



## Bankee: modelação e implementação de transferências interbancárias com Business Process Management

ORLANDO LUÍS LOPES NEVES

Outubro de 2017

# **Bankee: modelação e implementação de transferências interbancárias com Business Process Management**

**Orlando Luís Lopes Neves | 1101166**

**Dissertação para obtenção do Grau de Mestre em  
Engenharia Informática, Área de Especialização em  
Sistemas Gráficos e Multimédia**

**Orientadora: Professora Doutora Isabel Azevedo**

**Supervisor: Engenheiro Patrick Machado**

Porto, outubro 2017



# Resumo

Atualmente, com o aumento exponencial de competitividade nos mercados, as empresas que se queiram manter na linha da frente devem adotar uma posição crítica e proativa na análise e otimização dos seus processos de negócio.

Face às inegáveis complexidade, competitividade e volatilidade do sector, a banca é um dos principais segmentos onde se recorre a Business Process Management (BPM) para cumprir com diferentes metas para a fluidez e segurança do negócio, recorrendo a técnicas desenvolvidas de forma a agilizarem os seus processos e a reduzirem o risco operacional inerente à sua atividade.

Neste sentido, foi realizado um estudo de caracterização na área de domínio de BPM tendo por base uma aplicação de *frontend* bancário em desenvolvimento para uma instituição bancária angolana. Objetivamente procedeu-se à identificação de diferentes padrões de modelação de processos de negócio, e de integração, orientados à resolução de problemas operacionais relacionados com a falta de intervenção de diferentes níveis hierárquicos, durante o processo de transferências interbancárias, com o intuito de reduzir o risco de fraude interna. Como resultado deste estudo desenvolveram-se todos os componentes do módulo de transferências interbancárias, bem como o modelo de processo de negócio em BPMN 2.0 (Business Process Model and Notation) que integra a funcionalidade que permite a intervenção dinâmica de diferentes indivíduos da hierarquia da instituição em pontos chave do processo de negócio.





# Abstract

Nowadays, with the exponential increasing in markets competitiveness, companies that want to remain at the forefront must adopt a critical and proactive position in the analysis and optimization of their business processes.

In view of the undeniable complexity, competitiveness and volatility of the sector, banking is one of the main segments where Business Process Management (BPM) is used to meet different goals for business fluency and security, using techniques developed to streamline processes and reduce the operational risk inherent in its activity.

In this sense, a characterization study was carried out in the BPM domain area, based on a bank frontier application under development for an Angolan banking institution. Objectively, different patterns of business process modeling and integration were identified, aimed at solving operational problems related to the lack of intervention at different hierarchical levels, during the interbank transfer process, in order to reduce the risk of internal fraud. As a result of this study, all the components of the interbank transfer module were developed, as well as the business process model in BPMN 2.0 (Business Process Model and Notation) that integrates the functionality to allow dynamic intervention of different individuals in the hierarchy of the institution key points of the business process.

**Keywords:** *Business Process Management, BPM, BPMN 2.0, hierarchical intervention, banking, interbank transfers*



# Agradecimentos

A entrega deste trabalho marca o término de um desafio que se prevê cumprido e para o qual muitas pessoas direta, ou indiretamente, tiveram um papel importante.

O maior reconhecimento vai para quem abdicou diariamente da minha companhia para que pudesse cumprir este desafio com sucesso, contribuindo com o apoio logístico, carinho e compreensão, a minha esposa Paula.

Aos meus pais, Luís e Paula e à minha sogra Maria Alice pelo crescente reconhecimento e apoio incondicional.

À Professora Doutora Isabel Azevedo, minha orientadora, pela energia dispendida e pelo empenho e dedicação na condução deste trabalho.

Ao Engenheiro Patrick Machado. Mais do que um supervisor, um amigo que contribui diariamente para o meu desenvolvimento pessoal e profissional de forma genuinamente altruísta.

À Critical Software e todas as pessoas desta imensa instituição pela forma como diariamente me recebem e contribuem para o desenvolvimento deste e de outros trabalhos.

Ao ISEP, a instituição de ensino que providenciou os meios que me permitem atingir este fim.

A todos os meus amigos que, de forma solidária, abdicaram da minha companhia durante estes últimos anos.

Por ti, Bernardo, que estás a chegar.



# Conteúdos

<b>1. Introdução</b>	<b>1</b>
1.1. Contexto	1
1.2. Problema	2
1.3. Objetivo	4
1.4. Abordagem Preconizada	5
1.5. Organização	5
<b>2. Estado da Arte</b>	<b>7</b>
2.1. Enquadramento Teórico	7
2.1.1. Banca	7
2.1.2. A Banca Angolana	8
2.1.3. Transferência Interbancária	9
2.1.4. Business Process Re-Engineering na Indústria Bancária	10
2.2. Enquadramento Tecnológico	11
2.2.1. Business Process Management	11
2.2.2. Business Process Modeling	11
2.2.3. Notações	12
2.2.4. Aspect Oriented BPM	17
2.2.5. Sistema de Gestão de Processos	19
<b>3. Análise de Negócio</b>	<b>23</b>
3.1. Aplicação Bankee	23
3.2. Atores	24
3.3. Requisitos Funcionais	24
3.3.1. Pressupostos	24
3.3.2. Casos de Uso	25
3.4. Requisitos Não Funcionais	29
3.5. Modelação do Domínio	32
3.6. Modelação Preliminar do Processo de Negócio	33
<b>4. Tecnologias Adotadas</b>	<b>37</b>
4.1. Tecnologias de Frontend	37
4.1.1. Tecnologia em Uso	37
4.1.2. Alternativas	38
4.2. Tecnologias de Backend	41
4.2.1. Tecnologia em Uso	41
4.2.2. Alternativa	44
4.3. Soluções de BPM	46
<b>5. Design Arquitetural</b>	<b>49</b>
5.1. Arquitetura da Aplicação Bankee	49
5.2. Limites de Autorização	50
5.2.1. Arquitetura do Módulo de Limites de Autorização	55

<b>6. Design e Implementação .....</b>	<b>57</b>
6.1. Tecnologias .....	57
6.1.1. Infraestrutura .....	57
6.1.2. Ferramentas de Desenvolvimento .....	59
6.2. Metodologia .....	62
6.3. Produto Final .....	64
6.3.1. Alternativas .....	64
6.3.2. Padrões Adotados .....	65
6.3.3. Modelo Final do Processo .....	69
6.3.4. Ecrãs e Navegação .....	93
6.3.5. Serviços .....	96
<b>7. Análise de Valor .....</b>	<b>99</b>
7.1. Elementos Chave do Modelo NCD .....	100
7.2. Proposta de Valor .....	102
7.2.1. Modelo de Negócio <i>Canvas</i> .....	103
7.3. Rede de Valor .....	103
7.3.1. Valor Percebido .....	104
<b>8. Avaliação .....</b>	<b>107</b>
8.1. Testes em Ambientes de Qualidade Internos .....	107
8.2. Testes de Aceitação do Cliente .....	114
8.3. Avaliação .....	115
8.3.1. Avaliação Ambiente Qualidade Interno .....	115
8.3.2. Avaliação no Cliente .....	119
<b>9. Conclusão .....</b>	<b>121</b>
9.1. Objetivos Alcançados .....	121
9.2. Contributos .....	121
9.3. Limitações e Trabalho Futuro .....	122
9.4. Apreciação final .....	123

# Lista de Figuras

Figura 1 – Alguns Elementos da notação BPMN .....	13
Figura 2 - Diferentes camadas da estrutura de BPMN 2.0.....	14
Figura 3 – Notação e Modelo de Processo de Negócio BPMN 2.0 .....	15
Figura 4 – Elementos da notação EPC.....	17
Figura 5 - Aspectos de negócio num modelo bancário de BPM .....	18
Figura 6 - Modelação no processo de transferência bancária .....	18
Figura 7 - Processo de transferência seguindo o princípio orientado ao aspeto.....	19
Figura 8 - O ciclo de vida de BPM.....	21
Figura 9 - Diagrama de casos de uso para o módulo proposto.....	25
Figura 10 – Modelo de domínio .....	32
Figura 11 - Diagrama do processo de negócio de transferências interbancárias, visão geral ....	33
Figura 12 - Diagrama da 1ª atividade, "Registar transferência" .....	34
Figura 13 - Diagrama da atividade de "Confirmar Envio Para Aprovação" .....	34
Figura 14 - Diagrama da atividade "Aprovar Transferência". Incluída na intervenção hierárquica dinâmica .....	35
Figura 15 - Diagrama da atividade "Validar Transferência". Incluída na intervenção hierárquica dinâmica .....	35
Figura 16 - Arquitetura MVC .....	39
Figura 17 - Alguns componentes de IceFaces .....	40
Figura 18 - Votação online informal no website DevRates.com .....	41
Figura 19 - Índice TIOBE com o top 10 das linguagens de <i>backend</i> .....	42
Figura 20 - Arquitetura JVM com os principais subsistemas e áreas de memória .....	43
Figura 21 - <i>Stack</i> da <i>framework</i> .NET .....	45
Figura 22- Arquitetura alto nível do módulo de transferências interbancárias.....	49
Figura 23 - Exemplo de parametrização dos limites de autorização para o grupo de autorizações de transferências .....	52
Figura 24 - Arquitetura alto nível do módulo de limites de autorização .....	56
Figura 25 - Comunicação da infraestrutura de qualidade e pré-produção na aplicação Bankee58	
Figura 26 – Visão geral do <i>IBM Process Designer</i> .....	60
Figura 27 – Fases do desenvolvimento da metodologia <i>Waterfall</i> .....	63
Figura 28 – Detalhe diagrama de colaboração limites de autorização .....	65
Figura 29 – <i>Sequence Pattern</i> representada no modelo.....	66
Figura 30 – Exemplo do padrão de <i>Deferred Choice</i> . .....	67
Figura 31 – <i>Structured loop pattern</i> no sub-processo de limites de autorização .....	68
Figura 32 - <i>Implicit Termination pattern</i> no sub-processo de associação de documentos .....	68
Figura 33 – Diagrama final do processo de transferências interbancárias .....	70
Figura 34 – Diagrama da primeira atividade, “Registar Transferência” .....	71
Figura 35 – Diagrama da atividade “Confirmar Envio para Aprovação” .....	72
Figura 36 – Enquadramento do subprocesso de limites de autorização no processo geral.....	74
Figura 37 – Diagrama geral do subprocesso de limites de autorização .....	75
Figura 38 – Detalhe do subprocesso de limites de autorização.....	76
Figura 39 – Cadeia de níveis de intervenção.....	77
Figura 40 -Diagrama geral da atividade “Autorizar” .....	78
Figura 41 – Diagrama geral da atividade “Validar” .....	79
Figura 42 – Diagrama geral da atividade “Dar Parecer” .....	80



Figura 43 – Diagrama da atividade “Tomar Conhecimento” .....	81
Figura 44 – Diagrama da atividade de “Erro Intervenção Não Suportada .....	82
Figura 45 – Exemplo de implementação de um <i>General System Service</i> .....	83
Figura 46 – Excerto da primeira atividade do modelo preliminar do processo de negócio .....	84
Figura 47 – Impressão de certificação. Planeado vs. Atual. ....	85
Figura 48 – Obtenção de níveis de intervenção. Planeado vs. Atual. ....	85
Figura 49 – Diagrama da atividade serviço de sistema “AgrupaServicosNiveisIntervencao” .....	86
Figura 50 - Limites de autorização. Planeado vs. Atual.....	87
Figura 51 – Primeira metade do subprocesso de limites de autorização. ....	88
Figura 52 – Diagrama da atividade de sistema “Inicializa Transacao Limites” .....	89
Figura 53 – Detalhe do script “Inicializa Dados” .....	89
Figura 54 – Segunda metade do subprocesso de limites de autorização. ....	90
Figura 55 – Subprocesso de Cadeia de Intervenção. ....	91
Figura 56 – Diagrama da atividade “Finalizar Limites Autorizacao” .....	92
Figura 57 – Ecrã de inserção de dados da transferência.....	93
Figura 58 – Ecrã de validação dos dados da transferência. ....	94
Figura 59 – Ecrã de validação de assinaturas.....	94
Figura 60 – Ecrã de confirmação do pedido de transferência .....	95
Figura 61 – Ecrã de aprovação de transferência interbancária .....	95
Figura 62 – Ecrã para registo de parecer .....	96
Figura 63 – Excerto de método de detalhe no WebService.....	97
Figura 64 – Excerto de método transacional do serviço de transferências .....	98
Figura 65 - Representação gráfica do modelo NCD .....	100
Figura 66 - Canvas do modelo de negócio do projeto Bankee.....	103
Figura 67 - Rede de valor de Verna Allee aplicada ao modelo de negócio da <i>Critical Software</i> .....	104
Figura 68 – Exemplo de um defeito blocker .....	116
Figura 69 – Total de defeitos do módulo por prioridade.....	116
Figura 70 – Distribuição normal de defeitos por prioridade.....	117
Figura 71 – Horas gasto na resolução de problemas .....	118
Figura 72 – Tempo de resolução de <i>issues</i> por prioridade .....	118
Figura 73 – Quantidade de <i>issues</i> externos por prioridade .....	119
Figura 74 – Tempo total de resolução dos <i>issues</i> externos .....	120

# Lista de Tabelas

<i>Tabela 1 - Casos de uso do ator "Balconista"</i> .....	26
<i>Tabela 2 - Casos de uso do ator "Aprovador"</i> .....	27
<i>Tabela 3 - Casos de uso do ator "Validador"</i> .....	28
<i>Tabela 4 - Casos de uso do ator "Consultor"</i> .....	29
Tabela 5 – Características da estação de trabalho .....	29
Tabela 6 – Características do Browser .....	30
Tabela 7 – Resolução de ecrã .....	30
Tabela 8 – Gestão de envio de documentos .....	30
Tabela 9 – Tratamento de assinaturas .....	31
Tabela 10 – Acessibilidade da aplicação Bankee .....	31
Tabela 11 - IBM BPM vs. Oracle BPM .....	46
Tabela 12 - Tipos de Intervenção possíveis .....	54
Tabela 13 - Alguns exemplos de ações dos limites de autorização .....	55
Tabela 14 - Benefícios e sacrifícios da proposta de valor .....	105
<i>Tabela 15 - Especificação de testes transferências interbancárias com autorização aleatória e recusa de assinaturas</i> .....	108
<i>Tabela 16 - Especificação de testes. Processo Transferência Interbancária com recusa na autorização</i> .....	109
<i>Tabela 17 - Especificação de testes. Transferência Interbancária com diferentes níveis de intervenção hierárquica.</i> .....	110
Tabela 18 - Testes de aceitação para o fluxo do processo com diferentes níveis de intervenção, incluindo a dada de parecer desfavorável e favorável .....	111
Tabela 19. Inventário de ativos intelectuais .....	122

# Lista de Acrónimos e Siglas

ABI	Application Binary Interface
API	Application Programming Interface
BPM	Business Process Management
BPMI	Business Process Management Initiative
BPMN	Business Process Model and Notation
BPMS	Business Process Management System
BSE	Business Rules Engines
CSW	Critical Software
DAO	Data Access Object
DO	Depósito à ordem
DOM	Document Object Model
DSI	Direção de Serviços de Informática
EPC	Event-driven Process Chain
FCL	Framework Class Library
FE	Front-End Module (Módulo de apresentação)
IaC	Infrastructure as Code
IBPM	IBM Business Process Management
IDE	Integrated Development Environment
INT	Integration Module (Módulo de integração)
IT	Information Technology
JDBC	Java Database Connectivity
JSF	Java Server Faces
JVM	Java Virtual Machine
LGPL	Lesser General Public License
MVC	Model View Controller
OMG	Object Management Group
PERT	Program evaluation and review technique
REST	Representational State Transfer

RPG	Report Program Generator
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
UI	User Interface
WAS	Websphere Application Server
WOCA	Write Once Compile Anywhere
WORA	Write Once, Run Anywhere
XHTML	Extensible Hypertext Markup Language
XUL	Extensible User-interface Language
ZUML	ZK User Interface Markup Language



# 1. Introdução

Neste capítulo contextualiza-se o trabalho a realizar com uma descrição do problema que se pretende resolver, os objetivos que se pretendem atingir, a abordagem preconizada e a estrutura em que se organiza este documento.

## 1.1. Contexto

Atualmente, os mercados de valores ditam que as organizações que queiram estar na linha da frente da sua área de negócio, devem procurar implementar ativamente uma gestão estratégica dos seus recursos, recorrendo a princípios de otimização dos seus processos de negócio.

É no início do século XX que *Henry Laurence Gantt*, com a criação dos Diagramas de *Gantt*, dá os primeiros passos na modelação de processos de negócio. Entretanto, novas técnicas foram surgindo ao longo desse século, tais como os Diagramas de Fluxo (1920), Diagramas de Fluxo de Blocos Funcionais e Diagramas PERT (ambos em 1950), entre outras. Assim, foram criados os alicerces para novas técnicas de modelação, automação, execução, controlo, medição e otimização de fluxos em atividades de negócio, que mais tarde, e já no início do século XXI, viria a ser chamado de *Business Process Management (BPM)*.

Por um lado, BPM pode ser considerado como uma evolução do conceito de *Workflow Management (WFM)*, focado essencialmente na automatização de processos de negócio. Contudo, o BPM possui um âmbito mais abrangente que passa também pela análise dos processos, gestão de operações, integração com sistemas de informação, redução de riscos operacionais e organização do trabalho. O BPM pode ser completamente dissociado da tecnologia, concentrando-se somente no aprimorar dos processos de negócio das organizações, no desenvolvimentos e implementação de modelos mais eficazes, recorrendo à capacidade de análise e simulação, permitindo que os cargos de gestão e administração sejam capazes de congeminar ideias para aumentar a produtividade e qualidade dos seus serviços e/ou produtos, ao mesmo tempo que lhes permite reduzir custos operacionais.

Também a emergente evolução tecnológica dos últimos anos permitiu que o BPM passasse a ser frequentemente associado a software que permite implementar, controlar, manter, expandir e dar suporte a processos de negócio, com integração de uma enorme variedade de tecnologias atuais, com todas as técnicas de modelação de processos operacionais já referidos, mas com a vantagem de que os sistemas de informação baseados em BPM se tornam mais ágeis, acompanhando a tendência imposta pela rápida ascensão tecnológica dos dias de hoje.

Assim, *Business Process Management* tornou-se uma disciplina madura com um conjunto de princípios bem estabelecidos, assim como métodos e ferramentas que combinam o conhecimento em tecnologias de informação, gestão e engenharia industrial com o propósito de aperfeiçoar os processos de negócio” [5].

Atualmente, a grande maioria das organizações, desde microempresas a multinacionais, adotaram a abordagem de olhar para os seus processos de negócio de forma crítica. Este facto tem impulsionado a aplicabilidade de BPM no desenvolvimento de novas técnicas de modelação de processo, estendendo-a a áreas críticas de negócio. Face às inegáveis complexidade, competitividade e volatilidade do sector, a banca é um dos principais segmentos onde se recorre a BPM para cumprir com diferentes metas para a fluidez e segurança do negócio, recorrendo a

técnicas desenvolvidas de forma a agilizarem os seus processos e a reduzirem o risco operacional inerente à sua atividade.

Cada vez mais as instituições bancárias têm de se adaptar às realidades sociais e culturais das zonas geográficas em que estão inseridas, assim, procuram desenvolver e aperfeiçoar técnicas de BPM que vão ao encontro do exigido informalmente pelos fatores internos e externos que impulsionam o negócio em diferentes direções, procurando colmatar descompensações que possam afetar a performance da sua atividade e a evolução do seu negócio.

O princípio de modelação de negócio do BPM é aberto e universal, a sua leitura e interpretação é acessível a pessoas com funções tão distintas como o administrador, o engenheiro de software ou o analista financeiro, permitindo assim que diferentes intervenientes de áreas distintas consigam participar ativamente nestas tarefas, contribuindo para o modelo comum que melhor se adequa às circunstâncias empresariais do momento e à realidade da instituição bancária.

O projeto “*Bankee*”, que será o estudo de caso descrito neste documento, é todo um novo sistema modular de *frontend* desenvolvido pela empresa *Critical Software* para um cliente angolano na área da banca. É um projeto de elevada dimensão e complexidade, está em desenvolvimento há aproximadamente seis anos por uma equipa com cerca de 30 pessoas, com diferentes responsabilidades e distribuídas por várias localizações, tais como Coimbra, Porto e Luanda.

Na aplicação “*Bankee*” tem-se uma arquitetura baseada em serviços, onde os processos de negócio são modelados e implementados recorrendo ao software de *Business Process Management* da IBM (IBPM).

Esta aplicação pretende substituir, progressivamente, o atual sistema de core bancário, *Banka*, implementado no sistema AS/400 da IBM, atualmente ainda em produção nos diferentes setores da organização.

O que despoletou a necessidade desta instituição implementar um novo sistema de software prende-se essencialmente com a carência de automatização e monitorização dos processos de negócio, que o atual sistema não consegue suplantar. Desta forma, foi definido um plano de execução em diferentes fases, para o desenvolvimento de módulos orientados à área de negócio, com o objetivo de substituir, gradualmente, o atual sistema de software bancário.

## **1.2. Problema**

No atual sistema bancário, as operações de transferências interbancária estão dispersas por operações unitárias, realizadas por intervenientes isolados que executam e aprovam a saída de valores da instituição para outras, sem intervenção de terceiros para o sucesso das operações. Este, é um problema que a instituição bancária cliente pretende resolver.

Uma transferência interbancária trata-se da operação – ou conjunto de operações – que permite ao cliente da instituição bancária transferir valores, desde a sua conta, para outra conta numa instituição bancária distinta, o que compreende a saída de valores de uma instituição para outra sua concorrente.

O módulo a desenvolver no âmbito desta dissertação deve permitir realizar transferências interbancárias nos dois sistemas existente na realidade bancária angolana, STC (Subsistema de Transferências a Crédito) e SPTR (Subsistema de Pagamentos em Tempo Real), descritos no

segundo capítulo deste documento. No entanto, incidirá somente no processo de transferências interbancárias em moeda nacional. Pelo motivo de se tratar de um processo distinto, com diferente especificação funcional, serão excluídas do posteriormente descrito, regras ou definições que se apliquem a transferências interbancárias em moeda estrangeira.

Com a implementação do processo de transferências interbancárias, a instituição cliente pretende não só resolver problemas genéricos tais como:

- A inadequação do sistema de autorizações numéricas do anterior sistema;
- Complexidade da interface do utilizador, no software Banka, e dispersão da informação por um grande número de ecrãs acedidos por percursos numéricos;
- Inexistência de suporte à automatização, controlo e monitorização de processos;
- Complexidade no registo de operações bancárias com múltiplas operações separadas no sistema e dependentes de processos manuais, resultando num esforço significativo de correção de erros nos serviços centrais;
- Ausência de integração de um sistema de gestão documental e de suporte aos processos bancários, o que impossibilita a consulta rápida de documentos relacionados com determinada transferência interbancária e agregação destes num único repositório central.

Mas também, concretamente, resolver o problema da falta de intervenção hierárquica dinâmica durante a execução do processo de transferência interbancária de valores.

A funcionalidade de intervenção hierárquica dinâmica permitirá integrar no processo de negócio, um módulo autocontido, caracterizado como uma funcionalidade de manutenção que permite inserir, duplicar, alterar, remover e simular a utilização de “Limites de Autorização”. Sendo assim possível executar todas as operações referidas anteriormente, criando regras de negócio que serão posteriormente utilizadas nos cálculos do encaminhamento do processo de transferências interbancárias, para os grupos de utilizadores com permissões para executar atividades de aprovação, autorização ou dada de parecer, em momentos chave deste.

A intervenção hierárquica dinâmica é de extrema importância na monitorização de movimentações de determinados valores, normalmente altos, e que, por este ou outros motivos definidos pela instituição bancária, requerem a atenção de intervenientes com cargos específicos na organização, bem como a sua participação em níveis crescentes de intervenção, acompanhando as regras de negócio configuradas previamente. Esta aproximação ao problema, envolvendo diferentes intervenientes da hierarquia, permite identificar princípios de fraude em diferentes pontos do processo de negócio, ao mesmo tempo que expande a responsabilidade da operação a diferentes intervenientes.

A existência de níveis de intervenção hierárquica dinâmicos, justifica-se pela natureza e quantidade dos montantes envolvidos em cada operação de transferência interbancária, atuando como uma medida ativa de prevenção à fraude através do envolvimento de diferentes intervenientes da hierarquia da instituição com função proeminente durante o processo de negócio que permitirá concluir, com sucesso, a operação.

Num negócio como o bancário, que movimenta valores de diferentes ordens, surge inevitavelmente a possibilidade de fraude. Notícias sobre fraude bancária são mais ou menos comuns na imprensa nacional e angolana [52] [53]. A fraude praticada nos níveis mais baixos da



hierarquia das instituições, onde a probabilidade de se concretizar em desvios monetários de maior ou menor dimensão durante a realização de operações comuns, tem vindo a ser combatida através de um combinar de técnicas que envolvem sistemas de vigilância e participação ativa de elementos mais altos da hierarquia na concretização de operações com maior nível de vulnerabilidade à fraude.

Já o combate à fraude nos níveis hierárquicos mais altos é um tema de elevada complexidade. Se por um lado as instituições bancárias vão atribuindo maiores responsabilidades e cargos de maior relevância a pessoas que se mostrem dignos de confiança, por outro espera-se que estas pessoas monitorizem movimentações que possam levantar suspeitas ou lesar, de alguma forma, a instituição que representam.

### 1.3. Objetivo

Com este trabalho pretende-se realizar um estudo de caracterização na área de domínio de *Business Process Management* com identificação de diferentes padrões de modelação de processos de negócio, e de integração, orientados à resolução de problemas operacionais gerais e específicos do negócio a que se destina, considerando fatores tais como a realidade sociocultural, objetivos internos da instituição bancária e geografia onde esta desenvolve atividade.

Pretende-se ainda:

- Propor um modelo do processo de transferências interbancárias;
- Desenvolver um módulo que, considerado o modelo definido, implemente as operações de transferência de valores interbancários em moeda nacional (AKZ) com a integração de um módulo, autocontido, já desenvolvido e que permite determinar dinamicamente a necessidade de intervenção hierárquica de elementos da organização em pontos-chave do processo. A implementação incluirá ainda o desenvolvimento dos ecrãs de UI (*user interface*) e os *webservices* necessários.

A solução a implementar deverá cumprir os objetivos de negócio da instituição bancária a que se destina, sendo que será um módulo integrante de toda uma solução de *Frontend* desenvolvida por uma equipa de engenharia da “*Critical Software*” (CSW), na qual o autor do trabalho descrito neste documento se encontra a exercer funções enquanto membro integrante da equipa de desenvolvimento.

No final deste trabalho é expectável que se encontre desenvolvido um módulo de transferências interbancárias e integrado na aplicação Bankee, cumprindo os objetivos descritos, tanto a nível de modelação do processo de negócio a que se destina, como na integração com o *Core* bancário e os diferentes componentes que fazem parte da estrutura da aplicação. Este módulo deverá incluir ainda uma solução para o problema previamente descrito, de forma que o encaminhamento do processo nos níveis de intervenção configurados nas regras de negócio, sejam atribuídos corretamente e resultem na participação dos utilizadores com os perfis necessários à aprovação, autorização e/ou fornecimento de parecer.

Após este processo, o módulo desenvolvido passará por uma *pipeline* de validações e testes, tanto nos ambientes de qualidade da CSW, como em ambientes de qualidade da instituição bancária cliente, ambiente de pré-produção, piloto e por fim a expansão para os restantes

balcões da instituição, efetivando-se a sua entrada em produção. A solução supra descrita será desenvolvida em IBPM, a suite de BPM da IBM, otimizada para a versão

## 1.4. Abordagem Preconizada

O desenvolvimento deste trabalho foi dividido em três fases. Começando por analisar e estudar o problema, procurando consultar metodologias desenvolvidas estrategicamente para a resolução de problemas idênticos na área da banca. Ainda durante este período, foi feito um levantamento dos requisitos funcionais considerando as necessidades gerais e específicas do cliente, bem como todas as considerações a nível de execução, performance e gestão do risco operacional para o módulo desenvolvido.

Na segunda fase foi imperativo realizar todo o levantamento tecnológico necessário à implementação do módulo, no qual se incluem as necessidades a nível de infraestrutura, levantamento de APIs de integração com o *Core* bancário e serviços externos.

Ainda nesta fase se inclui todo o trabalho de desenvolvimento do módulo, o que envolveu a modelação final do processo de transferências interbancárias em BPM, desenvolvimento dos ecrãs, desenvolvimento dos *web services* necessários à execução das operações bancárias e de integração com a aplicação, criação de recursos de segurança, configurações e desenvolvimento de APIs de integração com o *Core* bancário em uso.

Por último foi realizada uma análise crítica e avaliação da solução implementada, apontando os pontos fortes e os pontos fracos com margem para melhoria.

## 1.5. Organização

Este documento está estruturado em 9 capítulos, com uma ordem sequencial que representa a evolução lógica deste trabalho.

Neste primeiro capítulo, *introdução*, é descrita uma breve apresentação do tema em desenvolvimento. É também feita uma exposição do problema, bem como a descrição dos objetivos que se pretendem alcançar com este estudo, fazendo um enquadramento do meio em que se desenvolve e como será organizado.

Durante o segundo capítulo, *Estado da Arte*, é feito o enquadramento teórico a diferentes níveis, histórico, de negócio e tecnológico. Neste capítulo é abordado de forma sequencial o desenvolvimento da banca Angolana durante as últimas décadas como suporte às necessidades que impulsionam o desenvolvimento deste trabalho, a contextualização do negócio bancário no que respeita em concreto às transferências interbancárias e um enquadramento tecnológico sobre BPM (Business Process Management).

O terceiro capítulo, *Análise de Negócio*, inicia com um pequeno contexto referente à solução a desenvolver, enquadrando-a no plano geral de desenvolvimento da aplicação Bankee. Nos subcapítulos que se seguem, é feito um levantamento dos Atores, dos requisitos funcionais englobando os pressupostos e casos de uso da solução proposta. São ainda identificados requisitos não funcionais genéricos na aplicação e específicos do módulo a desenvolver. É ainda apresentado um modelo do domínio como facilitador do enquadramento do leitor no negócio,

sendo por fim apresentados os modelos iniciais, e de estudo, do processo de negócio de transferências interbancárias, bem como das suas atividades.

No quarto capítulo, *Avaliação Tecnológica*, é realizado uma avaliação das soluções tecnológicas utilizadas na implementação e integração do processo de negócio de transferências interbancárias. Aqui são descritas as tecnologias em uso para o desenvolvimento da solução, bem como tecnologias alternativas que permitiriam alcançar objetivos idênticos. Neste capítulo são apresentadas e descritas algumas soluções para o desenvolvimento da camada de apresentação, servidor e modelação de processos de negócio.

O quinto capítulo, Design Arquitetural, é apresentada e descrita a arquitetura da aplicação Bankee, bem como a descrição pormenorizada e arquitetura do módulo de limites de autorização enquadrando a sua importância na qualidade da solução a desenvolver.

O sexto capítulo, Design e Implementação, é dedicado à fase de desenvolvimento da solução sendo dividido em 3 secções. A primeira secção – Tecnologias – é dedicada, nas suas sub-secções respetivamente, à apresentação da infraestrutura intrínseca ao projeto Bankee bem como à descrição e apresentação das ferramentas usadas durante o processo de desenvolvimento. A segunda secção – Metodologia – descreve a metodologia de desenvolvimento adotada para a implementação da solução, bem como a relevância de cada uma das suas fases. A terceira secção – Produto Final – é dedicada á descrição da solução desenvolvida, fazendo um levantamento de alternativas, padrões adotados, apresentação dos modelos finais do processo, ecrãs desenvolvidos e serviços implementados.

No sétimo capítulo - *Análise de Valor* - é efetuado um levantamento dos elementos chave do modelo NCD<sup>1</sup>, formalizada a proposta de valor da solução através de um modelo de negócio Canvas e identificada a respetiva rede de valor.

O oitavo capítulo – *Avaliação* - é dedicado à descrição de todas as componentes do processo de avaliação da solução implementada. Aqui se encontram descritos os testes realizados nos ambientes de qualidade internos, testes de aceitação do cliente e o resultado formal da avaliação do processo de transferências interbancárias com intervenção hierárquica dinâmica, através de um levantamento de defeitos nos ambientes de qualidade internos da Critical Software, e os defeitos nos ambientes de qualidade externos do cliente, correlacionando estes valores com a avaliação geral da qualidade da solução desenvolvida.

O novo capítulo - *Conclusão* – descreve em balanço a análise os objetivos alcançados e contributos do trabalho realizado, bem como limitações e trabalho futuro.

---

<sup>1</sup> NCD – New Concept Development, Peter Koen

## 2. Estado da Arte

Neste capítulo é realizado, no âmbito de estudo e investigação de *Business Process Management*, um estado da arte compreensivo orientado à implementação desta tecnologia ao serviço da banca. Inicia com um enquadramento teórico do negócio e o contexto em que se desenvolve, após o qual são abordadas metodologias de modelação, conceitos relacionados, exemplos de implementação de técnicas de BPM e adaptações destas abordagens num enquadramento tecnológico da área de negócio em estudo.

### 2.1. Enquadramento Teórico

Torna-se relevante enquadrar numa perspetiva teórica, e histórica, o negócio bancário e a sua relação com a tecnologia.

O contexto sociocultural em que se enquadra a banca angolana, e como esta se tem vindo a desenvolver nas últimas décadas, posicionando-a numa geografia com características próprias que influenciam ativamente o desenvolvimento de software para áreas críticas de negócio, como é o caso da área financeira.

Importa ainda durante este subcapítulo clarificar o leitor sobre a definição de transferência interbancária, e de que forma esta se processa no âmbito da banca angolana.

#### 2.1.1. Banca

Desde o momento em que surge moeda na história, os atos de emprestar, guardar e aplicar dinheiro tornaram-se cada vez mais comuns. As primeiras moedas, com características semelhantes às atuais, objetos metálicos cunhados, com um peso e valor definidos, surgiram na Lídia no século VI a.c. e com elas, as primeiras operações bancárias da história [37]. Desde esse tempo até à data atual, a evolução da indústria bancária foi considerável, contudo a revolução tecnológica que inicia em 1937 com o início do desenvolvimento daquele que em 1942 viria a ser o primeiro computador digital [39], o “*Atanasoff-Berry Computer*”, vem marcar o início de uma rápida evolução de tendências e na constante quebra de paradigmas dentro da indústria.

As operações bancárias tradicionais, tais como depósitos, transferências, levantamento de cheques ou até aberturas de conta, realizadas habitualmente no meio físico da agência bancária, mudaram drasticamente com o aparecimento e evolução da internet. As novas, e emergentes tecnologias, geram clientes cada vez mais exigentes [40]. O leque de necessidades é mais dinâmico, e as instituições bancárias são obrigadas a oferecer os mesmos serviços através de diferentes meios e/ou dispositivos.

À medida que a evolução tecnológica o permitiu, a indústria bancária foi lançando dispositivos e meios com o objetivo de facilitar as operações bancárias tradicionais, dentro e fora dos balcões, exemplo disso é a *Automated Teller Machine* (ATM, 1966) criada com a premissa de disponibilizar a distribuição de dinheiro fora do horário de atendimento dos balcões. Atualmente, estas máquinas permitem executar praticamente todas as operações de rotina sem que os clientes necessitem de se deslocar a um balcão físico, inclusive a realização de transferências interbancária, com as limitações inerentes, tais como a concretização da

operação em tempo real, um limite máximo permitido e a necessidade de intervenção humana em situações pontuais.

Atualmente, os bancos necessitam de adotar tecnologias disruptivas [38] para fortalecer as expectativas dos seus clientes com a garantia de serviços de ponta altamente eficientes. Tal como na indústria de retalho, competem entre si oferecendo uma grande variedade de produtos e serviços, acompanhados de pontos de acesso facilitados. Este fenómeno intensificou-se após a crise financeira mundial de 2007, fazendo com que aumentasse ferozmente a concorrência ao mesmo tempo que alguns dos maiores bancos mundiais reduziam os seus recursos humanos em até 25% [40].

Consequentemente, muitos bancos foram impulsionados no sentido de investir no desenvolvimento de soluções que lhe permitissem providenciar serviços através da internet, estendendo aqueles que eram já fornecidos nos balcões ou dispositivos de ATM [41]. Desta forma, ao mesmo tempo que se estendem os serviços tradicionais e se reduzem custos operacionais, são oferecidas soluções vincadamente tecnológicas que apelam a uma maior comodidade e proximidade com o cliente, apesar de grande parte, ser remotamente.

As evoluções dos sistemas de comunicação móvel vieram acentuar largamente esta tendência. Com o aparecimento de *smartphones* e *tablets* os consumidores apreciam cada vez mais a capacidade de poderem realizar diferentes tipos de tarefas a partir destes dispositivos [42]. Estima-se em 2017, que o número de utilizadores no Médio Oriente e África de dispositivos móveis para acesso remoto, através de dispositivos móveis inteligentes, ultrapasse os 80 milhões [43]. A crescente adesão dos consumidores é explicada pela expansiva aplicação prática desta abordagem tecnológica, não se limitando à concretização de operações bancárias tradicionais, mas abrindo também a possibilidade de introduzir aconselhamento financeiro remoto através de videoconferência ou até a autenticação de documentos através de leituras biométricas realizadas pelos próprios dispositivos.

Toda esta evolução tecnológica de *frontend* requer um adequado acompanhamento de *backend*, permitindo que todos estes processos de negócio possam ser automatizados e monitorizados através de sistemas de *Business Process Management*.

### **2.1.2. A Banca Angolana**

Com o intuito de contextualizar o que motiva, em parte, o desenvolvimento do módulo de transferências interbancárias com intervenções hierárquicas dinâmicas, importa transmitir para este trabalho, todo o contexto social e económico em que se desenrola o negócio bancário em Angola.

Devido à frequente instabilidade social do país, a realidade da banca Angolana sofreu algumas alterações de base em vários momentos da história ao longo do séc. XX.

Segundo dados do BNA (Banco Nacional de Angola), a primeira instituição bancária com autorização para se instalar em Luanda, o Banco Nacional Ultramarino, data de 1865, sendo o primeiro a emitir moeda nacional exclusivamente, até ao ano de 1926, altura em que devido ao descontrolo da emissão monetária e consequente descalabro na situação financeira do país, as autoridades coloniais tiveram de iniciar uma reforma monetária, que conduziu à criação do “Banco de Angola”, um banco emissor independente[46]. Este banco, detém até ao ano de 1957 o direito exclusivo de comércio bancário em Angola, e apesar do nome, o seu direito pertencia

a Portugal colonial, e a sua sede instalada em Lisboa onde supostamente estaria mais próximo da fiscalização do Ministro das Colónias e afastado das influências locais [45].

Em 1957 surge o “Banco Comercial de Angola”, com participação estritamente angolana. E desta data até 1975, implementam-se em Angola mais três bancos comerciais, o “Banco de Crédito Comercial e Industrial”, o “Banco Totta Standard de Angola”, “Banco Pinto & Sotto Mayor” e o “Banco Inter Unido”, bem como com quatro novas instituições de crédito, o “Instituto de Crédito de Angola”, o “Banco de Fomento Nacional”, a “Caixa de Crédito Agro-Pecuária” e o “Montepio de Angola”.

Após cerca de quatro meses da revolução dos cravos, que fez cair o regime ditatorial Português em 25 de Abril de 1974, a 14 de Agosto do mesmo ano, acontece também um evento marcante na banca angolana, a “tomada da banca”, que consistiu na ocupação física das instalações dos bancos comerciais em Luanda e a destituição dos quadros em vigência até à data, com o objetivo de implementar um plano delineado pelo Ministério do Plano e Finanças para evitar o desmoronamento de todo o sistema bancário e financeiro [45]. Como consequência, o governo Angolano confiscou todo o ativo e passivo do Banco de Angola e criou o Banco Nacional de Angola (BNA) que até à presente data mantém e acumula as funções de banco emissor, caixa de tesouro e de comércio bancário.

É importante perceber que toda a instabilidade social do país nas últimas décadas também influencia em grande número as estratégias de desenvolvimento de software. Torna-se necessário encontrar fórmulas robustas que se adaptem à realidade do país, tanto a nível de recursos como de meios, onde atualmente a distribuição elétrica nacional é bastante deficitária e o acesso à internet ainda se faz via satélite em muitos pontos do país.

### **2.1.3. Transferência Interbancária**

Transferências interbancárias são operações de movimentação de valores entre contas à ordem de clientes de instituições bancárias distintas, com a execução de uma ou mais operações de negócio que permitem ao cliente ordenante de uma instituição bancária transferir valores monetários, desde a sua conta, para outra do cliente beneficiário de uma instituição bancária concorrente. O cliente poderá reunir as qualidades de beneficiário e ordenante, conforme o papel que este assume na operação a executar [44].

No sistema bancário angolano as transferências interbancárias podem ser divididas em dois grupos: transferências interbancárias do “Subsistema de Pagamentos em Tempo Real” (SPTR) e transferências interbancárias do “Subsistema de Transferências a Crédito” (STC) [44]:

- SPTR – Trata-se de um “subsistema do sistema de pagamentos de Angola, que permite a liquidação interbancária de instruções de pagamento, em tempo real e por bruto (operação a operação)”. [44]
- STC – É “um subsistema da Câmara de Compensação Automatizada de Angola, que assegura a compensação interbancária de transferências eletrónicas a crédito”. [44]

Podem ainda ser em moeda nacional, Kwanza ou estrangeira.

As regras de negócio de execução de transferências interbancárias são definidas dentro da própria instituição, competindo a cada instituição bancária definir perentoriamente os níveis de intervenção durante o processo de transferência, associados ao montante a transferir. Estes

aspectos são descritos no regulamento interno de cada entidade financeira, e transcritos posteriormente para a implementação interna dos seus processos de negócio.

Estão, no entanto, definidas normativas nacionais gerais que regulam a execução destas operações entre as diferentes entidades. O seu objetivo é salvaguardar os interesses dos seus clientes e normalizar o fluxo de valores entre as diferentes instituições. O documento [44] descreve regras de regulamentação da atividade onde define os prazos máximos para a execução de transferências bem como as condições para a disponibilização do valor de transferências.

#### **2.1.4. Business Process Re-Engineering na Indústria Bancária**

Os competitivos mercados de hoje, ditam que as empresas devem estar em constante atualização da qualidade dos serviços e/ou produtos que comercializam, bem como a relação desta com o preço dos mesmos e a sua pronta disponibilidade. Também o constante desenvolvimento pessoal dos recursos humanos e a inovação tecnológica são pontos a considerar para o aumento da competitividade.

A atividade de reengenharia dos processos de negócio está diretamente relacionada com a marca de posição da empresa no mercado através da sua competitividade e implica a transformação de certos procedimentos e processos dentro da instituição, com o intuito de aumentar a inovação nas soluções de negócio, a atualização das tecnologias utilizadas e o investimento em mais e melhor formação dos seus recursos humanos. Estas transformações, resultam numa reestruturação organizacional ao nível da sua estrutura, afetando direta, ou indiretamente, os processos de negócio de toda a organização [12].

É caracterizada pelo repensar e remodelar de processos de negócio com o objetivo de alcançar melhorias na eficiência e qualidade do serviço, bem como na redução de custos.

Num mercado financeiro como o de Hong Kong foram identificados 4 fatores críticos de sucesso na prática de BPR na área da banca. O compromisso dos altos cargos de gestão, a comunicação eficaz da mudança em conjunto com o foco no cliente e recurso às tecnologias de informação na implementação do plano delineado para a reengenharia do processo de negócio. Estudos como [12] mostram que o maior peso recai no uso ativo das tecnologias de informação neste processo, seguido da efetiva comunicação do plano, alinhando os recursos de todos os intervenientes na hierarquia da instituição bancária com os objetivos traçados para os novos processos de negócio.

O peso das tecnologias da informação neste processo é justificado pela substituição de processos manuais por automação, aumentando a eficiência e produtividade das instituições permitindo desta forma ir ao encontro das necessidades imediatas do processo dentro dos recursos das instituições. Outros estudos [59, 61] consideram também que as tecnologias de informação são a chave para a relação de sucesso com a reengenharia dos processos de negócio. Face à evolução tecnológica é ainda proposta a reengenharia radical de processos, o que implica o redesenhar completo dos processos de negócio das instituições a fim de se adaptarem à realidade tecnológica da atualidade, permitindo eliminar pontos de estagnação que no contexto tecnológico atual não fazem sentido [62].

A perspetiva do cliente na análise do processo de negócio é valorizada, pois considera-se que só faz sentido repensar o processo com base no que acrescenta valor ao cliente final da instituição,

permitindo assim que os indicadores de sucesso acompanhem a tendência das necessidades dos clientes [63].

## 2.2. Enquadramento Tecnológico

### 2.2.1. Business Process Management

*Business Process Management* (BPM) é toda uma disciplina orientada à análise, definição, execução, monitorização e administração de processos de negócio. O objetivo principal de BPM está diretamente relacionado com a melhoria contínua do desempenho corporativo, através da otimização dos processos de negócio de uma determinada empresa.

