MergeArea.Value = "( X ) FOPA Bronze"

End If

'CLAUSULAS DO IR

If clausulas_ir = "1" Then
    Sheets("0795").Range("Form_clausulas_ir_A").MergeArea.Value = "( X )"
    Sheets("0795").Range("Form_clausulas_ir_B").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_C").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_D").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_E").MergeArea.Value = "( )"

Elseif clausulas_ir = "2" Then
    Sheets("0795").Range("Form_clausulas_ir_A").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_B").MergeArea.Value = "( X )"
    Sheets("0795").Range("Form_clausulas_ir_C").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_D").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_E").MergeArea.Value = "( )"

Elseif clausulas_ir = "3" Then
    Sheets("0795").Range("Form_clausulas_ir_A").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_B").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_C").MergeArea.Value = "( X )"
    Sheets("0795").Range("Form_clausulas_ir_D").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_E").MergeArea.Value = "( )"

Elseif clausulas_ir = "4" Then
    Sheets("0795").Range("Form_clausulas_ir_A").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_B").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_C").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_D").MergeArea.Value = "( X )"
    Sheets("0795").Range("Form_clausulas_ir_E").MergeArea.Value = "( )"

Else
    Sheets("0795").Range("Form_clausulas_ir_A").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_B").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_C").MergeArea.Value = "( )"

```



```

    Sheets("0795").Range("Form_clausulas_ir_D").MergeArea.Value = "( )"
    Sheets("0795").Range("Form_clausulas_ir_E").MergeArea.Value = "( X )"
End If
'IOF
If aliquota_iof = 0 Or aliquota_iof = "0" Then
    Sheets("0795").Range("Form_aliquota_iof").MergeArea.Value = 0
ElseIf aliquota_iof = "0,38" Or aliquota_iof = 0.0038 Then
    Sheets("0795").Range("Form_aliquota_iof").MergeArea.Value = 0.0038
ElseIf aliquota_iof = "1,10" Or aliquota_iof = 0.011 Then
    Sheets("0795").Range("Form_aliquota_iof").MergeArea.Value = 0.011
End If
On Error GoTo ERRO
pdfName = "Contrato_Cambio"
ChDir "C:\Users\Fernando\Downloads"
fileSaveName = "Contrato_Cambio" & ".pdf"
If fileSaveName <> False Then
    Sheets("0795").ExportAsFixedFormat Type:=xlTypePDF, FileName:= _
        fileSaveName _
        , Quality:=xlQualityStandard, IncludeDocProperties:=True, IgnorePrintAreas _
        :=False, OpenAfterPublish:=True
End If
Gerar_PDF.Hide
With Ultima_Tela
    .StartUpPosition = 0
    .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
    .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
    .Show
End With
Sheets("0795").Visible = False
Application.ScreenUpdating = True
Exit Sub
ERRO:
    MsgBox "Por favor feche o contrato anterior antes de gerar um novo.", vbExclamation, "Gerar PDF"
End Sub
Private Sub UserForm_Activate()
    lblSysVersion.Caption = SYSTEM_VERSION

```

```

End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)

    Dim formulario As UserForm

    For Each formulario In UserForms

        Unload formulario

    Next formulario

End Sub

```

7.1.10 Formulário de encerramento do programa

```

Private Sub Canal_Compra_Click()

End Sub

Private Sub CommandButton1_Click()

    Ultima_Tela.Hide

    With Gerar_PDF

        .StartPosition = 0

        .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)

        .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)

        .Show

    End With

End Sub

Private Sub CommandButton2_Click()

    Dim formulario As UserForm

    For Each formulario In UserForms

        Unload formulario

    Next formulario

End Sub

Private Sub UserForm_Activate()

    lblSysVersion.Caption = SYSTEM_VERSION

End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)

    Dim formulario As UserForm

    For Each formulario In UserForms

        Unload formulario

    Next formulario

End Sub

```

7.1.11 Módulo externo de declaração das variáveis globais

Public Nome_Funcionario, Email_Funcionario, Telefone_Funcionario, tipo_operacao *As String*,
exibir_clausula, Nome_Cliente, CPF_Cliente, Telefone_Cliente, Endereco_Cliente, Cidade_Cliente,
Pais_Cliente, Agencia_Cliente, Conta_Cliente, segmento_cliente, natureza_da_operacao, forma_entrega,
nome_beneficiario, banco_exterior, cidade_pais, iban, swift_exterior, info_adicional, nome_ordernante,
pais_ordernante, info_adicional_ordernante, clausulas_ir, origem_op, usuario *As String*

Public codigo_me, email_cliente, Extenso_ME *As String*, Extenso_MN *As String*, taxa_manual *As Boolean*

Public valor_operacao, valor_tarifa, valor_me, cotacao_dolar, taxa, aliquota_iof,
aliquota_iof_forma_entrega, vet, valor_iof *As Double*

Public KYC *As Integer*

Public Const SYSTEM_VERSION *As String* = "v20161128"

Sub Cotacao_Click()

On Error Resume Next

 usuario = Application.UserName

 'Escrever todas as variáveis no contrato de câmbio

 Sheets("0795").Unprotect

 'DADOS FUNCIONARIO

 Sheets("0795").Range("Form_Nome_Funcionario").MergeArea.Value = ""

 Sheets("0795").Range("Form_Telefone_Funcionario").MergeArea.Value = ""

 Sheets("0795").Range("Form_Email_Funcionario").MergeArea.Value = ""

 Sheets("0795").Range("Form_Compra").MergeArea.Value = "() Compra"

 Sheets("0795").Range("Form_Venda").MergeArea.Value = "() Venda"

 'DADOS CLIENTE

 Sheets("0795").Range("Form_Nome_Cliente").MergeArea.Value = ""

 Sheets("0795").Range("Form_CPF_Cliente").MergeArea.Value = ""

 Sheets("0795").Range("Form_Telefone_Cliente").MergeArea.Value = ""

 Sheets("0795").Range("Form_Endereco_Cliente").MergeArea.Value = ""

 Sheets("0795").Range("Form_Cidade_Cliente").MergeArea.Value = ""

 Sheets("0795").Range("Form_Pais_Cliente").MergeArea.Value = ""

 Sheets("0795").Range("Form_Agencia_Cliente").MergeArea.Value = ""

 Sheets("0795").Range("Form_Conta_Cliente").MergeArea.Value = ""

 Sheets("0795").Range("Form_email_cliente").MergeArea.Value = ""

 'DADOS OPERACAO

 Sheets("0795").Range("Form_codigo_me").MergeArea.Value = ""

 Sheets("0795").Range("Form_valor_me").MergeArea.Value = 0

 Sheets("0795").Range("Form_Extenso_ME").MergeArea.Value = ""

 Sheets("0795").Range("Form_taxa").MergeArea.Value = 0

