

UERJ

Dissertação de Mestrado em Engenharia de Computação
Área de Concentração em Geomática

**SISTEMA DE INFORMAÇÃO GEOGRÁFICA PARA APOIO À
TOMADA DE DECISÃO NO SETOR DE COBRANÇA –
SIGATDSC**

Autora: Fabiana Viana Salmaso

Orientador: Orlando Bernardo Filho
Co-orientador: João Araujo Ribeiro

Programa de Pós-Graduação em Engenharia de Computação
Área de Concentração em Geomática

Dezembro - 2007



Faculdade de Engenharia

SISTEMA DE INFORMAÇÃO GEOGRÁFICA PARA APOIO À TOMADA DE DECISÃO NO SETOR DE COBRANÇA – SIGATDSC

Fabiana Viana Salmaso

Dissertação submetida ao corpo docente da Faculdade de Engenharia da Universidade do Estado do Rio de Janeiro – UERJ, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Computação – área de concentração Geomática.

Orientador: Orlando Bernardo Filho, D.Sc.

Co-orientador: João Araújo Ribeiro, Dr.

Programa de Pós-Graduação em Engenharia de Computação – Área de Concentração Geomática

Rio de Janeiro
Dezembro – 2007

SALMASO, FABIANA VIANA

Sistema de Informação Geográfica para Apoio à
Tomada de decisão no Setor de Cobrança – SIGATDSC
[Rio de Janeiro] 2007

xi, 114 p. 29,7 cm, (FEN/UERJ, M.Sc.,
Engenharia de Computação – Área de Concentração
Geomática, 2007)

Dissertação – Universidade do Estado do Rio de
Janeiro, FEN

1. Sistema de Informação Geográfica
2. Cobrança
3. Lógica Nebulosa

I. FEN/UERJ II. Título

FOLHA DE JULGAMENTO

Título: Sistema de Informação Geográfica para Apoio à Tomada de Decisão no Setor de Cobrança - SIGATDSC

Candidata: Fabiana Viana Salmaso

Programa de Pós-Graduação em Engenharia de Computação Área de Concentração em Geomática

Data da defesa: 10/12/2007

Aprovada por:

Orientador: Orlando Bernardo Filho, D.Sc., UERJ

Co-orientador: João Araujo Ribeiro, Dr., UERJ

Flávio Joaquim de Souza, D.Sc, UERJ

Jorge Lopes de Souza Leão, Dr. Ing., UFRJ

Dedicatória

À minha família pelo total
apoio nas horas difíceis durante
o curso e no decorrer deste trabalho.

AGRADECIMENTOS

À UERJ.

Ao departamento de Engenharia de Sistemas e Computação e ao departamento de Engenharia Cartográfica, pelo padrão do curso.

Ao professor orientador Orlando Bernardo Filho, por ter sido paciente com os meus problemas pessoais e por ser um orientador exemplar.

Ao amigo Daniel Brasil Floriano, pelo apoio e ajuda na configuração e no início da programação.

Aos demais professores e amigos do curso de Geomática, cada um ao seu jeito contribuiu para minha formação.

Aos amigos que aqui deixo, Orlando, Neide e Adelaide que nunca esquecerei nas próximas jornadas e etapas da vida.

Resumo da Dissertação apresentada a FEN/UERJ como parte dos requisitos necessários para obtenção do grau de Mestre em Ciências (M. Sc.).