BPM coloca em prática um conjunto de ações perante uma perspetiva holística de cada processo de negócio num meio corporativo de forma a aprimorar cada um dos processos. Há uma integração estratégica e objetiva por parte das organizações focadas no crescimento das necessidades dos seus clientes, através de processos de negócio ponta-a-ponta. Com utilização de “estratégias, objetivos, cultura, estruturas organizacionais, papéis, políticas métodos e tecnologias para analisar, desenhar, implementar, gerir desempenho, transformar e estabelecer a gestão de processos de negócio” [11].

A disciplina BPM proporciona os meios para que as organizações sejam mais eficazes, instituindo um conjunto de princípios e práticas que lhes permitem executar os seus processos de negócio de forma mais eficiente. Assumindo implicitamente que as organizações serão bem-sucedidas na sua implementação, sempre que tenham capacidade de gerir os seus processos de negócio de acordo com estes princípios.

O conceito de *Business Process Management*, talvez pela representação gráfica e fonética do acrónimo, é regularmente confundido com *Business Process Modeling*, que por sua vez é uma das partes integrantes, talvez a mais crítica, do conceito em definição.

### 2.2.2. Business Process Modeling

Desde o início do século XX que se conhecem diferentes técnicas para a modelação de processos de negócio, é, contudo, no final dos anos 60, mais concretamente em 1967 que o termo “*Business Process Modeling*” surge pela primeira vez [4]. Neste estudo, do âmbito da engenharia de sistemas, o autor sugere que técnicas usadas para obter melhor compreensão de sistemas de controlo físico poderiam ser de igual forma aplicadas aos processos de negócio empresariais, contudo, só nos anos 90 é que este termo se torna popular.

*Business Process Modeling* refere-se à representação gráfica, por via de diagramas de fluxo que descrevem sequências de atividades no processo de negócio tais como eventos, ações e pontos de interação numa ligação ponto-a-ponto [15]. É uma metodologia multifuncional que implica regularmente a combinação de esforços entre os diferentes departamentos de uma organização, ou em situações mais complexas, poderá incluir atividades de organizações externas.



Nas organizações de maior escala, tal como as instituições bancárias, as operações de negócio são analisadas com maior granularidade [34] e representadas em maior detalhe, algo que está diretamente relacionado com a complexidade e criticidade do negócio, que se refletirá na representação dessas operações, durante a modelação dos processos.

Com a aceitação de *Business Process Modeling* como uma técnica de modelação de processos válida para um grande número de organizações, sendo as áreas da banca, finança e seguros os maiores utilizadores desta metodologia, foi necessário definir *standards* para a sua utilização na indústria, evoluindo para o que é hoje conhecido como *Business Process Model and Notation (BPMN)*.

### **2.2.3. Notações**

Atualmente existe um conjunto de notações que seguem o mesmo princípio de modelação de processos de negócio, são definidas pela descrição de atividades que seguem uma ordem modelada pela descrição de um conjunto de dependências, de propriedades e uso de dados nas decisões dos modelos.

#### **2.2.3.1. Business Process Model and Notation (BPMN)**

Com o intuito de formalizar o uso dos diagramas de fluxo para gerir a modelação de processos de negócio através de técnicas de *Business Process Modeling* (descrito no ponto anterior), foi necessário evoluir no sentido da uniformização desta técnica.

É assim que, no início do século XXI, desenvolvido pela *Business Process Management Initiative (BPMI)*, um consórcio constituído essencialmente por empresas de software, nasce o *Business Process Modeling and Notation*. A primeira especificação foi desenvolvida por uma equipa liderada por Stephen A. White da IBM, e publicada em 2004. Contudo somente em 2006 é que a versão 1.0 é oficialmente aceite pela *Object Management Group (OMG)*. Em 2011, e após a inserção de novas extensões, surge a versão 2.0 que se mantém até aos dias de hoje.

Nasce assim uma notação orientada à modelação de processos de negócio que cumpre com os requisitos necessário para uma representação precisa, de todos os aspetos relevantes à modelação de processos de grande dimensão.

A criação de uma notação padronizada permite assim que se crie uma linguagem universal, de forma que todos os envolvidos na criação e modelação de processos possam entender as abordagens dos demais. Através da criação de um conjunto de elementos tais como eventos, atividades, portais, fluxos de sequência e fluxos de mensagem, *lanes*, bem como os respetivos subgrupos que se encontram representados na Figura 1 e na Figura 3.

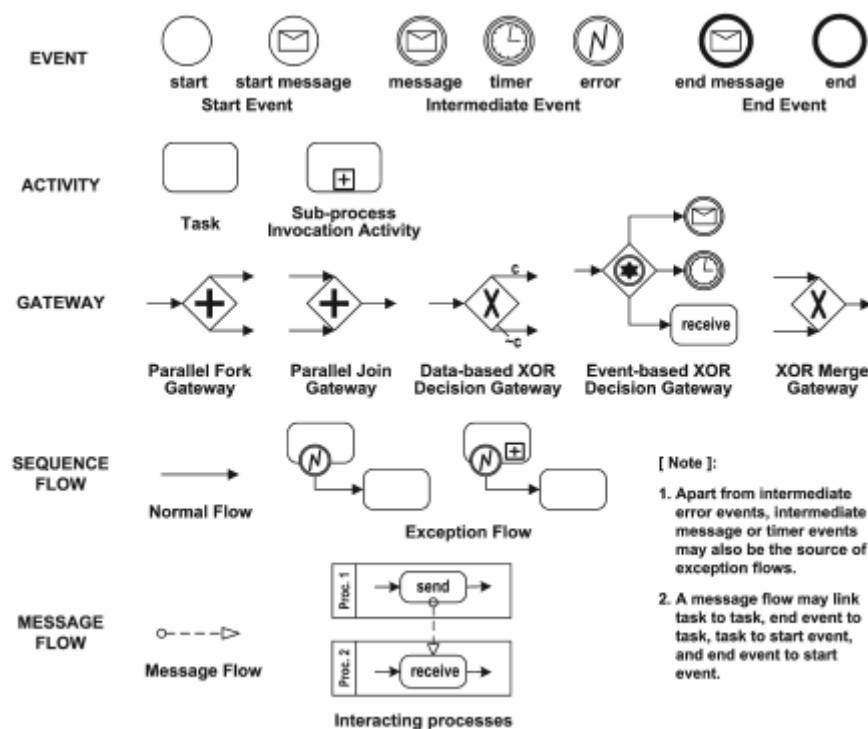


Figura 1 – Alguns Elementos da notação BPMN

Fonte: [19]

A versão 2.0 de BPMN fornece três tipos de diagramas que se encontram representadas na segunda camada da Figura 2:

- **Diagrama de processo ou colaboração** – Refere-se ao comum diagrama de modelação do fluxo de negócio e pode incluir atividades, decisores, eventos e fluxos paralelos. Os diagramas de processo ou colaboração são em tudo idênticos a nível de modelação e dos elementos utilizados. A designação “diagrama de processo” refere-se à modelação que só contém um único processo, quando a modelação inclui a conexão com outros processos, designa-se de diagrama de colaboração.
- **Diagrama de coreografia** – Este tipo de diagrama não existia na primeira versão de BPMN, sendo introduzido na versão 2.0. Representa a modelação da troca de dados entre diferentes intervenientes, tal como nos diagramas de colaboração, sendo que nestes diagramas a troca de dados dá-se ao nível das atividades [6]. Permitem analisar de que forma os participantes trocam informação de forma a coordenar as suas interações
- **Diagrama de conversação** – Foi também introduzido na versão 2.0 de BPMN e descrito como a representação do uso particular de uma descrição informal de um diagrama de colaboração. Refere-se a um novo diagrama, inserido como uma versão simplificada do diagrama de colaboração [47], este facto leva a considerar que se trata de uma extensão do diagrama de colaboração [64].



Figura 2 - Diferentes camadas da estrutura de BPMN 2.0  
Fonte [47]

## Atividades

- Tarefa**: Uma Tarefa é uma unidade de trabalho, a tarefa a ser realizada. O símbolo [T], em uma tarefa, indica um Subprocesso, uma atividade que pode ser decomposta em (sub-)tarefas.
- Transação**: Uma Transação é um conjunto de atividades, logicamente relacionadas; ela pode seguir um protocolo transacional específico.
- Subprocesso de Evento**: Um Subprocesso de Evento se situa no interior de outro (sub-)processo. Ele é ativado quando seu evento de início é disparado e executa até seu final ou enquanto o processo que o contém estiver ativo. Ele pode interromper o contexto do processo que o contém ou executar em paralelo a este (sem interrompê-lo), dependendo do evento de início.
- Atividade de Chamada**: A Atividade de Chamada é uma referência a um Subprocesso ou Tarefa definido globalmente e reutilizado no processo atual.

### Marcadores de Atividade

Marcadores denotam o comportamento específico de uma atividade durante sua execução, diferenciando tipos de atividades:

- Marcador de Subprocesso
- Marcador de Repetição
- Marcador de Instâncias Múltiplas em Paralelo
- Marcador de Instâncias Múltiplas em Sequência
- Marcador de Atividade Ad Hoc
- Marcador de Atividade de Compensação

### Tipos de Tarefas

Tipos determinam a natureza da tarefa a ser executada:

- Tarefa de Envio
- Tarefa de Recebimento
- Tarefa de Usuário
- Tarefa Manual
- Tarefa de Regra de Negócio
- Tarefa de Invocação de Serviço
- Tarefa de Execução de Script

### Fluxo de Seqüência

define a ordem de execução das atividades.

### Fluxo Padrão

é o caminho padrão a ser seguido, caso todas as outras condições retornem falso.

### Fluxo Condicional

possui uma condição associada, a qual define se o caminho será seguido ou não.

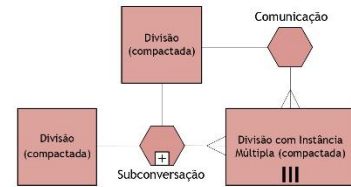
## Desvios

- Desvio Condicional Exclusivo (OU Exclusivo)**: Em um ponto de ramificação, seleciona exatamente um caminho de saída dentre as alternativas existentes. Em um ponto de convergência, basta a execução completa de um braço de entrada para que seja ativado o fluxo de saída.
- Desvio Condicionado por Evento**: Em seus fluxos de saída só são permitidos eventos ou tarefas de recepção; ativa somente o caminho, cujo evento ou recepção ocorrer antes.
- Ativação Incondicional em Paralelo**: Em um ponto de ramificação, todos os fluxos de saída são ativados simultaneamente. Em um ponto de convergência de fluxos, espera que todos os caminhos de entrada completem, antes de disparar o fluxo de saída.
- Ativação Inclusiva Condicional**: É um ponto de ramificação, após avaliar condições, um ou mais caminhos são ativados. Em um ponto de convergência de fluxos, espera que todos os fluxos de entrada ativos tenham completado para ativar o fluxo de saída.
- Desvio Exclusivo baseado em Eventos (gerador de instâncias)**: A cada ocorrência de um dos eventos subsequentes, inicia uma nova instância do processo.
- Desvio Paralelo baseado em Eventos (gerador de instâncias)**: Na ocorrência de todos os eventos subsequentes, se cria uma nova instância do processo.
- Desvio Complexo**: Comportamento complexo de ramificação ou convergência que não pode ser capturado por outros tipos de desvio.

## Conversações

- Uma **Comunicação** define um conjunto de trocas de mensagens logicamente relacionadas. Quando marcada com o símbolo [C], indica uma Sub-conversação, um elemento de conversação composto.
- Um **Link de Conversação** conecta Comunicações e Participantes.
- Um **Link de Conversação Ramificado** conecta Comunicações a múltiplos Participantes.

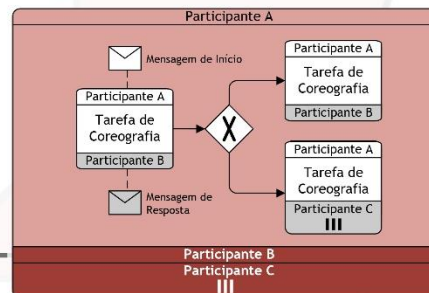
### Diagrama de Conversação



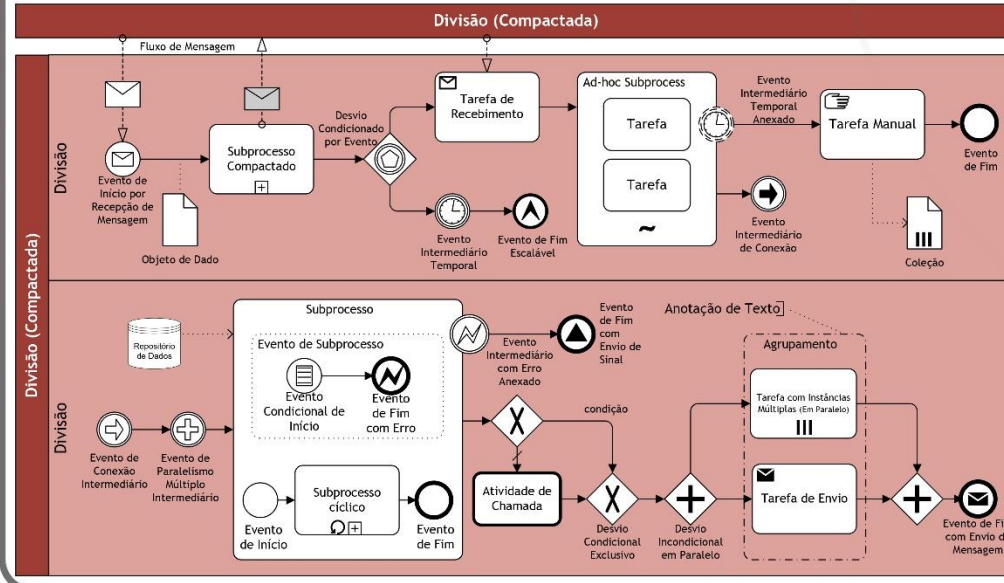
## Coreografias

- Uma **Tarefa de Coreografia** representa uma interação (Troca de Mensagem) entre dois Participantes.
- Uma **Marca de Participantes Múltiplos** Indica um conjunto de Participantes de um mesmo tipo.
- Uma **Coreografia de Sub-processo** contém uma coreografia refinada em interações.

### Diagrama de Coreografia



## Diagrama de Colaboração



## Divisões

- Divisões e Compartimentos de Responsabilidade** representam as entidades responsáveis pelas atividades, ou seja os participantes do processo, podendo ser uma organização, um papel, um ator humano ou um sistema automatizado. Compartimentos subdividem Divisões ou outros.
- Fluxo de Mensagem** simboliza fluxos de informação que transpõem fronteiras internas e externas de uma organização. Podem ser conectados a Divisões, atividades ou eventos de mensagem.
- A Ordem da Troca de Mensagens** no processo pode ser especificada através de combinações de fluxos de mensagem e fluxos de seqüência.

## Eventos

Evento de Início	Eventos Intermediários				Evento de Fim
Evento de Alto Nível	Captura	Interrupção a execução da instância em um compartimento	Não interrompe a execução da instância em uma divisão ou compartimento	Lançamento	
<b>Simples</b> : Eventos sem tipo indicam pontos de início, de fim e mudanças de estado.					
<b>Mensagem</b> : Recebimento e envio de mensagens.					
<b>Temporal</b> : pontos no tempo, instante no tempo, intervalo de tempo, limite de tempo. Podem ser eventos únicos ou cíclicos.					
<b>Eskalável</b> : ativa mudança para um nível mais alto de responsabilidade.					
<b>Condicional</b> : Reação a alterações nas condições de negócio ou a regras de negócio.					
<b>Conector</b> : Conector entre páginas. Dois eventos de conexão equivalem a um fluxo de seqüência.					
<b>Erro</b> : Captura ou inserção de erros pré-identificados.					
<b>Cancelamento</b> : reagem ao cancelamento de uma transação ou ativam cancelamento.					
<b>Compensação</b> : Tratamento ou ativação de ação de compensação.					
<b>Sinal</b> : Emitem sinais entre processos. Um mesmo sinal pode ser capturado várias vezes.					
<b>Múltiplo</b> : Ou capturam um dentro um conjunto de eventos, ou lançam um ou mais eventos de qualquer dos tipos definidos.					
<b>Múltiplo Paralelo</b> : capturam, de uma só vez, todos os eventos de um conjunto de eventos que ocorrem em paralelo.					
<b>Final</b> : Ativam a terminação imediata de um processo.					

## Dados

- Um **Dado de Entrada** é um evento externo ao processo. Pode ser lido por uma atividade.
- Um **Dado de Saída** é uma variável disponível como resultado da execução de um processo completo.
- Um **Objeto de Dado** representa informação que transita ao longo do processo, tal como documentos, correio eletrônico ou cartas.
- Uma **Coleção de Objetos de Dado** representa uma coleção de informações como, por exemplo, uma lista de itens de compra.
- Um **Repositório de Dados** é um local onde o processo pode ler e escrever dados como, por exemplo, uma base de dados ou um sistema de arquivos. O repositório de dados persiste, além do tempo de vida da instância de processo que o acessa.
- Um objeto do tipo **Mensagem** é usado para representar o conteúdo de uma comunicação entre dois Participantes do processo.

Figura 3 – Notação e Modelo de Processo de Negócio BPMN 2.0

Fonte [67]

### 2.2.3.2. Outras Notações

Apesar do presente trabalho descrever a modelação de processos de negócio com a notação BPMN 2.0, importa referir que existem outras notações passíveis de utilização na modelação de processos.

Notações como Petry Nets, UML (Unified Modeling Language) e EPC (Event-driven Process Chain) possuem características que as colocam como alternativas viáveis para a modelação de processos de negócio. Todas estas têm em comum a característica de descrever processos recorrendo a atividades e, eventualmente, sub-processos [56].

Petry Nets surgem em 1962 com 4 elementos gráficos e uma semântica formal que permite a sua utilização em diferentes domínios. É essencialmente utilizada em técnicas de análise estrutural de sistemas distribuídos e é considerada como uma notação precursora de BPMN [55]. A mesma publicação sugere que uma análise, mas aprofundada a técnicas de BPM, que estas usam Petri nets para representações internas, alegando que em casos como a modelação de fluxos básicos e de mais baixo nível, processos orientados ao estado (*state-based*) face aos orientados a eventos (*event-based*), esta notação é a mais indicada. Disponibiliza ainda, devido à sua simplificada semântica operacional, uma grande variedade de técnicas de análise, desde a verificação de modelos até à simulação de processos.

UML, tal como BPMN 2.0, é uma linguagem de modelação padronizada e aceite para a modelação de processos de negócio, transversal a diferentes propósitos, na área da engenharia de software. A notação de UML pertence, no entanto, a um âmbito mais alargado, com 14 tipos de diagramas diferentes, sendo que o diagrama de atividades é usado com a função de modelar processos de negócio, já BPMN é uma notação com a função específica de modelação deste tipo de processos. A grande diferença entre ambas prende-se com o facto de o UML ser uma notação orientada a objetos (*object-oriented*) enquanto que BPMN é orientada ao processo.

EPC surge no início de 1990 dentro da framework ARIS (*Architecture of Integrated Information Systems*) com o intuito de modelar fluxos de processos de negócio orientados a eventos. Permite ordenar um conjunto de eventos e funções, permitindo também a execução alternativa e paralela de processos com recurso a operadores lógicos [57]. Apesar de não ter tanta expressão como o BPMN, por suportar um número inferior de padrões e estar associada a um menor número de ferramentas de software do que este, esta é uma notação que tem grande expressividade junto de comunidades como a SAP <sup>2</sup> e a ARIS. Permite capturar e expor visualmente diferentes processos de negócio. Possui vários elementos para a modelação de processos de negócio tal como se pode verificar na Figura 4 e é, alegadamente, mais expressiva do que a BPMN na representação de elementos como recursos, resultados do processo e os riscos nos modelos [58] devido à granularidade que os seus elementos permitem demonstrar nos modelos realizados.

Equipas que trabalhem com ARIS ou SAP vão, tendencialmente, preferir modelar o seu processo de negócio com recurso a esta notação, defendendo o descrito no ponto anterior. Já equipas que trabalham, por exemplo, com IBM vão defender que a modelação de processos com BPM é mais eficaz, no entanto ambas as abordagens são corretas e resultam para a grande maioria dos propósitos [57].

---

<sup>2</sup> <https://www.sap.com/portugal/index.html>



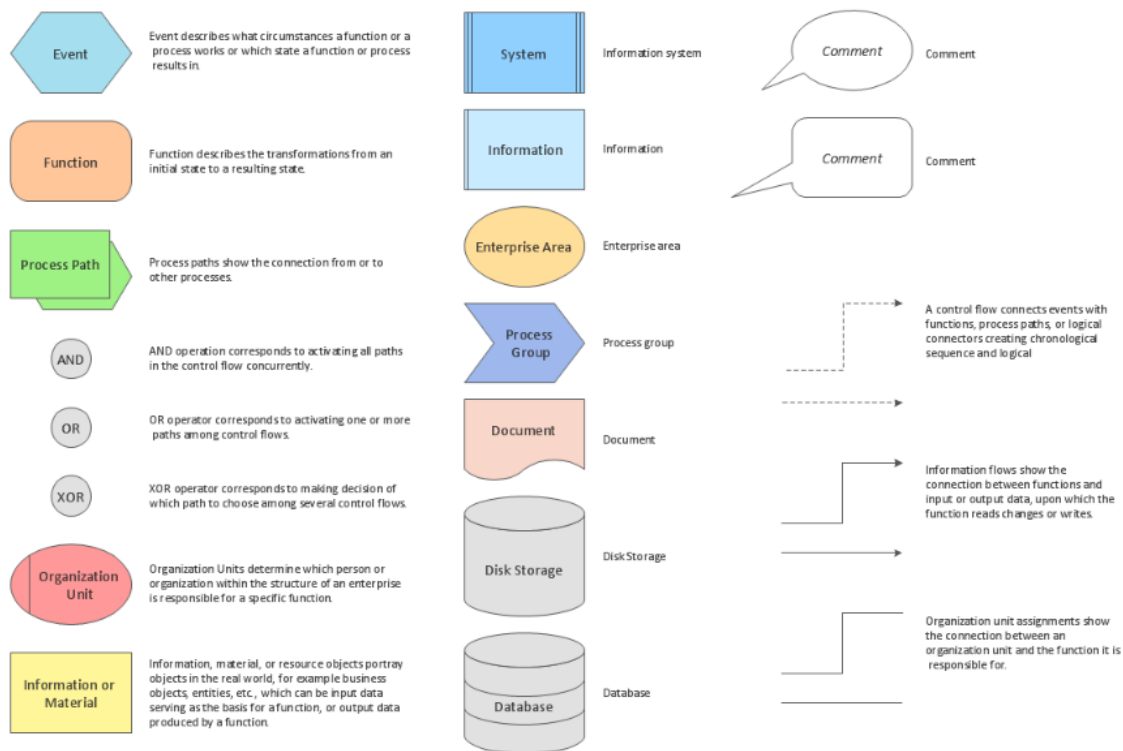


Figura 4 – Elementos da notação EPC

#### 2.2.4. Aspect Oriented BPM

Atualmente, a redução da complexidade no domínio de BPM é um tema em debate dentro da comunidade científica. A origem desta complexidade é normalmente intrínseca a uma variedade de aspetos que um processo de negócio aborda, tais como a segurança ou a privacidade. Tendo em consideração que estes aspetos podem estar sujeitos a alterações ao nível da organização, e não só especificamente do processo, podendo abranger diferentes processos dentro da organização, idealmente, estes devem ser definidos separadamente dos processos de negócio.

Estudos como [7, 8] defendem que pontos relevantes de um processo de negócio, relacionado normalmente com a segurança, devem ser transpostos para atividades genéricas passíveis de reutilização noutros pontos do mesmo processo, ou até dentro de outros processos da mesma organização.

Na Figura 5, retirada de [8], mostra quatro aspetos relevantes da área de negócio bancário, autenticação, segurança, auditoria e rastreabilidade implementados como parte integrante do processo. Esta abordagem além de aumentar a complexidade da solução, impossibilita a sua reutilização, tornando-a inflexível e não reutilizável.

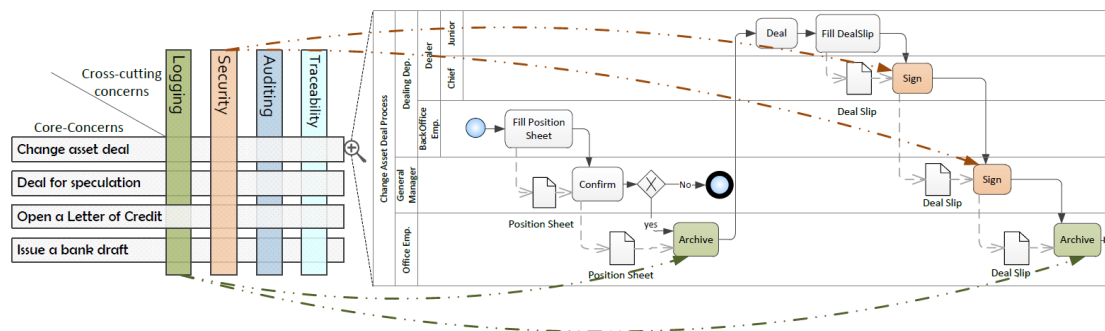


Figura 5 - Aspectos de negócio num modelo bancário de BPM

Fonte: [8]

À semelhança do que acontece no paradigma da programação orientada ao aspeto, propõe fragmentar a implementação em diferentes partes, proporcionando um nível de abstração que permita construir a solução agregando diferentes unidades funcionais autónomas.

O caso de estudo apresentado em [8], um processo simples de transferência bancária (Figura 5) podemos verificar que existem 2 aspetos a ter em consideração, autenticação e segurança. Na Figura 6 as atividades relacionadas com estes aspetos estão marcadas com cores diferentes para que possam ser distinguidas da componente exclusiva de negócio deste processo.

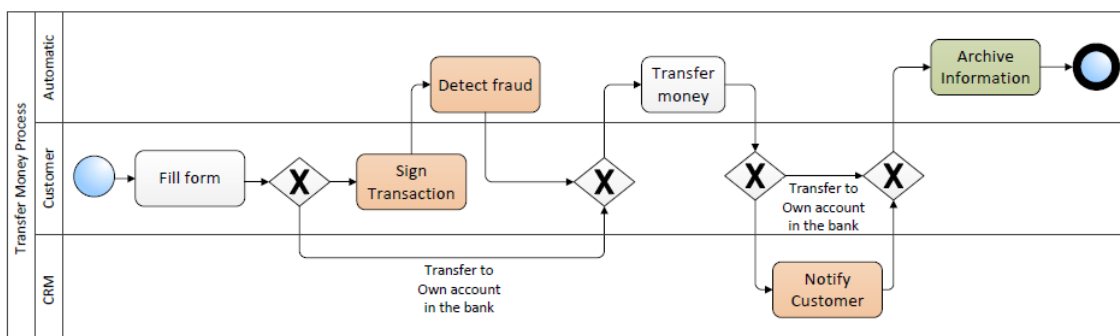


Figura 6 - Modelação no processo de transferência bancária

Fonte: [8]

A Figura 6 pretende demonstrar a modelação do mesmo processo, agora de acordo com a técnica de modelação orientada ao aspeto. Como referido anteriormente, esta abordagem implica que os aspetos relevantes sejam removidos do *core* de negócio deste processo, transformando-se em unidades funcionais independentes.

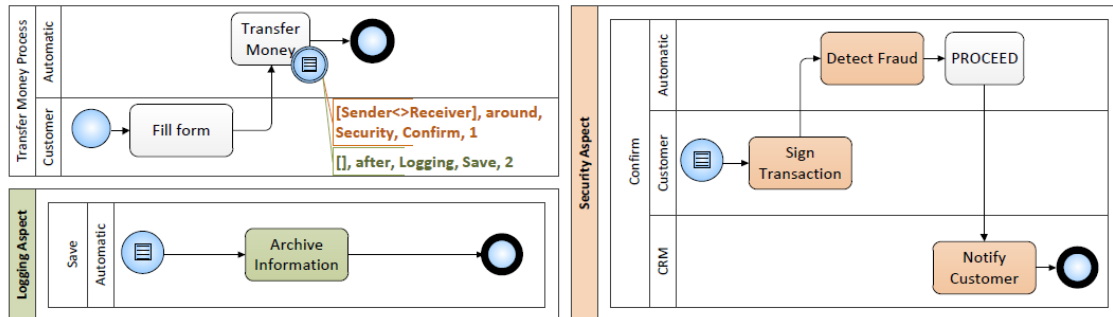


Figura 7 - Processo de transferência seguindo o princípio orientado ao aspecto

Fonte: [8]

## 2.2.5. Sistema de Gestão de Processos

É comum encontrar a sigla *BPM* junto à palavra “tecnologia” ou a sua variante no plural. De facto, o que aproxima a disciplina de BPM das práticas e benefícios das tecnologias da informação, são os sistemas de gestão de processo (*Business Process Management System/Suite* - BPMS).

Estes sistemas são ferramentas integradas que permitem às organizações, conceber processos centrados em soluções de tecnologia e engenharia de software. São usadas no desenho, implementação, manutenção e suporte do conjunto de atividades que se conjugam para formar diferentes processos de negócio nas organizações.

Este tipo de ferramentas possuem um forte carácter agregador [13], permitindo que os esforços dos diferentes intervenientes do processo, tais como o gestor, o engenheiro de software e o analista de negócio, se possam harmonizar na melhoria contínua do processo de negócio através ferramentas de modelação, *business rules engine* (BSE) e ferramentas de testes e simulação.

A origem do conceito não é recente, e emerge a partir dos *Workflow Management Systems* (WMS) em meados dos anos 90. Estes produtos surgem, na época, da necessidade emergente em reduzir o esforço de implementação de sistemas de informação capazes de lidar com a definição, administração, personalização e avaliação de processos de negócio de estruturas organizacionais em ambientes heterogéneos e distribuídos, sendo assim indicados como a primeira geração dos BPMS [14].

### 2.2.5.1. Evolução

A implementação tradicional de soluções de software empresarial, demasiado centradas em atividades unitárias e sem suporte ao processo como um todo, conduzem no início dos anos 70, os pioneiros Skip Ellis, Anatol Holt e Michael Zisman a iniciarem a sua investigação na área dos sistemas de informação de escritório (*office information systems*) [15, 16, 17, 18]. Estes sistemas sugeriam explicitamente, e de forma independente, a criação de modelos de processo para descrever negócios empresariais. Durante a década de meados de 80 a meados de 90, a investigação nesta área ficou bastante latente, pois poucas das aplicações desenvolvidas no



âmbito destas investigações, vingaram com sucesso. O insucesso vivido nos anos 80 é justificado por Skip Ellis por falhas a nível conceptual e técnico, sendo apontados fatores como a praticamente inexistência de redes, interfaces gráficas adequadas ou a falta de ferramentas de desenvolvimento adequadas [15]. No entanto van der Aalst defende que outros problemas se levantam, tais como a falta de um standard unificado de modelação de processos e sistemas demasiado rígidos [15].

Em meados dos anos 90, com a evolução ao nível dos sistemas de informação, devido essencialmente ao reforço das infraestruturas de rede e ao surgimento de novas ferramentas de apoio ao desenvolvimento de sistemas de informação empresariais, dá-se uma nova e vibrante fase de investigação na área dos WMS, sistemas percursos dos BPMS aqui em análise.

Van der Aalst identifica quatro fases no ciclo de vida dos sistemas BPM [15]:

- 1 – Desenho do processo;
- 2 – Configuração do sistema;
- 3 – Promulgação do processo;
- 4 – Diagnóstico.

Durante a fase do desenho do processo pretende-se a modelação dos objetivos de negócio que este pretende cumprir com atenção às diferentes perspetivas de abrangência, tais como o fluxo de dados, o modelo organizacional ou os aspetos operacionais da organização.

Na configuração do sistema, é necessário idealizar o sistema centrado no processo de negócio e incluir as configurações necessárias para que se materialize com base no desenho do processo na fase anterior.

Durante a promulgação o sistema de informação referido nas fases anteriores, é efetivamente utilizado.

A última fase, de diagnóstico, está diretamente relacionada com as possíveis melhorias identificadas. Durante esta fase novas tecnologias poderão vir a ser exploradas, bem como técnicas para aumentar a performance e produtividade.

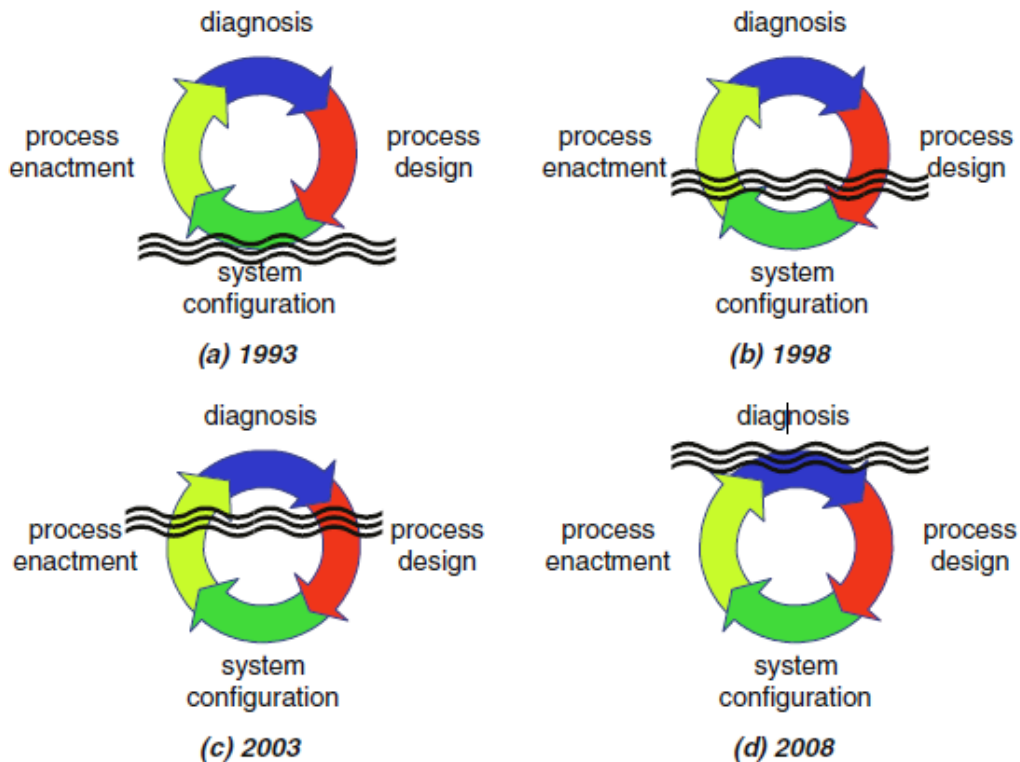


Figura 8 - O ciclo de vida de BPM  
 Fonte: [15]

### 2.2.5.2. Melhorias dos BPMS

Entre as melhorias mais relevantes que os BPMS trouxeram relativamente às tradicionais implementações de sistemas de informação, está o aumento da proximidade entre os diferentes envolvidos na implementação do processo de negócio, aproximando os *process owners* dos analistas funcionais e dos analistas técnicos [13].

Numa implementação tradicional, o(s) analista(s) de negócio procuram junto do cliente (*process owner*), fazer o levantamento dos requisitos de negócio que o sistema de informação procurará resolver. Após este trabalho, o(s) analista(s) de negócio transmitem o resultado do seu trabalho ao(s) analista(s) funcional(ais) que deve fazer o levantamento funcional do sistema a desenvolver bem como identificar necessidades específicas a nível funcional [11, 14]. Por sua vez, este(s) transmite(m) o resultado da sua análise ao analista técnico, que terá como principal função toda a configuração do sistema de acordo com as especificações levantadas da análise funcional e desenho de especificações técnicas para que possam ser implementadas.

Como é possível notar, desde o cliente (*process owner*) até à implementação, existem diferentes camadas de comunicação, com características próprias. Este facto vai impulsionar, naturalmente, entropia na cadeia de comunicação, resultando na implementação de um sistema que não corresponde à real necessidade do negócio e que foi levantado pelo *process owner* na primeira camada de todo este procedimento.

Com o BPMS é possível criar uma via de comunicação comum entre o *process owner* e todos os restantes intervenientes na implementação, envolvendo todos estes atores no desenho da solução.

## 3. Análise de Negócio

Neste capítulo são apresentados os detalhes de negócio inerentes à solução desenvolvida. Na secção 3.1 é feita uma introdução à aplicação no qual será integrado o módulo desenvolvido, seguida da identificação e caracterização dos atores na secção 3.2.

Na secção 3.3 é realizado um levantamento dos requisitos funcionais com a identificação de pressupostos e casos de uso nas subsecções seguintes.

Na secção 3.4 são identificados os requisitos não funcionais. Na secção 3.5 é feita uma introdução aos principais conceitos do domínio do problema, representados num diagrama de modelo de domínio com os seus atributos e relações.

Depois dos conceitos caracterizados, apresenta-se na secção 3.6 uma modelação inicial e simplificada dos processos de negócio com interesse para o trabalho proposto com o propósito de apresentar brevemente as principais atividades envolvidas.

### 3.1. Aplicação Bankee

No âmbito desta dissertação, propôs-se desenvolver um módulo, parte integrante de toda uma solução de *frontend*, aplicação “Bankee”, em desenvolvimento para uma instituição bancária angolana.

O desenvolvimento do projeto “Bankee”, iniciado há cerca de 6 anos, surge da necessidade por parte da instituição bancária de implementar um novo sistema de *frontend* que, progressivamente, lhe permitisse integrar e uniformizar o acesso aos principais sistemas de gestão económica, gestão documental, monitorização e execução de processos, atendimento ao público, gestão de níveis de intervenção, autorização e validação disponibilizado à massa orgânica do Banco, tanto na rede de balcões como no *backoffice*, bem como o sistema de gestão pessoal disponibilizado online a clientes.

Com este novo sistema, a instituição bancária pretende resolver e endereçar problemas reais da sua operação, estendendo o atual núcleo bancário em AS400, “BANKA<sup>3</sup>”.

A “Banka” embora suporte a grande maioria das operações de negócio necessárias à operação do cliente, é um sistema sem suporte para a implementação de fluxos de negócio complexos. A falta de suporte para a integração desses mesmos fluxos em processos de negócio mais complexos, como por exemplo, a recolha de documentos e mecanismos de aprovação e validação de processos, conduzem à existência de situações de risco operacional, que, em última instância este risco poderá conduzir a instituição a perdas significativas provocadas por eventos de fraude interna.

É neste contexto que se eleva o trabalho desenvolvido. No seguimento do plano de implementação modular previsto, a instituição bancária levantou prioridade ao desenvolvimento dos módulos de transferências que incluem, o módulo de transferências interbancárias. Adicionalmente, pretendia-se que fosse incluído um mecanismo dinâmico de

---

<sup>3</sup> Banka - produto de core bancário desenvolvido em IBM-AS400.

intervenção hierárquica durante a execução dos processos de transferências interbancárias, com o objetivo de consumir dados de um outro módulo autocontido, “Limites de Autorização”, para possibilitar o controlo de fluxo do processo de negócio na atribuição de tarefas de autorização e validação a utilizadores com perfis específicos dentro da hierarquia da instituição.

As regras de negócio a aplicar na modulação do processo de transferências interbancárias, com o intuito de calcular dinamicamente intervenção hierárquica, são configuradas no módulo de “Limites de Autorização”. Este módulo não foi desenvolvido no âmbito deste trabalho.

Além da implementação integrada de todos os ecrãs e serviços deste módulo de transferências interbancárias, o foco esteve centrado na modelação e implementação do processo de negócio através de BPMN 2.0, bem como na integração da funcionalidade de intervenção hierárquica dinâmica com vista a resolver o problema previamente descrito neste documento.

## 3.2. Atores

Para o processo de negócio em estudo, foram identificados diferentes atores. Não é fácil descrever com precisão todos os atores, pois será a funcionalidade de intervenção hierárquica dinâmica que determinará a quantidade de atores envolvidos. No entanto, através das atividades de intervenção hierárquica, autorizar, validar e dar parecer, poderemos extrapolar a envolvimento de, pelo menos, quatro atores diferentes: o utilizador que inicia o processo (balconista), o aprovador, validador e consultor. No entanto, cada uma das atividades envolvidas poder-se-á repetir por um número variável de vezes, para atores com perfis distintos.

Procede-se, de seguida, à identificação das funções gerais de cada um dos atores identificados:

- **Balconista** – É o responsável pela interação direta com o cliente, e por inserir todos os dados relevantes para o processo, tais como o número de conta, valor a transferir, *International Bank Account Number* (IBAN) da conta de destino, documentação necessária.
- **Aprovador** – Tem o poder de aprovar ou rejeitar a execução da operação. É o ator com maior relevância durante todo o processo, pois a execução da operação só efetiva com o seu aval.
- **Validador** – Acede aos dados do processo com a finalidade de validar a documentação envolvida bem como a(s) assinatura(s) do(s) cliente(s). Este utilizador poderá recusar, ou não, a validação, mas não poderá aprovar a execução do processo.
- **Consultor** - Este utilizador tem a função única, de emitir parecer por escrito sobre a operação do processo em curso, para possa ser analisada pelos restantes intervenientes do processo.

## 3.3. Requisitos Funcionais

### 3.3.1. Pressupostos

Os requisitos das funcionalidades listadas de seguida, serão os mesmo que se encontram já em utilização na aplicação “Bankee”, e decorrentes das fases de implementação antecedentes:

- Associação de documentos;
- Aba de imagens na visualização de documentos;
- Arquivo físico;
- Pesquisa de contas;
- Alertas e mensagens;
- Validação de assinaturas;
- Configuração de limites de autorização;

### 3.3.2. Casos de Uso

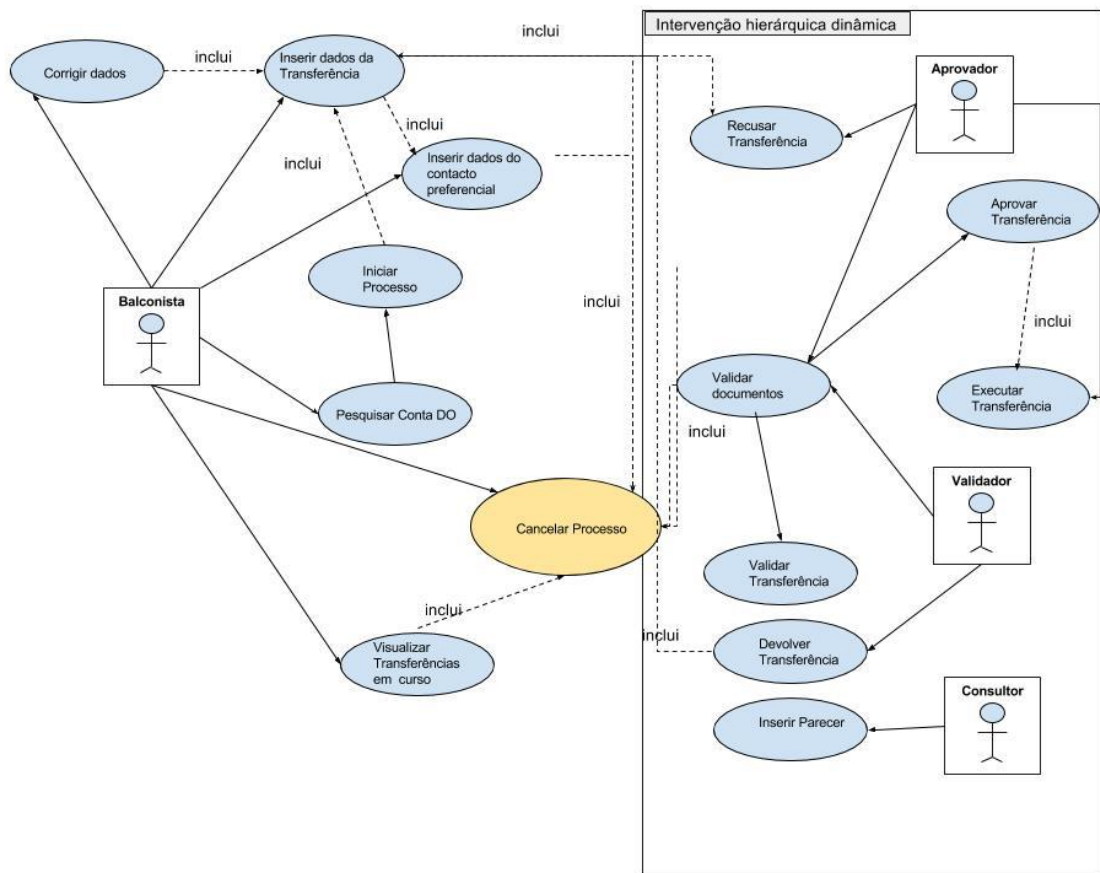


Figura 9 - Diagrama de casos de uso para o módulo proposto

Na Figura 9 estão representados, em diagrama, os casos de uso identificados para o módulo de transferências interbancárias com intervenção hierárquica dinâmica. Na tabela 3 serão descritos cada um dos casos de uso identificados para o desenvolvimento deste processo. O âmbito da análise funcional a este módulo ainda não se encontra fechado, pelo que poderão surgir algumas alterações durante o decorrer deste estudo.