```

Sheets("0795").Range("Form_valor_operacao").MergeArea.Value = 0
Sheets("0795").Range("Form_valor_tarifa").MergeArea.Value = 0
Sheets("0795").Range("Form_aliquota_iof").MergeArea.Value = ""
Sheets("0795").Range("Form_valor_iof").MergeArea.Value = ""
Sheets("0795").Range("Form_vet").MergeArea.Value = ""
Sheets("0795").Range("Form_forma_entrega").MergeArea.Value = ""
Sheets("0795").Range("Form_natureza_da_operacao").MergeArea.Value = ""
Sheets("0795").Range("Form_Extenso_MN").MergeArea.Value = ""
'DADOS CANAL BANCARIO
Sheets("0795").Range("Form_Canal_nome_beneficiario").MergeArea.Value = ""
Sheets("0795").Range("Form_Canal_banco_exterior").MergeArea.Value = ""
Sheets("0795").Range("Form_Canal_cidade_pais").MergeArea.Value = ""
Sheets("0795").Range("Form_Canal_iban").MergeArea.Value = ""
Sheets("0795").Range("Form_Canal_swift_exterior").MergeArea.Value = ""
Sheets("0795").Range("Form_Canal_info_adicional").MergeArea.Value = ""
Sheets("0795").Range("Form_VGPF").MergeArea.Value = "( ) VG/PF"
Sheets("0795").Range("Form_Select").MergeArea.Value = "( ) Select"
Sheets("0795").Range("Form_Funcionario").MergeArea.Value = "( ) Funcionário"
Sheets("0795").Range("Form_Diplomata").MergeArea.Value = "( ) Diplomata"
Sheets("0795").Range("Form_FOPA_Ouro").MergeArea.Value = "( ) FOPA Ouro"
Sheets("0795").Range("Form_FOPA_Prata").MergeArea.Value = "( ) FOPA Prata"
Sheets("0795").Range("Form_FOPA_Bronze").MergeArea.Value = "( ) FOPA Bronze"
Sheets("0795").Range("Form_clausulas_ir_A").MergeArea.Value = "( )"
Sheets("0795").Range("Form_clausulas_ir_B").MergeArea.Value = "( )"
Sheets("0795").Range("Form_clausulas_ir_C").MergeArea.Value = "( )"
Sheets("0795").Range("Form_clausulas_ir_D").MergeArea.Value = "( )"
Sheets("0795").Range("Form_clausulas_ir_E").MergeArea.Value = "( )"
SLAs.Show
With Dados_Func
    .StartUpPosition = 0
    .Left = Application.Left + (0.5 * Application.Width) - (0.5 * .Width)
    .Top = Application.Top + (0.5 * Application.Height) - (0.5 * .Height)
    .Show
End With
End Sub

```



```

.SavePassword = False
.SaveData = True
.AdjustColumnWidth = True
.RefreshPeriod = 0
.WebSelectionType = xlAllTables
.WebFormatting = xlWebFormattingNone
.WebPreFormattedTextToColumns = True
.WebConsecutiveDelimitersAsOne = True
.WebSingleBlockTextImport = False
.WebDisableDateRecognition = False
.WebDisableRedirections = False
.Refresh BackgroundQuery:=False
End With

Sheet11.Range("a1").QueryTable.Refresh BackgroundQuery:=False
Elseif InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0 Then
With Sheet9.QueryTables.Add(Connection:= _
"URL;file://xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx" _
, Destination:=Sheet9.Range("$A$1"))
.Name = "ESPECIE_CLIE"
.FieldNames = True
.RowNumbers = False
.FillAdjacentFormulas = False
.PreserveFormatting = True
.RefreshOnFileOpen = False
.BackgroundQuery = True
.RefreshStyle = xlOverwriteCells
.SavePassword = False
.SaveData = True
.AdjustColumnWidth = True
.RefreshPeriod = 0
.WebSelectionType = xlAllTables
.WebFormatting = xlWebFormattingNone
.WebPreFormattedTextToColumns = True
.WebConsecutiveDelimitersAsOne = True
.WebSingleBlockTextImport = False
.WebDisableDateRecognition = False

```

```

        .WebDisableRedirections = False

        .Refresh BackgroundQuery:=False

    End With

    Sheet9.Range("a1").QueryTable.Refresh BackgroundQuery:=False

    With Sheet10.QueryTables.Add(Connection:= _
        "URL;file://xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx" _
        , Destination:=Sheet10.Range("$A$1"))

        .Name = "ESPECIE_FUNC"