SISTEMA DE INFORMAÇÃO GEOGRÁFICA PARA APOIO À TOMADA DE DECISÃO NO SETOR DE COBRANÇA – SIGATDSC

Fabiana Viana Salmaso

Dezembro/2007

Orientador: Orlando Bernardo Filho, D.Sc., UERJ

Co-orientador: João Araujo Ribeiro, Dr., UERJ

Programa de Pós-Graduação em Engenharia de Computação – Área de Concentração
Geomática

Esta dissertação apresenta o desenvolvimento de um Sistema de Informação Geográfica (SIG) que irá consultar as informações dos inadimplentes de um sistema de cobrança. As informações sobre os inadimplentes são do estado do Rio de Janeiro. O SIG desenvolvido oferecerá consultas sobre o grau de risco da cobrança em um determinado bairro após adotar uma estratégia de cobrança, avaliar a distribuição geográfica dos inadimplentes por estado e cidades e o índice de pagamento. O SIG empregou um sistema de inferência nebuloso para auxiliar na gestão do acordo dentro do processo de cobrança. Tal sistema fornece como saída o grau de risco da aplicação de uma estratégia de cobrança, levando em conta o resultado do contato com o inadimplente e o efetivo recebimento do débito.

Palavras-Chave: Sistema de Informação Geográfica, Cobrança, Lógica Nebulosa.

Abstract of Dissertation presented to FEN/UERJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.).

GEOGRAPHIC INFORMATION SYSTEM TO AID DECISION-MAKING IN THE BILLING AREA

Fabiana Viana Salmaso

December/2007

Advisors: Orlando Bernardo Filho, D.Sc., UERJ

João Araujo Ribeiro, Dr., UERJ

Program Computing Engineering – Geomatic

This thesis shows the development of a Geographical Information System (GIS) that will consult the information of the debtors of a billing system. The debtors' information is about Rio de Janeiro state. The Geographical Information System developed will offer consultations on the risk degree of the billing in a certain neighborhood after adopting a billing strategy, to evaluate the geographical distribution of the debtors for state and cities and the payment index. The GIS used a fuzzy inference system to aid inside in the administration of the agreement of the billing process. Such system supplies as exit the risk degree of the billing strategy application, taking into account the result of the contact with the debtors and the effective debit received.

Keywords: Geographic Information System, Billing, Fuzzy Logic.

SUMÁRIO

CAPÍTULO 1	1
Introdução	1
1.1 APRESENTAÇÃO.....	1
1.2 OBJETIVO	2
1.3 CONTRIBUIÇÕES	3
1.4 ORGANIZAÇÃO	3
CAPÍTULO 2	5
Fundamentos Teóricos	5
2.1 INTRODUÇÃO.....	5
2.2 O PROCESSO DE COBRANÇA	5
2.3 LÓGICA NEBULOSA.....	8
2.3.1 Teoria dos Conjuntos Nebulosos.....	9
2.3.2 Sistema de Inferência.....	15
2.4 SISTEMA DE GERENCIAMENTO DE BANCO DE DADOS (SGBD).....	16
2.5 SISTEMA DE INFORMAÇÃO GEOGRÁFICA (SIG).....	19
2.5.1 Relacionamento entre informações de fontes diferentes	22
2.5.2 Aquisição de dados.....	23
2.5.3 Integração de dados	23
2.5.4 Projeções e registros	24
2.5.5 Estruturas de dados.....	24
2.5.6 Modelagem de dados	26
2.6 MAPSERVER	26
2.6.1 História	27
2.6.2 Formatos de entrada e Saída.....	28
2.6.3 Funcionalidades	29
2.6.4 Linguagens e Plataformas.....	30
2.6.5 Funcionamento	30
2.6.6 Modos de Utilização do MapServer	33
2.7 COMENTÁRIOS	34
CAPÍTULO 3	35
Modelagem do SIGATDSC	35
3.1 INTRODUÇÃO.....	35
3.2 MODELOS DE ANÁLISE	35
3.2.1 Modelo de Casos de Uso	35