*Tabela 1 - Casos de uso do ator "Balconista"*

<i>Ator</i>	<i>Caso de Uso</i>	<i>Descrição</i>
	Pesquisa conta DO (depósito à ordem)	Requer a introdução do número de conta à ordem e evocação dos serviços necessário à obtenção de todos os dados da mesma.
	Iniciar Processo	A ação de iniciar o processo de transferências interbancárias a partir de uma conta à ordem.
	Inserir dados da transferência	Refere-se à ação de inserção de dados referentes à transferência. O que inclui o IBAN de destino, montante, tipo de transferência, entre outros.
	Inserir dados do contacto preferencial	Inserção dos dados de contacto preferencial para o cliente que realiza o pedido de transferência.
<i>Balconista</i>	Visualiza transferências em curso	Visualização de lista de transferências em curso para o mesmo cliente.
	Cancela Processo	Ação de cancelar o processo em curso, terminando todas as atividades inerentes ao mesmo.

*Tabela 2 - Casos de uso do ator "Aprovador"*

<i>Ator</i>	<i>Caso de uso</i>	<i>Descrição</i>
<b>Aprovador</b>	Valida documentos	Ato de validar os documentos anexos ao processo.
	Aprova transferência	A ação de aprovação da operação de transferência interbancária.
	Executa transferência	O ato de validação da operação que se traduz na execução efetiva da mesma.
	Recusar transferência	Ação que remete o processo novamente ao utilizador inicia processo, na atividade de registar transferência e consequentemente ao ecrã de inserção de dados da transferência.



Tabela 3 - Casos de uso do ator "Validador"

<i>ATOR</i>	<i>CASO DE USO</i>	<i>DESCRIÇÃO</i>
<i>VALIDADOR</i>	Valida documentos	Ato de validar os documentos anexos ao processo.
	Valida transferência	Ação que confirma a validação do IBAN a creditar, o valor a transferir, as assinaturas do cliente e que a transferência em curso não é duplicada.
	Devolve transferência	Ação de devolução da transferência acompanhando de comentário obrigatório a indicar o motivo. O processo é devolvido à atividade de registrar transferência no ecrã inserir dados de transferência.

Tabela 4 - Casos de uso do ator "Consultor"

<i>ATOR</i>	<i>CASO DE USO</i>	<i>DESCRIÇÃO</i>
<i>CONSULTOR</i>	Inserir parecer	Ação de inserir parecer face à análise dos dados da operação de transferência em curso, sobre a forma de comentário, obrigatório, ao processo.

### 3.4. Requisitos Não Funcionais

Nesta secção é feito um levantamento de requisitos não funcionais. Alguns destes, tais como os referidos na Tabela 8 e Tabela 9, estão garantidos de base na aplicação e disponíveis para reutilização nos novos módulos. Já aqueles referidos na Tabela 5, Tabela 6, Tabela 7 e Tabela 10 devem ser garantidos durante o processo de desenvolvimento de novos módulos, e como tal, também no módulo de transferências interbancárias.

Tabela 5 – Características da estação de trabalho

<b>Nome</b>	Características da estação de trabalho		
<b>Categoria</b>	Sistema	<b>Prioridade</b>	Alta
<b>Descrição</b>	<p>O parque informático da instituição bancária é composto essencialmente por estações de trabalho com sistema operativo "Windows 8.1".</p> <p>Para este sistema operativo são aplicadas todas as atualizações lançadas pelo fabricante.</p> <p>A aplicação Bankee, tal como cada um dos módulos que a constitui, deve manter o normal funcionamento após cada atualização do sistema operativo, tanto a nível funcional como de performance.</p>		

Tabela 6 – Características do Browser

<b>Nome</b>	Características do Browser		
<b>Categoria</b>	Sistema	<b>Prioridade</b>	Alta
<b>Descrição</b>	Sendo a aplicação <i>Bankee</i> um sistema web baseado em comunicação entre cliente (browser) – servidor. Atendendo à uniformização do parque informático do cliente, esta deve estar otimizada para IE (Internet Explorer) versão 11.		

Tabela 7 – Resolução de ecrã

<b>Nome</b>	Resolução de ecrã		
<b>Categoria</b>	Sistema	<b>Prioridade</b>	Alta
<b>Descrição</b>	<p>Os monitores existentes no parque informático da instituição bancária são de marcas e modelos variáveis, pelo que se torna imperativo definir uma resolução de ecrã padrão e transversal a todos estes.</p> <p>A aplicação <i>Bankee</i>, bem como todos os módulos que a integram, deverão estar otimizados para uma resolução de ecrã de 1024x768px.</p>		

Tabela 8 – Gestão de envio de documentos

<b>Nome</b>	Envio de Documentos Para o Repositório Central		
<b>Categoria</b>	Sistema	<b>Prioridade</b>	Alta
<b>Descrição</b>	<p>As limitações impostas pelo terreno onde algumas agências estão implementadas, tais como acesso a energia elétrica, disponibilidade de rede e largura de banda implicam a existência de um mecanismo de gerenciamento do envio de documentos para o repositório central de forma assíncrona.</p> <p>Assim, deve ser implementado um proxy que faça a gestão do envio assíncrono de documentos para o repositório central, gerindo o envio dependendo da capacidade de rede disponível.</p> <p>O carregamento e envio de documentos não deverá interromper o fluxo normal dos processos de negócio nem impedir o normal funcionamento dos balcões.</p>		

Tabela 9 – Tratamento de assinaturas

<b>Nome</b>	Tratamento de assinaturas dos clientes		
<b>Categoria</b>	Sistema	<b>Prioridade</b>	Alta
<b>Descrição</b>	<p>Com o intuito de manter os processos de negócio ágeis, é necessário garantir que as assinaturas dos clientes são recolhidas, tratadas e armazenadas para posterior utilização.</p> <p>As assinaturas dos clientes devem ser recolhidas, tratadas e armazenadas num repositório central para que possam ser acedidas pelo sistema, no âmbito dos diferentes processos de negócio e respetivas validações, sempre que se mostre necessário.</p>		

Tabela 10 – Acessibilidade da aplicação Bankee

<b>Nome</b>	Acessibilidade da aplicação Bankee		
<b>Categoria</b>	Sistema	<b>Prioridade</b>	Alta
<b>Descrição</b>	A aplicação Bankee deve cumprir as recomendações de acessibilidade para aplicações web descritos pela W3C <sup>4</sup> .		

<sup>4</sup> World Wide Web Consortium - <https://www.w3.org/>

### 3.5. Modelação do Domínio

Nesta secção será apresentada a modelação de domínio para a interação do pedido de transferência com o módulo de Limites de Autorização descrito mais aprofundadamente na secção 5.2.

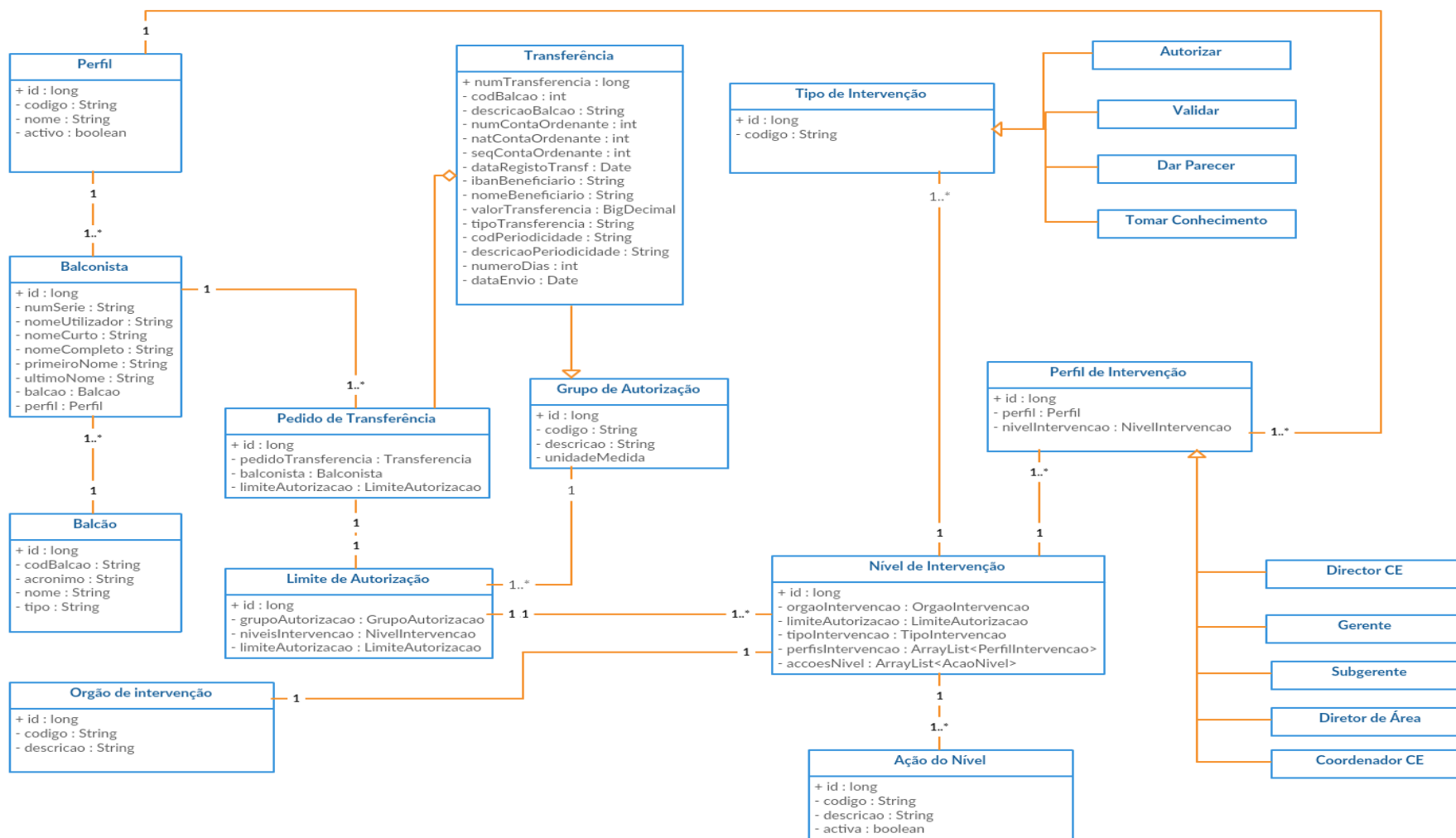


Figura 10 – Modelo de domínio

### 3.6. Modelação Preliminar do Processo de Negócio

Nesta secção serão apresentados os modelos preliminares de estudo para a implementação do processo de negócio de transferências interbancárias recorrendo à notação BPMN 2.0. Tendo em consideração que, aquando da modelação dos presentes modelos, a implementação ainda não estaria em curso, estes devem ser considerados os primeiros modelos que serviram de base para o desenvolvimento dos modelos finais apresentados na secção 6.3.3, tanto no que compete ao processo, como a cada uma das suas atividades.

A modelação dos modelos aqui apresentados teve uma participação ativa do cliente e da equipa de análise funcional, tendo sido desenvolvidos durante a fase de *design* do processo de desenvolvimento da solução.

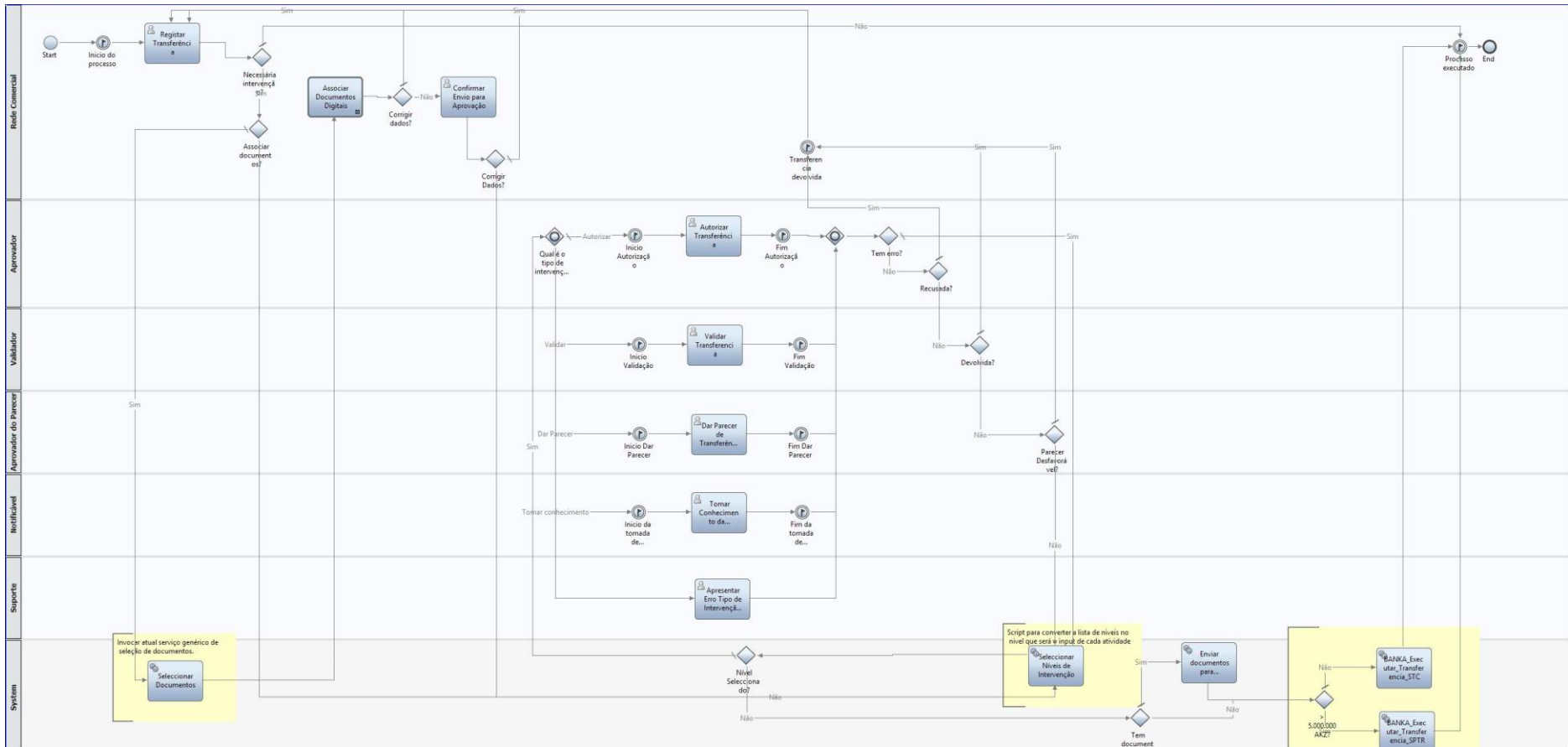


Figura 11 - Diagrama do processo de negócio de transferências interbancárias, visão geral

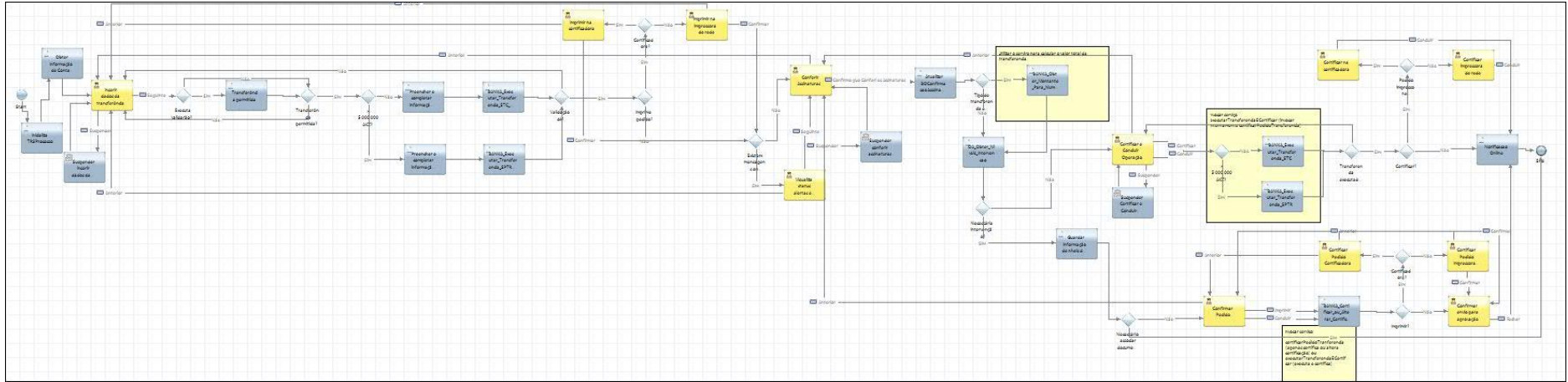


Figura 12 - Diagrama da 1ª atividade, "Registrar transferência"

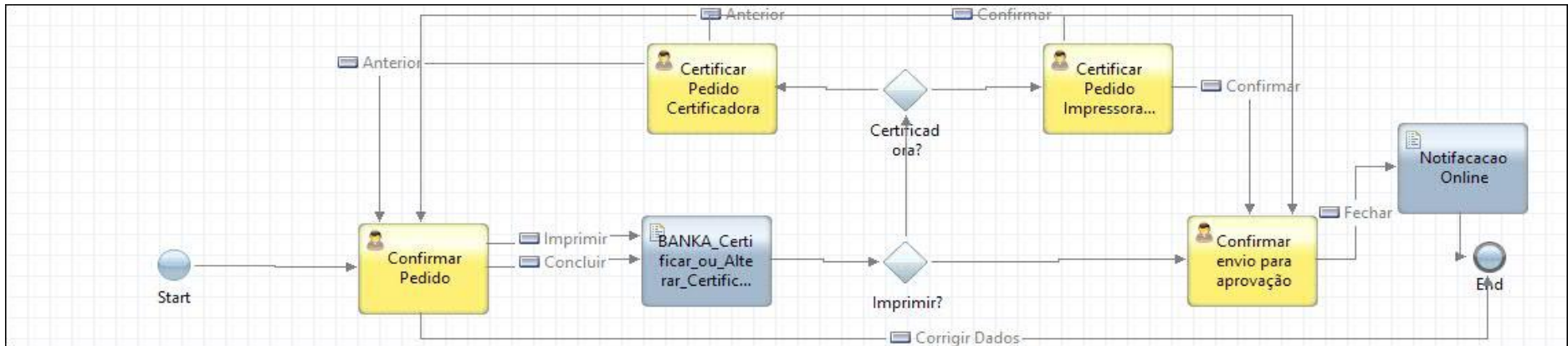


Figura 13 - Diagrama da atividade de "Confirmar Envio Para Aprovação"

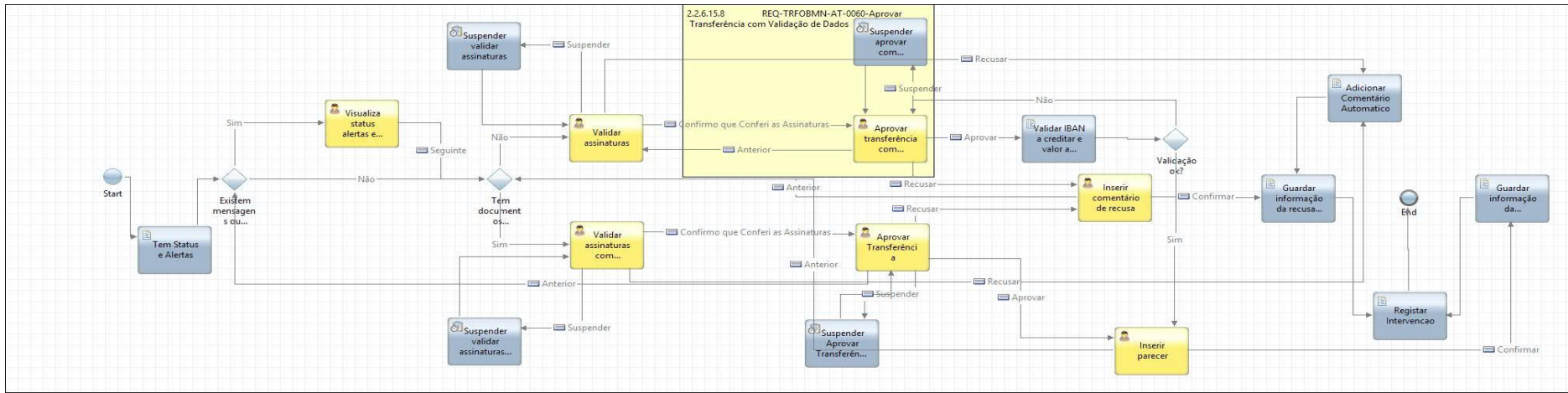


Figura 14 - Diagrama da atividade "Aprovar Transferência". Incluída na intervenção hierárquica dinâmica

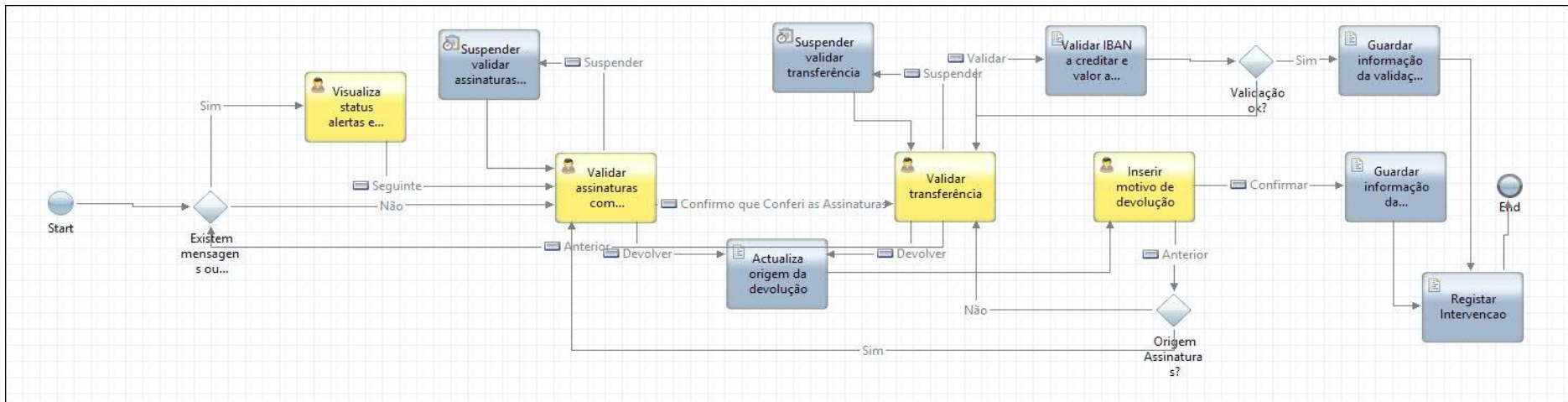


Figura 15 - Diagrama da atividade "Validar Transferência". Incluída na intervenção hierárquica dinâmica





## 4. Tecnologias Adotadas

O módulo desenvolvido no âmbito desta dissertação, pela sua envolvência, área de negócio a que se destina, e por se tratar de uma solução à medida que será parte integrante da aplicação *Bankee* não deve ser analisado de forma isolada no que compete à identificação e avaliação de soluções e abordagens existentes. É imperativo referir que as tecnologias a utilizar no desenvolvimento do módulo têm de ser as mesmas da aplicação em que este será integrado.

Assim, e partindo deste pressuposto, são analisadas as soluções e abordagens que estão a ser aplicadas em cada um dos componentes que compõem a arquitetura apresentada no capítulo 5.1, comparativamente com outras que poderiam cumprir a mesma função.

Devem-se considerar três componentes principais na arquitetura da aplicação *Bankee* que, conseqüentemente, serão os mesmos pilares em que assenta o módulo desenvolvido no âmbito deste estudo. Serão portanto avaliadas soluções de *Frontend*, *Backend* e *Business Process Management Suite* (BPMS).

### 4.1. Tecnologias de Frontend

A componente de *frontend* do módulo desenvolvido compreende os ecrãs com os quais o utilizador irá interagir e executar todas as ações de *input* necessárias ao normal decorrer do processo de negócio de transferências interbancárias. Ações tais como a inserção do número de conta, montante a transferir, moeda a transferir, entre outras, serão elementos integrantes de todo este conjunto. Também as validações e lógica de cliente, estão desenvolvidas neste componente.

Existe, atualmente, uma grande variedade de soluções para implementação de *frontend*, no entanto e considerando que o *core* de desenvolvimento da aplicação *Bankee* é baseado em JAVA EE, serão consideradas algumas *frameworks* de *user interface* (UI), que se considerem mais relevantes e tenham esta como tecnologia de base.

#### 4.1.1. Tecnologia em Uso

Nesta secção será descrita a tecnologia atualmente em uso no desenvolvimento da camada de *frontend* do módulo de transferências interbancárias.

##### 4.1.1.1. Zkoss

Esta é a solução atualmente em uso para o desenvolvimento desta camada da arquitetura da aplicação *Bankee*. O desenvolvimento da camada de *frontend* do módulo de transferências interbancárias foi também, conseqüentemente, desenvolvido com recurso a esta *framework*.

A principal motivação para a escolha desta *framework* face a outras analisadas neste capítulo, prende-se essencialmente com questões de performance e consumo de recursos em que o ZK mostrou resultados mais satisfatórios face às restantes *frameworks* [35].

Também conhecida na indústria pela abreviatura “ZK”, esta é uma *framework* com distribuição *open source* sob a licença LGPL (*Lesser General Public License*), bem como distribuição comercial.

Esta *framework* é escrita em JAVA e o seu *core* consiste num mecanismo baseado em AJAX e orientado a eventos. O design de cada página de *frontend* é implementado através de componentes ricos escritos em XHTML (*Extensible Hypertext Markup Language*) ou XUL (*Extensible User-interface Language*), que por sua vez, são manipulados através de eventos lançados pela interação dos utilizadores. A descrição dos componentes em linguagem baseadas em XML facilita a implementação das vistas e agiliza o processo de desenvolvimento de ecrãs.

A componente de JavaScript está embebida na *framework* através de *AJAX*, permitindo que os componentes sejam implementados sem necessidade de conhecer, ou escrever código nesta linguagem.

Implementa o padrão de interface gráfica do utilizador orientado a eventos [26] através da invocação de manipuladores de eventos que correm no servidor em resposta a eventos do DOM (*Document Object Model*), lançados através do browser.

Possui perto de 200 componentes suporta HTML5 e CSS3 e bibliotecas externas, tal como *Bootstrap*, permitindo criar designs responsivos.

Suporta diferentes padrões, tal como o MVC (*Model-View-Controller*), MVP (*Model-View-Presenter*) e MVVM (*Model-View-ViewModel*).

Disponibiliza ainda anotações em ZUML (*ZK User Interface Markup Language*) que permitem automatizar a sincronização de estado e automatizar as operações de criação, leitura, atualização e remoção (CRUD) de forma transparente entre a vista, o controlador e o modelo de dados. Suporta também o uso de java, como linguagem de programação unificada no desenvolvimento da interface de utilizador bem como no servidor, graças ao “*Beanshell*” embebido. Não obstante, suporta também a utilização de outras linguagens JVM (*Java Virtual Machine*) para desenvolvimento no servidor, tais como “*JRuby*” e “*Jython*”.

Por fim, destaca-se a possibilidade de criação de componentes personalizadas bem como macro componentes com a possibilidade de reutilização por diferentes ecrãs da aplicação.

#### **4.1.2. Alternativas**

Na presente secção, serão descritas tecnologias com características que lhes permitiriam também estas ser usadas em alternativa para o desenvolvimento da camada de frontend do módulo de transferências interbancárias.

##### **4.1.2.1. Framework JSF**

Java Server Faces (JSF) é uma *framework* orientada ao desenvolvimento de aplicações web em JAVA que implementa o *design pattern* “*Model-View-Controller*” (MVC). O objetivo da JSF é agilizar o desenvolvimento de aplicações web, minimizando o esforço para a sua manutenção. Possibilita integrar outras tecnologias de desenvolvimento de *frontend*, tal como CSS, HTML, validação e conversão de dados no desenvolvimento de novos componentes

A arquitetura MVC, representada na Figura 16, permite abstrair a camada de negócio, da camada de apresentação, dividindo o sistema em 3 partes diferentes, com responsabilidades distintas:

- **Model** – Na camada do modelo, armazenam-se os dados que por sua vez mantêm o estado da aplicação, disponibilizando o acesso aos mesmo sempre que necessário. Quando há alterações de dados, tipicamente, esta camada notifica os seus controladores da alteração, por forma a expandir os novos dados para as restantes camadas.
- **View** - Esta camada é a responsável por exibir os dados, normalmente após tratamento, que são obtidos na camada *model*. Na camada de *view* também se captam os *inputs* do utilizador relativos aos dados apresentados.
- **Controller** – Na camada do controlador, é feita a ligação entre as outras duas camadas desta arquitetura. Trata de atualizar a camada *view* sempre que há alterações nos dados transmitidos pela camada *model*, bem como recebe os *inputs* do utilizador, atualizando a *model* sempre que este manipula a *view*. É também nesta camada que se desenvolve alguma lógica de negócio e de validação de *inputs* e *outputs*.

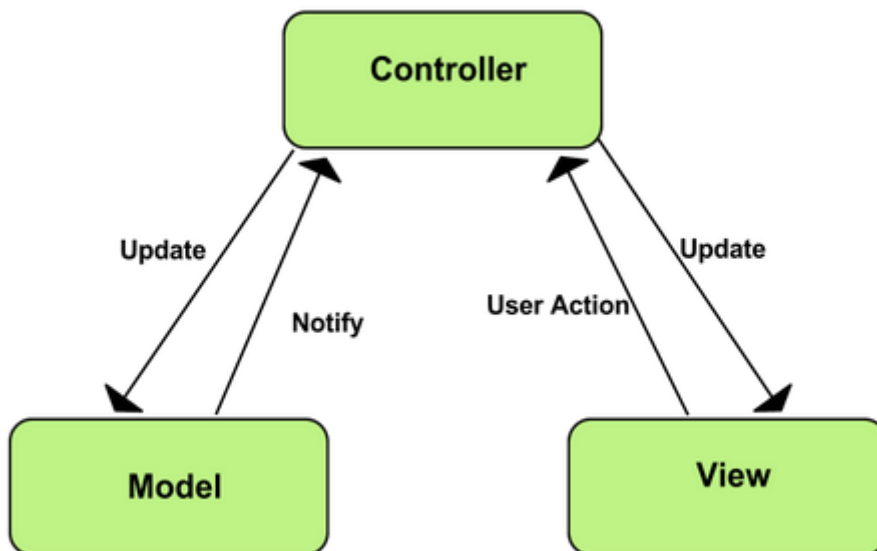


Figura 16 - Arquitetura MVC  
Fonte: [25]

#### 4.1.2.2. ICEfaces

IceFaces é uma *framework open-source* para JAVA EE baseado em *JavaServer Faces (JSF)* que implementa alguns componentes padrão, alguns representados na Figura 17, através AJAX incorporado, com o intuito de criar aplicações RIA (*Rich Internet Application*). É uma arquitetura orientada ao componente e implementado a definição do UI (*user interface*) com recurso a *tags* declarativas, suportando um sistema de *data binding* dinâmico.

As bibliotecas desta *framework* suportam técnicas de design responsivo e adaptativo, são baseadas em HTML5, e além disso, integra com a tecnologia *Bridgelt*, que permite desenvolver aplicações móveis híbridas, baseadas em HTML5.

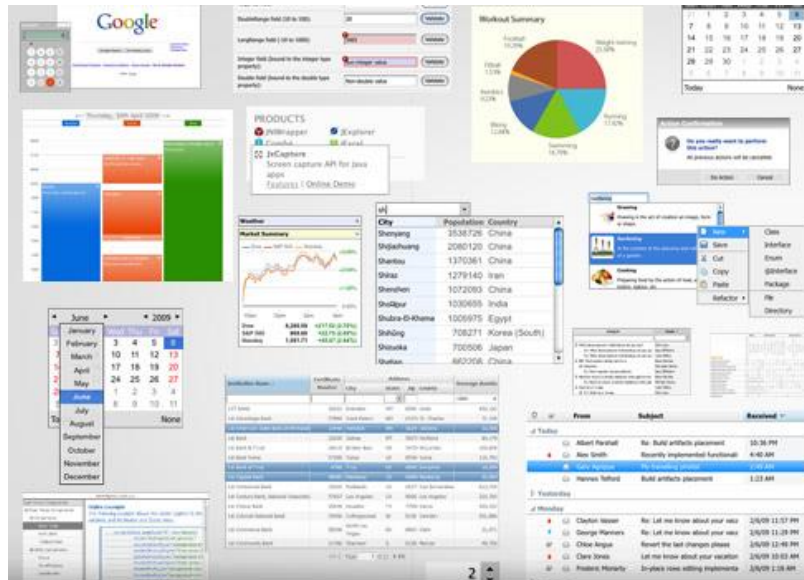


Figura 17 - Alguns componentes de IceFaces  
Fonte: [21]

Possui uma biblioteca com cerca de oitenta componentes [22] para aplicações web, que combinam técnicas de renderização baseadas no servidor e no cliente, com foco no uso balanceado do processamento em rede e do servidor, de forma a oferecer uma experiência rica e eficiente ao utilizador. A Figura 17 mostra alguns destes componentes, além disso possui também uma biblioteca de componentes orientados ao desenvolvimento de aplicações móveis [23] com diferentes características de HTML5 e CSS3 suportadas em mobile, bem como acesso a funcionalidades de input, multimédia e geolocalização, entre outros.

Além da versão *open source*, está ainda disponível uma versão comercial, com componentes mais robustos e orientados a versões de software certificadas para a industria, facto corroborado pela total cobertura SLA (*Service Level Agreements*).

#### 4.1.2.3. PrimeFaces

Tal como a *IceFaces*, a *PrimeFaces* é uma *framework open source* para *JAVA EE* baseada no standard JSF. Possui mais de 100 componentes de UI (*User interface*) de grande qualidade, compatíveis com HTML5, que incorporam *AJAX* abstraindo o uso de *JavaScript* na implementação de cada componente. Os componentes são responsivos, adaptativos e suportados pela maioria dos browsers e dispositivos móveis.

Além dos componentes padrão que *PrimeFaces* disponibiliza para o desenvolvimento de interfaces de utilizador, também importa referir as extensões "*PrimeUI*". Estas, que derivam

para uma *spin-off* de *PrimeFaces* e disponibiliza uma coleção de *widgets* JavaScript baseados na biblioteca *Jquery*. A versão “*PrimeUI*” 3.0 já se encontra disponível e é composta por “*PrimeElements*”, uma biblioteca de componentes web para criar interfaces de utilizador de forma declarativa através de elementos HTML personalizados.

*PrimeFaces*, disponibiliza ainda o Prime NG, que se trata de uma coleção de componentes de UI para AngularJS2 para o desenvolvimento de interfaces RIA.

Esta é uma *framework* bastante completa e com um nível de maturidade bastante elevado. Tal como o gráfico da Figura 18 mostra, fruto de uma votação online informal, lançada pelo website “*DevRates*”, *Primefaces* é a *framework* mais votada.



Figura 18 - Votação online informal no website DevRates.com

Fonte: [24]

Este resultado [24] deve-se à grande disponibilidade de componentes que as diferentes bibliotecas da *framework* disponibilizam, notando-se que em número, são superiores aos da *framework* analisada anteriormente, *IceFaces*. A simplicidade, facilidade de implementação e a grande disponibilidade de documentação com exemplos práticos, aliados a uma grande comunidade de utilizadores, têm contribuído para o sucesso que esta tem alcançado junto dos utilizadores.

## 4.2. Tecnologias de Backend

À semelhança da realidade face às soluções de *frontend* existentes no mercado, existem também diversas alternativas para o desenvolvimento da camada de *backend* da nossa aplicação, e conseqüentemente o módulo de transferências interbancárias.

### 4.2.1. Tecnologia em Uso

Tendo em consideração a magnitude da aplicação *Bankee*, é imperativo analisar algumas das tecnologias de desenvolvimento *backend* existentes, com o objetivo de identificar diferentes características que satisfaçam as necessidades mais importantes, fiabilidade, estabilidade, eficiência e robustez.

Existem diferentes alternativas no mercado que cumprem com todos os pressupostos suprarreferidos, e em última análise, a escolha, JAVA, dependeu de outros fatores tais como, a organização onde se produz a aplicação, o nível de *expertise* dos seus colaboradores nesta tecnologia, as bibliotecas disponíveis e até da sua popularidade junto da comunidade.

Existem diferentes índices de referência como [27] que mostram as linguagens de programação com maior popularidade no mercado, contudo após breve análise a alguns destes verifica-se que o TOP 5, representado na Figura 19, raramente muda. A razão para esta evidência, reside no facto de todas estas linguagens já terem atingido um nível de maturidade e fiabilidade que se reflete na opção, frequentemente tomada, por diferentes profissionais e organizações para a implementação dos seus projetos.

Para a síntese desta análise às linguagens de programação de *backend*, o foco estará em Java e .Net, ambas no top das mais populares, tendo sido JAVA a escolhida para o desenvolvimento da aplicação *Bankee*, e perpetuada no desenvolvimento do caso em estudo neste documento.

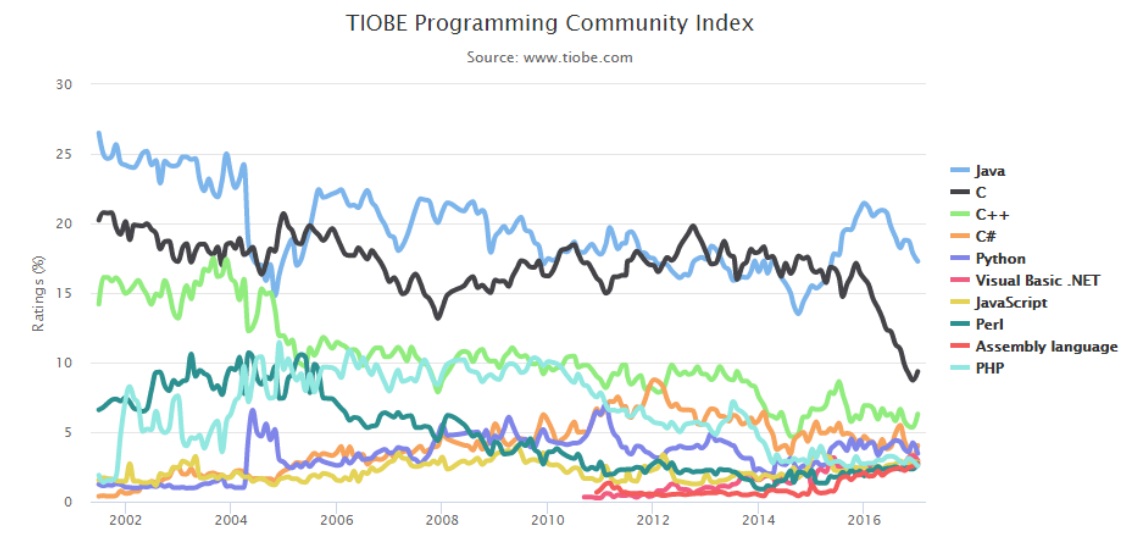


Figura 19 - Índice TIOBE com o top 10 das linguagens de *backend*  
Fonte: [27]

#### 4.2.1.1. JAVA

Java é uma linguagem de programação relativamente recente, criada em 1995 pela “Sun Microsystems”. É uma linguagem de alto nível, multiplataforma e orientada a objetos.

A sua sintaxe deriva de C++, sendo estas bastante idênticas, contudo o seu modelo de objetos tem menos recursos de baixo nível, tornando-se assim mais simples de implementar. As aplicações escritas em Java, são compiladas para *bytecode* e executadas dentro da JVM (*Java Virtual Machine*).

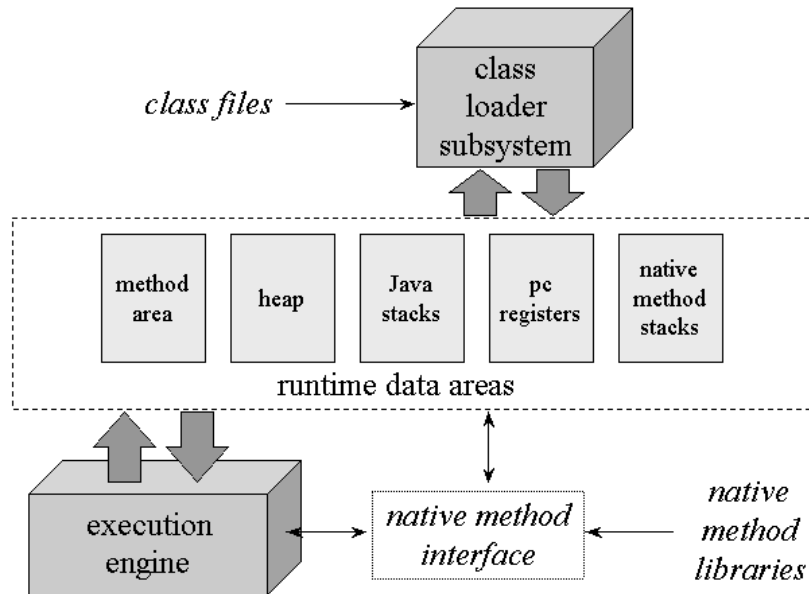


Figura 20 - Arquitetura JVM com os principais subsistemas e áreas de memória  
Fonte: [30]

A JVM refere-se a uma máquina virtual abstrata, que carrega o *bytecode* gerado pelo compilador, através do *class loader*, e o converte em código executável. É responsável pela execução da aplicação Java, bem como por toda a gestão de memória durante a sua execução. Diferentes implementações da JVM, podem conter diferentes limitações relativas à gestão de memória. No entanto, a natureza abstrata da sua especificação, permite que a implementação da JVM numa grande variedade de dispositivos e computadores seja eficaz.

A JVM permitiu ainda que a “Sun Microsystems” trouxesse uma nova filosofia para descrever a natureza multiplataforma desta linguagem, a WORA (*Write Once, Run Anywhere*). Tal significa que o código fonte de uma aplicação Java, sendo somente dependente do interprete na JVM, só necessita de ser escrito uma vez para compilar e executar em diferentes sistemas sem a necessidade de adaptação ou uso de bibliotecas externas, tal como acontece com C#.

Desde a versão 8, que também suporta os paradigmas de programação procedimental e funcional, é, contudo, em grande parte indicada para a programação orientada a objetos.

Devido à dimensão da aplicação “Bankee”, bem como a complexidade inerente à implementação de cada um dos módulos, a arquitetura de Java, as suas características multiplataforma e a disponibilidade de uma equipa com *know how* relevante nesta tecnologia, tornaram-na como a opção para o desenvolvimento de todo o *backend* desta aplicação, e consequentemente, de cada um dos seus módulos.



#### 4.2.2. Alternativa

É importante sublinhar neste ponto o facto da aplicação Bankee se encontrar já em avançado estado de desenvolvimento, o que por si só significa que a tecnologia de *backend* usada na sua implementação - JAVA - foi definida no início do projeto, mantendo-se este plano para os módulos a desenvolver no futuro. Esta secção pretende, no entanto, realçar as características de outra tecnologia existente no mercado que poderiam justificar a sua utilização no desenvolvimento da camada de *backend* desta aplicação e consequentemente do módulo de transferências interbancárias.

##### 4.2.2.1. Plataforma .Net

A plataforma .Net surge em 2002 como parte da estratégia da Microsoft para criar uma plataforma única de desenvolvimento e execução de aplicações. Permite a utilização de diferentes linguagens de programação nativas, tais como C# e VB.NET e não nativas, tal como PHP desde que usadas em máquinas Windows.

A plataforma .Net é composta por 5 componentes principais:

5. Sistema operativo Windows;
6. Servidores Windows Server;
7. Os elementos principais da Framework .Net. Common Language Runtime (CLR) e Framework Class Library (FCL).
8. .Net Building Block Services;
9. Ambiente integrado de desenvolvimento *Visual Studio*.

As linguagens de programação de .NET são linguagens orientadas a objetos, permitindo o uso de classes, herança e polimorfismo no desenvolvimento de projetos.

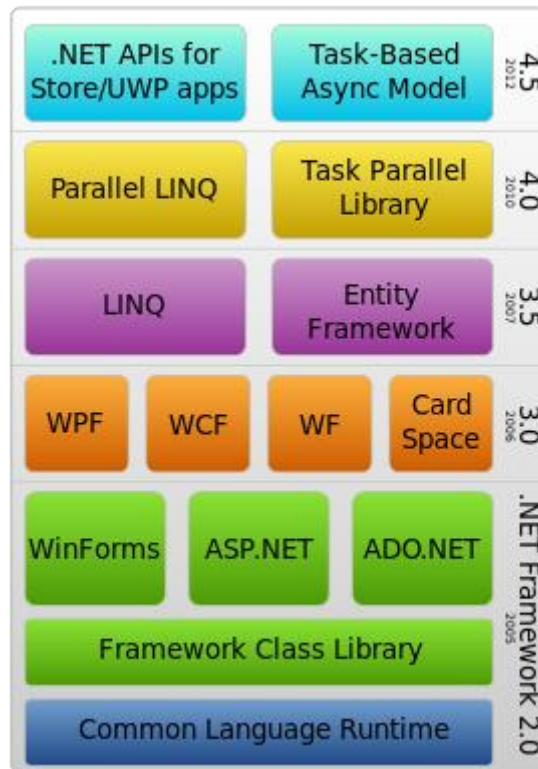


Figura 21 - Stack da framework .NET  
Fonte: [49]

Assegura a possibilidade de desenvolvimento modular, permitindo a reutilização de classes, bem como o uso de dependências externas dentro de projetos proprietários. A um nível mais baixo, a própria FCL da framework .NET já carrega componentes, módulos e controladores para os projetos desenvolvidos com esta tecnologia, permitindo assim que estes tenham desde o primeiro momento funcionalidade disponível e previamente integrada no sistema.