        .FieldNames = True

        .RowNumbers = False

        .FillAdjacentFormulas = False

        .PreserveFormatting = True

        .RefreshOnFileOpen = False

        .BackgroundQuery = True

        .RefreshStyle = xlOverwriteCells

        .SavePassword = False

        .SaveData = True

        .AdjustColumnWidth = True

        .RefreshPeriod = 0

        .WebSelectionType = xlAllTables

        .WebFormatting = xlWebFormattingNone

        .WebPreFormattedTextToColumns = True

        .WebConsecutiveDelimitersAsOne = True

        .WebSingleBlockTextImport = False

        .WebDisableDateRecognition = False

        .WebDisableRedirections = False

        .Refresh BackgroundQuery:=False

    End With

    Sheet10.Range("a1").QueryTable.Refresh BackgroundQuery:=False

End If

End Sub

```

7.1.13 Módulo de cálculo das taxas

Function consulta_taxa_ordem_consulado(ByVal moeda As String) As Double
Dim objMoed As Object, objDolar As Object


```

If tipo_operacao = "Compra" Then col = "e" Else col = "f"

With Sheet11

Set objDolar = .Range("a1:a200").Find(what:="USD", LookIn:=xlValues, MatchCase:=True)
Set objMoed = .Range("a1:a200").Find(what:=moeda, LookIn:=xlValues, MatchCase:=True)

If Not (objMoed Is Nothing) Then

    If moeda = "USD" Then

        If valor_me <= 30000 Then

            taxa = .Cells(objMoed.Row, col).Value

            If taxa > 1000 Then taxa = taxa / 1000

        ElseIf valor_me > 30000 And valor_me <= 60000 Then

            taxa = .Cells(objMoed.Row + 1, col).Value

            If taxa > 1000 Then taxa = taxa / 1000

        ElseIf valor_me > 60000 And valor_me < 100000 Then

            taxa = .Cells(objMoed.Row + 2, col).Value

            If taxa > 1000 Then taxa = taxa / 1000

        Else

            MsgBox "Erro - Para operações com valor à partir de USD 100.000,00 entrar em contato com a mesa.", vbExclamation, "Validação de parâmetros"

            taxa = 0

        End If

    Else

        cotacao_dolar = .Cells(objDolar.Row, col).Value

        If cotacao_dolar > 1000 Then cotacao_dolar = cotacao_dolar / 1000

        taxa = .Cells(objMoed.Row, col).Value

        If taxa > 1000 Then taxa = taxa / 1000

        If valor_me * taxa / cotacao_dolar <= 30000 Then

            taxa = taxa

            If taxa > 1000 Then taxa = taxa / 1000

        ElseIf valor_me * taxa / cotacao_dolar > 30000 And valor_me * taxa / cotacao_dolar <= 60000

Then

            taxa = .Cells(objMoed.Row + 1, col).Value

            If taxa > 1000 Then taxa = taxa / 1000

        ElseIf valor_me * taxa / cotacao_dolar > 60000 And valor_me * taxa / cotacao_dolar < 100000

Then

            taxa = .Cells(objMoed.Row + 2, col).Value

            If taxa > 1000 Then taxa = taxa / 1000

    End If

End With

```

```

Else

    MsgBox "Erro - Para operações com valor à partir de USD 100.000,00 entrar em contato com
a mesa.", vbExclamation, "Validação de parâmetros"

    taxa = 0

End If

End If

End If

End With

If taxa <> vbNullString Then If taxa > 1000 Then taxa = taxa / 1000

consulta_taxa_ordem_consulado = If(taxa = "", 0, taxa)

Set objMoed = Nothing

Set objDolar = Nothing

End Function

Function consulta_taxa_ordem_clie(ByVal moeda As String) As Double

    Dim objMoed As Object, objDolar As Object

    If tipo_operacao = "Compra" Then col = "e" Else col = "f"

    With Sheet3

        Set objDolar = .Range("a1:a200").Find(what:="USD", LookIn:=xlValues, MatchCase:=True)

        Set objMoed = .Range("a1:a200").Find(what:=moeda, LookIn:=xlValues, MatchCase:=True)

        If Not (objMoed Is Nothing) Then

            If moeda = "USD" Then

                If valor_me <= 30000 Then

                    taxa = .Cells(objMoed.Row, col).Value

                    If taxa > 1000 Then taxa = taxa / 1000

                ElseIf valor_me > 30000 And valor_me <= 60000 Then

                    taxa = .Cells(objMoed.Row + 1, col).Value

                    If taxa > 1000 Then taxa = taxa / 1000

                ElseIf valor_me > 60000 And valor_me < 100000 Then

                    taxa = .Cells(objMoed.Row + 2, col).Value

                    If taxa > 1000 Then taxa = taxa / 1000

                Else

                    MsgBox "Erro - Para operações com valor à partir de USD 100.000,00 entrar em contato com
a mesa.", vbExclamation, "Validação de parâmetros"