3.2.2 Diagrama de Classe do SIGATDSC.....	42
3.2.3 Diagrama de Classe Detalhado.....	43
3.2.4 Diagrama de Sequência	43
3.2.5 Diagrama de Estado.....	48
3.3 MODELO DO SISTEMA DE INFERÊNCIA NEBULOSA.....	49
3.4 DEFINIÇÃO DAS REGRAS DE INFERÊNCIA NEBULOSA	51
3.5 COMENTÁRIOS	52
CAPÍTULO 4.....	53
Implementação do SIGATDSC	53
4.1 INTRODUÇÃO.....	53
4.2 PROCESSO DE IMPORTAÇÃO DOS DADOS	53
4.3 INTERFACE DO SIGATDSC.....	57
4.3.1 Funcionamento Básico do Sistema.....	57
4.3.2 Definição das Interfaces com o Usuário.....	58
4.3.2.1 A página Index.php	58
4.3.2.2 A página consulta.php	59
4.3.2.3 A Página LNebulosa.php.....	60
4.3.2.4 A Página Posto_Mapa.php	62
4.4 FUNCIONAMENTO LÓGICO DO SIGATDSC.....	64
4.4.1 A classe usuario_validar.php.....	64
4.4.2 A Classe DefuzzyBean e Defuzzy.php.....	64
4.5 BASE DE DADOS	66
4.5.1 Descrição das tabelas utilizadas pelo SIGTDSC.....	67
4.5.2 Descrição das tabelas do Sistema de Inferência Nebuloso - SIN	76
4.6 ALGORITMO DE INFERÊNCIA NEBULOSA.....	77
4.6.1 Classe DeFuzzyBean	77
4.6.1.1 Método setResultadoFinal	78
4.6.2 Classe DeFuzzy	78
4.6.2.1 Método PercorrerRegras.....	78
4.6.2.1.1 Método VerificarTermoRegra	79
4.6.2.1.2 Método AvaliarMi	80
4.6.2.1.3 Método CalculaAlfa	80
4.6.2.1.4 Método VerificarTermoConsRegra.....	81
4.6.2.1.5 Método CalculaCentroideRegra	81
4.6.2.1.6 Método CalculaCentroide.....	82

4.6.2.1.7 Método CalculaArea.....	82
4.7 COMENTÁRIOS	83
CAPÍTULO 5	84
Testes e Resultados	84
5.1 INTRODUÇÃO.....	84
5.2 TESTE PARA CALCULAR O GRAU DE RISCO.....	84
5.3 TESTE PARA ANALISAR A CONCENTRAÇÃO DE DEVEDORES	85
5.4 TESTE PARA ANALISAR O ÍNDICE DE PAGAMENTO	86
5.5 COMENTÁRIOS	88
CAPÍTULO 6	89
Conclusões	89
Referências Bibliográficas	91
APÊNDICES	96
APÊNDICE A1	96
Funções de Pertinência do SLN.....	96
APÊNDICE A2	104
Regras definidas para o SLN.....	104
APÊNDICE A3	107
Codificação: Classe DeFuzzyBean	107
APÊNDICE A4	108
Codificação: Classe DeFuzzy	108
APÊNDICE A5	112
MapFile: posto.map.....	112

CAPÍTULO 1

Introdução

1.1 APRESENTAÇÃO

O mundo ideal de todo negócio é vender um serviço ou produto e receber por ele, mas esse ciclo se estende, pois a maioria das empresas sofre com os consumidores inadimplentes, sendo que esse é um dos fatores que mais atingem os setores de comércio e de serviços.

A inadimplência [1] pode ser causada por dificuldades financeiras pessoais, desemprego, falta de controle nos gastos, compras para terceiros, atraso de salário, comprometimento de renda com outras despesas, redução de renda e doenças. O descumprimento dessas obrigações acaba encarecendo todo o sistema. A empresa acaba não tendo caixa suficiente para cumprir suas operações. Muitas vezes, optando por empréstimos bancários, aumentando seus custos e, por conseguinte, repassando esses custos para seus produtos ou serviços. No atual mundo globalizado [2,3,4], de abertura de mercado, das quedas de barreiras alfandegárias e da formação de blocos econômicos, aliados à lei da oferta e da procura, da livre iniciativa e da economia perfeita, a competitividade dos produtos fica mais do que nunca atrelada ao seu custo final. Além disso, a maioria dos consumidores que antes não se importavam tanto com preços, já não se deixa impressionar apenas pela qualidade de um produto e a beleza de sua embalagem, principalmente quando está em jogo o seu orçamento familiar. A competitividade dos seus produtos é o que vai determinar a capacidade da empresa em disputar espaço nesse mercado de constantes mutações e acirrada competição.