Permite o desenvolvimento de aplicações empresariais para um âmbito alargado de plataformas, tais como web, *desktop* e *mobile*.

Proporciona uma grande capacidade para o desenvolvimento e integração de *webservices* o que permite tornar as aplicações mais extensíveis e interoperáveis.

De uma forma geral, a framework .NET apesar de ser mais recente do que JAVA, já atingiu uma maturidade que, tal como JAVA, a dotam com um conjunto de características que a tornariam também numa boa opção para o desenvolvimento do Projeto Bankee.

### 4.3. Soluções de BPM

Devido à crescente preocupação das empresas em manter os seus processos de negócio ágeis, otimizados e monitorizados, a implementação de soluções de BPM a este nível tem vindo a aumentar significativamente, e com este as ferramentas de apoio à implementação desta disciplina. Atualmente existem mais de 150 ferramentas [31] disponíveis para a prática de BPM e das atividades que a constroem.

Para este estudo em concreto, importa que as soluções existentes a analisar cumpram com os requisitos fundamentais da aplicação, tais como uma arquitetura orientada a serviços (SOA), possibilidade de integração de sistemas de *backend*, integração de sistemas de *frontend*, controlo e gestão de processos de negócio, integração com sistemas mobile e possibilidade de monitorização em tempo real.

Atualmente, existem três grandes concorrentes neste mercado, a IBM, Oracle e PegaSystems, com produtos direcionados ao desenvolvimento, implementação, gestão, monitorização e integração de processos de negócio através das suas ferramentas integradas. Sendo que a Oracle e a IBM são empresas de reconhecida qualidade, ambas comercializam produtos consistentes para a implementação de BPM em processos de negócio de áreas tão sensíveis e exigentes como a banca, serão objeto de análise e comparação na Tabela 11.

Tabela 11 - IBM BPM vs. Oracle BPM

	IBM BPM	Oracle BPM
<b>Características</b>	<ul style="list-style-type: none"> <li>- Disponibiliza três edições do produto. Edição Express, Standard e Advanced.</li> <li>- Na edição Advanced permite a utilização com o Websphere Process Server (WPS), providenciando capacidades de integração avançadas;</li> <li>- Disponibiliza um Process Designer compatível com BPMN 2.0;</li> <li>- Monitorização com relatórios em tempo real;</li> <li>- Armazenamento de dados de desempenho;</li> </ul>	<ul style="list-style-type: none"> <li>- Disponibiliza uma única edição, a Oracle Business Process Management Suite 12C;</li> <li>- Inclui o Oracle BPM Studio para a realização da modulação de processos, criação de user interfaces (UI), e definir as integrações do processo;</li> <li>- Gestor de conteúdos integrado que permite adicionar diferentes tipos de conteúdos ao processo e criar documentação em tempo real;</li> <li>- Monitorização de atividade de negócio, através de um monitor de eventos, transmitindo em tempo real para um dashboard.</li> </ul>

Tabela 11 - IBM BPM vs. Oracle BPM

<p><b>Principais Clientes [32]</b></p>	<ul style="list-style-type: none"> <li>- Possibilidade de desenvolver UI para dispositivos móveis;</li> <li>- Ferramentas de design de integração (SOA / BPEL / ESB);</li> <li>- Suporte à transacionalidade;</li> </ul>	<ul style="list-style-type: none"> <li>- Possibilidade de desenvolver UI para dispositivos móveis;</li> </ul>
	<ul style="list-style-type: none"> <li>- Barclays;</li> <li>- Emericon;</li> <li>- Banca Popolare di Milano;</li> <li>- CST Consulting;</li> <li>- KeyBank;</li> <li>- Prolifics;</li> <li>- State of Alaska;</li> <li>- Humana S.A.;</li> <li>- Banco Espirito Santo;</li> </ul>	<ul style="list-style-type: none"> <li>- State revenue Office Victoria;</li> <li>- New South Wales Health;</li> <li>- REDISA;</li> <li>- Panduit;</li> <li>- Total E&amp;P Indonesie;</li> </ul>

O projeto Bankee implementa os seus processos de negócio através do produto IBM BPM (IBPM) versão *Advanced*, e esta foi a solução usada para o desenvolvimento do caso de estudo desta dissertação. Considera-se, contudo, que a suite da Oracle também permitiria cumprir os objetivos traçados e pretendidos para o desenvolvimento e implementação dos processos de negócio da aplicação Bankee.

A razão que motiva a escolha do produto da IBM para a implementação dos processos de negócio, e subseqüentes funcionalidades, é anterior ao início do trabalho documentado nesta dissertação e prende-se titularmente com a relação de confiança que a IBM tem vindo a desenvolver com o cliente e a fiabilidade demonstrada através do reconhecimento obtido ao longo dos anos, junto de clientes importantes dentro da indústria bancária [32].



## 5. Design Arquitetural

O presente capítulo é dedicado ao design arquitetural da solução proposta. Na secção 5.1 é demonstrada a arquitetura alto nível da aplicação Bankee com detalhe sobre os componentes que a integram.

A secção 5.2 descreve o módulo de limites de autorização, fulcral para a resolução do problema, bem como a sua arquitetura.

### 5.1. Arquitetura da Aplicação Bankee

A arquitetura de alto nível do projeto Bankee está representada no diagrama representado na Figura 22. É possível ter uma noção genérica de todos os componentes que fazem parte desta aplicação, bem como as estratégias de comunicação utilizadas.

Consequentemente, o módulo de transferências interbancárias a desenvolver seguirá o mesmo padrão arquitetural.

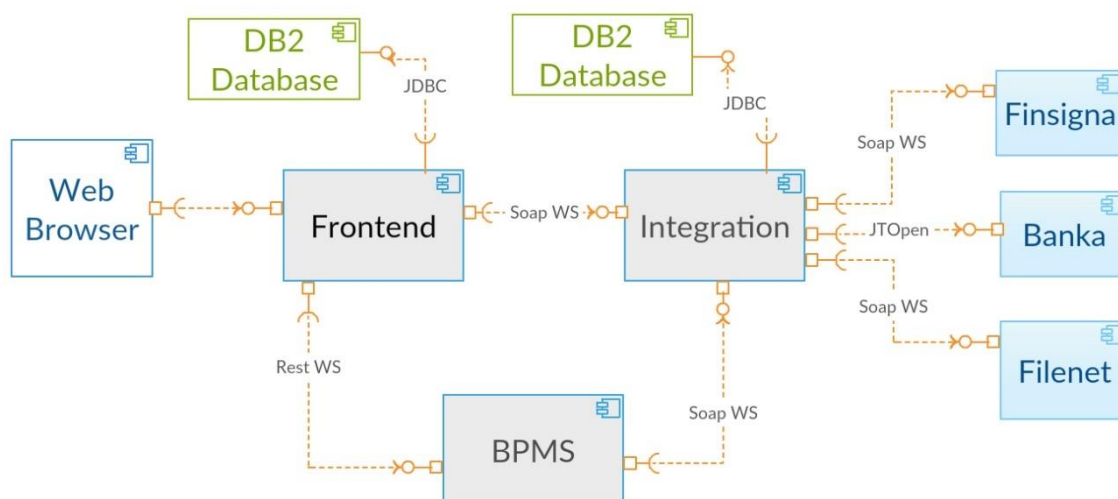


Figura 22- Arquitetura alto nível do módulo de transferências interbancárias

Os principais componentes são os seguintes:

- O *Frontend* refere-se a toda a camada de interface de utilizador, desenvolvido com base na *framework Zkoss*, já descrita durante o terceiro capítulo deste estudo. Aqui são implementados todos os elementos dos ecrãs de interação com o utilizador, bem como as respetivas validações no cliente. Comunica por *JDBC (Java Database Connectivity)* com uma base de dados local em *DB2*, o sistema de bases de dados relacionais da IBM, onde guarda dados e configurações a consumir nesta camada.
- O *Integration* é a referência a toda a camada de *backend*, onde se encontram implementados os serviços *SOAP (Simple Object Access Protocol)* de integração entre o core bancário e os componentes da arquitetura, *Frontend* e *BPM*. À semelhança do que acontece com o *Frontend*, também este comunica por *JDBC* com uma base de dados *DB2* com o intuito de guardar dados e configurações a consumir nesta camada.

- *Finsigna* trata-se do repositório digital de assinaturas. Apesar de ter sido incluído no desenvolvimento do módulo proposto, não será alvo de análise mais detalhada durante este documento pelo motivo de se tratar de uma funcionalidade previamente implementada na aplicação Bankee, havendo somente lugar à integração de algo já criado e configurado. O mesmo acontece com o *Filenet*, o repositório digital de documentos. Através destas tecnologias é possível armazenar e consultar assinaturas e documentos de clientes respetivamente, resolvendo assim dois problemas já endereçados no primeiro capítulo, nomeadamente a ausência de um sistema de gestão documental e de suporte aos processos bancários e um repositório digital de assinaturas, os quais motivam a instituição bancária cliente a investir num projeto como o Bankee.
- A “*Banka*” representa o *core* bancário atual, com o qual a aplicação integra funcionalmente para executar operações de negócio nucleares e movimentar dados sensíveis. Este sistema, também abordado durante o capítulo 1, é baseado em AS400 e disponibiliza *APIS RPG (Report Program Generator)* que comunicam com o módulo de integração através de *JTOpen*, uma biblioteca de Java que suporta ligações cliente/servidor a sistemas que correm em IBM i, tal como o AS400, podendo ser usado por *applets*, *servlets* e aplicações para aceder aos dados e recursos destes sistemas.
- O *BPMS (Business Process Management System)* é a representação da camada de modelação e implementação de processos de negócio. Como referido no capítulo 4.3, a solução em uso é a *IBM BPM Advanced*, que se interliga com a componente de integração através de *webservices SOAP (Simple Object Access Protocol)* e com a componente de Frontend através de *webservices REST (Representational State Transfer)*.

Apesar do módulo de transferências interbancárias incluir tarefas de desenvolvimento nas componentes de *Frontend*, *Integration* e *BPMS*, o foco principal será na modelação do processo de negócio, em *BPMN*, que inclui a integração de uma funcionalidade de intervenção hierárquica dinâmica essencial na resolução do problema previamente descrito. Importa para esta funcionalidade, descrever com algum detalhe o módulo de limites de autorização, o cursor que permite configurar as regras de negócio necessárias para que se cumpra o objetivo proposto.

## 5.2. Limites de Autorização

Trata-se de um módulo autocontido, uma funcionalidade de manutenção com o objetivo de inserir, duplicar, alterar, remover e simular a utilização de limites de autorização indexados a distintos processos e operações de negócio da entidade bancária, de acordo com parâmetros idênticos aos representados na Figura 23.

O módulo aqui descrito não será desenvolvido no âmbito desta dissertação, trata-se, no entanto, de um ponto chave na realização deste trabalho que não poderia passar despercebido. Será através da interação deste módulo com o processo de transferências interbancárias que serão calculados os atores da hierarquia da instituição com um papel ativo nas tarefas de aprovação, validação e dada de parecer na funcionalidade de intervenção hierárquica dinâmica.

O limite de autorização, define um escalão de valor ao qual é indexado um conjunto de regras para o processo de negócio que venha a lidar com um montante inferior ou igual ao que aí foi definido.

Exemplo:

Limite 1 – 290.000,00 AKZ

Limite 2 – 2.000.000,00 AKZ

Se o montante da operação for de 290.000,00 AKZ será aplicada a definição do Limite 1.

Se o montante da operação for de 290.001,00 AKZ será aplicada a definição do Limite 2.

### **Níveis de Intervenção**

Os níveis de intervenção estão intrinsecamente ligados aos limites de autorização, pois nenhum destes poderá existir sem o outro.

Os níveis de intervenção referem-se a momentos ordenados de intervenção no processo, também estes com regras de negócio específicas e configuráveis. Cada limite de autorização poderá ter configurados entre 1-N níveis de intervenção, sendo que o valor numérico com que estes são configurados determina a ordem com que são executados (Figura 23). É também permitido o paralelismo, ou seja, níveis de intervenção a executar em paralelo durante o processo.

### **Grupo de Autorização**

Como referido, o módulo de limites de autorização permite configurar limites monetários ou percentuais, dependendo do grupo de autorização em causa. O objetivo dos grupos de autorização é permitir definir limites de autorização para um grupo de operações, como por exemplo, transferências ou depósitos a prazo. No caso das transferências, o grupo de operações engloba as operações de transferências interbancárias e intrabancárias.

O limite de autorização é caracterizado pela unidade de medida do grupo de autorizações ao qual pertence, por exemplo, no caso de o limite ser configurado para o grupo de autorizações de transferências, a unidade de medida a utilizar será monetária, sendo necessário definir um limite monetário mínimo, a partir do qual, haverá necessariamente, de forma aleatória ou contínua, a intervenção de um dos atores da hierarquia, ou seja, é definido um limite monetário e a transação cujo valor seja inferior ou igual a este, deverá cumprir as regras definidas na configuração desse limite de autorização.



Grupo de Autorizações	Tipo OE	Código do Órgão	Limite	Ações no Registo	Tipo de Autorização	Nível Intervenção	Condição de Execução	Tipo de Intervenção	Ações Nível de Intervenção	Órgão Intervenção	Tipo OD	Código do Órgão	Perfis	
Transferências	Agencia	N/A	300.000		Aleatória	10		Autorizar	Telefonar ao Cliente se Não Presencial	Órgão Executor	N/A	N/A	GERENTE	
														SUBGERENTE
			2.500.000		Sempre	10		Autorizar	Telefonar ao Cliente se Não Presencial	Órgão Executor	N/A	N/A	GERENTE	
														SUBGERENTE
			20.000.000	1-Associar pedido do cliente	Sempre	10		Autorizar		Órgão Executor	N/A	N/A	GERENTE	
						20		Validar		Outro	N/A		BO TRANSFERENCIAS	
						30		Autorizar	Telefonar ao Cliente	Órgão Domicilio	Agencia	N/A	DIRECTOR AREA	
											Centro de Empresas	N/A	DIRECTOR_CE	
													COORDENADOR_CE	
											Centro de Investimento	N/A	GERENTE	
													SUBGERENTE	

Figura 23 - Exemplo de parametrização dos limites de autorização para o grupo de autorizações de transferências

### **Tipo de Aprovação**

O tipo de aprovação identifica a abordagem como as operações de negócio são selecionadas para aprovação, tendo dois valores possíveis:

Aleatória – As transações que se incluem no limite são selecionadas por amostragem, recorrendo a uma função *random*, são selecionadas para aprovação as operações em que o resultado da execução desta função seja menor ou igual a um valor configurado por omissão de 0.20.

Sempre – As operações em que o montante se encontre dentro do intervalo de valores pertencentes ao limite de autorização, serão sempre selecionadas para tarefas de intervenção hierárquica.

### **Tipo de Intervenção**

Representa o tipo de intervenção que será aplicado ao nível de intervenção definido. O tipo de intervenção ditará a posição do interveniente, e a ação que este deverá tomar durante a intervenção hierárquica dinâmica no decorrer do processo de negócio.

Os valores possíveis para os tipos de intervenção estão definidos na Tabela 12, no entanto, é da responsabilidade do processo tratar do encaminhamento do fluxo para o tipo de intervenção adequado bem como garantir que toda a funcionalidade inerente a este é disponibilizada.

Tabela 12 - Tipos de Intervenção possíveis

<b>VALOR</b>
Autorizar
Validar
Dar Parecer
Tomar Conhecimento

### **Ação**

Refere-se à ação configurada para o momento do registo do limite de autorização ou para algum dos seus níveis de intervenção e que será executada pelo utilizador em determinada fase do processo, sendo, no entanto, da responsabilidade do processo disponibilizar a funcionalidade necessária à execução da ação definida. É possível definir múltiplas ações tanto para o momento do registo, como para cada um dos níveis de intervenção. As ações disponíveis para configuração na versão atual do módulo são as descritas na Tabela 13.

Tabela 13 - Alguns exemplos de ações dos limites de autorização

<b>CÓDIGO</b>	<b>DESCRIÇÃO</b>
<b>01</b>	Telefonar ao Cliente
<b>02</b>	Telefonar ao Cliente se não presencial
<b>03</b>	Associar Pedido do Cliente
<b>04</b>	Enviar SMS para Cliente Autorizar

### **5.2.1. Arquitetura do Módulo de Limites de Autorização**

O módulo de limites de autorização segue o mesmo princípio de arquitetura orientada a serviços que caracteriza a aplicação *Bankee*, extraindo, contudo, o componente BPM.

Tal como referido anteriormente, este trata-se de um módulo autocontido incluído na árvore do projeto, que integra com a restante aplicação através de uma SOAP API que expõe serviços para os módulos cliente.

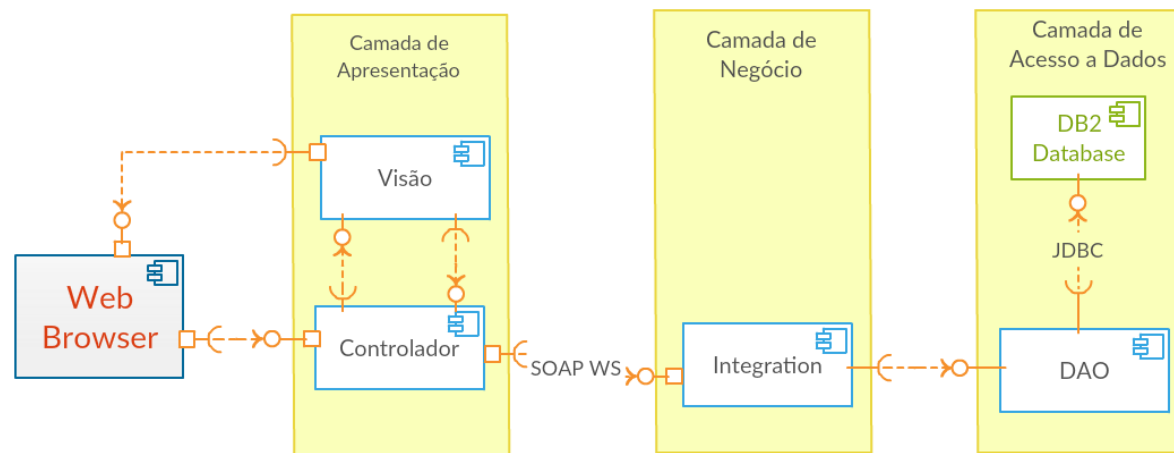


Figura 24 - Arquitetura alto nível do módulo de limites de autorização

A funcionalidade de configuração dos limites de autorização parte da mesma base arquitetural da aplicação *Bankee*, com a diferença de que a Figura 24 apresenta maior granularidade na representação dos componentes da arquitetura dos limites de autorização, enquanto que a Figura 22 mostra uma visão mais genérica da aplicação *Bankee*.

Há, no entanto, dois pontos a destacar nesta arquitetura relativamente à arquitetura geral da aplicação. É possível observar a inexistência do componente de BPM, que apesar deste módulo integrar com a aplicação através do BPMS da IBM em uso no âmbito do presente estudo, não faz parte da sua arquitetura de base.

A camada de dados, é, no caso do módulo de limites de autorização, de tratamento proprietário, enquanto que na aplicação *Bankee* os dados são fornecidos pelo *core* bancário, Banka<sup>5</sup> com recurso ao canal de comunicação “JOpen”.

<sup>5</sup> Software de *core* bancário atualmente em produção paralelamente ao projeto *Bankee*. <https://www.exictos.com/home/software-services/banking/software/pfs-core-banking/pfs-core-banking>

## 6. Design e Implementação

Durante o presente capítulo são descritos os detalhes relacionados com a implementação da solução proposta. Serão abordados temas como as tecnologias utilizadas no desenvolvimento, metodologia e diferentes fases do processo de implementação do módulo de transferências interbancárias.

### 6.1. Tecnologias

O processo de implementação do módulo de transferências interbancárias depende de uma série de tecnologias que inevitavelmente estão presentes em diferentes fases do desenvolvimento. O presente subcapítulo pretende descrever os ambientes que fazem parte da infraestrutura da aplicação *Bankee*, bem como as ferramentas utilizadas na aplicação das tecnologias descritas pormenorizadamente no capítulo 4 e destacar a sua relevância no processo de implementação e desenvolvimento.

#### 6.1.1. Infraestrutura

A infraestrutura necessária à implementação do módulo de transferências interbancárias é a mesma de toda a aplicação *Bankee* e devem ser destacados 5 componentes fundamentais: o ambiente local de desenvolvimento, servidor aplicacional, servidor de processos, servidor de base de dados e *core* bancário.

Existem outros elementos da infraestrutura que são essenciais para todo o processo de desenvolvimento da aplicação *Bankee*, tais como os servidores com os ambientes de teste em qualidade, proxy<sup>6</sup> para gestão de envio de documentos para o repositório central, ambientes FileNet<sup>7</sup> para a gestão e armazenamento de documentos e o ambiente de base de dados.

---

<sup>6</sup> Maior detalhe no subcapítulo 5.4.

<sup>7</sup> <http://www-03.ibm.com/software/products/pt/filecontmana>

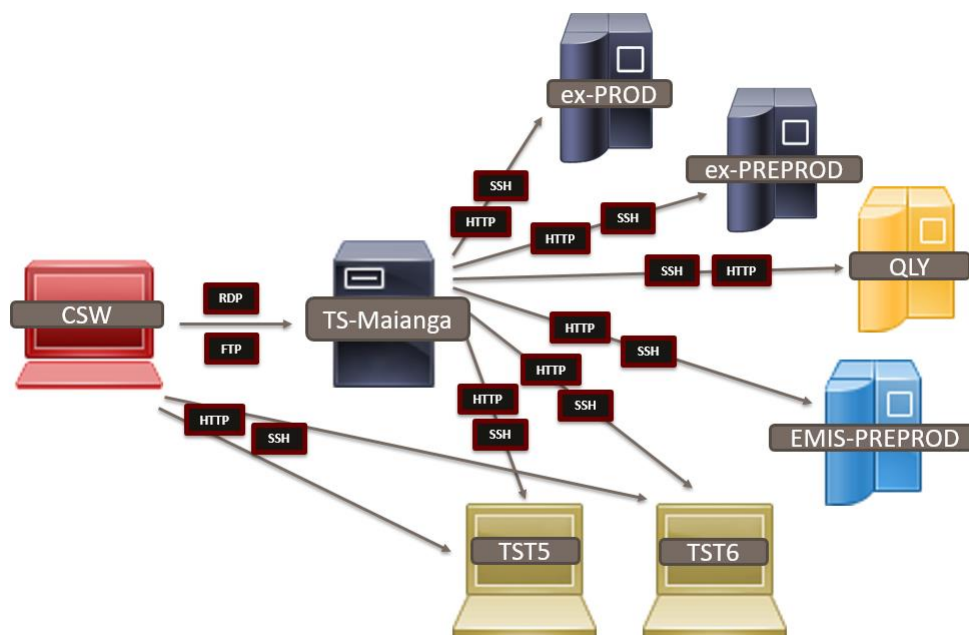


Figura 25 - Comunicação da infraestrutura de qualidade e pré-produção na aplicação Bankee

A Figura 25 representa de forma genérica todas as máquinas que fazem parte da infraestrutura de qualidade de pré-produção da aplicação Bankee. É ainda importante referir que à data da escrita deste documento o projeto iniciou o processo de migração da infraestrutura para a nuvem através de uma iniciativa de “*Infrastructure as Code*” (IaC).

### Ambiente local de desenvolvimento

A estação de trabalho a partir da qual é realizado todo o desenvolvimento deste trabalho é um PC baseado em x64, com processador Intel® Core™ I7-4710MQ CPU @ 2.50GHz, 2494 Mhz com 4 Cores, 16GB de memória RAM com sistema operativo Microsoft Windows 8.1 Pro 64bit<sup>8</sup>.

Nesta máquina está instalado o software Vagrant para suporte à criação e manutenção do ambiente virtual com o servidor aplicacional local. A ferramenta Vagrant<sup>9</sup> permite configurar, num único fluxo, a criação de máquinas virtuais, bem como todo o software necessário de forma automática, permitindo que ambientes completos de desenvolvimento possam ser criados numa questão de minutos. O objetivo desta ferramenta é precisamente providenciar uma forma ágil de criar ambientes virtuais de desenvolvimento para suporte a projetos de desenvolvimento de software de dimensão elevada.

O comando “vagrant up” inicializa o fluxo de instalação do ambiente de desenvolvimento local, lança a *build* de uma máquina virtual com uma distribuição Linux Ubuntu<sup>10</sup> 64 bit. Neste processo é instalado o servidor aplicacional IBM Websphere<sup>11</sup>.

<sup>8</sup> <https://www.microsoft.com/pt-pt/software-download/windows8>

<sup>9</sup> <https://www.vagrantup.com>

<sup>10</sup> <https://www.ubuntu.com/>

<sup>11</sup> <http://www-03.ibm.com/software/products/pt/appserv-was>

Toda a camada de servidor da aplicação Bankee corre na versão 8.5 do *Websphere Application Server* (WAS).

Paralelamente, e para tornar os ambientes de desenvolvimento local mais leves foi criado um ambiente de desenvolvimento partilhado, pcenter-bpm85, que está alojado na máquina CSW8. Esta encontra-se sob gestão da DSI da Critical Software nas instalações de Coimbra. É equipada com um processador Intel® Core™ I7 a 2.6 GHz com 4 CPUs e 32 GB de RAM e alberga os servidores de base de dados, servidor BPM / APP e servidor de FileNet<sup>1</sup>.

### 6.1.2. Ferramentas de Desenvolvimento

Na Critical Software cada estação de trabalho pessoal é entregue ao colaborador com uma configuração personalizado para as necessidades do projeto e do colaborador dentro do mesmo. O projeto Bankee não é exceção, e à semelhança do que acontece com os demais projetos a estação de trabalho é entregue com a configuração de base de desenvolvimento JAVA.

#### IntelliJ IDEA

A ferramenta que adquire maior relevância na estação de trabalho de um programador JAVA é o IDE (*Integrated Development Environment*), neste caso concreto trata-se do IntelliJ IDEA<sup>12</sup> na versão *Ultimate*. É uma ferramenta muito completa, com diferentes funcionalidades que permitem ao programador desenvolver código de forma mais ágil e num fluxo contínuo, com funcionalidades avançadas de *intelligent code completion* e análise de controlo de fluxos de código não só para JAVA com também para uma grande variedade de outras linguagens tais como JavaScript, TypeScript, HTML, CSS, SQL, entre outras. Tem ainda disponível uma quantidade alargada de plugins de integração com praticamente todas as *frameworks* existentes, tal como para Zkoss, a *framework* de UI usada no projeto Bankee para o desenvolvimento das interfaces com o utilizador. Todas estas características, entre outras que não foram destacadas, fazem deste software a ferramenta ideal para o programador do projeto Bankee.

#### IBM Process Designer

O “*Process Designer*” da IBM trata-se de uma ferramenta disponibilizada pela IBM para o desenho, modelação e compilação de processos de negócio. É implementada sob a tecnologia *Open Source* Eclipse<sup>13</sup> numa arquitetura cliente-servidor. Permite também orquestrar a colaboração de diferentes intervenientes humanos, bem como interação entre sistemas.

O *Process Designer* tem uma componente gráfica poderosa que permite a interação de intervenientes de distintas competências durante a implementação de processos, tornando-se

---

<sup>12</sup> <https://www.jetbrains.com/idea/>

<sup>13</sup> <http://www.eclipse.org>



na ferramenta mais interdisciplinar em uso nesta equipa. A um nível mais alto, é possível permitir a intervenientes sem conhecimentos técnicos de programação modelar processos de negócio com a notação BPMN 2.0, bastando para tal mover os elementos da notação, previamente disponibilizados, para a tela através de *drag-and-drop*<sup>14</sup> e estabelecer as ligações.

A um nível mais baixo, o engenheiro de software recorre à mesma ferramenta para integrar com diferentes sistemas, transformar lógica de negócio em código e garantir que a aplicação reflete, na íntegra, o processo de negócio modelado. Permite também verificar o estado do processo em tempo real e realizar operações de *debug* em caso de erro durante a execução.

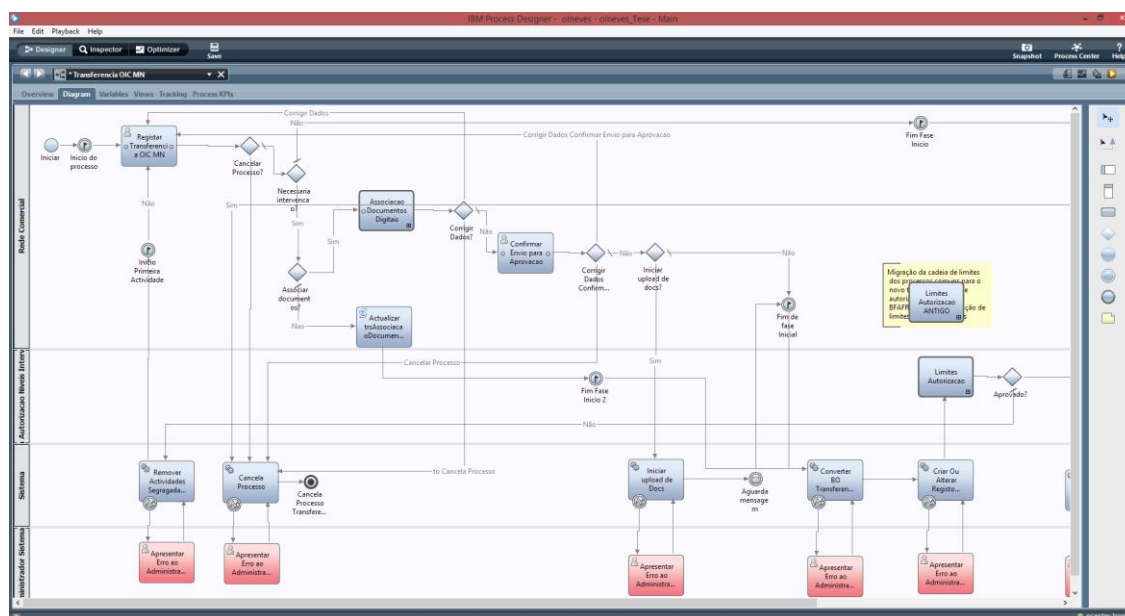


Figura 26 – Visão geral do IBM Process Designer

## SOAP UI<sup>15</sup>

Trata-se de uma ferramenta Open Source de teste para serviços SOAP e REST que fornece suporte a testes funcionais de serviços SOAP e REST. Esta é também uma ferramenta de grande relevância considerando a arquitetura orientada a serviços do projeto Bankee. Permite testar a implementação de novos serviços, e alterações realizadas a serviços existentes, imediatamente após o seu *deployment*, evitando a necessidade de percorrer o processo até ao ponto onde estes são usados, permitindo desta forma agilizar o desenvolvimento e aumentar a qualidade das APIs implementadas.

Esta ferramenta permite ainda realizar testes de carga. Criados habitualmente a partir de testes funcionais, é possível validar a performance de serviços sob diferentes cenários e manter validação funcional permitindo verificar se não há uma quebra de performance sob carga. É ainda possível realizar vários testes de carga em simultâneo, permitindo verificar de que forma

<sup>14</sup> Ação de clicar num objeto e arrastá-lo para posição diferente no ecrã

<sup>15</sup> <https://www.soapui.org/>

estes se influenciam entre si. Outras funções tais como testes à segurança das APIs, suporte à virtualização e simulação, testes baseados em dados e obtenção de métricas estatísticas para a medição quantitativa dos testes efetuados, são também suportadas por esta ferramenta.

### **DB Visualizer<sup>16</sup>**

Esta ferramenta é um gestor de bases de dados que suporta a grande maioria dos sistemas de base de dados disponíveis, tais como SQL Server, Oracle, SQLite, MySQL, DB2, entre outros. Uma solução transversal para a gestão de bases de dados, orientada a programadores, analistas e administradores de bases de dados.

Possui um conjunto de funcionalidades tais como a gestão de objetos específicos de base de dados, ações visuais para ações como “CREATE”, “ALTER”, “DROP”, “RENAME”, entre outras. Permite ainda a comparação de conjuntos de resultados e exportação de esquemas de base de dados.

Pela capacidade na criação de diferentes ligações em simultâneo e suporte a DB2<sup>17</sup>, o sistema de base de dados da IBM em utilização no projeto Bankee, foi usado no desenvolvimento do módulo em estudo para a gestão de dados tanto nos esquemas locais de base de dados, como na gestão e consulta de dados da BANKA para suporte ao desenvolvimento e gestão de APIs de integração com este sistema.

### **Bare Tail Pro<sup>18</sup>**

Trata-se de um software de monitorização de ficheiros de log em tempo real. Permite a visualização de vários ficheiros em simultâneo sem limite de tamanho, sendo possível navegar para qualquer ponto do ficheiro através de *scroll* instantâneo. Possui várias configurações, tais como o destacar de linhas em diferentes cores, tamanhos de letra, espaçamento e expansão de texto, por forma a facilitar a apresentação de texto e fragmentação do conteúdo do ficheiro permitindo uma melhor visualização de diferentes linhas com diferentes origens no LOG.

Suporta ficheiros com diferentes formatos (ficheiros de texto Windows / DOS, Unix, Microsoft IIS log files, entre outros) bem como diferentes codificações (Unicode<sup>19</sup>, UTF-8, ANSI e ASCII).

Com a opção “Follow Tail” é possível acompanhar em tempo real a evolução do ficheiro de log para um ou vários ficheiros em simultâneo. Esta opção replica a ação do comando “tail -f” usado nos sistemas Unix.

Possui ainda um poderoso sistema de pesquisa e filtragem que permite a utilização de texto livre com a opção de “Match Case” ou a utilização de REGEX<sup>20</sup>.

---

<sup>16</sup> <https://www.dbvis.com/>

<sup>17</sup> <https://www.ibm.com/analytics/us/en/db2/>

<sup>18</sup> <https://baremetalsoft.com/baretailpro/>

<sup>19</sup> <http://unicode.org/>

<sup>20</sup> <http://www.regular-expressions.info/>

### **JRebel<sup>21</sup>**

Esta é uma ferramenta que permite visualizar, em tempo real, alterações inseridas no código JAVA, tais como novas funcionalidades ou alterações à lógica implementada, sem necessidade de um novo *deploy* e reinicialização do servidor aplicacional, permitindo aumentar a produtividade do programador durante o desenvolvimento, eliminando uma fração significativa de tempo gasto no *deploy* do novo código a cada alteração efetuada [51].

É uma das ferramentas mais importantes no desenvolvimento do projeto Bankee, pois face à sua dimensão, o tempo de compilação e instalação de cada módulo poderá variar em média entre os 15 e os 30 minutos, o JRebel tem a capacidade de eliminar grande parte deste tempo durante o desenvolvimento, tornando este processo muito mais ágil.

### **XRebel<sup>22</sup>**

Dos mesmos criadores do JRebel, o XRebel é uma ferramenta de análise de performance em tempo real. É usada pela equipa de programadores durante o processo de desenvolvimento da aplicação Bankee com o intuito de identificar, em tempo real, situações comuns que podem levantar problemas de performance na aplicação.

É uma ferramenta desenhada exclusivamente para o *developer*, permite acompanhar o tempo gasto em cada pedido ao servidor, faz a monitorização de mensagens e trabalhos em segundo plano, possibilita a monitorização da atividade da base de dados identificando métodos com consultas pesadas, identifica exceções escondidas e permite encontrar informação no log, emitida em tempo real, para um pedido concreto sem necessidade de pesquisa.

### **Tortoise SVN<sup>23</sup>**

Trata-se de um cliente Apache Subversion (SVN) <sup>24</sup> implementado como uma extensão *Shell* de Windows. Possui uma interface de utilizador intuitivo e de fácil utilização evitando a necessidade de utilização de um cliente em linha de comandos para executar as operações de SVN necessárias para garantir o controlo de versionamento.

## **6.2. Metodologia**

O Projeto Bankee, e consequentemente o módulo de transferências interbancárias desenvolvido no âmbito deste estudo, seguem a metodologia de desenvolvimento de software em *Waterfall*<sup>25</sup>.

---

<sup>21</sup> <https://zeroturnaround.com/software/jrebel/>

<sup>22</sup> <https://zeroturnaround.com/software/xrebel/>

<sup>23</sup> <https://tortoisesvn.net/>

<sup>24</sup> <http://subversion.apache.org/>

<sup>25</sup> <http://www.umsl.edu/~hugheyd/is6840/waterfall.html>

A adoção do modelo *Waterfall* para a implementação do projeto Bankee está relacionada com diferentes fatores internos e externos, mas justificada no âmbito da gestão de projeto com o acordado entre a Critical Software e o cliente.

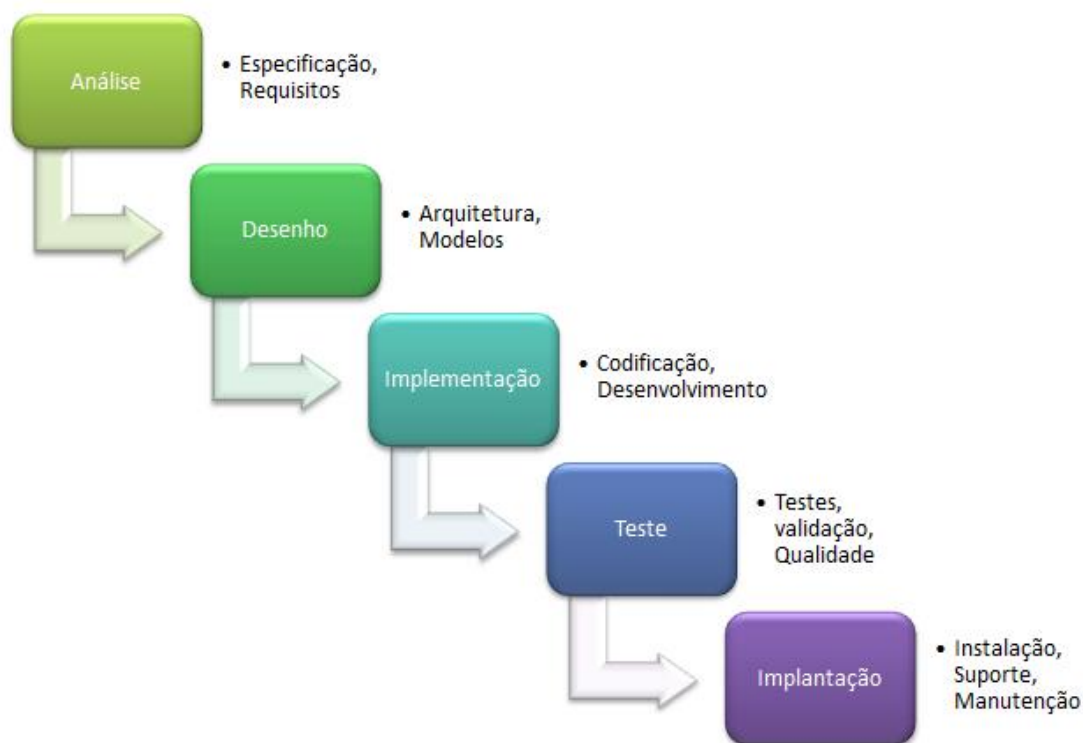


Figura 27 – Fases do desenvolvimento da metodologia *Waterfall*

Fonte: [52]

Cada uma das fases de desenvolvimento de cada módulo do projeto Bankee, tal como aconteceu com o módulo de transferências interbancárias, parte de um documento de requisitos finalizado e aprovado pelo cliente, e cada uma das fases seguintes só inicia após o término da fase anterior, de forma sequencial tal como mostra a Figura 27.

O desenvolvimento do módulo de transferências interbancárias começou na fase de análise onde a equipa funcional, em conjunto com o cliente, definiu todo o âmbito do módulo a desenvolver num documento de requisitos. Este documento, após aprovação final pelo cliente, serve de base ao trabalho desenvolvido na fase seguinte, desenho.

Durante a fase de desenho é modelado o processo de negócio em BPMN 2.0 que servirá de base para o processo a implementar na fase seguinte. Foi ainda durante esta fase, que no início do projeto se definiu a arquitetura geral do projeto. No desenvolvimento de módulos autocontidos, com arquiteturas próprias, é também durante esta fase que se procede ao desenho das mesmas.

A fase de implementação partiu do legado transmitido da fase anterior e procedeu-se à codificação do módulo. Durante este período foram desenvolvidos os ecrãs de UI da aplicação, serviços, implementação de lógica de negócio, mapeamento de objetos BPM e afinações ao modelo inicial do processo de negócio com o intuito de implementar, na íntegra, o que se encontra descrito do documento de requisitos que resulta da primeira fase deste modelo.

Depois de fechado o âmbito da implementação, o módulo passou para a equipa de testes e controlo de qualidade, com o intuito de passarem por uma bateria de testes funcionais e de validação. O *developer* teve ainda um papel ativo durante esta fase, pois todos os defeitos que surgiram da validação foram resolvidos por este.

No final da cascata, compreende-se que o módulo está pronto para ser instalado nos ambientes do cliente. No caso concreto do projeto Bankee, no momento em que o módulo é entregue ao cliente, esta fase desdobra-se em três momentos distintos. O módulo é instalado numa fase inicial no ambiente de qualidade interno do cliente, aqui, uma equipa de *testers* corre novos testes de validação. De seguida, o módulo será instalado num ambiente de pré-produção, habitualmente numa agência piloto para, por último, ser expandido para produção e instalado em todas as dependências bancárias. Neste momento, a equipa responsável, dentro da Critical Software, pela gestão do módulo é a manutenção.

Até à data de entrega deste documento, o módulo desenvolvido encontra-se no primeiro momento desta fase, sendo expectável que passe por todos os restantes e entre em produção tão breve quanto possível.

Importa ainda referir que o autor deste estudo esteve somente envolvido na fase de implementação e nas iterações necessárias ao *bug fixing* durante a fase de testes.

## 6.3. Produto Final

No presente subcapítulo pretende-se apresentar o resultado do trabalho de implementação e desenvolvimento dos diferentes componentes que compõem o módulo de transferências interbancárias. Serão, de seguida, apresentadas as modelações finais do processo em BPMN 2.0 com especial ênfase na solução do problema previamente descrito.

Posteriormente, serão apresentados e descritos os ecrãs de UI (*user interface*) desenvolvidos, bem como *webservices* e APIs de integração com a “Banka”.

### 6.3.1. Alternativas

No contexto do desenvolvimento do produto final foram sendo ponderadas alternativas essencialmente no que compete à integração do módulo de limites de autorização descrito na secção 5.2.

A alternativa à integração do módulo de limites de autorização seria organizar cada uma das atividades humanas num único diagrama de colaboração, em diferentes faixas, sendo a interação entre estas seria definida por regras de negócio provenientes da configuração inserida nos limites de autorização. Esta abordagem que se encontra apresentada no diagrama

preliminar de estudo da Figura 11 e com maior detalhe na Figura 28, aqui é possível observar as diferentes atividades humanas de intervenção hierárquica distribuídas pelas faixas dos tipos de intervenção hierárquicos descritos na secção Limites de Autorização 5.2, ambas assinaladas a contorno vermelho.

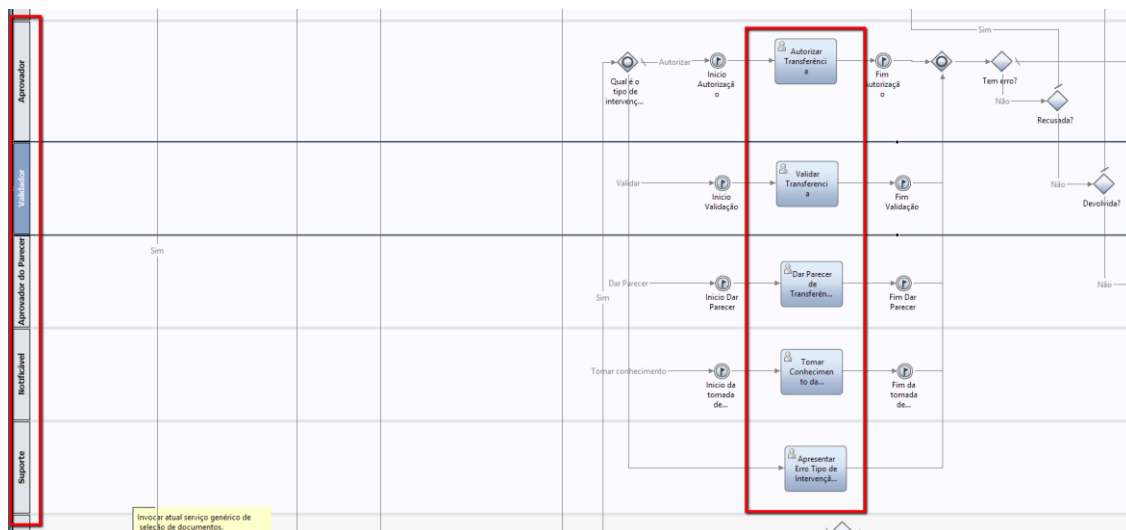


Figura 28 – Detalhe diagrama de colaboração limites de autorização

Esta alternativa, apesar de aumentar a complexidade do modelo do processo de negócio seria viável para a resolução do problema descrito na secção 1.2, no entanto seria uma solução atômica não passível de reutilização. A análise funcional ao módulo previa que módulos futuros pudessem necessitar de implementar esta funcionalidade, facto que mais tarde veio a ser confirmado pelo cliente, motivando assim o abandono a esta abordagem, tendo vindo a ser substituída pela criação de um subprocesso.