                    taxa = 0

                End If

            Else

                cotacao_dolar = .Cells(objDolar.Row, col).Value

            End If

        End If

    End With

End Function

```

```

    If cotacao_dolar > 1000 Then cotacao_dolar = cotacao_dolar / 1000
    taxa = .Cells(objMoed.Row, col).Value
    If taxa > 1000 Then taxa = taxa / 1000
    If valor_me * taxa / cotacao_dolar <= 30000 Then
        taxa = taxa
        If taxa > 1000 Then taxa = taxa / 1000
    ElseIf valor_me * taxa / cotacao_dolar > 30000 And valor_me * taxa / cotacao_dolar <= 60000
Then
        taxa = .Cells(objMoed.Row + 1, col).Value
        If taxa > 1000 Then taxa = taxa / 1000
    ElseIf valor_me * taxa / cotacao_dolar > 60000 And valor_me * taxa / cotacao_dolar < 100000
Then
        taxa = .Cells(objMoed.Row + 2, col).Value
        If taxa > 1000 Then taxa = taxa / 1000
    Else
        MsgBox "Erro - Para operações com valor à partir de USD 100.000,00 entrar em contato com
a mesa.", vbExclamation, "Validação de parâmetros"
        taxa = 0
    End If
End If
End If
End With
If taxa <> vbNullString Then If taxa > 1000 Then taxa = taxa / 1000
consulta_taxa_ordem_clie = If(taxa = "", 0, taxa)
Set objMoed = Nothing
Set objDolar = Nothing
End Function
Function consulta_taxa_ordem_func(ByVal moeda As String) As Double
    Dim objMoed As Object, objDolar As Object
    If tipo_operacao = "Compra" Then col = "c" Else col = "d"
    With Sheet5
        Set objDolar = .Range("a1:a200").Find(what:="USD", LookIn:=xlValues, MatchCase:=True)
        Set objMoed = .Range("a1:a200").Find(what:=moeda, LookIn:=xlValues, MatchCase:=True)
        If Not (objMoed Is Nothing) Then
            If moeda = "USD" Then
                If valor_me < 100000 Then
                    taxa = .Cells(objMoed.Row, col).Value
                    If taxa > 1000 Then taxa = taxa / 1000

```

```

Else

    MsgBox "Erro - Para operações com valor à partir de USD 100.000,00 entrar em contato com
a mesa.", vbExclamation, "Validação de parâmetros"

    taxa = 0

End If

Else

    cotacao_dolar = .Cells(objDolar.Row, col).Value

    If cotacao_dolar > 1000 Then cotacao_dolar = cotacao_dolar / 1000

    taxa = .Cells(objMoed.Row, col).Value

    If taxa > 1000 Then taxa = taxa / 1000

    If valor_me * taxa / cotacao_dolar < 100000 Then

        taxa = .Cells(objMoed.Row, col).Value

        If taxa > 1000 Then taxa = taxa / 1000

    Else

        MsgBox "Erro - Para operações com valor à partir de USD 100.00,00 entrar em contato com
a mesa.", vbExclamation, "Validação de parâmetros"

        taxa = 0

    End If

End If

End If

End With

If taxa <> vbNullString Then If taxa > 1000 Then taxa = taxa / 1000

consulta_taxa_ordem_func = If(taxa = "", 0, taxa)

Set objMoed = Nothing

Set objDolar = Nothing

End Function

Function consulta_taxa_especie_clie(ByVal moeda As String) As Double

    Dim objMoed As Object, objDolar As Object

    If tipo_operacao = "Compra" Then col = "e" Else col = "f"

    With Sheet9

        Set objDolar = .Range("a1:a200").Find(what:="USD", LookIn:=xlValues, MatchCase:=True)

        Set objMoed = .Range("a1:a200").Find(what:=moeda, LookIn:=xlValues, MatchCase:=True)

        If Not (objMoed Is Nothing) Then

            If moeda = "USD" Then

                If valor_me >= 1 And valor_me <= 4999 Then

                    taxa = .Cells(objMoed.Row - 1, col).Value

                    If taxa > 1000 Then taxa = taxa / 1000

                End If

            End If

        End If

    End With

End Function

```

```

ElseIf valor_me >= 5000 And valor_me <= 9999 Then
    taxa = .Cells(objMoed.Row + 0, col).Value
    If taxa > 1000 Then taxa = taxa / 1000
ElseIf valor_me >= 10000 And valor_me < 100000 Then
    taxa = .Cells(objMoed.Row + 1, col).Value
    If taxa > 1000 Then taxa = taxa / 1000
Else
    MsgBox "Erro - Para operações com valor à partir de USD 100.000,00 entrar em contato com
a mesa.", vbExclamation, "Validação de parâmetros"
    taxa = 0
End If

ElseIf moeda = "EUR" Then
    cotacao_dolar = .Cells(objDolar.Row, col).Value
    If cotacao_dolar > 1000 Then cotacao_dolar = cotacao_dolar / 1000
    taxa = .Cells(objMoed.Row, col).Value
    If taxa > 1000 Then taxa = taxa / 1000

    If valor_me * taxa / cotacao_dolar <= 4999 Then
        taxa = .Cells(objMoed.Row - 1, col).Value
        If taxa > 1000 Then taxa = taxa / 1000
    ElseIf valor_me * taxa / cotacao_dolar >= 5000 And valor_me * taxa / cotacao_dolar <= 9999
Then
        taxa = .Cells(objMoed.Row + 0, col).Value
        If taxa > 1000 Then taxa = taxa / 1000
    ElseIf valor_me * taxa / cotacao_dolar >= 10000 And valor_me * taxa / cotacao_dolar < 100000
Then
        taxa = .Cells(objMoed.Row + 1, col).Value
        If taxa > 1000 Then taxa = taxa / 1000
    Else
        MsgBox "Erro - Para operações com valor à partir de USD 100.000,00 entrar em contato com
a mesa.", vbExclamation, "Validação de parâmetros"
        taxa = 0
    End If
Else
    cotacao_dolar = .Cells(objDolar.Row, col).Value
    If cotacao_dolar > 1000 Then cotacao_dolar = cotacao_dolar / 1000
    taxa = .Cells(objMoed.Row, col).Value
    If taxa > 1000 Then taxa = taxa / 1000

```

```

    If valor_me * taxa / cotacao_dolar < 100000 Then
        taxa = taxa
    Else
        MsgBox "Erro - Para operações com valor à partir de USD 100.000,00 entrar em contato com a mesa.", vbExclamation, "Validação de parâmetros"
        taxa = 0
    End If
End If
End If
End With

If taxa <> vbNullString Then If taxa > 1000 Then taxa = taxa / 1000
consulta_taxa_especie_clie = If(taxa = "", 0, taxa)
Set objMoed = Nothing
Set objDolar = Nothing
End Function

Function consulta_taxa_especie_func(ByVal moeda As String) As Double
    Dim objMoed As Object, objDolar As Object
    If tipo_operacao = "Compra" Then col = "c" Else col = "d"
    With Sheet10
        Set objDolar = .Range("a1:a200").Find(what:="USD", LookIn:=xlValues, MatchCase:=True)
        Set objMoed = .Range("a1:a200").Find(what:=moeda, LookIn:=xlValues, MatchCase:=True)
        If Not (objMoed Is Nothing) Then
            If moeda = "USD" Then
                If valor_me < 100000 Then
                    taxa = .Cells(objMoed.Row, col).Value
                    If taxa > 1000 Then taxa = taxa / 1000
                Else
                    MsgBox "Erro - Para operações com valor à partir de USD 100.000,00 entrar em contato com a mesa.", vbExclamation, "Validação de parâmetros"
                    taxa = 0
                End If
            Else
                cotacao_dolar = .Cells(objDolar.Row, col).Value
                If cotacao_dolar > 1000 Then cotacao_dolar = cotacao_dolar / 1000
                taxa = .Cells(objMoed.Row, col).Value
                If taxa > 1000 Then taxa = taxa / 1000
                If valor_me * taxa / cotacao_dolar < 100000 Then

```

```

        taxa = .Cells(objMoed.Row, col).Value

        If taxa > 1000 Then taxa = taxa / 1000

    Else

        MsgBox "Erro - Para operações com valor à partir de USD 100.000,00 entrar em contato com a mesa.", vbExclamation, "Validação de parâmetros"

        taxa = 0

    End If

End If

End If

End With

If taxa <> vbNullString Then If taxa > 1000 Then taxa = taxa / 1000

consulta_taxa_especie_func = If(taxa = "", 0, taxa)

Set objMoed = Nothing

Set objDolar = Nothing

End Function

Sub pega_taxa()

    Call refresh_conn

    If segmento_cliente = "Funcionário" Or segmento_cliente = "FOPA Ouro" Then

        If codigo_me = "220 - USD - Dólar Americano" Then

            taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0, consulta_taxa_especie_func("USD"), consulta_taxa_ordem_func("USD"))

            ElseIf codigo_me = "978 - ECU - Euro" Then

                taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0, consulta_taxa_especie_func("EUR"), consulta_taxa_ordem_func("EUR"))

                ElseIf codigo_me = "470 - JPY - Iene Japonês" Then

                    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0, consulta_taxa_especie_func("JPY"), consulta_taxa_ordem_func("JPY"))

                    ElseIf codigo_me = "540 - GBP - Libra" Then

                        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0, consulta_taxa_especie_func("GBP"), consulta_taxa_ordem_func("GBP"))

                        ElseIf codigo_me = "425 - CHF - Franco Suíço" Then

                            taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0, consulta_taxa_especie_func("CHF"), consulta_taxa_ordem_func("CHF"))

                            ElseIf codigo_me = "165 - CAD - Dólar Canadense" Then

                                taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0, consulta_taxa_especie_func("CAD"), consulta_taxa_ordem_func("CAD"))

                                ElseIf codigo_me = "065 - NOK - Coroa Norueguesa" Then

                                    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0, consulta_taxa_especie_func("NOK"), consulta_taxa_ordem_func("NOK"))

                                    ElseIf codigo_me = "055 - DKK - Coroa Dinamarquesa" Then

```

```

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
Elseif codigo_me = "070 - SEK - Coroa Sueca" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_func("SEK"), consulta_taxa_ordem_func("SEK"))

Elseif codigo_me = "150 - AUD - Dólar Australiano" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_func("AUD"), consulta_taxa_ordem_func("AUD"))

Elseif codigo_me = "195 - SGD - Dólar de Singapura" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_func("SGD"), consulta_taxa_ordem_func("SGD"))

Elseif codigo_me = "741 - MXN - Pesos Mexicanos" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_func("MXN"), consulta_taxa_ordem_func("MXN"))

Elseif codigo_me = "642 - TRY - Lira Turca" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_func("TRY"), consulta_taxa_ordem_func("TRY"))

Elseif codigo_me = "205 - Dólar de Hong Kong" Then

    If tipo_operacao = "Compra" Then

        taxa = "ERRO - CONTATAR MESA"

    Elseif tipo_operacao = "Venda" Then

        taxa = "ERRO - CONTATAR MESA"

    End If

End If

Elseif segmento_cliente = "VG/PF" Then

    If codigo_me = "220 - USD - Dólar Americano" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_func("USD"), consulta_taxa_ordem_func("USD"))

Elseif codigo_me = "978 - ECU - Euro" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_func("EUR"), consulta_taxa_ordem_func("EUR"))

Elseif codigo_me = "470 - JPY - Iene Japonês" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_func("JPY"), consulta_taxa_ordem_func("JPY"))

Elseif codigo_me = "540 - GBP - Libra" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_func("GBP"), consulta_taxa_ordem_func("GBP"))

Elseif codigo_me = "425 - CHF - Franco Suíço" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_func("CHF"), consulta_taxa_ordem_func("CHF"))

Elseif codigo_me = "165 - CAD - Dólar Canadense" Then

```



```

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
        consulta_taxa_especie_clie("CAD"), consulta_taxa_ordem_clie("CAD"))

    ElseIf codigo_me = "065 - NOK - Coroa Norueguesa" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
        consulta_taxa_especie_clie("NOK"), consulta_taxa_ordem_clie("NOK"))

    ElseIf codigo_me = "055 - DKK - Coroa Dinamarquesa" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
        consulta_taxa_especie_clie("DKK"), consulta_taxa_ordem_clie("DKK"))