### 6.3.2. Padrões Adotados

Com o intuito de formalizar recorrências durante a modelação de processos em diferentes sistemas baseados em fluxo funcional de processo, surgiu em 1999 a *Workflow Patterns Initiative*, resultante de um intenso e contínuo trabalho de pesquisa, levantamento e documentação de padrões na modelação de processos. É auto-descrita pelos seus autores como “o delinear de requisito fundamentais que surgem durante a modelação recorrente de processos de negócio, descrevendo-os de uma forma imperativa” [65].

Foram inicialmente descritos 20 padrões de controlo de fluxo através de formalismos na modelação de processos de negócio, mais tarde em 2006 é realizada uma revisão a estes padrões com descrições formais e são identificados 23 novos padrões complementando assim o cardápio disponível no website oficial desta iniciativa [66].

Para a modelação do processo de negócio de transferências interbancárias foram adotados alguns dos padrões aqui descritos, e serve a presente secção o objetivo de os identificar no modelo disponibilizado na secção 6.3.3.

Os padrões adotados para este trabalho foram:

- Padrão 1 - *Sequence Pattern*;
- Padrão 16 – *Deferred Choice*;
- Padrão 21 – *Structured Loops*;
- Padrão 43 - *Explicit Termination*;

### Padrão 1 – *Sequence Pattern*

É possível identificar o padrão de sequência em diversos pontos do modelo de processo de negócio de transferências interbancárias desenvolvido.

A Figura 29 mostra um dos diferentes exemplos de aplicação do padrão de sequência, onde as tarefas seguintes dependem sempre da finalização das anteriores para que possam ser inicializadas. Nesta imagem está identificada numericamente, e de forma crescente, a sequência deste fluxo.

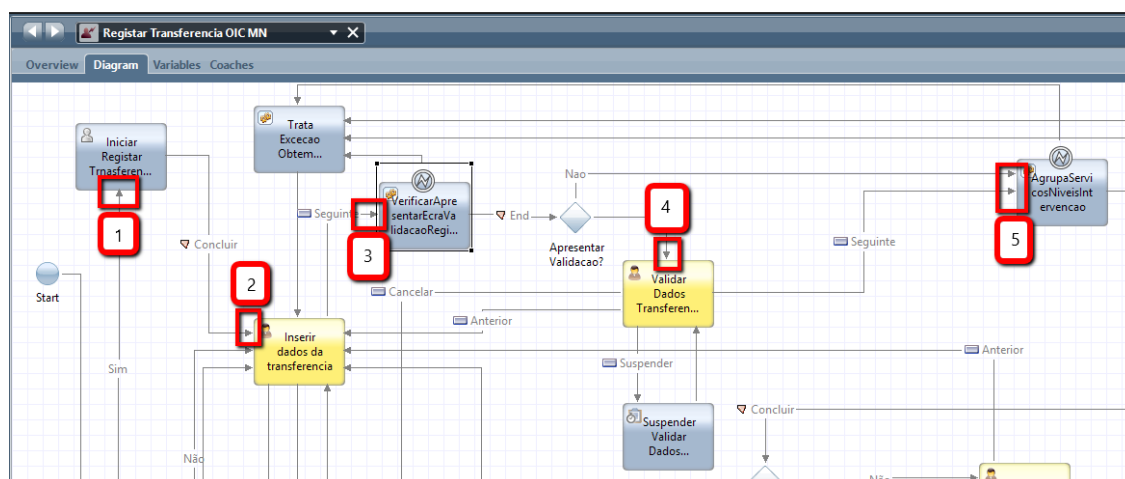


Figura 29 – *Sequence Pattern* representada no modelo

### Padrão 16 - *Deferred Choice*

O padrão de escolha deferida refere-se ao fluxo que disponibiliza vários ramos com possibilidade de escolha de um destes a partir de um *gateway XOR*<sup>26</sup> exclusivo que interage, através de dados, com o estado do processo para determinar qual dos diferentes caminhos do fluxo este deve tomar.

Pode ser observada a aplicação deste padrão no sub-processo da cadeia de autorização e de níveis de intervenção apresentado na Figura 30. Aqui é possível visualizar o *gateway XOR* exclusivo que avalia as condições do ambiente para reencaminhar o fluxo para uma das atividades de autoriza, validar, dar parecer, tomar conhecimento ou apresentar erro tipo de intervenção.

<sup>26</sup> <https://www.lucidchart.com/pages/bpmn/gateways>

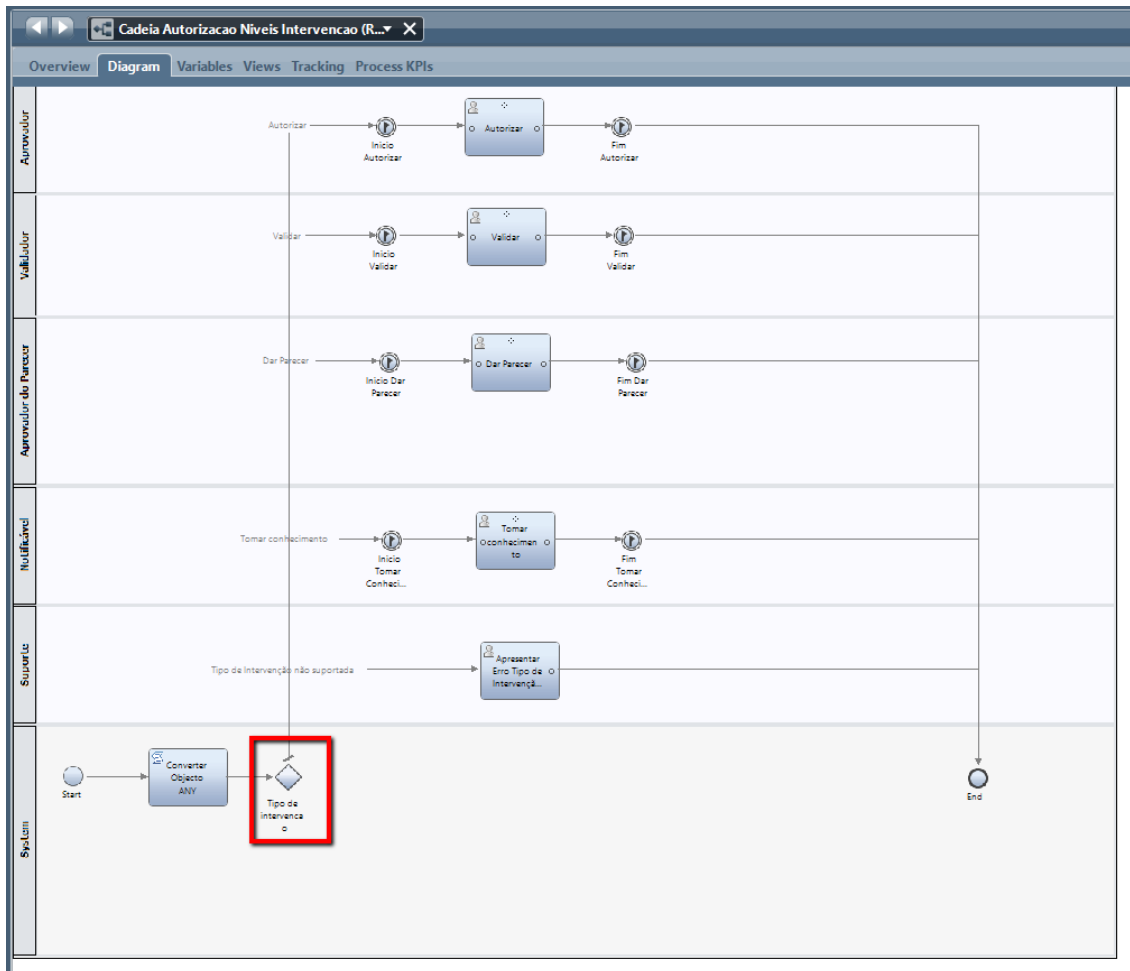


Figura 30 – Exemplo do padrão de *Deferred Choice*.

### Padrão 21 – Structured Loop

Este padrão representa a estrutura de um *loop* condicional, onde uma atividade ou subprocesso tem a capacidade de executar continuamente enquanto a pré ou pós condição de execução continuar assim o determinar.

No modelo do processo de transferências interbancárias, mais concretamente no sub-processo “Limites de Autorizacao” foi implementado um loop estruturado, onde as pós-condições assinaladas com o número 1 na Figura 31 se existem, ou não, mais níveis de intervenção a executar. Havendo mais níveis a executar o fluxo é encaminhado novamente para a atividade de sistema “Inicializa Transaccão Limites Autorizacao” seguindo novamente para o sub-processo de níveis de intervenção – “Cadeia Autorizacao Niveis Intervencao” marcado com o número 2 na Figura 31. Enquanto existirem mais níveis de intervenção a processar este ciclo repetir-se-á num loop estruturado.



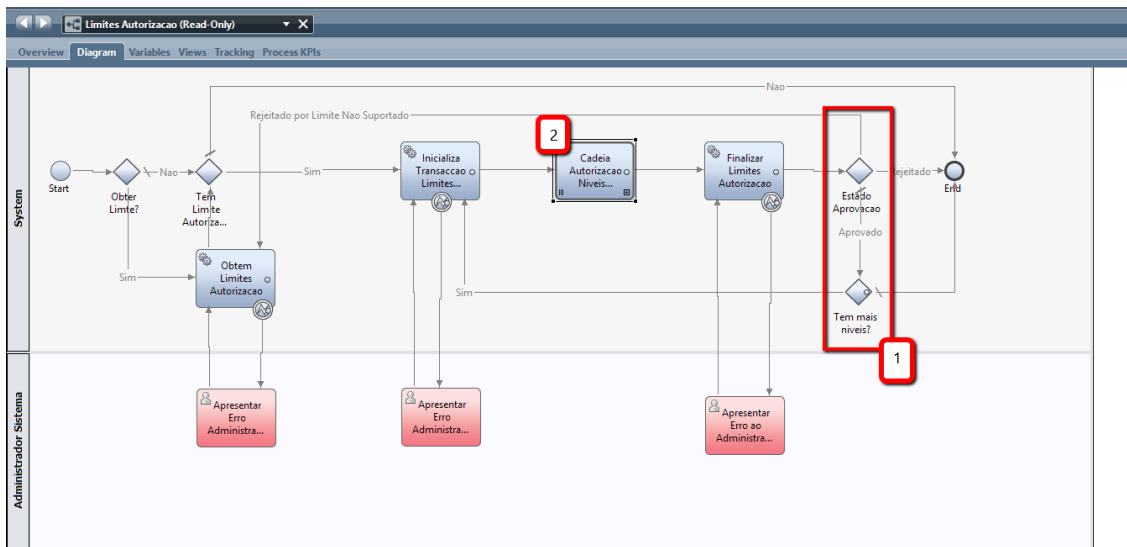


Figura 31 – *Structured loop pattern* no sub-processo de limites de autorização

### Padrão 43 - Explicit Termination

O padrão de término explícito representa a finalização com sucesso de um processo ou sub-processo após este atingir um determinado estado, normalmente representado por um evento finalizador. Atingindo um estado que o marca como terminado, mesmo que ainda existam atividades por processar estas serão canceladas e o processo terminado com sucesso.

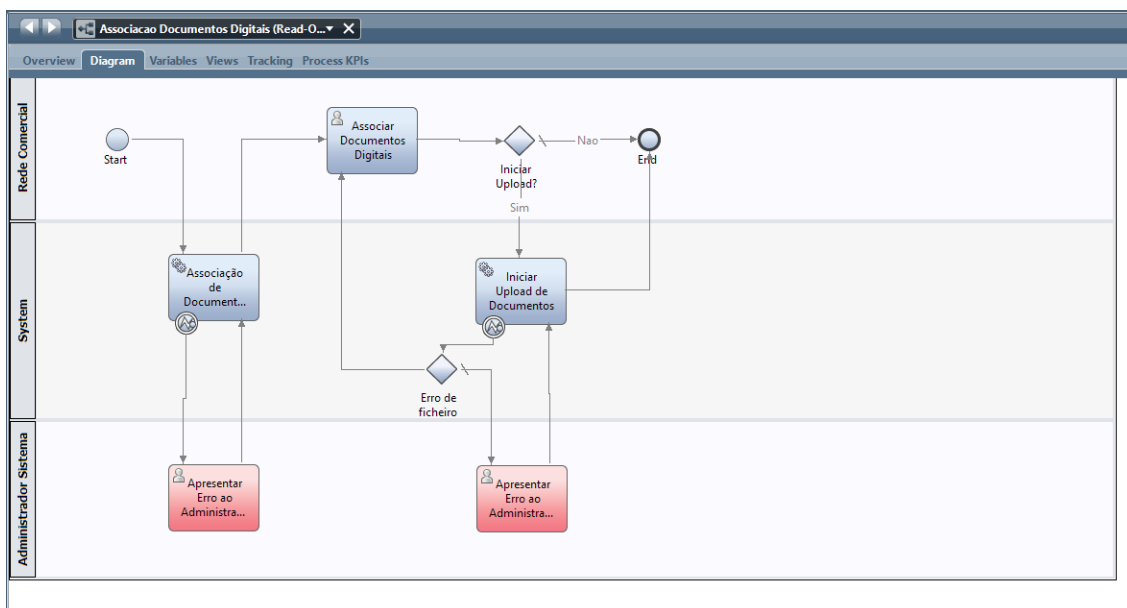


Figura 32 - *Implicit Termination pattern* no sub-processo de associação de documentos

Este padrão está representado em 4 momentos do processo de transferências interbancárias. No subprocesso de associação de documento representado na Figura 32 – “Associacao

Documentos Digitais”, nos sub-processos de limites de autorização na Figura 31 e de níveis de intervenção na Figura 30. Também o processo geral de transferências interbancárias termina somente de forma explícita.

### **6.3.3. Modelo Final do Processo**

Partindo dos modelos de processo inicialmente desenvolvidos no âmbito da fase de desenho na metodologia de desenvolvimento waterfall e previamente descritos no capítulo 5, foram incluídas durante a fase de implementação, naturalmente, alterações que visam a eficiência do processo de negócio e que se coadunam com a lógica necessária à implementação de funcionalidade levantada em caderno de requisitos.

Será assim demonstrado, nesta secção, o resultado final da modelação do processo de negócio, com a representação gráfica de diferentes atividades e subprocessos que deste fazem parte, havendo ainda lugar à descrição técnica das opções tomadas e em que medida contribuíram para a resolução do problema.

Na secção 6.3.3.1 é dedicada à exposição do modelo do processo de negócio de transferências interbancárias com intervenção hierárquica dinâmica. Nesta secção é possível ter uma visão geral do modelo do processo de negócio desenvolvido, bem como dos seus sub-processos e atividades, através da representação gráfica do mesmo.

A secção 6.3.3.2 é dedicada à análise do fragmento do modelo do processo de negócio de transferências interbancárias onde é representada a funcionalidade de intervenção hierárquica dinâmica, nomeadamente os sub-processos de limites de autorização e níveis de intervenção.

Durante a secção 6.3.3.3 é feita uma análise detalhada e descritiva à modelação do processo de transferências interbancárias, fazendo a ponte entre o modelo planeado e o implementado justificando as alterações introduzidas no modelo final.

### 6.3.3.1. Processo de Transferências Interbancárias

No presente subcapítulo são apresentados os diagramas finais obtidos da modelação do processo de transferências interbancárias. Podem ser observadas diferentes atividades e componentes do processo, modelados em BPMN 2.0. A Figura 33 mostra o diagrama final do processo de transferências interbancárias que cumpre com o primeiro objetivo descrito na secção 1.3, este diagrama será abordado com maior detalhe na secção

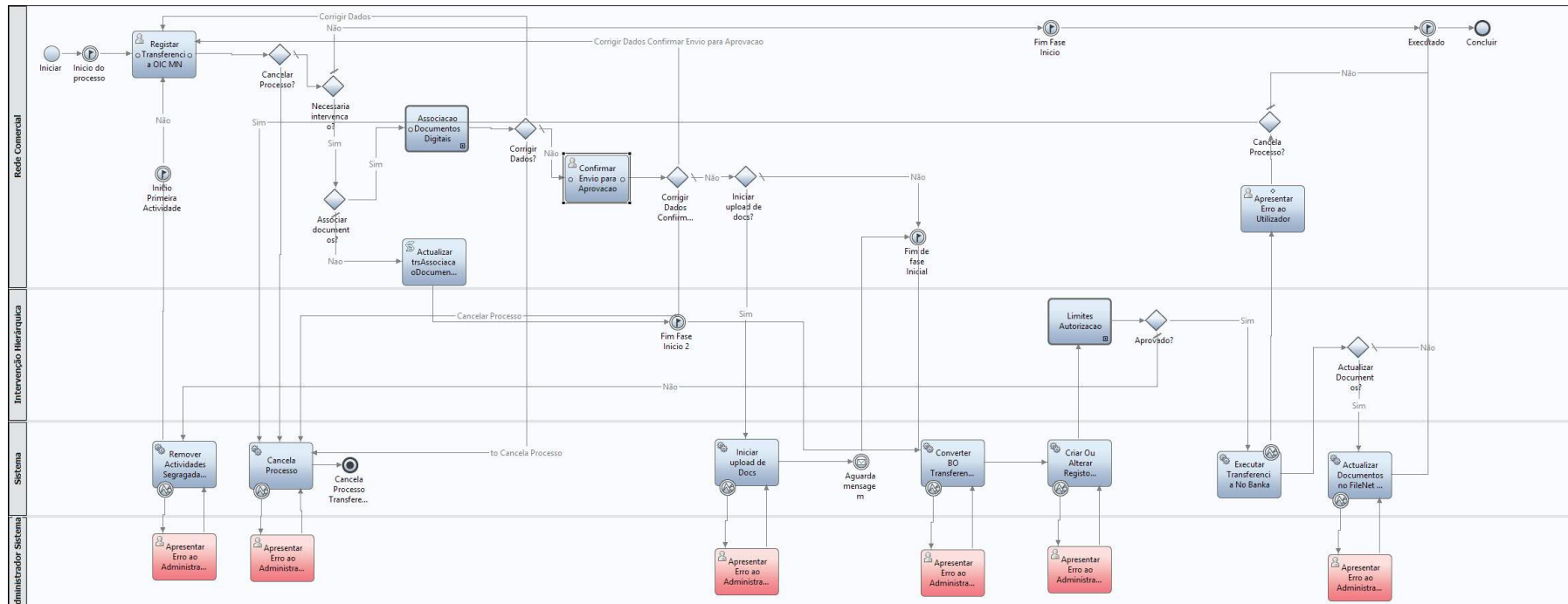


Figura 33 – Diagrama final do processo de transferências interbancárias

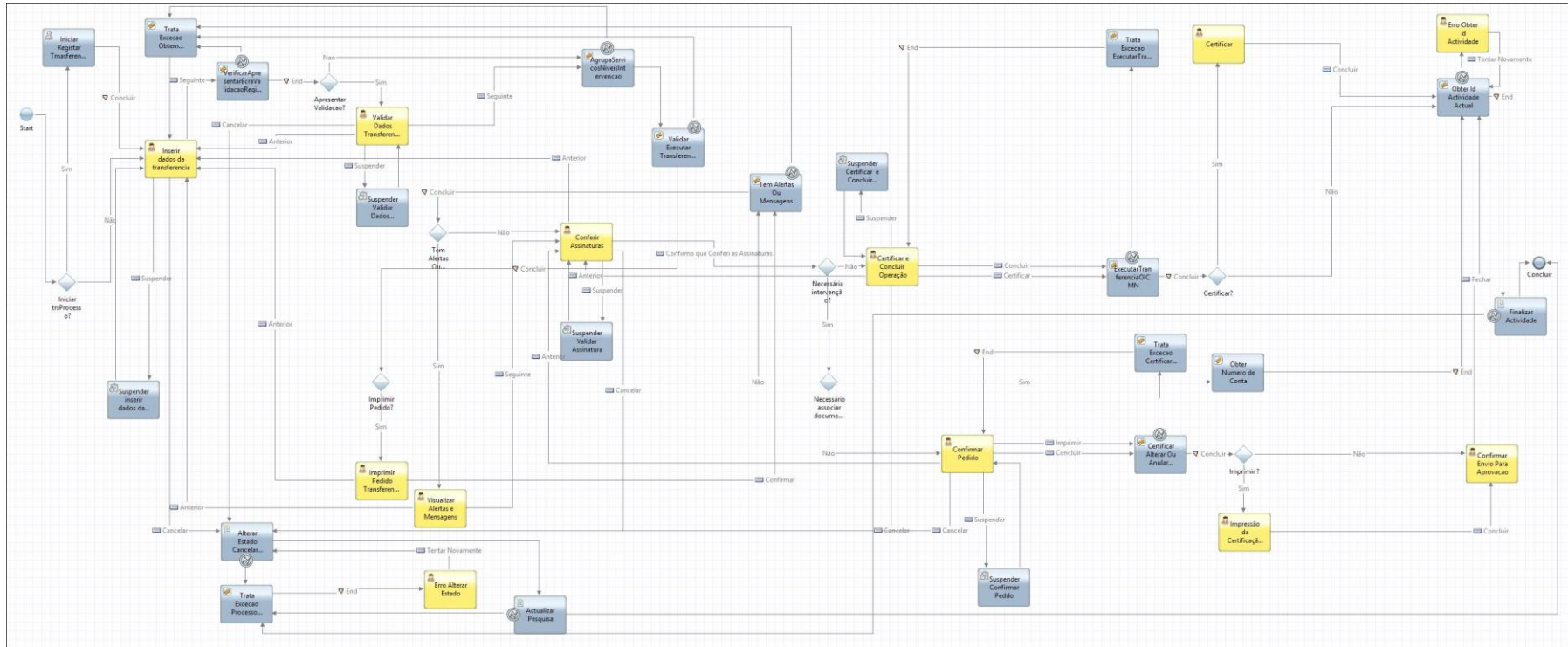


Figura 34 – Diagrama da primeira atividade, “Registrar Transferência”

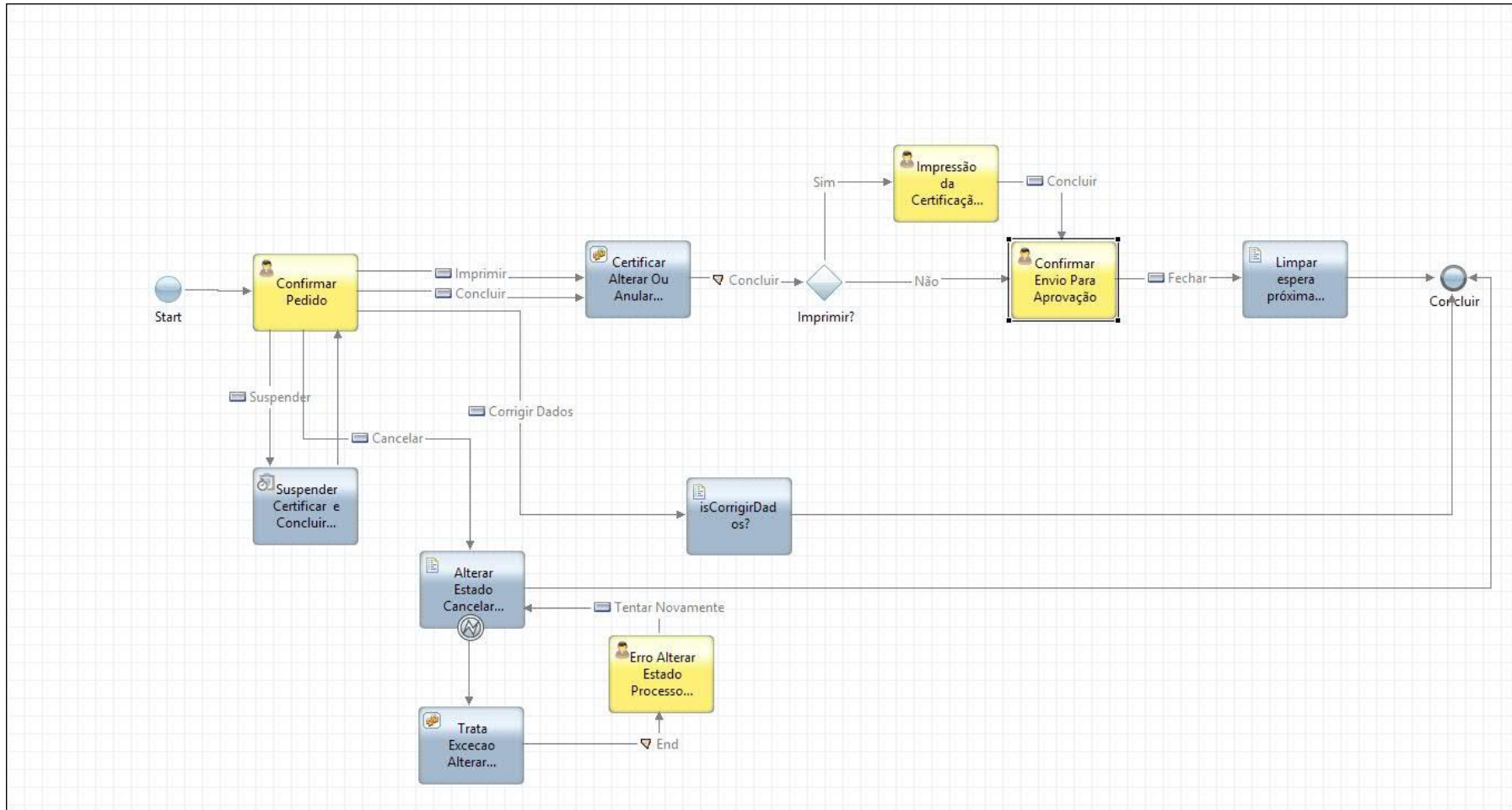


Figura 35 – Diagrama da atividade “Confirmar Envio para Aprovação”

### 6.3.3.2. Intervenção Hierárquica Dinâmica

Durante este capítulo será possível observar com algum nível de detalhe a abordagem utilizada na introdução do módulo de limites de autorização no processo de transferências de forma a colmatar o problema, previamente descrito, de falta de intervenção hierárquica dinâmica, durante o processo de transferências interbancárias em moeda nacional. Na

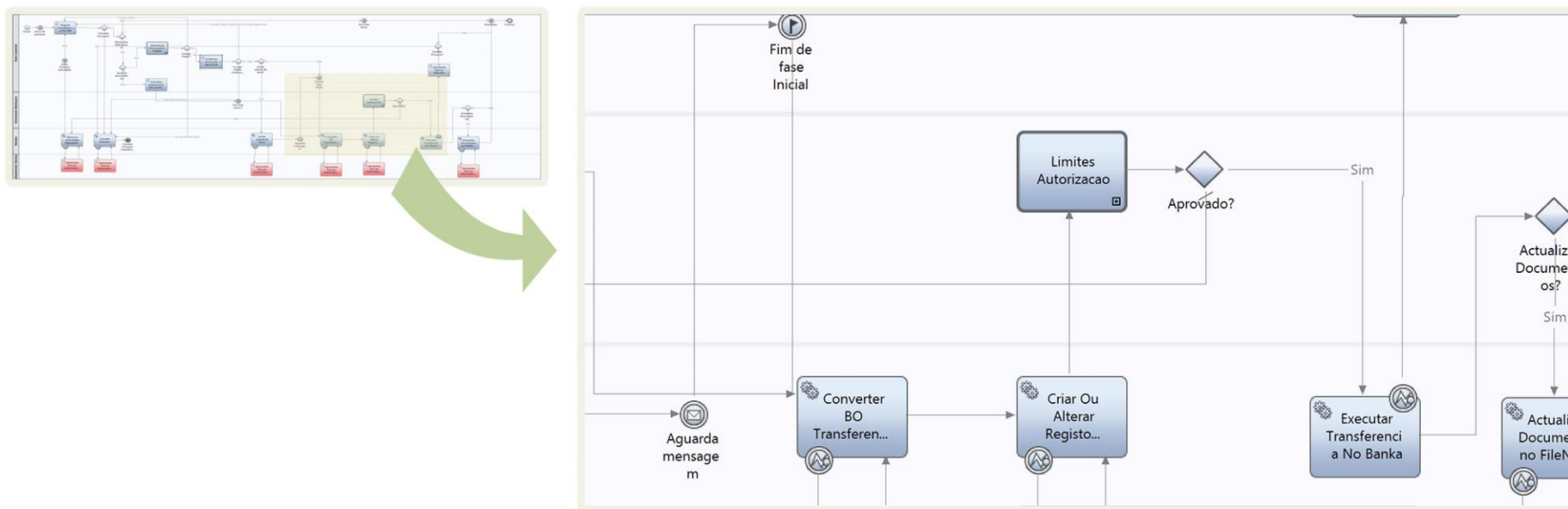


Figura 36 é possível observar, com algum detalhe, o subprocesso de Limites de Autorização incluído no diagrama geral do processo.

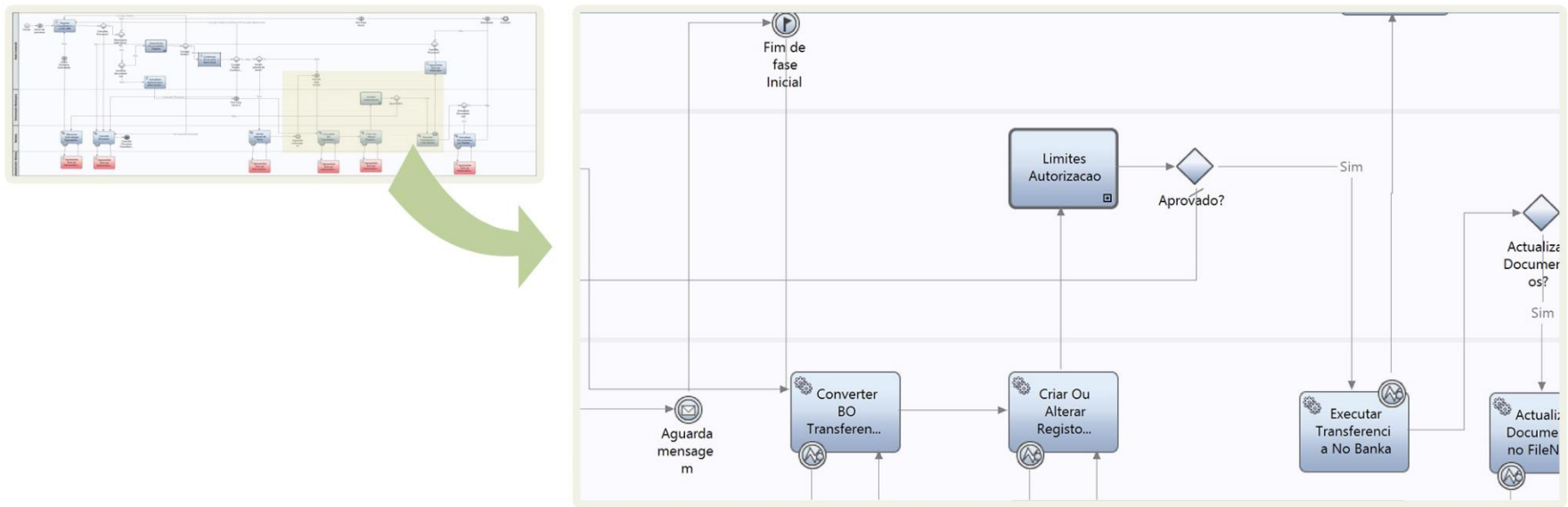


Figura 36 – Enquadramento do subprocesso de limites de autorização no processo geral

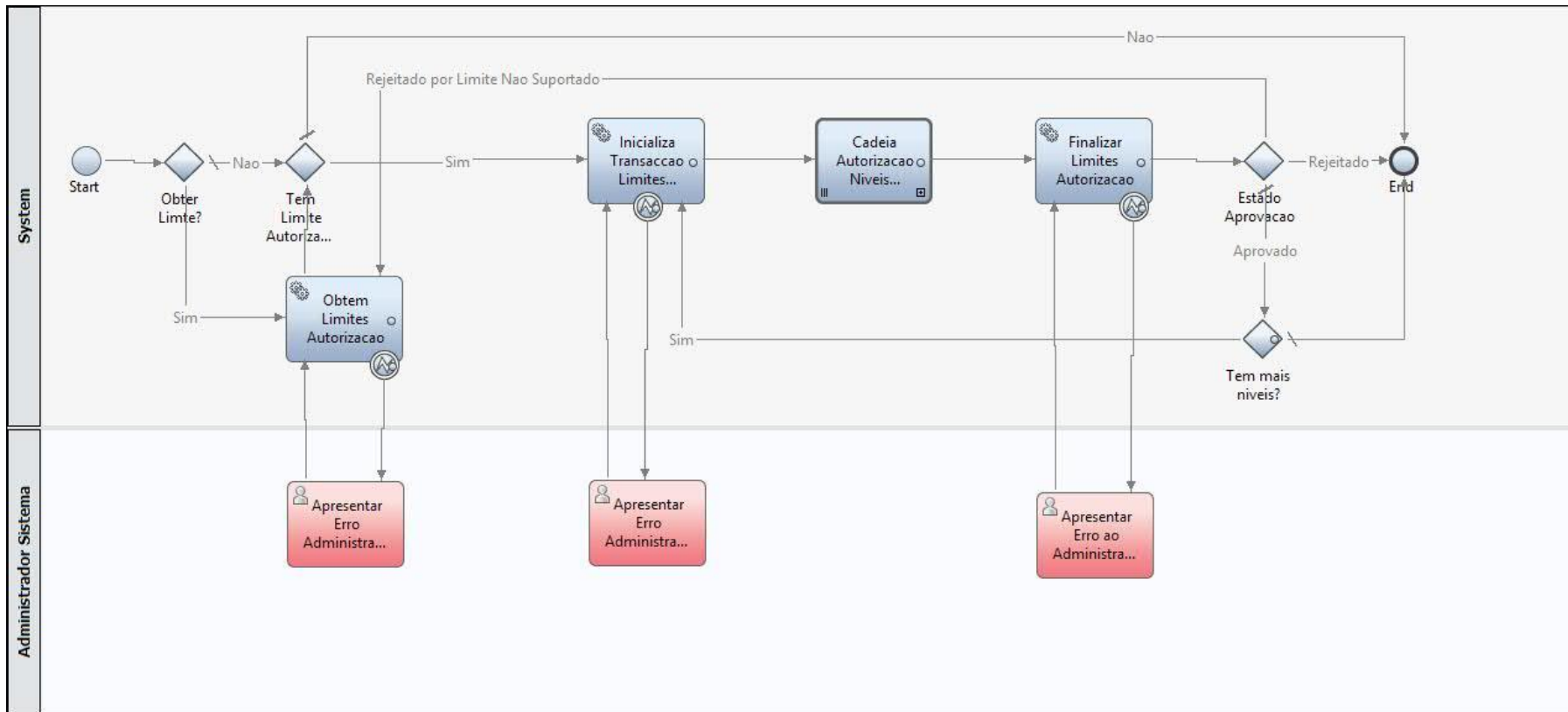


Figura 37 – Diagrama geral do subprocesso de limites de autorização



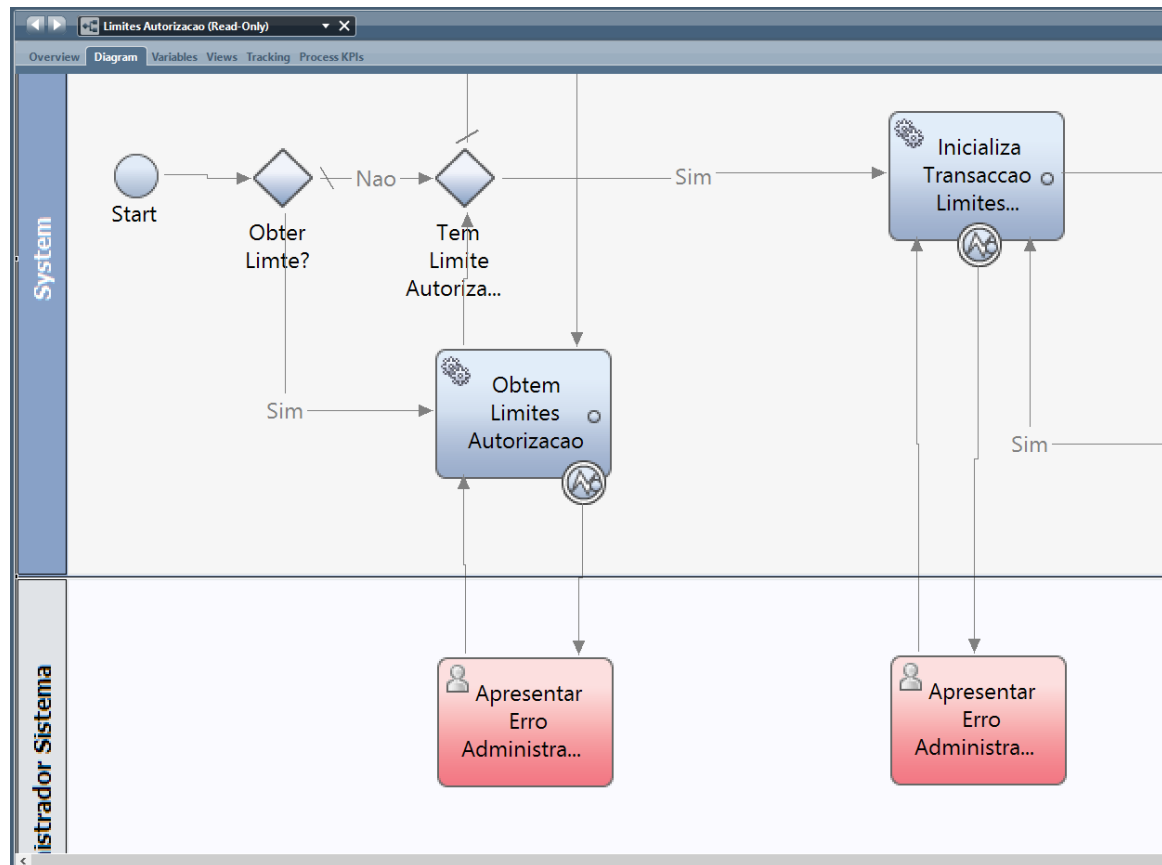


Figura 38 – Detalhe do subprocesso de limites de autorização

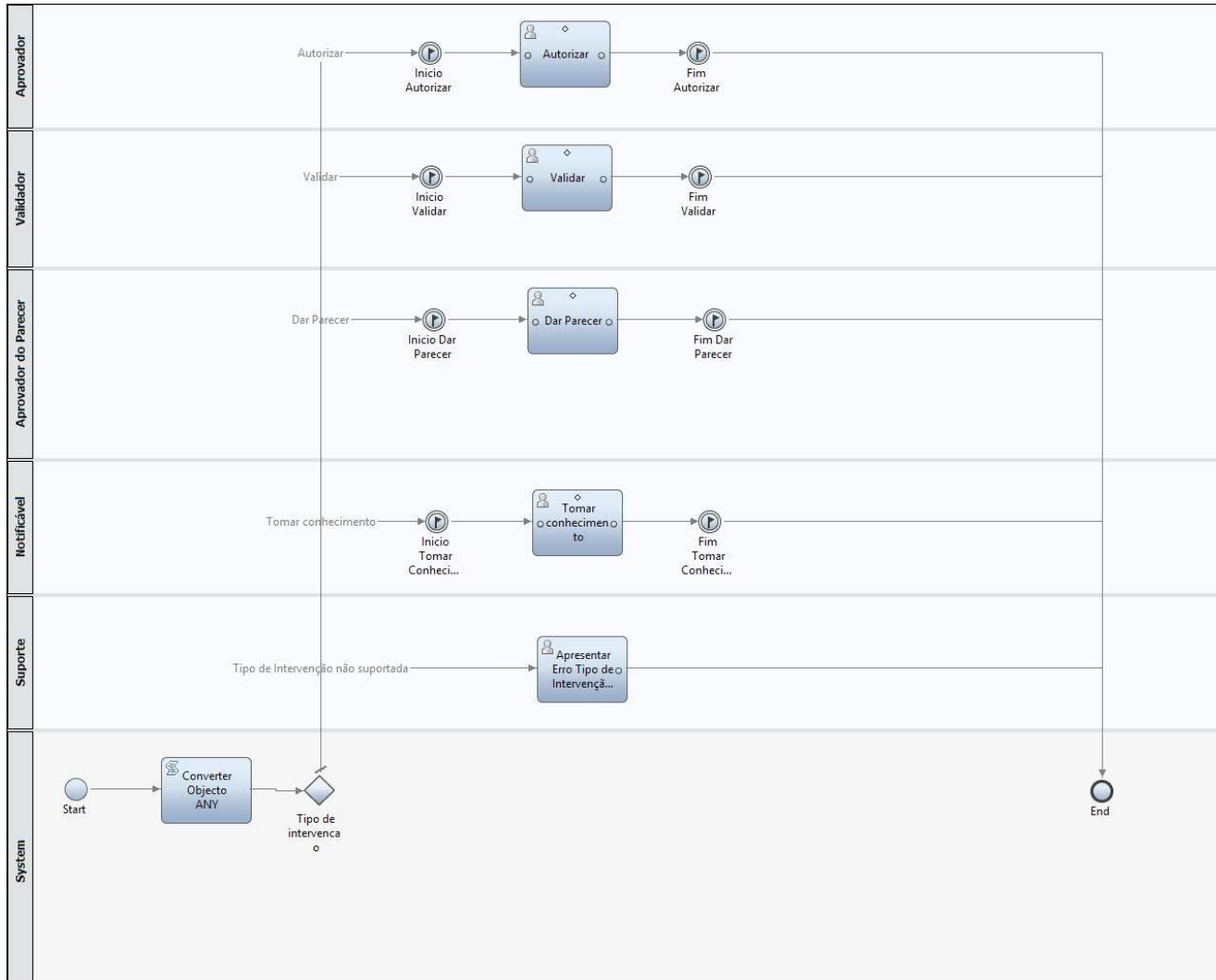


Figura 39 – Cadeia de níveis de intervenção

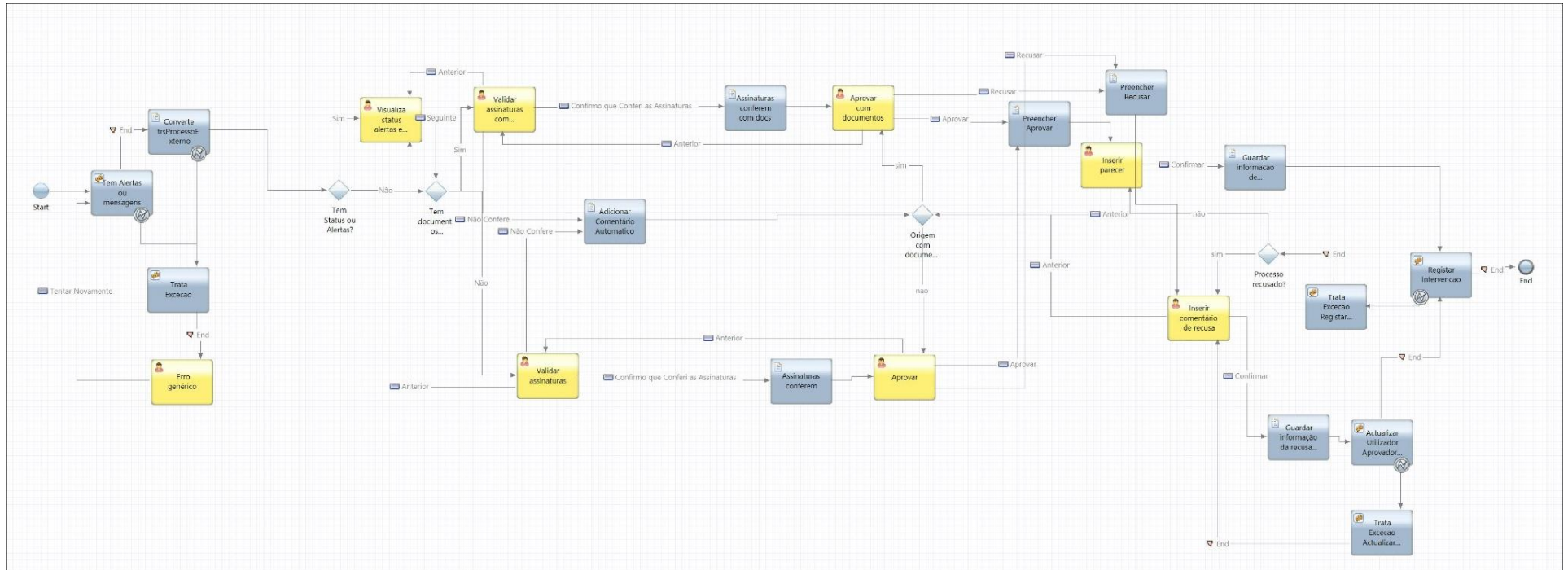


Figura 40 -Diagrama geral da atividade “Autorizar”