    ElseIf codigo_me = "070 - SEK - Coroa Sueca" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
        consulta_taxa_especie_clie("SEK"), consulta_taxa_ordem_clie("SEK"))

    ElseIf codigo_me = "150 - AUD - Dólar Australiano" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
        consulta_taxa_especie_clie("AUD"), consulta_taxa_ordem_clie("AUD"))

    ElseIf codigo_me = "195 - SGD - Dólar de Singapura" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
        consulta_taxa_especie_clie("SGD"), consulta_taxa_ordem_clie("SGD"))

    ElseIf codigo_me = "741 - MXN - Pesos Mexicanos" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
        consulta_taxa_especie_clie("MXN"), consulta_taxa_ordem_clie("MXN"))

    ElseIf codigo_me = "642 - TRY - Lira Turca" Then

        taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
        consulta_taxa_especie_clie("TRY"), consulta_taxa_ordem_clie("TRY"))

    ElseIf codigo_me = "205 - Dólar de Hong Kong" Then

        If tipo_operacao = "Compra" Then

            taxa = "ERRO - CONTATAR MESA"

        ElseIf tipo_operacao = "Venda" Then

            taxa = "ERRO - CONTATAR MESA"

        End If

    End If

    ElseIf segmento_cliente = "Select" Or segmento_cliente = "Diplomata" Or segmento_cliente = "FOPA
    Prata" Or segmento_cliente = "FOPA Bronze" Then

        If codigo_me = "220 - USD - Dólar Americano" Then

            taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
            consulta_taxa_especie_clie("USD"), consulta_taxa_ordem_consulado("USD"))

        ElseIf codigo_me = "978 - ECU - Euro" Then

            taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
            consulta_taxa_especie_clie("EUR"), consulta_taxa_ordem_consulado("EUR"))

        ElseIf codigo_me = "470 - JPY - Iene Japonês" Then

            taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
            consulta_taxa_especie_clie("JPY"), consulta_taxa_ordem_consulado("JPY"))

```

```

Elseif codigo_me = "540 - GBP - Libra" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_clie("GBP"), consulta_taxa_ordem_consulado("GBP"))

Elseif codigo_me = "425 - CHF - Franco Suíço" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_clie("CHF"), consulta_taxa_ordem_consulado("CHF"))

Elseif codigo_me = "165 - CAD - Dólar Canadense" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_clie("CAD"), consulta_taxa_ordem_consulado("CAD"))

Elseif codigo_me = "065 - NOK - Coroa Norueguesa" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_clie("NOK"), consulta_taxa_ordem_consulado("NOK"))

Elseif codigo_me = "055 - DKK - Coroa Dinamarquesa" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_clie("DKK"), consulta_taxa_ordem_consulado("DKK"))

Elseif codigo_me = "070 - SEK - Coroa Sueca" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_clie("SEK"), consulta_taxa_ordem_consulado("SEK"))

Elseif codigo_me = "150 - AUD - Dólar Australiano" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_clie("AUD"), consulta_taxa_ordem_consulado("AUD"))

Elseif codigo_me = "195 - SGD - Dólar de Singapura" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_clie("SGD"), consulta_taxa_ordem_consulado("SGD"))

Elseif codigo_me = "741 - MXN - Pesos Mexicanos" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_clie("MXN"), consulta_taxa_ordem_consulado("MXN"))

Elseif codigo_me = "642 - TRY - Lira Turca" Then

    taxa = If(InStr(forma_entrega, "Espécie") > 0 Or InStr(forma_entrega, "Traveller") > 0,
consulta_taxa_especie_clie("TRY"), consulta_taxa_ordem_consulado("TRY"))

Elseif codigo_me = "205 - Dólar de Hong Kong" Then

    If tipo_operacao = "Compra" Then

        taxa = "ERRO - CONTATAR MESA"

    Elseif tipo_operacao = "Venda" Then

        taxa = "ERRO - CONTATAR MESA"

    End If

End If

End If

End Sub

```

7.1.14 Módulo de conversão de numeral em texto

```
Public Function Extenso_Valor(valor As Double) As String
    Dim strMoeda As String
    Dim cents As Variant
    Dim decimalSep As String
    ' Se o valor for igual ou maior que 1 quatrilhao
    ' passar erro e sair da funcao
    If valor > 999999999999999# Then
        Extenso_Valor = "Valor excede 999.999.999.999.999"
        Exit Function
    End If
    ' Se valor for igual a 1, a unidade está no singular
    If WorksheetFunction.RoundDown(valor, 0) = 1 Then
        ' a string da moeda no singular
        strMoeda = " real"
        ' Se for maior que 1 a unidade está no plural
        Elseif WorksheetFunction.RoundDown(valor, 0) > 1 Then
            strMoeda = " reais"
        End If
    ' Remove os centavos
    cents = valor - WorksheetFunction.RoundDown(valor, 0)
    ' Remove os centavos do valor
    valor = valor - CDb1(cents)
    ' Passo o extenso dos centavos
    cents = centavos(CDb1(cents) * 100)
    ' End If
    ' Caso a string seja diferente de branco e valor seja maior ou igual a 1
    If cents <> "" And valor >= 1 Then
        ' acrescentar uma vírgula antes do extenso
        cents = " e " & cents
    End If
    ' Iniciar o processo de conversao dos valores longos
    strMoeda = Trim(Trilhoes(valor)) & strMoeda & cents
    strMoeda = Replace(strMoeda, ", e", " e")
    strMoeda = Replace(strMoeda, ", r", " r")
End Function
```

```

If Left(strMoeda, 2) = "e " Then
    strMoeda = Mid(strMoeda, 3, Len(strMoeda))
    'Elseif Left(strMoeda, 5) = "mil e" Then
        ' strMoeda = Mid(strMoeda, 5, Len(strMoeda))
End If

vzz = "00000000000000000000"
vtam = Len(Trim(Mid(Trim(valor), 2, 100)))

If Right(vzz + vzz + vzz + vzz, vtam) = Mid(Trim(valor), 2, 100) And InStr(UCase(strMoeda), UCase("es
")) > 0 Then
    vetor = Split(strMoeda, " ")
    vtrocar = vetor(UBound(vetor))
    strMoeda = Replace(strMoeda, vtrocar, "de " + vtrocar)
End If

Extenso_Valor = strMoeda
End Function

Private Function centavos(valor As Double) As String
    Dim dezena    As Integer
    Dim unidade   As Integer
    ' Passa o valor para base decimal
    valor = Round(CDbl(valor / 100), 2)
    ' Se for um centavo, escrever valor e sair da funcao
    If valor = 0.01 Then
        centavos = "um centavo"
        Exit Function
    End If
    ' Repassa valor para dezenas
    valor = valor * 100
    ' Se nao houver dezenas no valor passado
    If dezenas(valor) = "" Then
        ' a string centavos fica em branco
        centavos = ""
    Else
        ' caso contrário, passar extenso das dezenas e concatenar
        ' com a palavra centavos
        centavos = dezenas(valor) & " centavos"
    End If
End Function
End Function

```

Private Function unidades(unidade As Double) As String

Dim unid(9)

' Define as unidades a serem usadas

unid(1) = "um": unid(6) = "seis"

unid(2) = "dois": unid(7) = "sete"

unid(3) = "três": unid(8) = "oito"

unid(4) = "quatro": unid(9) = "nove"

unid(5) = "cinco"

' Retorna a string referente a unidade passada para

' esta funcao

unidades = Trim(unid(unidade))

End Function

Private Function dezenas(dezena As Double) As String

Dim dezes(9)

Dim dez(9)

Dim intDezena As Double

Dim intUnidade As Double

Dim tmpStr As String

' Define as dezenas a serem utilizadas

dezes(1) = "onze": dezes(6) = "dezesesseis"

dezes(2) = "doze": dezes(7) = "dezessete"

dezes(3) = "treze": dezes(8) = "dezoito"

dezes(4) = "quatorze": dezes(9) = "dezenove"

dezes(5) = "quinze"

dez(1) = "dez": dez(6) = "sessenta"

dez(2) = "vinte": dez(7) = "setenta"

dez(3) = "trinta": dez(8) = "oitenta"

dez(4) = "quarenta": dez(9) = "noventa"

dez(5) = "cinquenta"

' Calcula o inteiro da dezena

intDezena = Int(dezena / 10)

' Calcula o inteiro da unidade

intUnidade = dezena Mod 10

' Se o inteiro da dezena for zero

If intDezena = 0 Then

' dezenas sao iguais as unidades

```

    dezenas = unidades(intUnidade)

Exit Function

Else:
' caso contrário, é igual a dez
    dezenas = dez(intDezena)

End If

' Se o inteiro da dezena for igual a 1 e
' o inteiro da unidade for zero, os valores estão
' entre 11 e 19
    If (intDezena = 1 And intUnidade > 0) Then
        dezenas = dezes(intUnidade)
    Else
' Caso contrário, valor está entre 20 e 90 inclusive
        If (intDezena > 1 And intUnidade > 0) Then
' Concatena a string da dezena com a string da unidade
            dezenas = dezenas & " e " & unidades(intUnidade)
        End If
    End If

    dezenas = dezenas

End Function

Private Function centenas(centena As Double) As String
    Dim tmpCento As Double
    Dim tmpDez As Double
    Dim tmpUni As Double
    Dim tmpUniMod As Double
    Dim tmpModDez As Double
    Dim centoString As String
    Dim cento(9)

' Define as centenas
    cento(1) = "cento": cento(6) = "seiscentos"
    cento(2) = "duzentos": cento(7) = "setecentos"
    cento(3) = "trezentos": cento(8) = "oitocentos"
    cento(4) = "quatrocentos": cento(9) = "novecentos"
    cento(5) = "quinhentos"

' Calcula o inteiro da centena
    tmpCento = Int(centena / 100)

```

```

' Calcula a parte da dezena
tmpDez = centena - (tmpCento * 100)

' Calcula o inteiro da unidade
tmpUni = Int(tmpDez / 10)

' Calcula o resto da unidade
tmpUniMod = tmpUni Mod 10

' Calcula o resto da dezena
tmpModDez = tmpDez Mod 10

' Se centena for cem, definir string como "cem " e sair
If centena = 100 Then
    centoString = "cem "
Else
' Caso contrário definir a string da centena
    centoString = cento(tmpCento)
End If

' Avalia se a unidade é maior ou igual a zero, se o resto da unidade é igual ou
' maior que zero, se a dezena é maior ou igual a um e se a centena é igual ou
' maior que 1. Se forem verdadeiros; então, adicionar " e " a string da centena
If (tmpUni >= 0 And tmpUniMod >= 0 And tmpDez >= 1 And tmpCento >= 1) Then
    centoString = centoString & " e "
End If

' Concatena a string do cento com a string da dezena
centenas = Trim(centoString & dezenas(tmpDez))

End Function

Private Function milhares(milhar As Double) As String
    Dim tmpMilhar As Double
    Dim tmpCento As Double
    Dim milString As String

' Calcula o inteiro da milhar
    tmpMilhar = Int(milhar / 1000)

' Calcula o cento dentro da milhar
    tmpCento = milhar - (tmpMilhar * 1000)

' Se milhar for zero, então a string da milhar fica em branco
    If tmpMilhar = 0 Then milString = ""