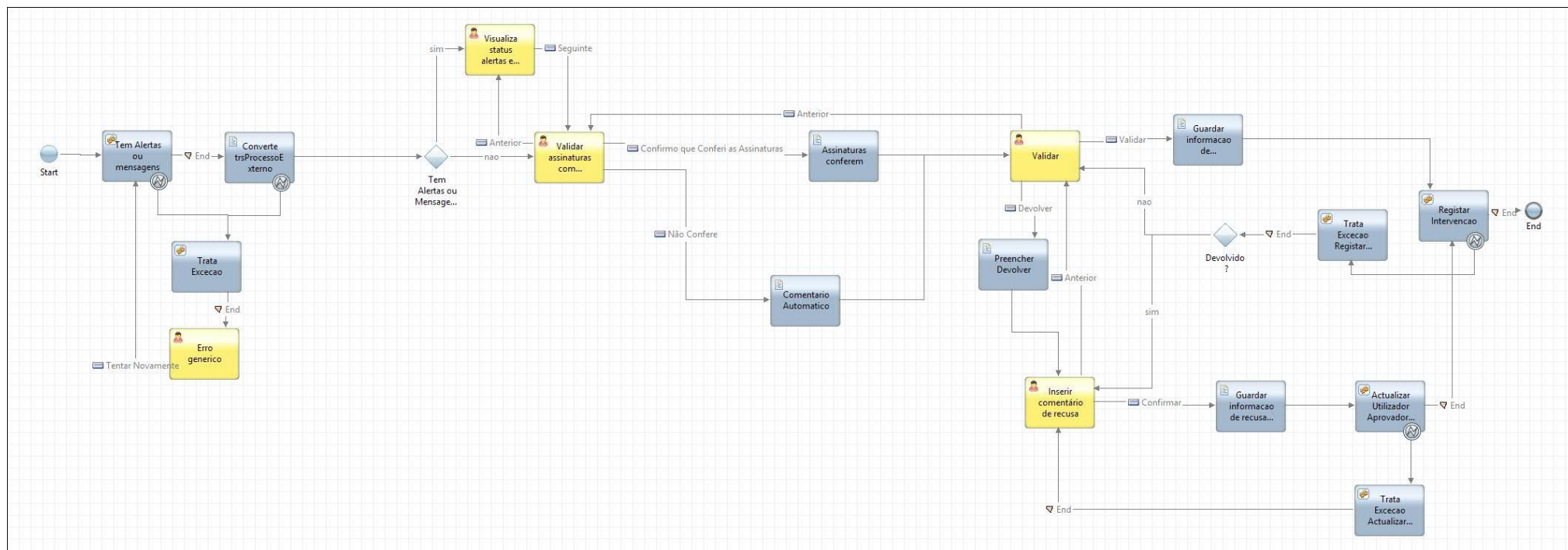


Figura 41 – Diagrama geral da atividade “Validar”

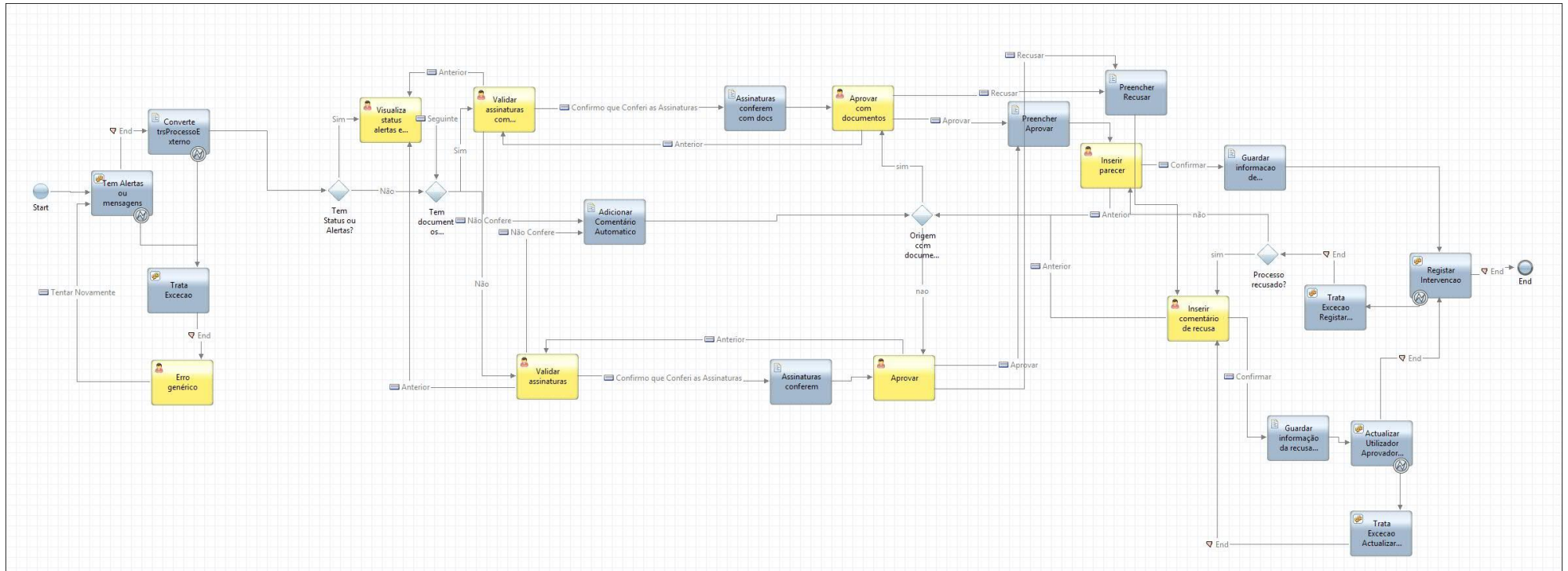


Figura 42 – Diagrama geral da atividade “Dar Parecer”

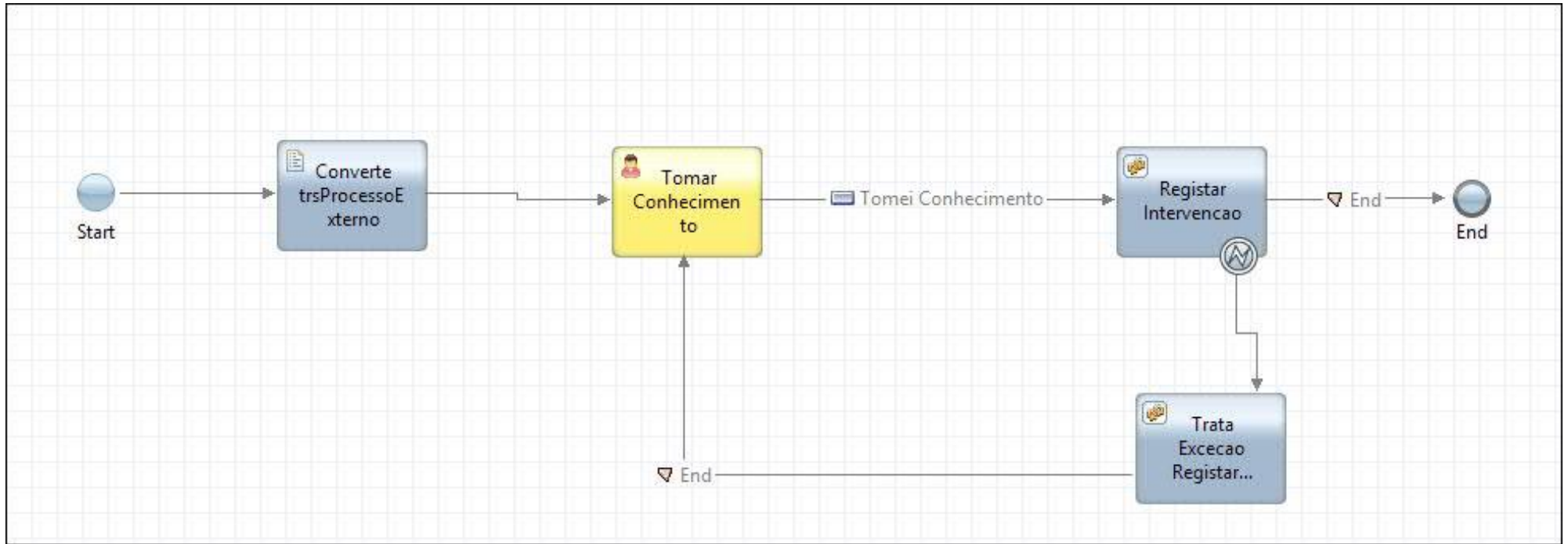


Figura 43 – Diagrama da atividade “Tomar Conhecimento”

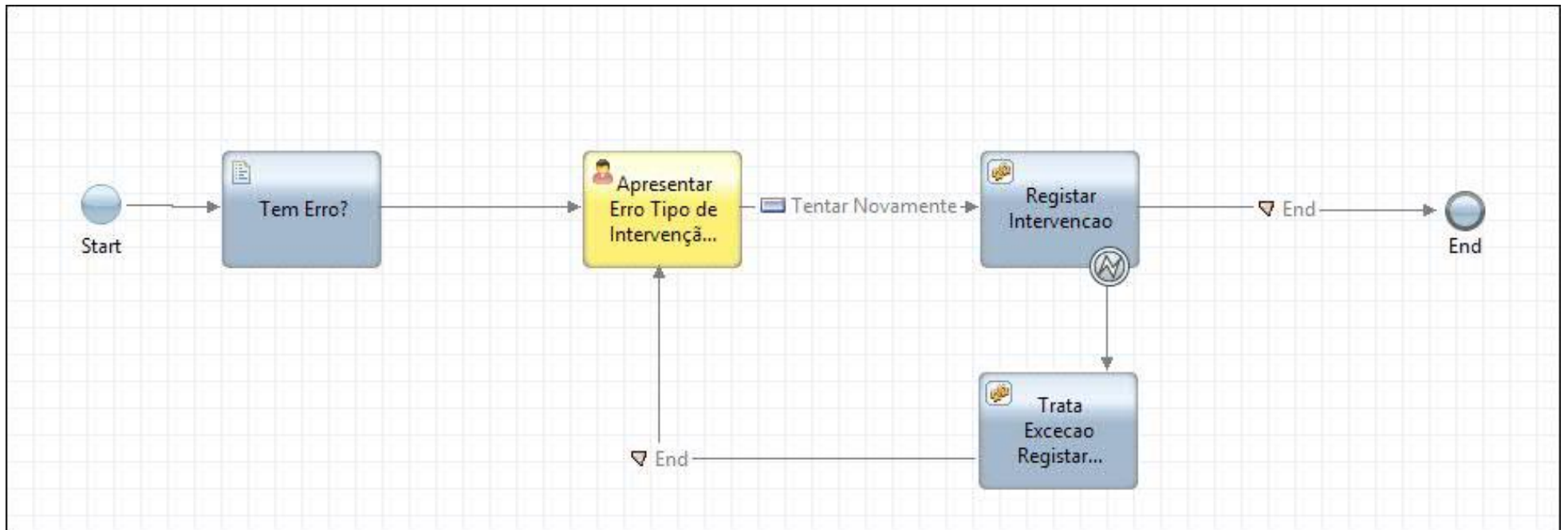


Figura 44 – Diagrama da atividade de “Erro Intervenção Não Suportada

### 6.3.3.3. Análise da Modelação de Processo

Durante os subcapítulos anteriores é possível verificar alguma diferenças face aos modelos iniciais. Este facto está, não só, relacionado com a funcionalidade e lógica de implementação, mas também pelo motivo de os modelos iniciais não incluírem, habitualmente, elementos do processo relacionados com detalhes do desenvolvimento, mas somente de fluxo de negócio.

#### Atividade “Registar Transferência”

No que compete ao modelo geral do processo de transferências interbancárias, pode-se observar a evolução na primeira atividade, “Registar Transferência”, entre a Figura 12 e a Figura 34.

No fluxo representado na Figura 34, imediatamente após o primeiro ecrã é o possível observar a inclusão de uma atividade de “*General System Service*”<sup>27</sup> que, tal como é possível verificar na Figura 45, agrupa um conjunto de outras atividades semelhantes, scripts e um “*Integration Service*”<sup>26</sup>. Este, tem a função de obter a variável booleana que indica ao decisor seguinte a necessidade, ou não, de apresentação do ecrã de validação dos dados de transferência.

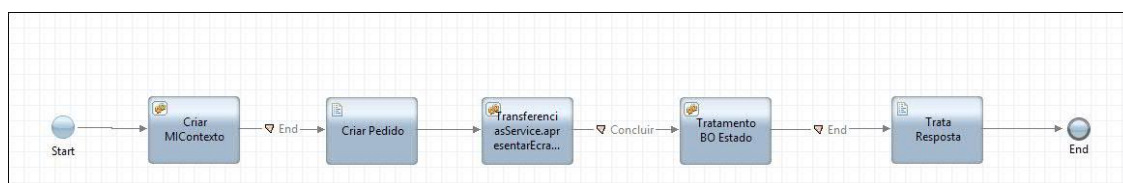


Figura 45 – Exemplo de implementação de um *General System Service*

Esta abordagem, visa a substituição da atividade de script que é possível encontrar na mesma posição na Figura 12. Torna-se assim um exemplo claro de alterações ao modelo do processo de negócio orientadas ao desenvolvimento e integração do mesmo com a aplicação Bankee que, não descurando o fluxo definido pelo negócio, se adapta aos objetivos funcionais da aplicação.

É facilmente identificável a aumentada complexidade descrita no diagrama da Figura 34, face ao previamente demonstrado na Figura 12, mas da mesma forma que por vezes, para incluir lógica e funcionalidade se torna necessário aumentar o número de elementos do diagrama, também é possível fazer precisamente o contrário, tal como podemos verificar na Figura 46. Esta, é um excerto pormenorizado do diagrama da Figura 12, e representa a lógica de negócio necessária para tratar a transferência, sendo categorizada pelo valor envolvido, em sistema STC ou SPTR<sup>28</sup>.

<sup>27</sup>[https://www.ibm.com/support/knowledgecenter/en/SSV2LR/com.ibm.wbpm.wle.editor.doc/topics/understanding\\_service\\_types.html](https://www.ibm.com/support/knowledgecenter/en/SSV2LR/com.ibm.wbpm.wle.editor.doc/topics/understanding_service_types.html)

<sup>28</sup> Ver capítulo 2.1.3



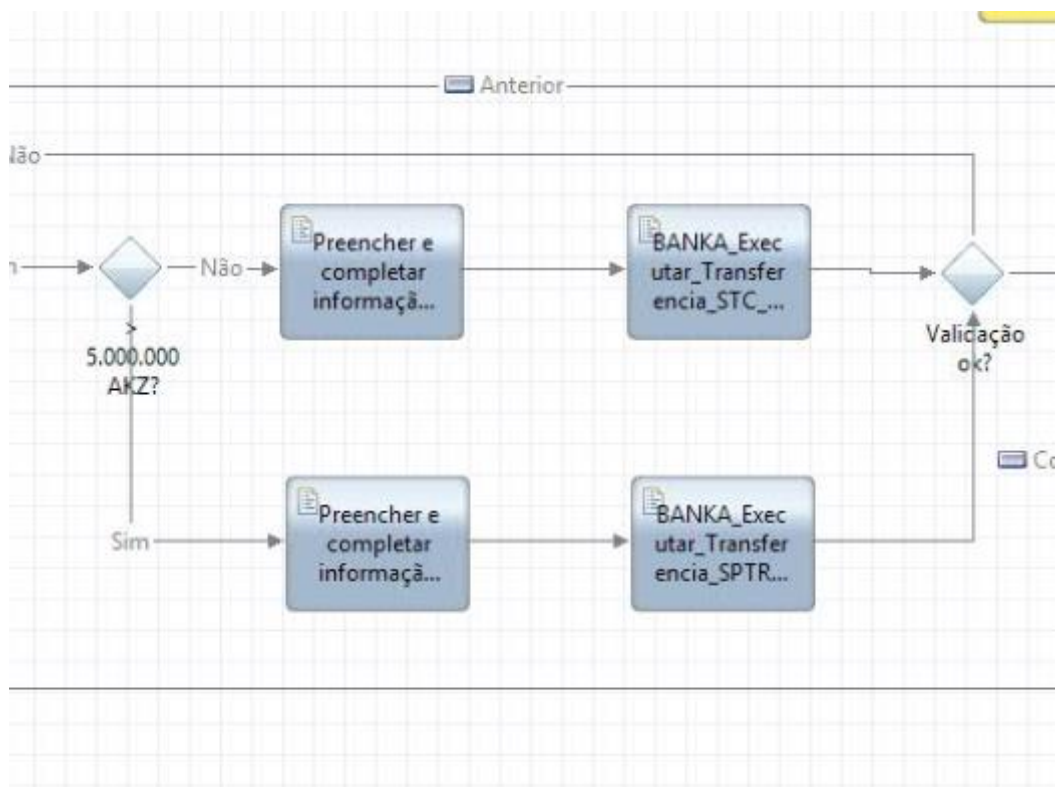


Figura 46 – Excerto da primeira atividade do modelo preliminar do processo de negócio

O bloco aqui representado foi substituído, no diagrama final desta atividade, por um único “General System Service” que recebe os dados da transferência, incluindo o valor, e os envia para o serviço de integração que trata de aplicar a lógica de negócio necessária para evocar o serviço de validação de transferência interbancária do tipo STC ou SPTR. Este pormenor de implementação permite manter a mesma lógica de negócio, reduzir a complexidade do diagrama e aumentar a eficiência da aplicação pois só necessitará de um único serviço de integração para executar transferências para os dois sistemas.

Outro exemplo de agregação de funcionalidade, pode ser observado no decisor assinalado à esquerda da Figura 47, também esta um excerto do diagrama da primeira atividade, que no atual diagrama foi substituído por um único ecrã de UI.

A impressão da certificação pode ser realizada em dois componentes de hardware diferentes, a impressora de rede ou a certificadora, sendo que estava previsto ser o próprio processo a controlar o fluxo para dois ecrãs diferentes no caso de haver necessidade de impressão. No diagrama atual, esta lógica é implementada dentro de um único ecrã, que se adapta ao seu objetivo tendo em consideração o input que recebe do processo. Desta forma é possível reduzir a complexidade do processo e transpor a funcionalidade para o próprio ecrã.

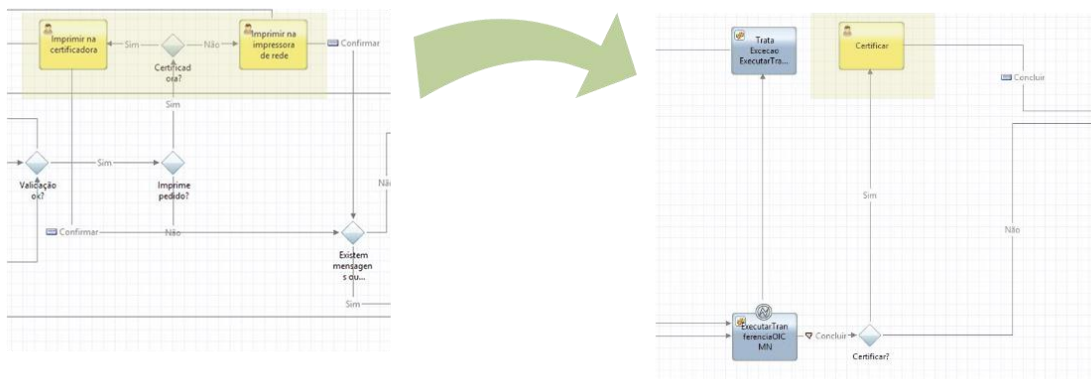


Figura 47 – Impressão de certificação. Planeado vs. Atual.

Por fim, é importante ainda realçar as alterações incluídas na primeira atividade do processo, face ao planeado, no que compete à inclusão de funcionalidade que permitirá a intervenção hierárquica dinâmica calculada a partir dos limites de autorização e respetivos níveis de intervenção previamente configurados.

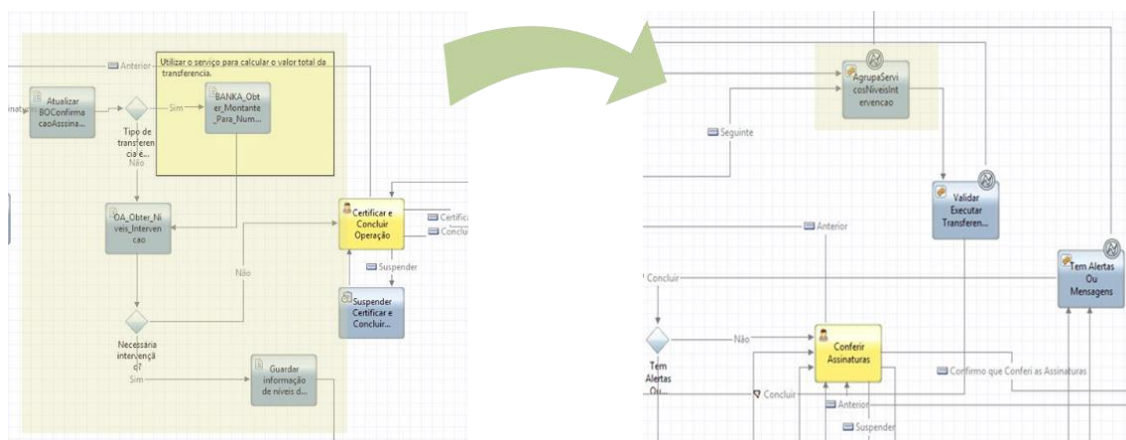


Figura 48 – Obtenção de níveis de intervenção. Planeado vs. Atual.

Para a obtenção dos níveis de intervenção que compõem o limite de autorização da transação em curso no processo de transferência interbancária, de entre os diversos parâmetros de *input* enviado ao serviço um dos mais importantes é o valor da transação, pois é este que vai determinar o limite de autorização aplicado à transação em curso. Quando a operação a executar se trata de uma transferência permanente<sup>29</sup>, é necessário evocar um serviço que calcule o montante total para o número de dias em que esta operação se repetirá, só assim será

<sup>29</sup> Que se repete durante um determinado período.

possível obter o montante total da operação para enviar ao serviço que permite determinar o limite de autorização.

É possível verificar, na Figura 48, que durante a fase de planeamento e design, toda essa lógica foi incluída no diagrama da primeira atividade, no entanto, durante o processo de desenvolvimento verificou-se ser possível agregar todas estas atividades, dentro de uma atividade do tipo “General System Service”, com o nome “AgrupaServicosNiveisIntervencao” permitindo assim a sua reutilização noutros pontos do processo, se assim se mostrar necessário, remover complexidade ao diagrama e encapsular a diferentes operações num único serviço de sistema. Isto permite tratar exceções lançadas individualmente como um todo.

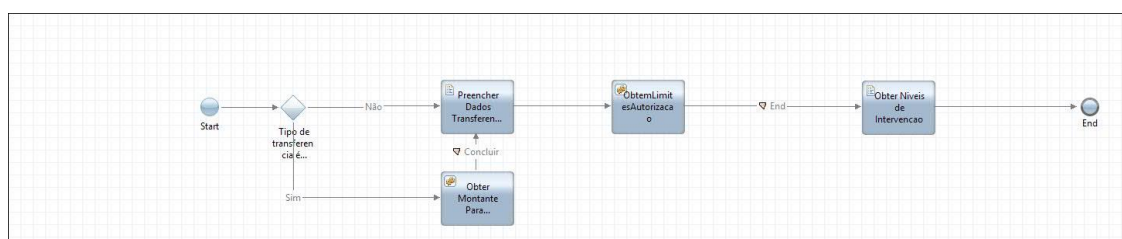


Figura 49 – Diagrama da atividade serviço de sistema “AgrupaServicosNiveisIntervencao”.

A Figura 49 mostra, à entrada, a implementação de um decisor que encaminhará o fluxo do processo para o serviço de integração “Obter Montante Para Numero de Dias” no caso de se tratar de uma transferência permanente, ou diretamente para o script com o preenchimento dos dados da transferência e seguidamente para o serviço de integração que permitirá obter o limite de autorização adequado e consequentes níveis de intervenção.

A partir desta atividade, o processo terá a informação dos perfis de utilizador que devem intervir no subprocesso de limites de autorização, permitindo assim implementar a funcionalidade de intervenção hierárquica dinâmica.

### Subprocesso de Limites de Autorização

O problema, da falta de intervenção hierárquica dinâmica no processo de transferências interbancárias, previamente identificado no subcapítulo 1.2 e que lança o mote para o desenvolvimento do presente estudo é resolvido, em grande parte, no subprocesso de limites de autorização. Este, talvez tenha sido o ponto do processo de transferências interbancárias que mais se afastou do inicialmente planeado.

Na Figura 50 é possível verificar a evolução desde a fase de desenho e planeamento, até à fase de implementação.

Inicialmente, foi planeado incluir as atividades de intervenção hierárquica identificadas no próprio diagrama do processo de transferências interbancárias, no entanto rapidamente se verificou que esta não seria a abordagem correta pois não permitia a reutilização da funcionalidade noutros processos.



Figura 50 - Limites de autorização. Planeado vs. Atual.

Tendo em consideração a vontade do cliente em reutilizar a funcionalidade de intervenção hierárquica dinâmica no processo de transferências intrabancárias (a desenvolver) bem como em outros módulos que serão desenvolvidos no futuro, considerou-se pertinente separar esta funcionalidade para um subprocesso com o nome de “Limites Autorização”, representado na Figura 37.

O subprocesso de limites de autorização recebe, do processo onde é integrado, um conjunto de parâmetros tais como o “Grupo de Autorização”<sup>30</sup>, o “Código da Operação”, “Código do Balcão”, entre outros. Recebe ainda um objeto complexo, “MILimiteAutorizacao”<sup>30</sup>, com o limite de autorização calculado na primeira atividade que contém, por sua vez, uma lista de “Níveis de Intervenção” e uma lista de “Ações”<sup>29</sup>.

Com o intuito de facilitar a compreensão do fluxo deste subprocesso que se pretende apresentar com algum detalhe, divide-se o diagrama em duas metades, abordando cada metade de forma isolada, tendo, no entanto, em mente que ambas se complementam entre si.

Na Figura 51 é possível observar a primeira metade do diagrama do subprocesso de limites de autorização. Salienta-se que todos os elementos estão posicionados em *lanes* de sistema, significando que não há qualquer intervenção humana do que se encontra instrumentado neste subprocesso.

À entrada do processo, encontra-se um elemento de decisão “Obter Limite?” que tem como principal objetivo verificar se o subprocesso recebeu o limite de autorização calculado no processo externo, se tal não aconteceu este instrumenta o fluxo do processo para uma atividade de sistema que implementa o serviço de integração necessário à obtenção do limite de autorização e respetivos níveis de intervenção. Da execução deste serviço resulta o objeto complexo “MILimiteAutorizacao”<sup>31</sup> que, entre os restantes objetos e variáveis que o compõem, possui uma variável booleana onde indica se o processo necessita de intervenção obrigatória, e guarda o resultado obtido para o tipo de aprovação, que tal como descrito no subcapítulo 5.6,

<sup>30</sup> Ver subcapítulo 5.6

<sup>31</sup> Prefixo “MI” é a *naming convention* para os DTO (*Data Transfer Object*) do componente de integração.

poderá ser do tipo “Aleatória” ou “Sempre”. Sendo do tipo aleatória, o módulo de limites de autorização determina se a aleatoriedade obriga, ou não, a intervenção hierárquica no processo.

O decisor seguinte, “Tem Limite Autorizacao”, lê o valor contido nesta variável e determina se o processo segue para a cadeia de intervenção hierárquica ou sai do subprocesso para terminar a sua execução sem necessidade de intervenção.

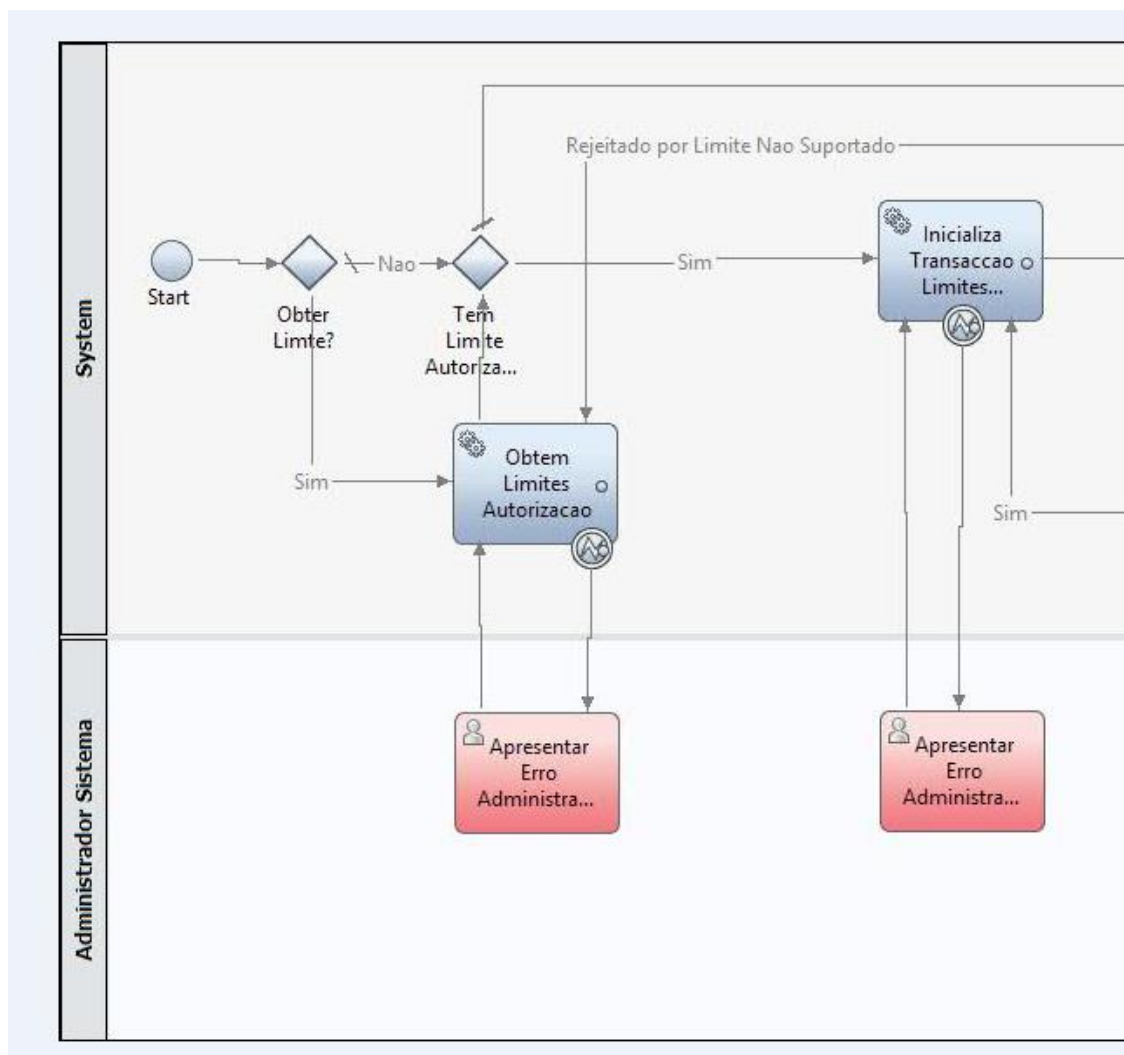


Figura 51 – Primeira metade do subprocesso de limites de autorização.

Tendo sido determinado que o processo deve incluir intervenção hierárquica, é iniciado o ciclo de iterações necessárias até que todos os intervenientes definidos nos níveis de intervenção do limite de autorização emitam o seu parecer, favorável ou desfavorável, à execução do processo de transferência interbancária.

A atividade de sistema seguinte - “Inicializa Transaccão Limites” – dá início ao ciclo de intervenções hierárquicas necessárias para o processo de transferência em curso. É possível, na Figura 52, verificar um conjunto de atividades de *scripting* que introduzem a lógica necessária e que permitirá à atividade seguinte realizar a atribuição correta da intervenção ao utilizador, ou grupo de utilizadores, com o perfil necessário para tal.

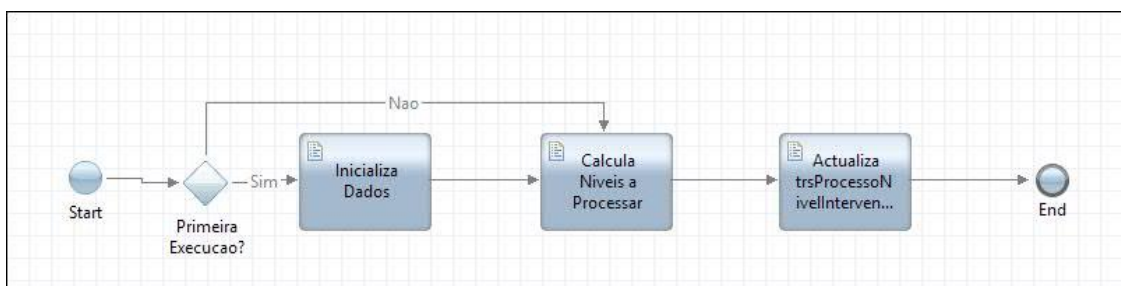


Figura 52 – Diagrama da atividade de sistema “Inicializa Transacao Limites”

À entrada do diagrama da atividade representado na Figura 52 está presente um decisor que verifica se é a primeira execução. Sendo a primeira execução, todos os dados necessários à execução dos níveis de intervenção são inicializados, como pode ser verificado na Figura 53.

Na Figura 53 é possível verificar que existe lista com os níveis de intervenção por processar, que faz a atribuição direta à lista de níveis de intervenção presente no objeto complexo “MiLimiteAutorizacao” e um objeto que guarda o valor mais baixo do nível a processar. O objetivo desta lógica é permitir guardar o valor do primeiro nível a processar obtido da lista inicial de níveis de intervenção obtida no processo. O fluxo só passará neste script na primeira iteração do subprocesso de limites de autorização.

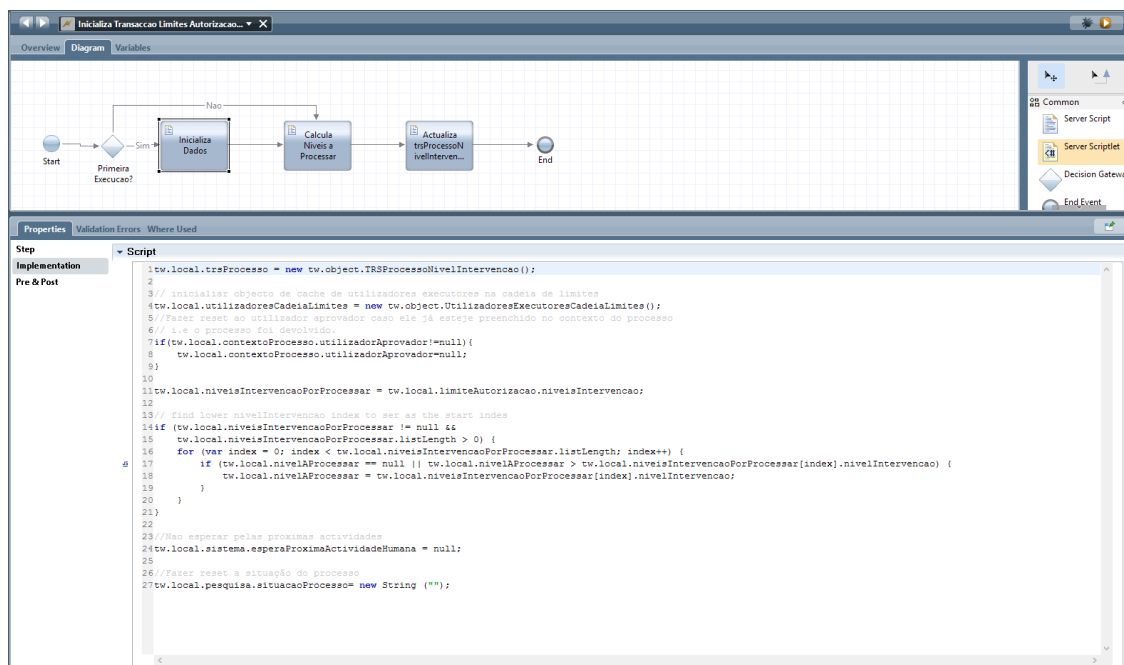


Figura 53 – Detalhe do script “Inicializa Dados”

O Script seguinte – “Calcula Níveis a Processar” – permite interagir com duas listas, a primeira de níveis de intervenção por processar e uma nova instância de uma lista de níveis de

intervenção a processar. Desta forma é possível incluir lógica para calcular a execução do nível seguinte, guardar a informação de processamento em paralelo de dois, ou mais, níveis distintos e a informação do valor do nível posterior ao seguinte. Esta informação servirá de suporte ao mapeamento das variáveis necessárias para o BO (*Business Object*) “TrsProcessoNivelIntervencao<sup>32</sup>” na atividade de *scripting* seguinte – “Atualiza TrsProcessoNivelIntervencao”.

Na verdade, não há qualquer razão que impeça de agregar as duas atividades de *scripting* numa única atividade, o motivo que motivou a criação de uma nova atividade foi a separação de funções para permitir uma leitura mais segmentada do processo e do próprio código.

Deste momento em diante, importa analisar a segunda metade do subprocesso de limites de autorização representada na Figura 54.

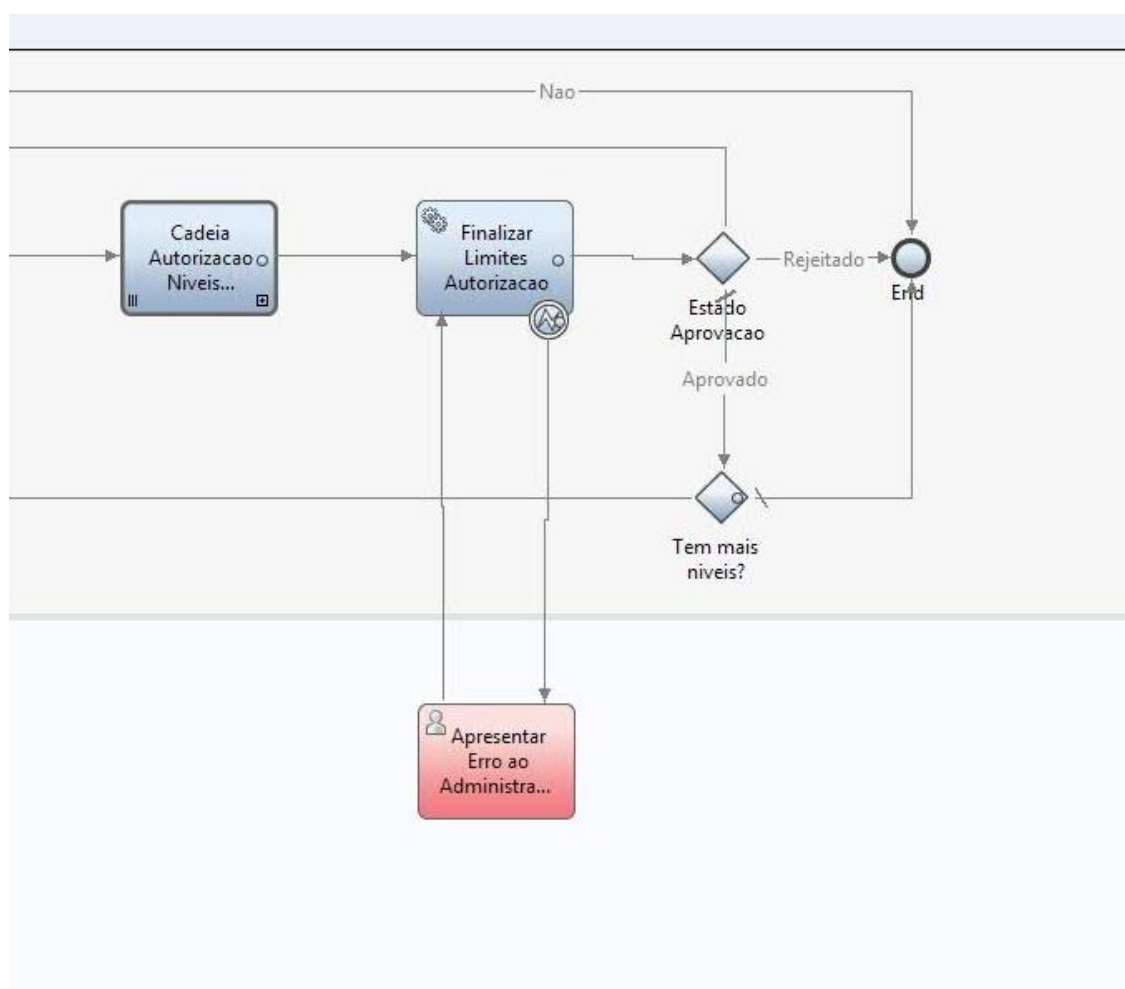


Figura 54 – Segunda metade do subprocesso de limites de autorização.

Como é possível verificar, o elemento seguinte no subprocesso de limites de autorização é um novo subprocesso com a cadeia de autorização de níveis de intervenção. O motivo de incluir a cadeia de níveis de intervenção num subprocesso, prende-se também com a necessidade de

<sup>32</sup> O prefixo “TRS” refere-se à *naming convention* para os *Business Objects* no projeto Bankee.

reutilização futura, ou possibilidade de definir diferentes cadeias para diferentes processos dependendo estes de níveis de intervenção iguais ou específicos.

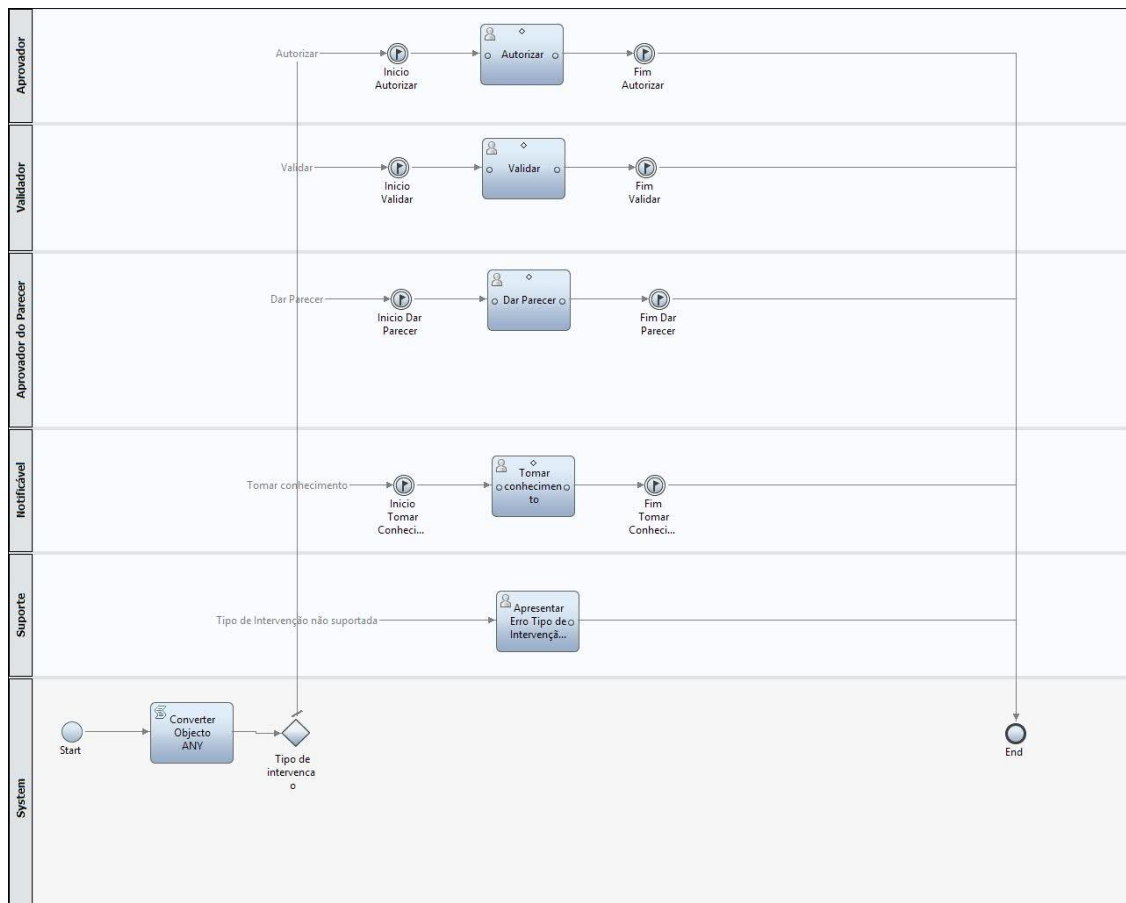


Figura 55 – Subprocesso de Cadeia de Intervenção.

À semelhança do que acontece no subprocesso anterior, o representado na Figura 55 recebe também um conjunto de objetos necessários para o sucesso da execução de cada uma das atividades, incluindo um objeto abstrato com os dados de específicos de input do processo que determina se é um processo de transferências interbancárias ou outro.

No decisor seguinte, é determinado o tipo de intervenção necessário e daí feito o encaminhamento para a atividade de intervenção hierárquica correspondente.

Como é possível verificar na Figura 55, cada uma destas atividades está localizada numa *lane* de utilizador, o que implica que em cada uma destas terá, obrigatoriamente, de participar um utilizador com o perfil adequado à função, cumprindo assim a tarefa de intervenção no processo determinada pela sua posição na hierarquia da instituição. A Figura 41, Figura 42, Figura 43 e Figura 44 mostram ao pormenor todos os elementos de cada uma destas atividades.

Os principais dados de *output* deste subprocesso são a posição do utilizador que intervencionou o processo, onde indica, positiva ou negativamente, se aprova, autoriza ou a emissão de



parecer, os comentários associados a cada uma das atividades e a informação pormenorizada sobre o utilizador da hierarquia que a emitiu.

No final de cada uma destas atividades, o processo volta ao subprocesso de Limites de autorização na Figura 51, onde é reencaminhado para a atividade de sistema “Finalizar Limites Autorizacao”.

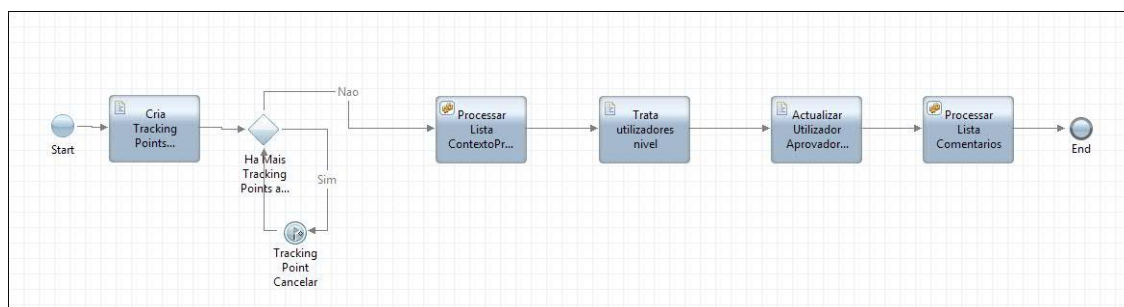


Figura 56 – Diagrama da atividade “Finalizar Limites Autorizacao”

No diagrama da atividade “Finalizar Limites Autorizacao” representado na Figura 56 surge uma primeira atividade de *scripting* que recolhe todos os *tracking points* em que o estado obtido na cadeia de níveis de intervenção é negativo, ou seja, onde o utilizador que interveio naquele ponto do processo emitiu um parecer negativo para efeitos de monitorização de processo. Após este passo seguem-se um conjunto de atividades com o objetivo de tratar o contexto do processo, mapear os utilizadores por nível de intervenção e processar a lista de comentários deixados recolhidos no nível de intervenção processado.

Após o ponto do subprocesso anteriormente descrito é encontrado um decisor que determinar o fluxo com base no estado de aprovação. Se o processo tiver sido recusado no nível de intervenção este sai do subprocesso e volta à atividade inicial de “Registar Transferência”, permitindo que o utilizador que iniciou o processo reveja os dados do mesmo, e se for necessário alterar algum destes ou mesmo cancelar o processo.

Se tiver sido aprovado, ou obtido parecer positivo no nível de intervenção, segue para outro decisor que verificará a existência de mais níveis de intervenção para processar. Não havendo mais níveis a processar significa que foram cumpridos com sucesso todos os níveis de intervenção hierárquica, sai do subprocesso, executa a operação de transferência interbancária e termina o processo com sucesso. Havendo mais níveis a processar, volta à atividade “Inicializa Transaccão Limites” da Figura 51 e faz uma nova iteração pelo previamente descrito até que sejam processados todos os níveis de intervenção hierárquico.

A salientar, que os níveis de intervenção hierárquica são dinâmicos pois comportam-se de acordo com o configurado no módulo de limites de autorização, permitindo reencaminhar o utilizador da hierarquia por diferentes níveis de intervenção com base nesses dados.

### 6.3.4. Ecrãs e Navegação

Pretende-se nesta secção demonstrar, graficamente, os elementos de UI e navegação que foram introduzidos no âmbito do desenvolvimento do módulo de transferências interbancárias, sendo possível identificar os elementos de interação com o utilizador nos diferentes momentos do processo.

Os ecrãs aqui apresentados foram desenvolvidos concretamente para este propósito com recurso à tecnologia descrita na secção 4.1.1.1. O desenvolvimento foi realizado com base em *mockups* desenvolvidos pela equipa de análise funcional durante a fase de design do processo de desenvolvimento adotado.

Alguns elementos são propositadamente omitidos do ecrã para preservar a identidade da instituição bancária e, eventualmente, do cliente.

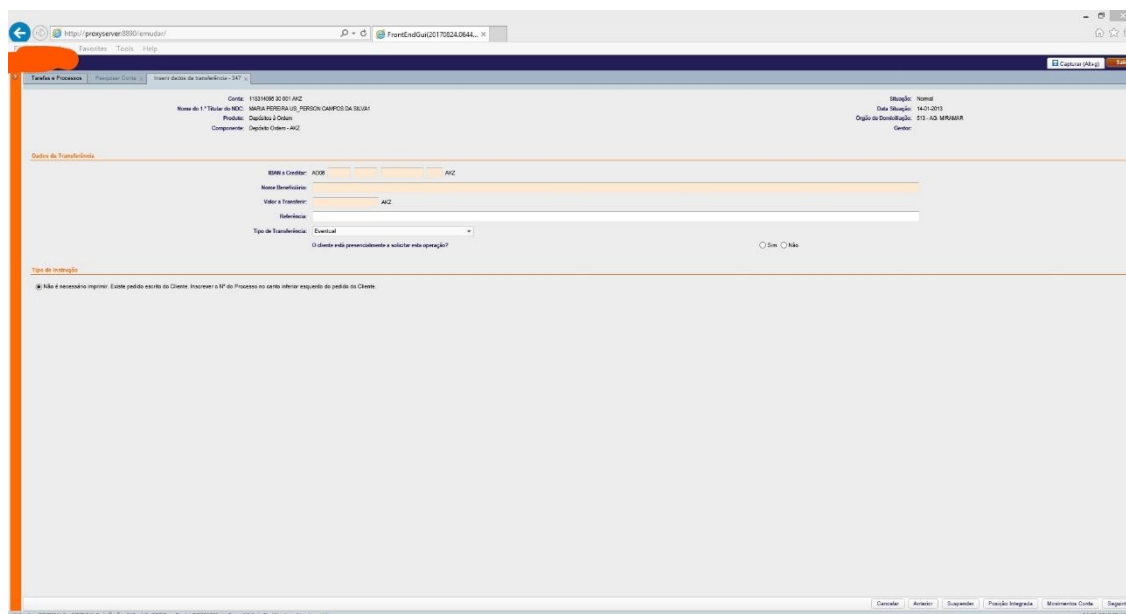


Figura 57 – Ecrã de inserção de dados da transferência.

Na Figura 57 é possível visualizar com detalhe o ecrã de inserção de dados da transferência. Demonstra o primeiro ecrã da primeira atividade do processo. Neste ecrã é possível consultar os dados da conta ordenante e inserir os dados relativos à conta beneficiária, que como é possível verificar, o campo disponível aceita e valida o IBAN <sup>33</sup>introduzido.

Seguindo a lógica de negócio do processo, se existirem transferências em curso entre as mesmas contas, ordenante e beneficiária, definidas no primeiro ecrã do processo o utilizador é reencaminhado para o ecrã apresentado na Figura 58.

<sup>33</sup> *International Bank Account Number*

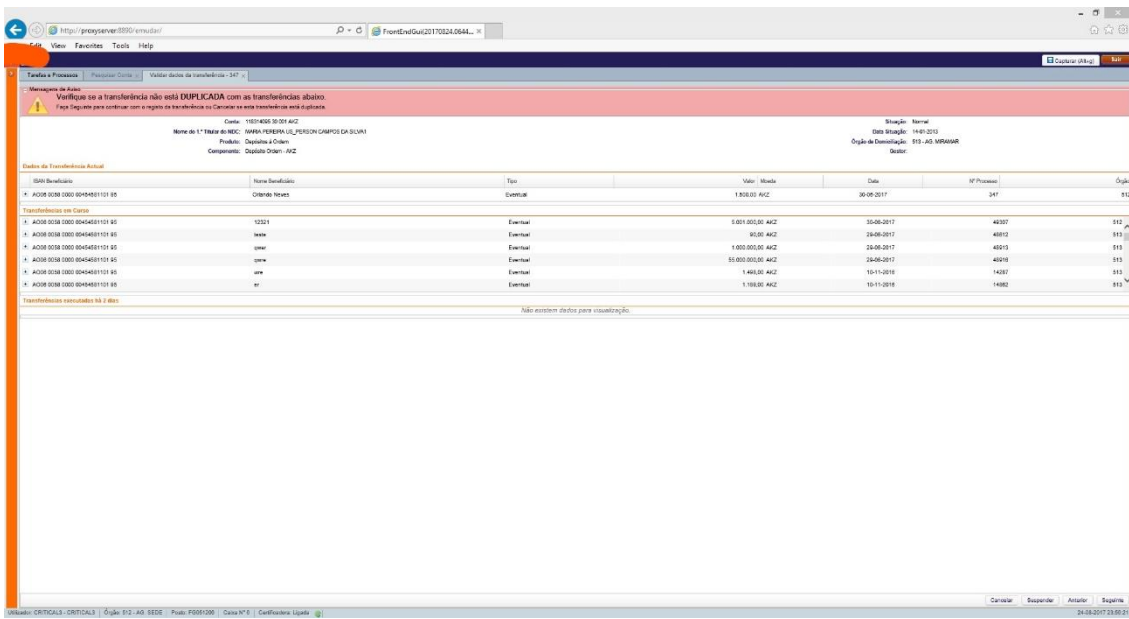


Figura 58 – Ecrã de validação dos dados da transferência.

Após validação dos dados da transferência o fluxo do processo segue até ao ecrã de validação de assinaturas do cliente, representado na Figura 59. Este é um ecrã genérico não desenvolvido no âmbito deste estudo.

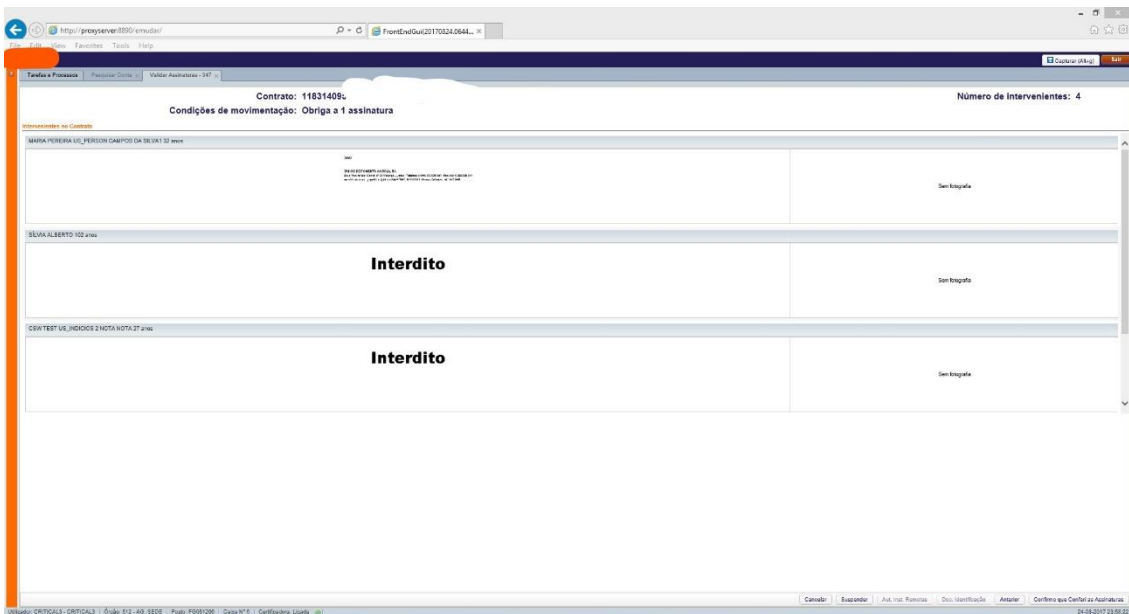


Figura 59 – Ecrã de validação de assinaturas.

Imediatamente antes da atividade de certificar o pedido, é possível confirmar os dados introduzidos para a transferência no ecrã da Figura 60. Após este ponto é inserido o registo de certificação, um registo de base de dados com toda a informação do processo para efeitos de auditoria, e o processo termina a primeira atividade de registo de transferência.

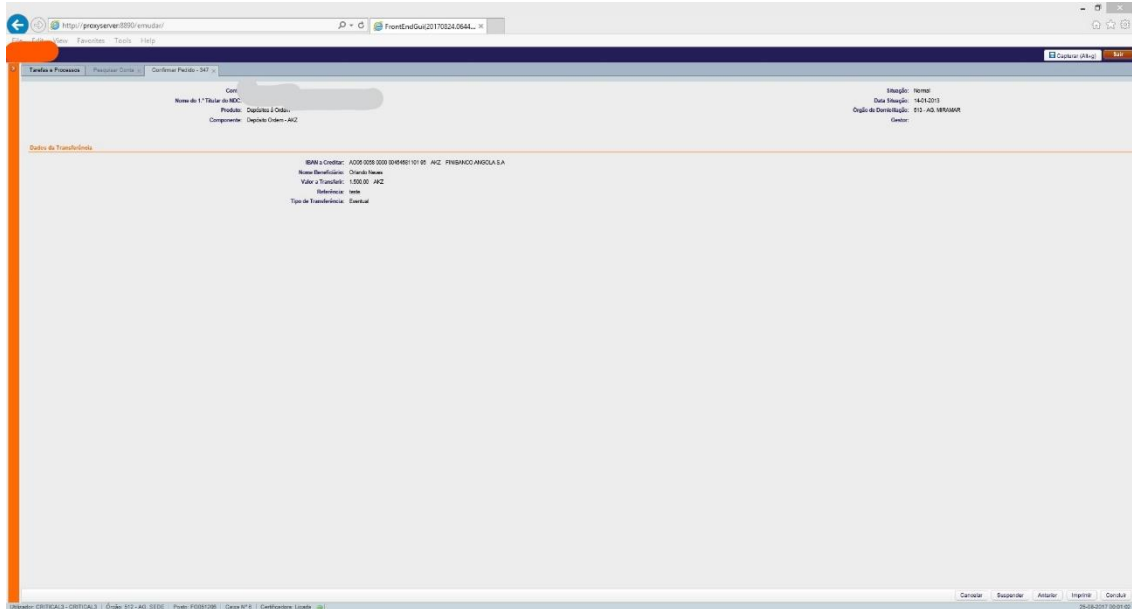


Figura 60 – Ecrã de confirmação do pedido de transferência

## Ecrãs dos Níveis de Intervenção

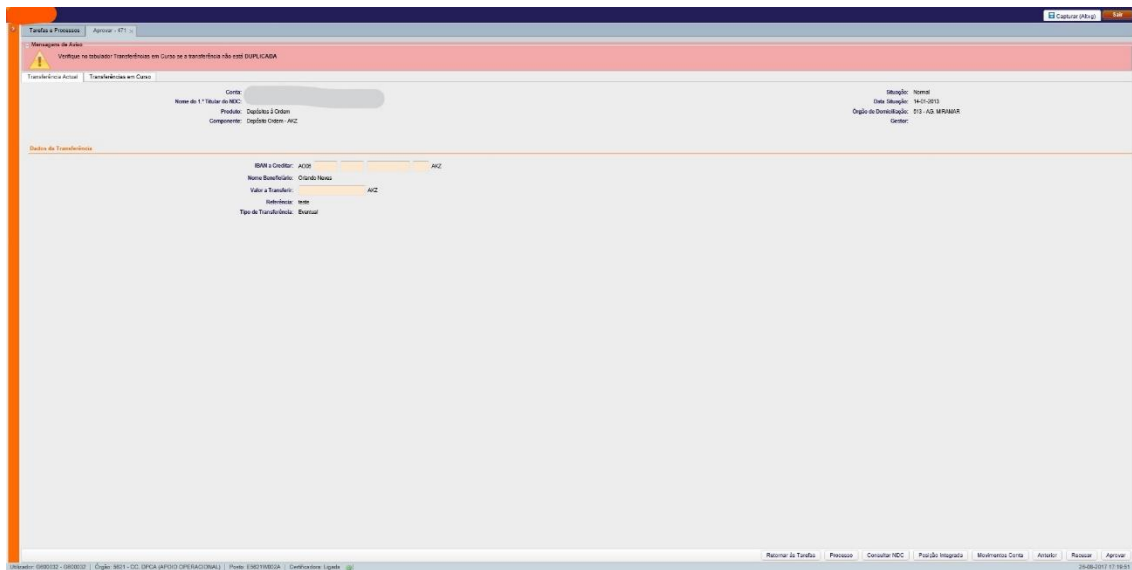


Figura 61 – Ecrã de aprovação de transferência interbancária

Na Figura 61 é possível visualizar o primeiro ecrã da atividade de aprovação no subprocesso de intervenção hierárquica. Neste ecrã o utilizador é obrigado a incluir novamente o IBAN da conta beneficiária e o valor da transação para conseguir aprovar o processo com sucesso.

Tem ainda a possibilidade de consultar todos os dados referentes ao processo, ao número de cliente, os movimentos da conta, entre outros, sendo que todas estas funcionalidades não foram desenvolvidas no âmbito deste estudo, mas reintroduzidas neste processo.

O primeiro ecrã da atividade de autorização do subprocesso de intervenção hierárquica é em tudo idêntico ao representado na Figura 61. No decorrer desta atividade o utilizador poderá transmitir um parecer não obrigatório num ecrã idêntico ao da Figura 62, mas sem obrigatoriedade no preenchimento do campo de “parecer”.

O primeiro ecrã da atividade “Dar Parecer” no subprocesso de intervenção hierárquica é idêntico ao representado na Figura 60. No entanto o fluxo do processo obriga o utilizador a um comentário obrigatório num ecrã com o da Figura 62.

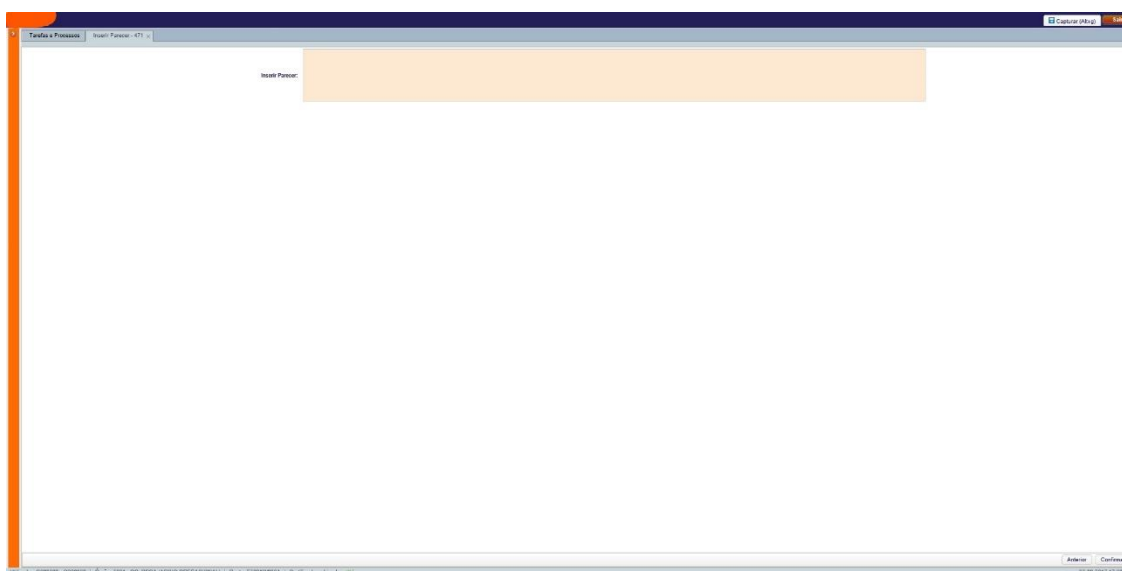


Figura 62 – Ecrã para registo de parecer

Na atividade “Tomar Conhecimento”, não havendo associação de documentos, o primeiro ecrã é idêntico ao da Figura 60 sendo que o utilizador tem a opção de recusar ou aprovar a tomada de conhecimento.

### 6.3.5. Serviços

Durante a implementação do processo de transferências interbancárias foi também necessário desenvolver alguns serviços da componente de integração demonstrada no diagrama da arquitetura da aplicação Bankee, representado na Figura 22 como “INT”.

No âmbito desta necessidade foram desenvolvidos serviços de integração com o sistema bancário central – Banka – através do consumo de API disponibilizadas para permitir a interação e comunicação de dados entre a aplicação Bankee e o sistema bancário central, sendo toda a transacionalidade gerida pela Banka.

Os serviços SOAP<sup>34</sup>, implementados em JAVA EE com recurso a algumas das ferramentas de desenvolvimento descritas em 6.1.2 foram desenvolvidos com base na *Service Encapsulation Pattern* descrita em [68] permitindo encapsular a lógica de negócio na componente de serviços permitindo que a funcionalidade do serviço possa ser disponibilizada além dos limites da aplicação.

Neste caso em concreto, os serviços desenvolvidos são na sua maioria consumidos nas componentes de *frontend (FE)* e BPM. Na componente componente de *frontend*, por norma, para efeitos de validação de dados inseridos no ecrã do cliente no servidor ou para obter elementos necessários para o preenchimento dos elementos do ecrã tal como o exemplo demonstrado do excerto da Figura 63.

```
296
297
298  /**
299   * Cobre o requisito: REQ-CNSTRF-OICMN-0113-Ação Detalhe - Transferência STC
300   * Apresenta os detalhes de uma transferencia STC
301   *
302   * @param contexto cont informacao de contexto
303   * @return resposta com o estado do pedido
304   */
305  @WebMethod
306  @WebResult(name = "resultado", targetNamespace = WsConstants.NAMESPACE)
307  MIObterDetalleTransferenciaOICResposta obterDetalleTransferenciasSTC(
308      @WebParam(name = "contexto", targetNamespace = WsConstants.NAMESPACE) MIContexto contexto,
309      @WebParam(name = "pedido", targetNamespace = WsConstants.NAMESPACE) MIObterDetalleTransferenciaPedido pedido);
310
```

Figura 63 – Excerto de método de detalhe no Webservice

Já quando o serviço é consumido no BPM, são tipicamente serviços transaccionais usados na execução de operações em modo de efetivação ou validação (da operação). Podem ser integrados em BPM como atividades de “General System Service” demonstradas na secção 6.3.3.3 onde correm de forma síncrona, ou em *lanes* de sistema correndo assincronamente.

Um exemplo de um método do serviço de transferências interbancárias que entra nesta categoria é o representado na Figura 64, o método que expõem a funcionalidade de executar a operação de transferência interbancária, bem como de agregar um conjunto de lógica de negócio necessária para o correcto registo da operação no sistema bancário central, a Banka.

<sup>34</sup> Simple Object Access Protocol

```
461
462
463  * Executa a Transferencia OIC pelo sistema STC ou SETE no BANKA = cria o registro de certificacao no Banks ou altera-lo
464  *
465  * Cobre os requisitos:
466  * REQ-TRFOMM-RIRF-0120-Transferencia não permitida
467  * REQ-TRFOMM-RIRF-0125-Executa validação de transferencia não permitida?
468  * REQ-TRFOMM-RIRF-0300-Executar no BANKA transferencia STC
469  * REQ-TRFOMM-RIRF-0130-Obter montante para executar a transferencia por SETE
470  * REQ-TRFOMM-RIRF-0310-Executar no BANKA transferencia SPR
471  * REQ-TRANSV-0250-CRIAR REGISTO DE CERTIFICACAO NO BANKA
472  *
473  *
474  * @param contexto contexto informacao de contexto
475  * @param dados contém os dados da transferencia, a informaçao do banco e os dados da certificacao
476  * @return resposta certificacao retorna a resposta de certificacao com o id do registro criado na BD
477  */
478
479  @WebMethod
480  @WebResult(name = "resultado", targetNamespace = WsConstants.NAMESPACE)
481  MICertificacaoExtResposta executarTransferenciaOIC(
482      @WebParam(name = "contexto", targetNamespace = WsConstants.NAMESPACE) MIContexto contexto,
483      @WebParam(name = "dados", targetNamespace = WsConstants.NAMESPACE) MIExecutarTransferenciaOICMNPedido dados);
484
485
```

Figura 64 – Excerto de método transacional do serviço de transferências

Em suma, neste contexto foi desenvolvido um serviço de transferências que agrega 28 métodos diferentes expostos segundo as necessidade que o processo de transferências interbancárias impõe.

## 7. Análise de Valor

Nos ambientes empresariais atuais, devido ao seu dinamismo e constante evolução, é importante ser-se competitivo. Assim, as empresas procuram desenvolver novos produtos ou serviços que procurem gerar valor para o cliente. Valor este calculado pela relação custo / benefício e na percepção que o cliente tem desta relação.

O cálculo do custo é geralmente mais simples de perceber ou até justificar, já o cálculo do benefício mostra-se mais complexo, pois as prioridades são tão mutáveis quando a diversidade de indivíduos com capacidade de avaliação crítica.

Para o cálculo e análise de valor, é necessário quantificar o custo e o benefício, bem como o seu peso relativo dentro da função de valor [1]. Tendo em consideração a complexidade inerente ao cálculo do valor absoluto de um benefício, deve-se aplicar o conceito de benefício relativo. Neste conceito, é atribuído um valor percentual baseado na avaliação relativa ao benefício total. Também a quantificação do custo será calculada com base nas percentagens relativas, que à *posteriori* serão mantidas em cada uma das funções de valor com um custo relativo associado.

A técnica de engenharia e análise de valor desenvolvida em 1947 por Lawrence D. Miles [2], defende ainda que a percepção de valor aumenta sempre que se diminuem custos ou pelo aumento de performance desde que o cliente queira, precise ou esteja disposto a pagar por esta.

O método de análise hierárquica (AHP) refere-se a uma técnica estruturada de lidar com decisões complexas na análise de valor e a quantificação da criação de valor.

Considerando o âmbito do projeto Bankee, é determinado que o objetivo será o desenvolvimento de um módulo para a execução de transferências interbancárias com níveis de intervenção hierárquica dinâmicos. Os critérios serão o suporte à automatização do processo de negócio, integrar com o *Core* bancário, integração transparente com os restantes módulos desenvolvidos, suporte à intervenção hierárquica e monitorização. As alternativas seriam diferentes soluções desenvolvidas à medida por outra equipa de engenharia de software, já que este é um módulo com necessidades específicas que será desenvolvido à medida. Relativamente ao processo de tomada de decisão teriam de ser realizadas comparações par a par de todos os critérios do género Eficiência *versus* Segurança, por exemplo. As pontuações devem ser então calculadas, preferencialmente para cada critério, após comparação baseada nos *inputs* do utilizador de acordo com a importância dada a cada critério relativamente aos restantes. Sobre estes dados é então aplicada uma fórmula que determina o coeficiente e indica a alternativa vencedora.



## 7.1. Elementos Chave do Modelo NCD

O modelo NDC<sup>35</sup> proposto por Peter Koen, define um modelo conceptual de primeira linha na definição de novos conceitos, dividindo-o em três áreas distintas, tal como representado na Figura 65.

1. O núcleo, ou *engine*, que se posiciona no centro do modelo, pretende representar a visão estratégica e a cultura que impulsionam os novos conceitos em desenvolvimento.
2. Elementos chave – No centro do modelo estão representados os 5 elementos chave de atividade do *front end*.
3. Elemento externo – Refere-se aos fatores ambientais externos que poderão influenciar cada um dos elementos chave.

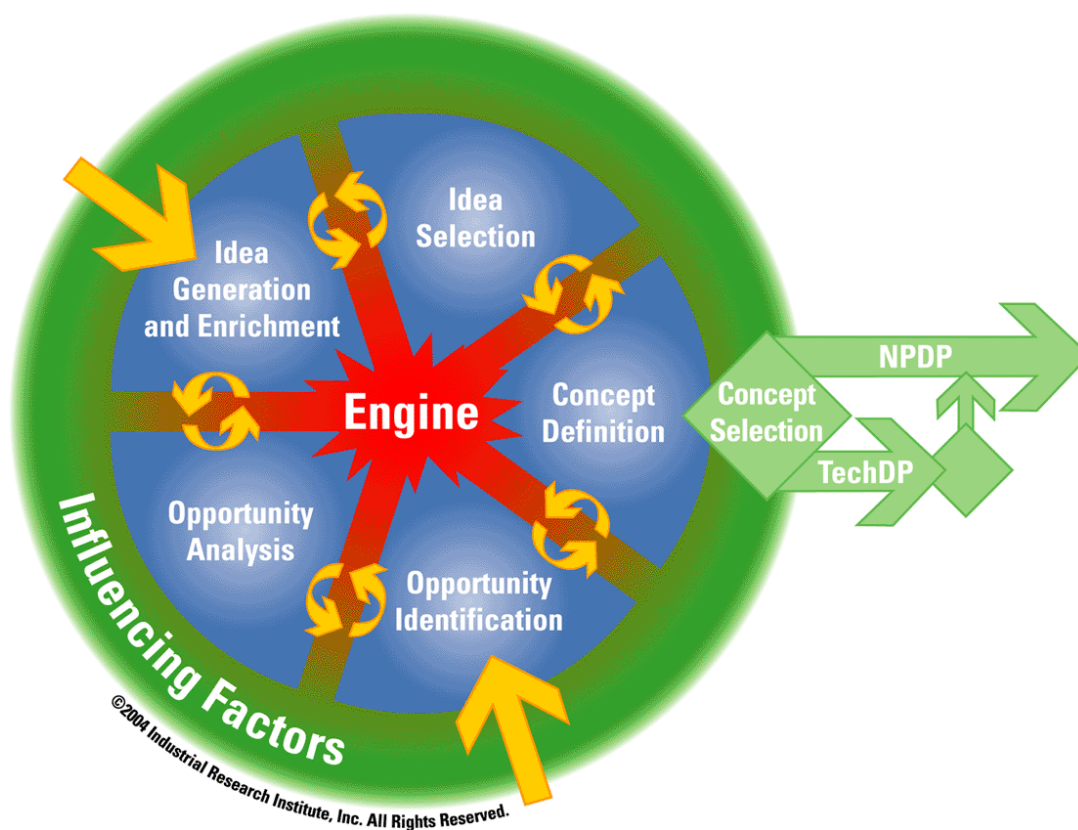


Figura 65 - Representação gráfica do modelo NCD

Fonte: [37]

### Oportunidade

É a partir deste elemento do modelo NDC que se inicia o desenvolvimento do projeto *Bankee* e cada um dos módulos que o constituem. Foi identificada a oportunidade de implementar um sistema baseado em processos de negócio ponto-a-ponto, transformando operações dispersas

<sup>35</sup> NCD - *New Concept Development Model* de Peter Koen.

e sem controlo de diferentes intervenientes hierárquicos, em processos de negócio bem definidos.

### **Análise da oportunidade**

É necessário ter em consideração que a instituição bancária, daqui em diante identificada como “cliente”, pretende com o projeto *Bankee* resolver diferentes problemas inerentes ao anterior sistema bancário, bem como problemas específicos que esse mesmo sistema trazia para cada módulo.

Para o presente caso, na análise da oportunidade de inovação, será considerado o módulo de transferências interbancárias.

O cliente tem uma grande necessidade de transformar operações dispersas, que permitem a execução de transferências de valores entre diferentes instituições bancárias, em processos de negócio eficientes e com capacidade de automatização, monitorização e fácil readaptação. Atribuiu alta prioridade à execução da oportunidade identificada e está efetivamente disposto a comprar uma solução que cumpra com os referidos pressupostos.

Existe, efetivamente, o *know how técnico* para que possam ser cumpridos esses objetivos, contudo, é necessário realizar uma análise de abordagens existentes com o intuito de perceber que tecnologias podem ajudar a obter o resultado esperado.

### **Gerar e enriquecer a ideia**

Com o objetivo de prender a oportunidade e a transformar numa ideia conceptualmente exequível, está definido um plano alto nível para a descrição da sua implementação:

- Perceber, junto do cliente, que acessos ao núcleo bancário este disponibiliza para que se possam cumprir os objetivos identificados.
- Através Da análise de negócio. Modelando em conjunto com o cliente, o processo de transferências interbancárias recorrendo à notação BPML 2.0. Este é um ponto valorizado pelo cliente, pois permite-lhe participar ativamente na definição do produto que necessita.
- Realização de pequenos casos de estudo, que permite estudar conceptualmente a ideia e desenvolver novas abordagens para a resolução dos problemas que daí possam emergir.
- Debater sobre pontos de vista e opiniões de colegas de equipa a trabalhar em diferentes áreas. Por vezes, as melhores ideias surgem nos momentos, locais e das pessoas mais inesperadas. É necessário estar atento a estímulos que possam surgir do exterior.

### **Seleção da ideia**

Durante esta fase, serão realizados alguns exercícios para identificar, de um portefólio de ideias que possam ter surgido no âmbito das atividades realizadas no ponto anterior, ou mesmo durante o mesmo – de forma iterativa – caso surja uma ideia destacadamente rica.

Na seleção da ideia, será necessário ter em consideração fatores como a probabilidade de sucesso, com conseqüente satisfação e aumento de confiança na relação com o cliente. Também se considera importante a realização pessoal e obtenção de novas competências que possam ser úteis para colocar em prática ideias futuras.

### **Definição do Conceito**

Este o ponto assume elevada relevância no modelo NCD, pois após o seletor de conceito, trata-se do - único - ponto de saída para o processo de desenvolvimento.

O conceito é baseado na arquitetura da aplicação Bankee, o que permite ter a estrutura previamente preparada para receber o novo módulo de transferências interbancárias. A modelação do processo incluirá atividade específicas que permitiram atingir os objetivos da operação de negócio, bem como consumir dados de um módulo autocontido, de definição de limites de autorização.

O módulo de limites de autorização disponibiliza funções de configuração, em *backoffice*, das regras de negócio que definem o correto percurso e atribuição à rede de intervenção hierárquica durante a execução do processo. Neste módulo poderão ser configurados valores de limites de autorização que contenham obrigatoriamente a definição de um ou mais níveis de intervenção, com identificação dos perfis necessários para a realização das diferentes intervenções hierárquicas durante o decorrer do processo de transferência interbancária.

## **7.2. Proposta de Valor**

O valor do módulo a desenvolver no seguimento da resolução do problema previamente especificado assume uma perceção mais significativa junto do cliente direto, a instituição bancária, não obstante, influenciará também a qualidade dos serviços prestados por esta aos seus clientes finais.

Para a instituição bancária, o valor está diretamente relacionado com as seguintes funções de valor:

- Automatizar o processo de negócio de transferências interbancárias;
- Otimizar fluxos de negócio;
- Controlar recursos;
- Melhorar a performance do processo;
- Identificar pontos de estagnação;
- Permitir que diferentes utilizadores intervenham durante a execução do processo;
- Adaptar, sem muito esforço, o processo de negócio a fatores externos e internos;
- Permitir que a própria instituição participe na modelação do processo;
- Reduzir riscos operacionais;
- Prevenir fraudes;
- Monitorizar o processo.

O produto que se pretende desenvolver, assumirá um papel deveras importante na execução das operações de negócio de transferências interbancárias. Permitirá um suporte ao processo de negócio de todas as atividades envolvidas na execução da referida operação, permitindo que a instituição tenha uma visão global do processo. Tal facto, permitirá um melhor acompanhamento ao serviço providenciado ao cliente (pela instituição bancária), reduzir o risco de fraude envolvendo diferentes intervenientes da hierarquia bancária durante o processo de negócio, melhorar e otimizar continuamente os fluxos de negócio.

### 7.2.1. Modelo de Negócio *Canvas*

A Figura 66 representa o modelo de *canvas* para a proposta de valor descrita, identificando os principais parceiros, atividades, recursos, clientes e canais de distribuição. É a representação lógica do modelo de negócio inerente ao módulo de transferências interbancárias. Também neste modelo são identificadas as proposições de valor inerentes a este módulo.

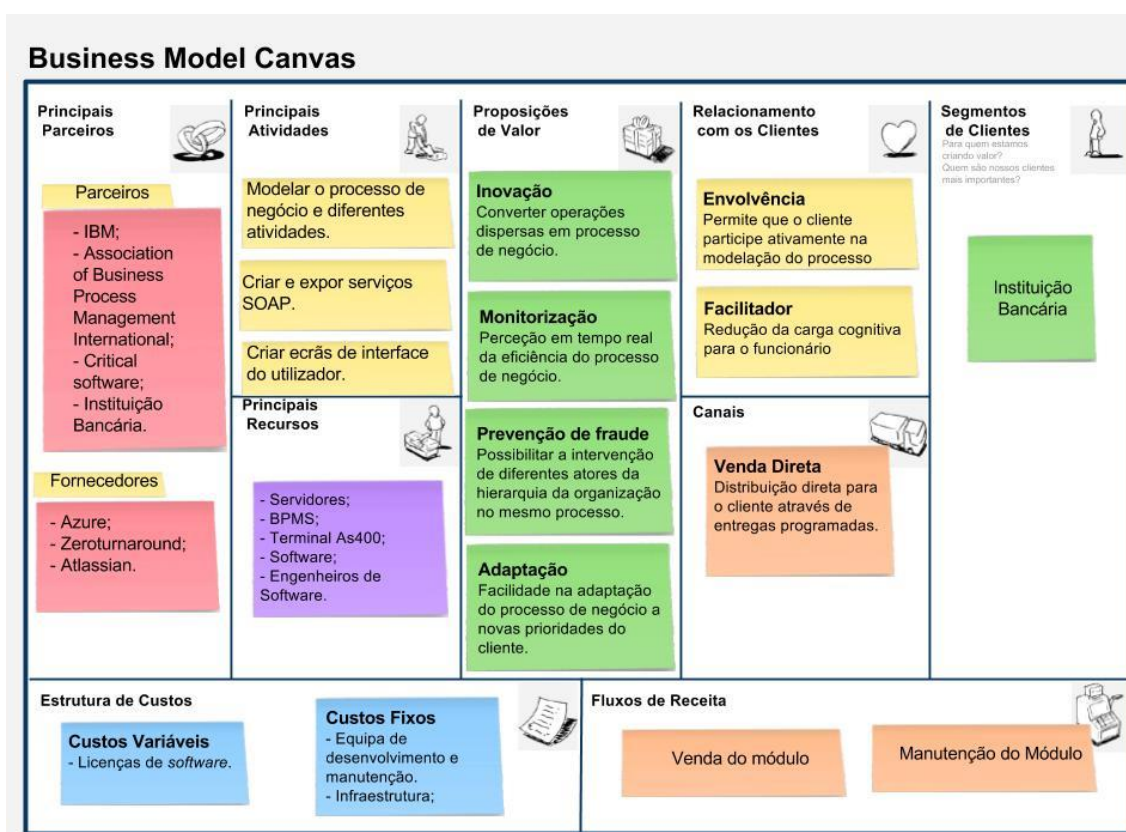


Figura 66 - Canvas do modelo de negócio do projeto Bankee

### 7.3. Rede de Valor

A interação entre os diferentes atores de um modelo de negócio é essencial para exponenciar a criação de valor. A relação resulta do envolvimento e massa criativa de um conjunto

organizado de cargos que se conjugam com um objetivo comum, permitindo assim abrir lugar à inovação através de uma rede de valor.

Tendo em consideração que a cadeia de valor de Porter se baseia num modelo linear de negócio, que invoca a imagem de uma linha de produção industrial, o modelo de Verna Allee mostra-se mais adequado à análise de valor do módulo a desenvolver.

É possível identificar toda uma dinâmica entre os diferentes intervenientes da rede de valor da Critical Software com o desenvolvimento da aplicação Bankee, permitindo perceber os tipos de valor tangíveis e intangíveis descritos por Verna Allee. Consequentemente, a mesma rede de valor se aplica ao desenvolvimento do módulo de transferências interbancárias, estando esta representada no diagrama da Figura 67.

Há todo um conjunto de relações entre os diferentes intervenientes que sugere a criação de valor tangível e intangível.

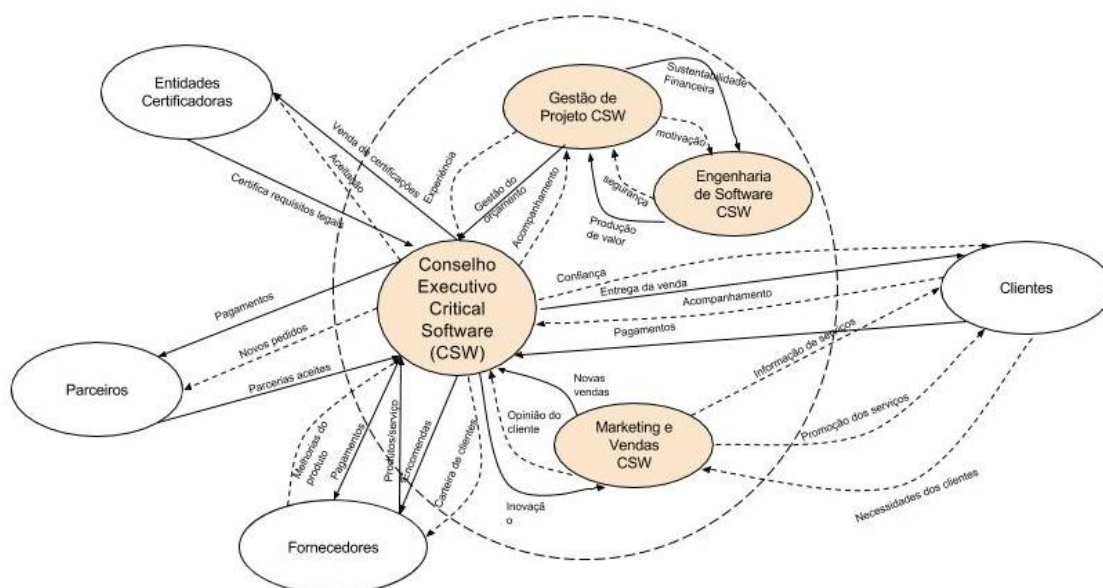


Figura 67 - Rede de valor de Verna Allee aplicada ao modelo de negócio da *Critical Software*

### 7.3.1. Valor Percebido

Como referido anteriormente, clientes distintos têm uma percepção distinta do valor. Além de clientes distintos terem uma percepção distinta do valor para o mesmo produto ou serviço também as empresas fornecedoras, poderão ter uma percepção diferente (da do cliente) do valor do seu produto ou serviço [3]. Para o desenvolvimento do módulo que sustenta o caso de estudo desta aplicação, na tabela 1 estão demonstrados os benefícios e sacrifícios que representam o valor para o cliente numa perspetiva longitudinal.

Tabela 14 - Benefícios e sacrifícios da proposta de valor

<div style="text-align: center;">Âmbito</div> <div style="text-align: left;">Domínio</div>	<div style="text-align: center;">Produto / Serviço</div>	<div style="text-align: center;">Relacionamento</div>
<div style="text-align: center;">Benefício</div>	<ul style="list-style-type: none"> <li>- Automatização;</li> <li>- Otimização;</li> <li>- Monitorização;</li> <li>- Adaptação;</li> <li>- Controlo;</li> <li>- Prevenção;</li> </ul>	<ul style="list-style-type: none"> <li>- Segurança;</li> <li>- Confiança;</li> <li>- Eficiência;</li> </ul>
<div style="text-align: center;">Sacrifício</div>	<ul style="list-style-type: none"> <li>- Preço;</li> <li>- Contextualização;</li> </ul>	<ul style="list-style-type: none"> <li>- Esforço;</li> <li>- Tempo;</li> </ul>

O sistema bancário atual, não suporta a modelação de processos de negócio, o que por si só, permite que se identifiquem facilmente diferentes benefícios no produto que se encontra em desenvolvimento progressivo (aplicação *Bankee*) e integrará o módulo de transferências interbancárias.

Destacam-se diferentes benefícios relacionados com o suporte a processos de negócio que até então eram inexistentes, a possibilidade de monitorização do processo, permitindo obter informação detalhada da atividade de negócio com o intuito de ajudar a melhorar a agilidade das operações envolvidas, otimizando assim os recursos e prevenindo desvios desnecessários.

Toda esta informação de gestão permite que haja um controlo em tempo real das atividades que compõem todo o processo, permitindo que, de uma forma rápida e sem prejuízo para as operações de negócio, se executem as adaptações necessárias com o objetivo de eliminar pontos de estagnação.

O ponto de relacionamento com estes benefícios prende-se essencialmente com a eficiência do processo de negócio e das operações de transferências interbancárias.

Paralelamente, a implementação de níveis de intervenção hierárquica durante a execução, impulsiona também a monitorização - do negócio - por parte dos intervenientes que, conseqüentemente envolve a notificação e intervenção de pessoas com diferentes cargos dentro da hierarquia, na execução de tarefas chave, prevenindo assim o risco operacional e no limite, a fraude. Estes benefícios têm pontos comuns de relacionamento na tabela (XX),

impulsionam a segurança interna e externa da instituição, elevando assim a confiança de clientes e parceiros.

Os sacrifícios relacionados com o produto, prendem-se com o custo da implementação e desenvolvimento, bem como com as ações de contextualização que terão de ser levadas a cabo pela instituição para que os seus funcionários se adaptem às novas ferramentas disponibilizadas pelo produto implementado. Estas ações, exigirão certamente esforço na adaptação, bem como tempo para que as novas funcionalidades sejam absorvidas.