' Se for igual a 1, então
' If (tmpMilhar = 1) Then

```

```

' string da milhar é igual a unidade e "mil"
  'milString = unidades(tmpMilhar) & "um mil "
' se maior que 1 e menor que dez, string igual a unidades
If (tmpMilhar >= 1 And tmpMilhar < 10) Then
    milString = unidades(tmpMilhar) & " mil, "
' Se for entre 10 e 100, entao string igual a dezenas
Elseif (tmpMilhar >= 10 And tmpMilhar < 100) Then
    milString = dezenas(tmpMilhar) & " mil, "
' Se for entre 100 e 1000, entao igual string centenas
Elseif (tmpMilhar >= 100 And tmpMilhar < 1000) Then
    milString = centenas(tmpMilhar) & " mil, "
End If
'If tmpCento = 1 Then milString = " e "
If (tmpCento >= 1 And tmpCento <= 100) Then milString = milString & "e "
milhares = Trim(milString & centenas(tmpCento))
End Function

Private Function milhoes(milhao As Double) As String
' Ver comentários para milhares acima
Dim tmpMilhao As Double
Dim tmpMilhares As Double
Dim miString As String
tmpMilhao = Int(milhao / 1000000)
tmpMilhares = milhao - (tmpMilhao * 1000000)
If tmpMilhao = 0 Then miString = ""
If (tmpMilhao = 1) Then
    miString = unidades(tmpMilhao) & " milhão, "
Elseif (tmpMilhao > 1 And tmpMilhao < 10) Then
    miString = unidades(tmpMilhao) & " milhões, "
Elseif (tmpMilhao >= 10 And tmpMilhao < 100) Then
    miString = dezenas(tmpMilhao) & " milhões, "
Elseif (tmpMilhao >= 100 And tmpMilhao < 1000) Then
    miString = centenas(tmpMilhao) & " milhões, "
End If
If milhao = 1000000# Then miString = "um milhão de "
milhoes = Trim(miString & milhares(tmpMilhares))
End Function

```



```

Private Function bilhoes(bilhao As Double) As String
' Ver comentários para milhares acima
    Dim tmpBilhao    As Double
    Dim tmpMilhao    As Double
    'Dim tmpMilhoes   As Double
    Dim biString     As String
    tmpBilhao = Int(bilhao / 1000000000)
    tmpMilhao = bilhao - (tmpBilhao * 1000000000)
    If (tmpBilhao = 1) Then
        biString = unidades(tmpBilhao) & " bilhão, "
    ElseIf (tmpBilhao > 1 And tmpBilhao < 10) Then
        biString = unidades(tmpBilhao) & " bilhões, "
    ElseIf (tmpBilhao >= 10 And tmpBilhao < 100) Then
        biString = dezenas(tmpBilhao) & " bilhões, "
    ElseIf (tmpBilhao >= 100 And tmpBilhao < 1000) Then
        biString = centenas(tmpBilhao) & " bilhões, "
    End If
    If bilhao = 1000000000# Then biString = "um bilhão de "
    bilhoes = Trim(biString & milhoes(tmpMilhao))
End Function

Private Function Trilhoes(Trilhao As Double) As String
' Ver comentários para milhares acima
    Dim tmpTrilhao    As Double
    Dim tmpBilhao     As Double
    Dim triString     As String
    tmpTrilhao = Int(Trilhao / 1000000000000#)
    tmpBilhao = Trilhao - (tmpTrilhao * 1000000000000#)
    If (tmpTrilhao = 1) Then
        triString = unidades(tmpTrilhao) & " trilhão, "
    ElseIf (tmpTrilhao > 1 And tmpTrilhao < 10) Then
        triString = unidades(tmpTrilhao) & " trilhões, "
    ElseIf (tmpTrilhao >= 10 And tmpTrilhao < 100) Then
        triString = dezenas(tmpTrilhao) & " trilhões, "
    ElseIf (tmpTrilhao >= 100 And tmpTrilhao < 1000) Then
        triString = centenas(tmpTrilhao) & " trilhões, "
    End If

```

```

    If Trilhao = 1000000000000# Then triString = "um trilhão de "
    Trilhoes = Trim(triString & bilhoes(tmpBilhao))
End Function

Function arredBaixo(valor)

    Dim tmpValor

    tmpValor = Round(CDbl(Right(Round(valor, 2) * 100, 2)) / 100, 2)

    arredBaixo = Round(Round(valor, 2) - tmpValor, 0)
End Function

Public Sub NavigateToURL(ByVal argURL As String, Optional argVisible As Boolean)

    On Error GoTo err

    Const READYSTATE_COMPLETE As Integer = 4

    Dim objIE As Object

    Set objIE = CreateObject("InternetExplorer.Application")

    With objIE

        .Visible = False

        .Silent = True

        .Navigate argURL

        Do Until .ReadyState = READYSTATE_COMPLETE

            DoEvents

        Loop

    End With

    If argVisible = False Then

        objIE.Quit

    Else

        objIE.Visible = True

    End If

ext:

    Set objIE = Nothing

    Exit Sub

err:

    GoTo ext

End Sub

Public Sub OpenHelpFile()

    On Error GoTo err

    Dim objPPT As Object

    Set objPPT = CreateObject("PowerPoint.Application")

```

```

objPPT.Visible = True

objPPT.Presentations.Open ActiveWorkbook.Path & "\Formulario_0795.ppt", ReadOnly:=True

MsgBox "ATENÇÃO " & vbCrLf & vbCrLf & "Verifique se um documento de ajuda foi carregado." &
vbCrLf & _

"(Formato PowerPoint)", vbInformation, ". : Formulário 0795 : ."

ext:

Set objPPT = Nothing

Exit Sub

err:

GoTo ext

End Sub

```