## **8. Avaliação**

A avaliação da solução proposta é um ponto essencial para o culminar deste estudo. Neste capítulo serão expostos os diferentes métodos de avaliação do módulo de transferências interbancárias com intervenção hierárquica dinâmica. A avaliação será efetuada com base nos testes ao módulo realizados nos ambientes de qualidade da CSW, bem como nos ambientes de qualidade e pré-produção do cliente.

### **8.1. Testes em Ambientes de Qualidade Internos**

A avaliação em causa consistirá essencialmente num conjunto de testes funcionais realizados em ambientes estanques com o objetivo de detetar falhas no funcionamento das diferentes funcionalidades que compõe o módulo desenvolvido.

Através dos resultados destes testes será possível auferir a quantidade de falhas e anomalias introduzidas durante o desenvolvimento.

Serão realizados diferentes testes de aceitação face a vários cenários possíveis no decorrer da execução do processo de transferências interbancárias com intervenção hierárquica dinâmica. No âmbito destes testes serão validadas as vertentes especificadas nas tabelas 7 e 8.



*Tabela 15 - Especificação de testes transferências interbancárias com autorização aleatória e recusa de assinaturas*

<i>Código</i>	<i>Título</i>	<i>Descrição</i>
<b>TA01</b>	Registrar Transferência Interbancária	Este teste verifica o registo de uma transferência Interbancária eventual com autorização, mas recusa das assinaturas.
<b>TA02</b>	Autorizar Transferência Interbancária - recusar assinaturas	Este teste verifica a autorização de uma transferência Interbancária eventual, onde se recusa das assinaturas.
<b>TA03</b>	Registrar Transferência Interbancária após recusa das assinaturas	Este caso de teste verifica o registo de uma transferência Interbancária eventual após a recusa das assinaturas.
<b>TA04</b>	Autorizar Transferência Interbancária - confirmar assinaturas	Este teste verifica a autorização online de uma transferência OIC MN eventual, com autorização da transferência e confirmação das assinaturas. Com este caso de teste o fluxo do processo de transferência, com autorização da transferência, mas recusa das assinaturas, termina.

Tabela 16 - Especificação de testes. Processo Transferência Interbancária com recusa na autorização

<i>Código</i>	<i>Título</i>	<i>Descrição</i>
<i>TB01</i>	Registrar Transferência Interbancária (com autorização)	Este teste verifica o registo de uma transferência Interbancária eventual com autorização.
<i>TB02</i>	Autorizar Transferência Interbancária- recusar	Este teste verifica a autorização de uma transferência Interbancária eventual. Neste caso é testado a recusa da transferência.
<i>TB03</i>	Registrar Transferência Interbancária (após recusa)	Este caso de teste verifica o registo de uma transferência Interbancária eventual após a recusa da transferência na autorização.
<i>TB04</i>	Autorizar Transferência Interbancária - autorizar	Este teste verifica a autorização de uma transferência Interbancária eventual, com autorização da transferência após a recusa. Com este caso de teste o fluxo do processo de transferência com recusa da transferência e posterior autorização termina.

Os casos de teste especificados anteriormente preveem a inclusão de somente um nível de intervenção hierárquica, autorização (ou aprovação), aplicado de forma aleatória e permanente. É, contudo, necessário prever cenários de teste para a implementação de outros níveis de intervenção hierárquica. Estes serão descritos nas tabelas 9 e 10.

*Tabela 17 - Especificação de testes. Transferência Interbancária com diferentes níveis de intervenção hierárquica.*

<i>Código</i>	<i>Título</i>	<i>Descrição</i>
<b>TC01</b>	Registrar Transferência Interbancária (com diferentes níveis de intervenção hierárquica)	Este teste verifica o registo de uma transferência Interbancária eventual com associação de documentos e três níveis de intervenção: autorização, validação e autorização.
<b>TC02</b>	Associação Documentos e Confirmação de Pedido	Este teste verifica a associação de documentos e confirmação de pedido de uma transferência Interbancária eventual com associação de documentos e três níveis de intervenção: autorização, validação e autorização.
<b>TC03</b>	Autorizar Transferência com documentos - aprovar	Este teste verifica a autorização de uma transferência Interbancária eventual. Neste caso é testada a aprovação da transferência.
<b>TC04</b>	Validar Transferência Interbancária	Este teste verifica a validação de documentos de uma transferência Interbancária eventual. Neste caso é

*Tabela 17 - Especificação de testes. Transferência Interbancária com diferentes níveis de intervenção hierárquica.*

		testado a validação dos documentos da transferência, sem devolução de nenhum dos documentos.
<b>TC05</b>	Autorizar Transferência com documentos - aprovar	Este teste verifica a autorização de uma transferência Interbancária eventual. Neste caso é testado a aprovação da transferência. Após este caso de teste o fluxo do processo termina.

*Tabela 18 - Testes de aceitação para o fluxo do processo com diferentes níveis de intervenção, incluindo a dada de parecer desfavorável e favorável*

<i>Código</i>	<i>Título</i>	<i>Descrição</i>
<i>TD01</i>	Registrar Transferência OIC MN	Este teste verifica o registo de uma transferência OIC MN eventual com aprovação, validação e devolução de documentos na validação.
<i>TD02</i>	Associação de Documentos e Confirmação de Pedido	Este teste verifica a associação de documentos e a confirmação de pedido de uma transferência OIC MN eventual com aprovação, validação e devolução de documentos na validação.
<i>TD03</i>	Autorizar Transferência com documento - aprovar	Este teste verifica a autorização online de uma

*Tabela 18 - Testes de aceitação para o fluxo do processo com diferentes níveis de intervenção, incluindo a dada de parecer desfavorável e favorável*

		transferência OIC MN eventual. Neste caso é testado a aprovação da transferência.
<b>TD04</b>	Validar Transferência OIC MN - devolver	Este teste verifica a validação de documentos de uma transferência OIC MN eventual. Neste caso é testado a validação dos documentos da transferência, com devolução de um dos documentos.
<b>TD05</b>	Registrar Transferência OIC MN	Este teste verifica o registo de uma transferência OIC MN eventual com aprovação e validação, após a devolução na validação da transferência. Este caso de teste inclui também a atividade associação de documentos e confirmação de pedido.
<b>TD06</b>	Tomar Conhecimento da Transferência	Este teste verifica a tomada de conhecimento de uma transferência OIC MN eventual com aprovação, validação e devolução de documentos na validação. Este caso de teste ocorre após a devolução na validação, onde os níveis de intervenção foram alterados para dois níveis, respetivamente a tomada de

*Tabela 18 - Testes de aceitação para o fluxo do processo com diferentes níveis de intervenção, incluindo a dada de parecer desfavorável e favorável*

		conhecimento da transferência e o dar parecer da transferência.
<i>TD07</i>	Dar Parecer de Transferência-desfavorável	Este teste verifica o nível de intervenção Dar Parecer de uma transferência OIC MN eventual, com parecer desfavorável. Inicialmente os níveis de intervenção foram aprovação e validação (devolução de documentos na validação). Este caso de teste ocorre após a devolução na validação. Ao voltar ao registo de transferência, os níveis de intervenção foram alterados para dois níveis de intervenção, respetivamente a tomada de conhecimento da transferência e o dar parecer de transferência. Neste caso de teste o parecer é desfavorável e o processo volta ao registo de transferência, onde se vai definir um valor a transferir com um limite de autorização associado, com um nível de intervenção: Dar Parecer.
<i>TD08</i>	Dar Parecer de Transferência-favorável	Este teste verifica o nível de intervenção “Dar Parecer” de uma transferência interbancária eventual. Inicialmente os níveis de intervenção foram aprovação, validação

*Tabela 18 - Testes de aceitação para o fluxo do processo com diferentes níveis de intervenção, incluindo a dada de parecer desfavorável e favorável*

(devolução de documentos na validação). Após a devolução o valor a transferir é alterado para um valor com um limite de autorização associado a dois níveis de intervenção: tomada de conhecimento e dar parecer. Este caso de teste ocorre após o dar parecer desfavorável. Ao voltar ao registo de transferência, os níveis de intervenção foram alterados para um nível de intervenção: o dar parecer de transferência. Com este caso de teste o fluxo aqui testado termina.

## **8.2. Testes de Aceitação do Cliente**

A metodologia utilizada com os testes de aceitação realizados pelo cliente nos seus ambientes é em tudo idêntica aos testes nos ambientes de qualidade da CSW. A diferença reside no facto da especificação ser efetuada pela equipa do cliente.

O acesso à especificação de testes do cliente, por motivos de garantia de uma análise e avaliação independente de fatores externos, está vedado a elementos externos, incluindo qualquer equipa da CSW.

No caso de falha, é aberto um *Issue* externo, no sistema JIRA da Atlassian, que será avaliado pela equipa de qualidade da CSW de forma a confirmar que o defeito se verifica em conformidade com a análise funcional. Verificando-se que sim, é aberto um *Issue* interno, sendo atribuída uma prioridade que pode ser do tipo “*Trivial*”, “*Low*”, “*Medium*”, “*High*” ou “*Blocker*”, dependendo do grau crescente de severidade verificada. Assim, poderá ser analisado e resolvido pela equipa de desenvolvimento.

## 8.3. Avaliação

A avaliação em cada um destes âmbitos terá como base o tratamento estatístico dos resultados obtidos pelos testes de aceitação aplicados em cada um dos universos referidos.

Pretende-se avaliar a quantidade de defeitos introduzidos durante o desenvolvimento, e identificados nos ambientes de qualidade internos, correlacionando com o tempo gasto na sua resolução, assumindo uma perspetiva de gestão, obtendo dados analíticos sobre o impacto destes no custo para o projeto.

Com base nos defeitos abertos pelo cliente, será possível extrapolar um grau de satisfação relativa.

É ainda importante reforçar que a solução desenvolvida no âmbito deste documento terá, obrigatoriamente, de cumprir os requisitos necessários que lhe permitam entrar nos ambientes de produção do cliente. Em última análise, o que define a qualidade da solução desenvolvida é esta aprovação que no momento já foi atribuída pelo cliente à Critical Software, validando assim a viabilidade da solução desenvolvida.

### 8.3.1. Avaliação Ambiente Qualidade Interno

A avaliação no ambiente de qualidade interno tem como base o resultado obtido pela equipa de testes e qualidade do projeto aquando da execução dos testes descritos na secção 8.1.

O procedimento descrito nas normas internas de qualidade da Critical Software prevê que no momento de deteção de defeitos seja aberto um *issue* do tipo *defect* na plataforma Jira <sup>36</sup> da Atlassian onde é descrito o problema, assinalada uma prioridade para a sua resolução e atribuído ao elemento da equipa com responsabilidade na mesma. Na Figura 68 pode ser visualizado o exemplo de um problema (*issue*) do tipo *Defect* com prioridade *Blocker* na plataforma Jira. Os defeitos têm 5 níveis crescentes de prioridade sendo estes *trivial*, *low*, *medium*, *high* e *blocker* respetivamente.

A plataforma Jira fornece ainda um conjunto de ferramentas que permitem fazer um levantamento estatístico dos defeitos abertos para cada módulo, bem como o número de horas gasto na sua resolução e conseqüentemente o impacto que estes significam para o projeto.

---

<sup>36</sup> <https://www.atlassian.com/software/jira>



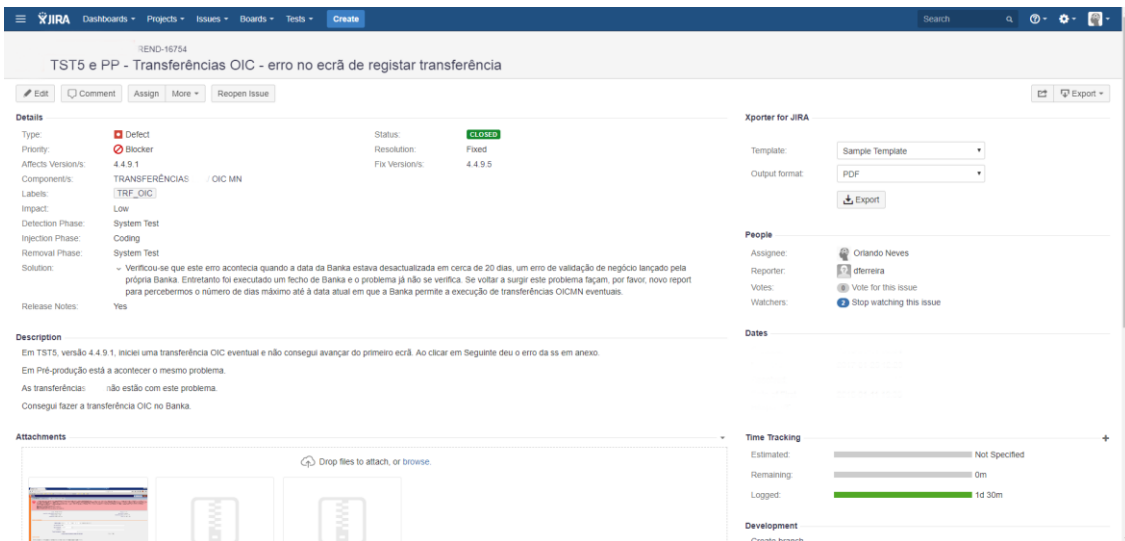


Figura 68 – Exemplo de um defeito blocker

O gráfico representado na Figura 69, obtido através de uma filtragem por query no Jira, mostra que foram abertos um total de 48 defeitos no âmbito dos testes de qualidade realizados. Destes, a maior percentagem (~35%) concentra-se nos defeitos de prioridade baixa (*Low*), seguidos de defeitos com prioridade média (~25%), prioridade alta (~23%) e bloqueantes (~15%).

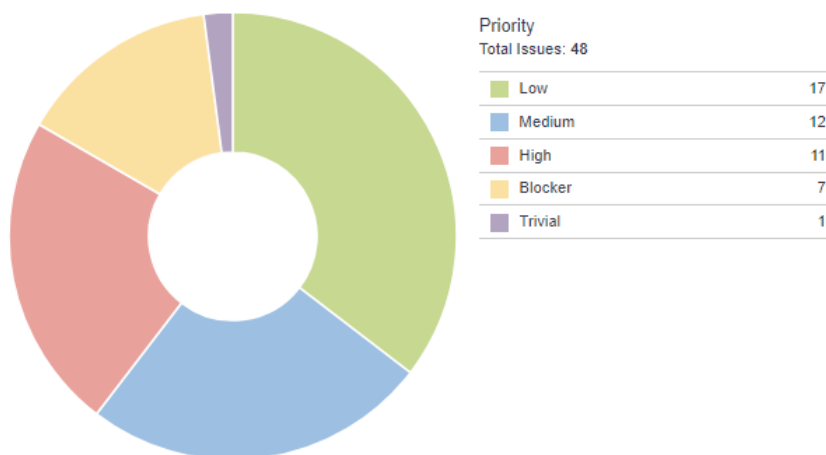


Figura 69 – Total de defeitos do módulo por prioridade

A distribuição normal de defeitos do módulo por prioridade demonstrada no gráfico de barras da Figura 70 mostra uma pirâmide de prioridade crescente. Este é um bom indicador pois o

número de defeitos é inversamente proporcional à prioridade, significando que existe um número de defeitos a diminuir à medida que aumenta a prioridade.

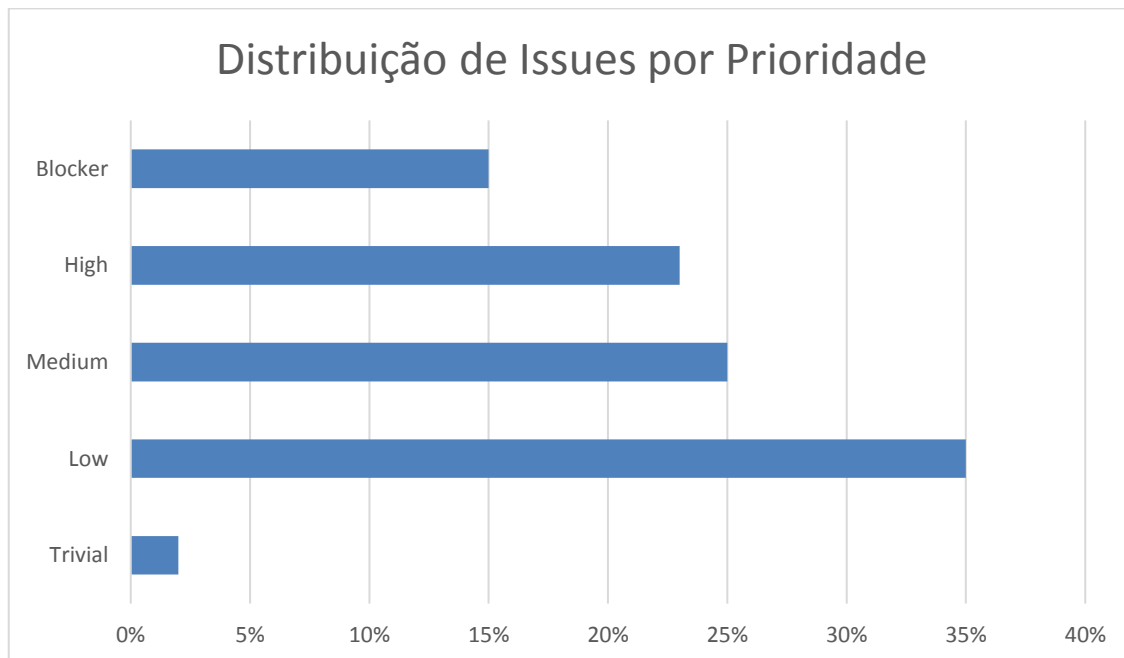


Figura 70 – Distribuição normal de defeitos por prioridade

A existência de defeitos após o processo de desenvolvimento de software é normal, no entanto o número e tempo necessário para a sua resolução poderão determinar custos não estimados que resultarão em desvios ao plano financeiro inicial.

É considerado um desvio de 35%, relativo ao tempo de execução inicialmente estimado, para a resolução de *bugs* e outros problemas que possam resultar do processo de desenvolvimento da solução.

Para o módulo de transferências interbancárias foi estimado um tempo de desenvolvimento, resultante da soma da estimativa de desenvolvimento para cada um dos componentes da arquitetura, 611,5 horas que significa cerca de 76 dias de trabalho em ciclos de 8 horas.

A Figura 71, retirada dos dados obtido no Jira, mostra que foram gastas aproximadamente 5 semanas 1 dia e 7 horas no total para a resolução de problemas encontrados na aplicação. Este valor traduz-se em 215 horas de trabalho, correspondendo a 35,16% do valor total estimado para o desenvolvimento da solução. Este facto significa que houve um ligeiro desvio de 0.16% ao originalmente estimado, significando que o projeto teve de assumir o custo de, aproximadamente, 1 hora de trabalho aplicada ao desenvolvimento de *bugs* na solução desenvolvida.

Apesar de se tratar de um desvio, considera-se este valor residual, assumindo que o número de horas gasto na resolução de problemas foi bem ajustado face à estimativa inicial.

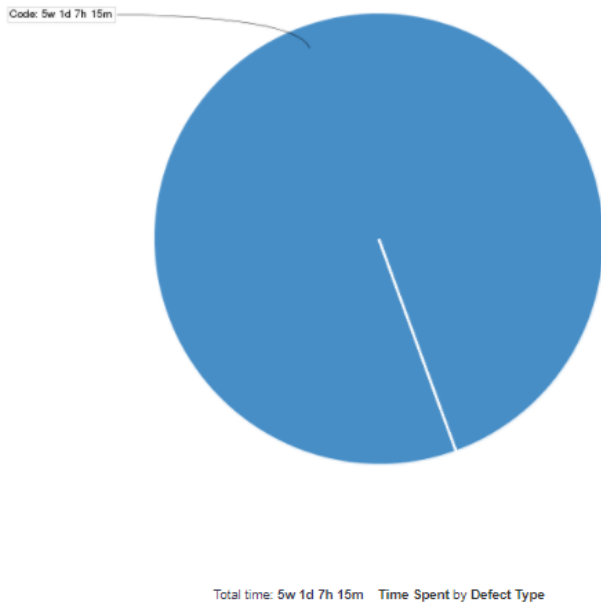


Figura 71 – Horas gasto na resolução de problemas

A Figura 72 resulta da fragmentação da Figura 71 para que seja possível analisar com maior granularidade o tempo gasto na reolução de *bugs* em cada nível de prioridade. Desta forma é possível obter uma análise mais detalhada desta métrica

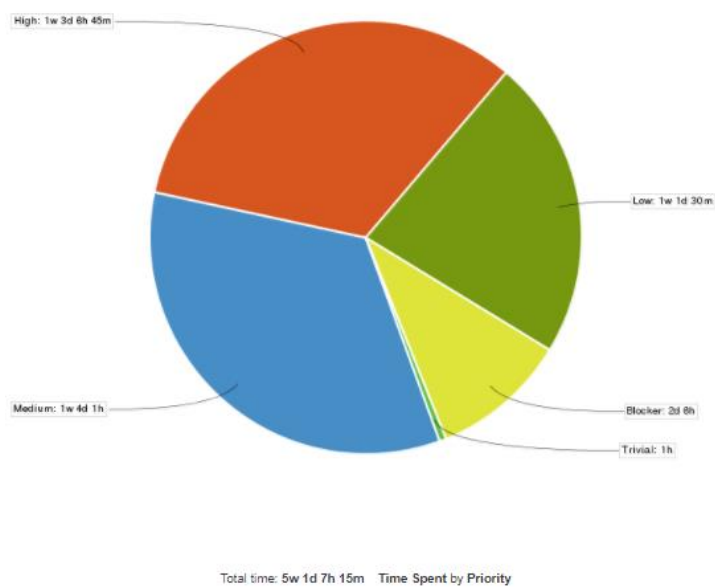


Figura 72 – Tempo de resolução de *issues* por prioridade

### 8.3.2. Avaliação no Cliente

Na sequência lógica de validação do módulo desenvolvido, após resolução de todos os problemas encontrados e referenciados na secção 8.3.1, o módulo foi instalado no ambiente de qualidade do cliente com a finalidade de ser submetido à bateria de testes de aceitação especificados pelas equipas de controlo de qualidade.

Este é um passo muito importante no primeiro nível de aceitação do cliente. O segundo nível passa por instalar o módulo num ambiente de pré-produção para se verificar a robustez e coerência no comportamento antes de entrar em produção.

Sempre que surge um problema nesta fase é aberto um *issue* externo no Jira que é validado por um, ou mais elementos da equipa de testes e qualidade da Critical Software para verificar a validade o problema reportado e fornecerem mais informação sobre a origem do problema.

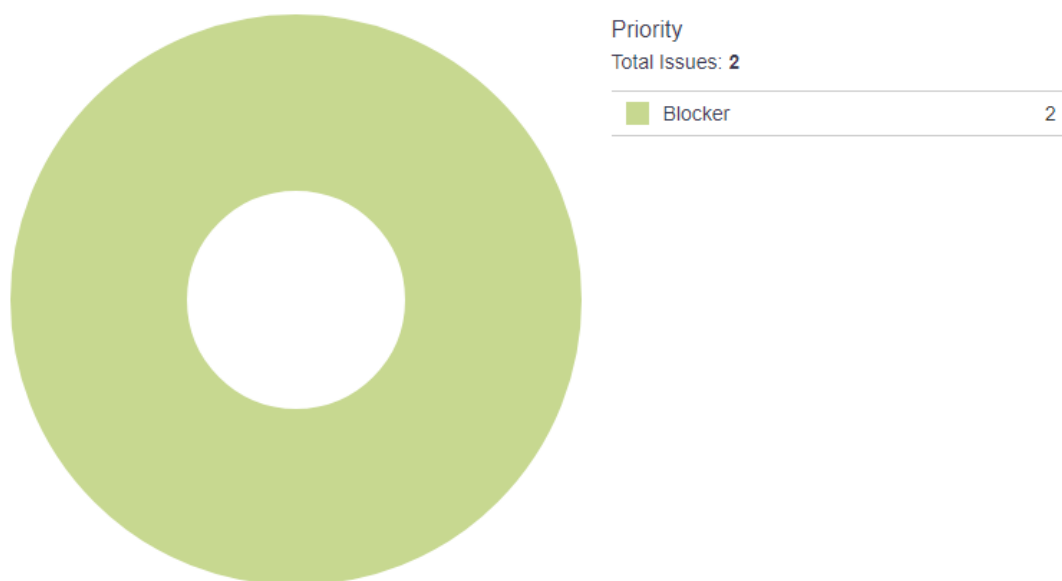


Figura 73 – Quantidade de issues externos por prioridade

Durante a avaliação realizada pelo cliente, tal como mostra a Figura 73, foram abertos 2 defeitos de prioridade *blocker*. Os defeitos em causa foram validados e face à prioridade atribuída, resolvidos no imediato.

Considerando que os 2 defeitos levantados pelo cliente são todos do mesmo nível de prioridade, a Figura 74 mostra um gráfico homogéneo que representa o tempo total gasto na resolução dos defeitos externos abertos.

Foram gastos 2 dias, 5 horas e 30 minutos na resolução destes dois problemas. Este tempo irá somar ao tempo total de desvio referido na secção 8.3.1 contribuindo para o deslize face à estimativa que aqui foi apontado.

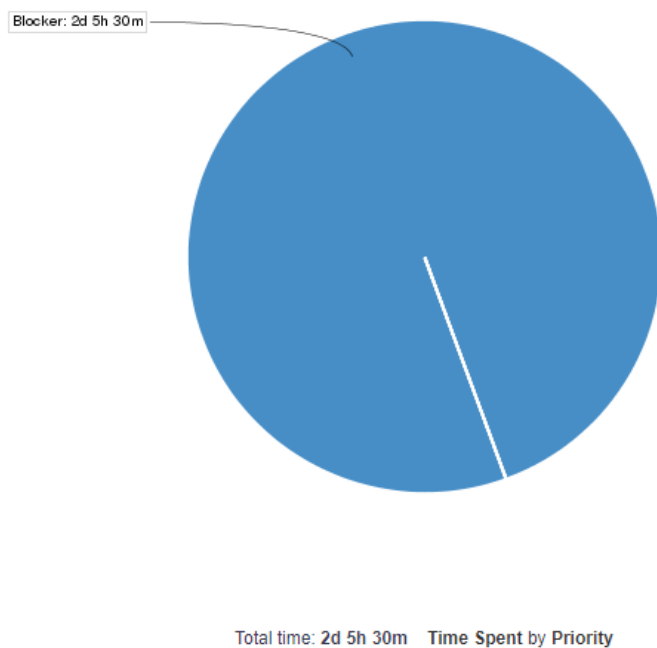


Figura 74 – Tempo total de resolução dos issues externos

Nesta fase de avaliação não é expectável que sejam encontrados muitos problemas, sendo que a inexistência destes é o cenário mais favorável para o projeto. Considera-se, contudo, que o número de issues externos aqui relatados não é significativo para os objetivos de gestão do projeto, considerando-se o desvio de 3,7% de tempo a mais do que o inicialmente estimado um valor sem peso significativo para os valores previstos pela gestão de projeto.

## 9. Conclusão

No presente capítulo é realizado um resumo sobre o trabalho realizado no âmbito deste estudo, realçando os objetivos alcançados, contributos, limitações e trabalho futuro.

### 9.1. Objetivos Alcançados

Com o presente trabalho foi possível cumprir, na íntegra, com os objetivos propostos na secção 1.3. Tal como expectável, realizou-se um estudo na área de domínio de BPM com o objetivo de propor uma solução para o desenvolvimento do processo de transferências interbancárias em moeda nacional com a integração do módulo de limites de autorização para a configuração e cálculo de níveis de intervenção hierárquicos dinâmicos durante a execução deste.

Gradualmente, partindo da primeira modelação do processo, e em conjunto com o cliente foi realizado um levantamento das necessidades em caderno de requisitos e modelado o processo de negócio percursor. A partir deste ponto foi possível seguir um conjunto de práticas de engenharia de software no desenvolvimento, modelação e validação da solução final. Estas práticas que incluem o cumprimento de padrões e recomendações no desenvolvimento de ecrãs de UI (*user interface*) e *web services*.

Como resultado, é possível iniciar e concluir com sucesso processos de transferências interbancárias com diferentes valores e conseqüentemente diferentes intervenientes na intervenção hierárquica obrigatória ao longo do processo e em pontos chave do mesmo.

### 9.2. Contributos

A Tabela 19 descreve as formas de conhecimento resultantes do trabalho desenvolvido, descritas neste documento e que são transferíveis como modelos, métodos, dados e software, entre outros. O processo de identificação destes ativos intelectuais seguiu a proposta do instituto Chalmers Innovationskontor<sup>37</sup>.

---

<sup>37</sup> <https://innovationskontor.chalmers.se/en/content/intellectual-assets-inventory>

Tabela 19. Inventário de ativos intelectuais

<b>Name do ativo</b>	<b>Tipo</b>	<b>Proprietário</b>	<b>Descrição</b>	<b>Relação com outros ativos</b>	<b>Descrição da utilização</b>
Módulo de Transferências Interbancárias	Software	Critical Software	Módulo de software desenvolvido em JAVA, com modelação de processo em BPM e integrante de um sistema de gestão bancário, aplicação Bankee.	Modelo de Transferências Interbancárias	Utilizado na execução de operações de transferências de valores entre instituições bancárias distintas.
Modelo de Transferências Interbancárias	Design	Critical Software	Modelo de processo de negócio de transferências interbancárias em moeda nacional com integração de sub-processo que integra a funcionalidade de intervenção hierárquica dinâmica.	Módulo de Transferências Interbancárias	Usado na implementação do processo de negócio, agregando todas as atividades e tarefas necessárias.

### 9.3. Limitações e Trabalho Futuro

Face aos objetivos propostos, as principais limitações encontradas relacionam-se com vários fatores, nomeadamente:

- Desenvolvimento de uma solução à medida para um cliente específico. Este fator não permitiu que pudessem ser exploradas funcionalidades diferentes das que foram previamente identificadas em caderno de requisitos.

- A solução ser parte integrante de uma aplicação em desenvolvimento que limita a utilização de tecnologias às que estão em uso.
- Abordagens de integração da aplicação Bankee com o sistema de BPM da IBM que limitou em parte a possibilidade de explorar algumas potencialidades da ferramenta, nomeadamente no desenvolvimento de ecrãs de UI.
- Elevado sentido de responsabilidade e compromisso no desenvolvimento de uma solução com uma timeline identificada e data determinada para entrada em produção, factor que não permite perder o foco, sair “fora da caixa” e explorar diferentes perspetivas.

No que compete ao trabalho futuro, também este está diretamente limitado pelas exigências do cliente face a novas funcionalidades dentro do processo de negócio.

Seria interessante aplicar alguns dos princípios identificados no capítulo 2, nomeadamente a transformação de atividades recorrentes em aspetos, seguindo o princípio de *aspect oriented BPM*.

## 9.4. Apreciação final

Considerando os objetivos inicialmente definidos balanceando com as limitações impostas, o resultado alcançado é francamente positivo. Foram cumpridos todos os pontos exigidos pelo cliente para o fluxo do processo de transferências interbancárias, incluída a funcionalidade de intervenção hierárquica dinâmica no fluxo do processo integrando o módulo de limites de autorização para que funcionalmente permitisse configurar o comportamento do processo com base no valor da operação.

Os testes e avaliação do módulo demonstraram um ligeiro desvio face ao previamente estimado, no entanto sem perdas relevantes para a gestão do projeto e com aval positivo do cliente, estando prevista a entrada em produção para um futuro próximo.



# Referências Bibliográficas

- [1] C. H. Usirono, “Engenharia e Análise de Valor”, Euax 2015, [Online]. Disponível: [http://www.euax.com.br/wp-content/uploads/2015/04/ebook\\_2015\\_010\\_Thiago\\_16\\_Paginas.pdf](http://www.euax.com.br/wp-content/uploads/2015/04/ebook_2015_010_Thiago_16_Paginas.pdf)
- [2] L. D. Miles, “Techniques of Value Analysis and Engineering: 3<sup>rd</sup> Edition”, Lawrence D. Miles Value Foundation, Março 9, 2015
- [3] W. Ulaga, A. Eggert, “Value-Based Differentiation in Business Relationships: Gaining and Sustaining Key Supplier Status”. *Journal of Marketing*: Janeiro 2006, Vol. 70, No. 1, pp. 119-136
- [4] Williams, S. (1967) "Business Process Modeling Improves Administrative Control" In: *Automation*. December, 1967, pp. 44 - 50.
- [5] W.M. van der Aalst, M. La Rosa, F. M. Santoro, “Business Process Management: Don't Forget to Improve the Process!”, Springer Fachmedien Wiesbaden 2015, Published online: January 4, 2016
- [6] Matthias Geiger, Simno Harrer, Jörg Lenhard, Guido Wirtz “BPMN 2.0: The state of support and implementation”, January 7, 2017
- [7] A. Jalali, P. Wohed, and C. Ouyang. “Dynamic Weaving of Aspects for Business Process Management Systems.” Technical report, Dept. of Computer and Systems Sciences, Stockholm University, March 2012.
- [8] A. Jalali, P. Wohed, and C. Ouyang. “Aspect Oriented Business Process Modelling with Precedence.” Technical report, Dept. of Computer and Systems Sciences, Stockholm University, 2012.
- [9] Anis Charfi, Heiko Witteborg, Mira Mezini. “Aspect-Oriented Business Process Modeling with AO4BPMN”, in Proc. of 6th European Conference on Modelling Foundations and Applications (EC-MFA), LNCS 6138, pp. 48-61, Springer, Paris, France, June 2010.
- [10] Nathaniel Palmer. “What Is BPM” 16 December 2016 [Online], Available at: <http://bpm.com/what-is-bpm>
- [11] ABPMP (Ed.). (2013).” Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento” (BPM CBOK V3.0) (1ª ed.).
- [12] T. C. E. Cheng, Ida S. F. Chiu. “Critical Success Factors of Business Process Re-engineering in the Banking Industry”. *Knowledge and Process Management* Volume 15 Number 4 pp 258-269 (2008). Department of Logistics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
- [13] James F. Chang. “Business Process Management Systems: Strategy and Implementation”. Auerbach Publications, Taylor & Francis Group, New York, USA, 2006.

- [14] Dimitris Karagiannis. "BPMS: Business Process Management Systems". *ACM SIGOIS Bulletin – Special issue: business process reengineering* Volume 16 Issue 1, New York, USA, Aug. 1995.
- [15] Wil M. P. Van Der Aalst. "Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management". *Lecture Notes in Computer Science*, pp 1-65, Springer Berlin Heidelberg, 2004.
- [16] C.A. Ellis. Information Control Nets: A Mathematical Model of Office Information Flow. In *Proceedings of the Conference on Simulation, Measurement and Modeling of Computer Systems*, pages 225–240, Boulder, Colorado, 1979. ACM Press.
- [17] A. W. Holt. Coordination Technology and Petri Nets. In G. Rozenberg, editor, *Advances in Petri Nets 1985*, volume 222 of *Lecture Notes in Computer Science*, pages 278–296. Springer-Verlag, Berlin, 1985.
- [18] M.D. Zisman. *Representation, Specification and Automation of Office Procedures*. PhD thesis, University of Pennsylvania, Warton School of Business, 1977.
- [19] "A core subset of BPMN elements. - Figure 1 of 22". Acedido a 2 de Janeiro de 2017 [Online], Disponível em: [https://www.researchgate.net/figure/228668926\\_fig1\\_Figure-1-A-core-subset-of-BPMN-elements](https://www.researchgate.net/figure/228668926_fig1_Figure-1-A-core-subset-of-BPMN-elements)
- [20] "ICEfaces JSF Framework Overview - ICEsoft Technologies" January 4, 2017 [Online], Available at: <http://www.icesoft.org/java/projects/ICEfaces/overview.jsf>
- [21] "Free JSF Library For Dynamic UIs: OpenFaces - Web Resources Depot" January 5, 2017 [Online], Available at: <https://webresourcesdepot.com/free-jsf-library-for-dynamic-uis-openfaces/>
- [22] "ACE Components - ICEsoft Technologies" January 5, 2017 [Online], Available at: <http://www.icesoft.org/java/projects/ICEfaces/ace-components.jsf>
- [23] "ICEmobile Components for Rich Mobile Web Applications - ICEsoft Technologies", January 5, 2017 [Online], Available at: <http://www.icesoft.org/java/projects/ICEfaces/mobi-components.jsf>
- [24] "PrimeFaces the number 1 framework – DevRates.com" January 5, 2017 [Online], Available at: <http://www.devrates.com/project/show/89675/PrimeFaces>
- [25] "MVC Architecture - Google Chrome" January 7, 2017 [Online], Available at: [https://developer.chrome.com/apps/app\\_frameworks](https://developer.chrome.com/apps/app_frameworks)
- [26] Massey, Simon. "Implementing event-driven GUI patterns using the ZK Java AJAX framework". 5 July, 2014 [Online], Available at: <http://www.ibm.com/developerworks/websphere/zones/portal/proddoc/zkjavaajax/>
- [27] "TIOBE Index | TIOBE - The Software Quality Company" January 11, 2017 [Online], Available at: <http://www.tiobe.com/tiobe-index>
- [28] "C++ Programming Language" January 15, 2017 [Online], Available at: <https://www.techopedia.com/definition/26184/c-programming-language>

- [29] “A list of open source C++ libraries” January 15, 2017 [Online], Available at: <http://en.cppreference.com/w/cpp/links/libs>
- [30] “Java Virtual Machine's Internal Architecture” January 16, 2017 [Online], Available at: <http://www.artima.com/insidejvm/ed2/jvm2.htmls>
- [31] “Best Business Process Management Software | 2017 Reviews of the Most Popular Systems” January 17, 2017 [Online], Available at: <http://www.capterra.com/business-process-management-software/>
- [32] “Compare Oracle BPM vs IBM BPM | IT Central Station” January 18, 2017 [Online], Available at: [https://www.itcentralstation.com/products/comparisons/ibm-bpm\\_vs\\_oracle-bpm](https://www.itcentralstation.com/products/comparisons/ibm-bpm_vs_oracle-bpm)
- [33] Niehaves, B., & Henser, J. “Business Process Management beyond Boundaries? – A Multiple Case Study Exploration of Obstacles to Collaborative BPM.” Proceedings of the 44th Hawaii International Conference on System Sciences, 2011.
- [34] Bhasin, J., & Parrey, A. H. “Correlating Business Process Management and Organizational Performance: A Case Study of J&K Bank” Jammu & Kashmir, India, 2013.
- [35] Potix Corporation, “ZK v. Ajax JSF Performance Report”, Julho 2012
- [36] “Optimizing FEI - Consortium for Corporate Entrepreneurship” 20 de fevereiro 2017 [Online], Available at: <https://web.stevens.edu/cce/NEW/optFEI.htm>
- [37] Banco de Cabo Verde, “O que é o dinheiro – Caderno nº2.”, Praia – Cabo Verde, 2012
- [38] “Evolution of technology in the banking sector” february 21, 2017 [Online], Available at: <http://gulfnnews.com/business/sectors/banking/evolution-of-technology-in-the-banking-sector-1.1905437>
- [39] “When was the first computer invented?” february 21, 2017 [Online], Available at: <http://www.computerhope.com/issues/ch000984.htm>
- [40] Prerna Kumar, “Perceptual Differences between Customers and Employees with respect to Corporate Image of Banks: Na Empirical Study”. Indra Publishing House, India, November 1, 2015
- [41] “Big four' banks cut 189,000 jobs worldwide in five years | Business | The Guardian” february 21, 2017 [Online], Available at: <https://www.theguardian.com/business/2013/may/28/big-four-banks-cut-jobs>
- [42] Lee, M.-C. (2009). Factors influencing the adoption of Internet banking: An inte-gration of TAM and TPB with perceived risk and perceived benefit. *ElectronicCommerce Research and Applications*, 8(3), 130–141.
- [43] “Which MENA Countries Lead in Digital Banking Adoption? ArabNet Digital Banking Report” February 21, 2017 [Online], Available at: <http://news.arabnet.me/mena-countries-lead-digital-banking-adoption-arabnet-report/>
- [44] Angola. Aviso N.º 09/2015. “Sistema de Pagamentos – Prazo de execução e de disponibilização de fundos aplicáveis aos movimentos de depósitos à ordem, transferências e remessas de valores”. *Diário da República*, I série, nº 53, de 20 de Abril.

- [45] Geremias, Rosa H. L., "O CRESCIMENTO DO SECTOR BANCÁRIO E A CONCESSÃO DE CRÉDITO AS EMPRESAS. O CASO DE ANGOLA." Tese (Mestrado em Ciências Empresariais) – Lisboa School of Economics & Management. Junho, 2014.
- [46] "Banco Nacional de Angola - O Banco" março 4, 2017 [Online], Available: [http://www.bna.ao/Conteudos/Artigos/detalhe\\_artigo.aspx?idc=139&idsc=168&idl=1](http://www.bna.ao/Conteudos/Artigos/detalhe_artigo.aspx?idc=139&idsc=168&idl=1)
- [47] OMG, Business process model and notation (BPMN 2.0), formal/2011-01-03, OMG, <http://www.omg.org/spec/BPMN/2.0> (Janeiro 2011), pp. 124-131
- [48] ".NET vs Java: How to Make Your Pick | Segue Technologies" July 14, 2017 [Online], Available at: <http://www.seguetech.com/net-vs-java/>
- [49] ".NET Framework - Wikipedia" July 15, 2017 [Online], Available at: [https://en.wikipedia.org/wiki/.NET\\_Framework](https://en.wikipedia.org/wiki/.NET_Framework)
- [50] ".NET vs Java: How to Make Your Pick | Segue Technologies" July 29, 2017 [Online], Available at: <http://www.seguetech.com/net-vs-java/>
- [51] "JAVA EE Productivity Report 2011", November 2, 2011 [Online], Available at: [https://zeroturnaround.com/wp-content/uploads/2010/11/Java\\_EE\\_Productivity\\_Report\\_2011\\_finalv2.pdf](https://zeroturnaround.com/wp-content/uploads/2010/11/Java_EE_Productivity_Report_2011_finalv2.pdf)
- [52] "Waterfall | Universo Projeto", August 21, 2017 [Online], Available at: <https://universoprojeto.wordpress.com/tag/waterfall/>
- [53] "Suspeitos de fraude de 100 milhões de dólares do Banco Nacional de Angola entregam-se - Banca & Finanças - Jornal de Negócios", September 10 2017 [Online], Available at: [http://www.jornaldenegocios.pt/empresas/banca---financas/detalhe/suspeitos\\_de\\_fraude\\_de\\_100\\_milhoes\\_de\\_dolares\\_do\\_banco\\_nacional\\_de\\_angola\\_entregam\\_se](http://www.jornaldenegocios.pt/empresas/banca---financas/detalhe/suspeitos_de_fraude_de_100_milhoes_de_dolares_do_banco_nacional_de_angola_entregam_se)
- [54] "Fraude - Investigação de fraude já recuperou quantias avultadas", September 10 2017 [Online], Available at: <http://www.dn.pt/economia/interior/investigacao-de-fraude-ja-recuperou-quantias-avultadas-1868599.html>
- [55] W.M. van der Aalst, "Business Process Management as the Killer App for Petri Nets". Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands;
- [56] W.M. van der Aalst, "Business Process Management: A Comprehensive Survey". Department of Mathematics and Computer Science, Technische Universiteit Eindhoven, 5612 AZ Eindhoven, The Netherlands, 2012.
- [57] Anni Tsai et al. (2006). "EPC Workflow Model to WIFA Model Conversion". In: 2006 IEEE International Conference on Systems, Man, and Cybernetics, Taipei, Taiwan, pp. 2758-2763].
- [58] "EPC vs. BPMN - the perfect flamewar | ARIS BPM Community", September 20, 2017 [Online], Available at: <http://www.ariscommunity.com/users/sstein/2010-04-15-epc-vs-bpmn-perfect-flamewar>

- [59] Markus, L.M., Benjamin, R.I., 1997. The magic bullet theory in IT-enabled transformation. *Sloan Management Review*, 55–68.
- [60] Davenport, T., Short, J., 1990. The new industrial engineering: Information technology and business process redesign. *Sloan Management Review* 31 (41), 11–27.
- [61] Terziovski M., Fitzpatrick P., O’Neill P., 2002. Successful predictors of business process reengineering (BPR) in financial services.
- [62] Cole, R., 1994. Reengineering the corporation: A review essay. *Quality Management Journal* 1 (4), 77–85.
- [63] Sheehy, B., 1997. Quality comeback. *Executive Excellence* 14 (5), 7–8.
- [64] J. Freund and B. Rücker, *Real-life BPMN: Using BPMN 2.0 to analyze, improve, and automate processes in your company*. Camunda, 2012
- [65] N. Russell, A.H.M. ter Hofstede, W.M.P. van der Aalst, and N. Mulyar. *Workflow Control-Flow Patterns: A Revised View*. BPM Center Report BPM-06-22, BPMcenter.org, 2006.
- [66] “Workflow Patterns | Patterns | Control”, October 6 2017 [Online] Available at: <http://www.workflowpatterns.com/patterns/control/>
- [67] “www.bpmb.de”, October 10 2017 [Online] Available at: [http://www.bpmb.de/images/BPMN2\\_0\\_Poster\\_PT.pdf](http://www.bpmb.de/images/BPMN2_0_Poster_PT.pdf)
- [68] “SOA Patterns | Design Patterns | Service Encapsulation”, October 20 2017 [Online] Available at: [http://soapatterns.org/design\\_patterns/service\\_encapsulation](http://soapatterns.org/design_patterns/service_encapsulation)