Para se manter competitiva, a empresa deve reduzir o índice de inadimplência, ou seja, ela deve possuir um setor ou departamento para administração e controle de crédito. Esse será responsável por verificar os clientes em débito e executar as regras para recuperação de crédito [5,6].

Outra opção seria contratar uma empresa especializada em recuperação de créditos (Assessorias de Cobrança). Fazendo uma analogia, a empresa funciona como um automóvel que precisa de combustível para manter o motor em funcionamento e se movimentar. Assim, as contas incobráveis são como um vazamento de combustível que prejudica a velocidade do automóvel, devendo ser eliminadas rapidamente para evitar que o motor fique completamente sem combustível e precise parar. Por esse motivo é necessário que as ações de cobrança

funcionem bem, injetando constantemente e de forma abundante o dinheiro na tesouraria, para que a empresa obtenha maior e contínua liquidez para funcionar bem.

Para realizar a cobrança, é necessária uma base de dados com informações dos inadimplentes como dados pessoais e da dívida. Pode-se extrair muito mais dessas informações quando se leva em conta o enfoque espacial para mostrar a concentração de devedores em uma dada região. Utilizando o componente territorial pode-se agilizar o processo de tomada de decisão em vários níveis, como, por exemplo, avaliar riscos de créditos, estratégias de cobranças, escolher a melhor localização para abertura de um posto de cobrança ou optar pelo seu fechamento. Gerir o setor de cobrança com uma ferramenta que contribua para o processo de tomada de decisão é fundamental, pois quanto mais rápida e eficaz a cobrança menor será o volume da dívida. Segundo Câmara e Davis, é possível dizer de forma genérica:

[7]: “Se onde é importante para seu negócio, então Geoprocessamento é sua ferramenta de trabalho”. Sempre que o onde aparece dentre as questões e problemas que precisam ser resolvidos por um sistema informatizado, haverá uma oportunidade para considerar a adoção de um SIG.”

Levando-se em conta as questões acima citadas, foi desenvolvido um Sistema de Informação Geográfica (SIG) para auxiliar o setor de cobrança no processo de tomada de decisão com base na distribuição geográfica dos inadimplentes. A elaboração desse projeto foi possível através do trabalho realizado em uma consultoria de sistemas voltada para o setor de cobrança.

Apesar de seu sistema de cobrança administrar e agilizar todo o processo operacional e gerencial, automatizar desde a captação das informações do devedor, passando por: distribuição da cobrança entre as filiais, acompanhamento dos acionamentos realizados, emissão de boletos e cartas de cobrança, realização de acordos; baixa de pagamento e emissão de recibos, carteira e comissionamento de recuperadores, fluxo de caixa e geração de relatórios diários e mensais até a prestação de contas ao contratante, tal sistema não possui um enfoque espacial.

1.2 OBJETIVO

Este trabalho teve como objetivo desenvolver um Sistema de Informação Geográfica que contenha informações sobre os inadimplentes no estado do Rio de Janeiro. Para fins de estudo, escolheu-se o município de Nova Iguaçu, pelo fato do tempo de desenvolvimento do

sistema ser reduzido e o Programa de Pós-Graduação em Engenharia de Computação já possuir as cartas digitais desse município.

O Sistema de Informação Geográfica desenvolvido oferecerá consultas sobre a distribuição geográfica dos inadimplentes por estado e cidades, assim como a alocação dos postos de cobrança da assessoria de cobrança. Essas informações buscam otimizar o processo de cobrança através de uma melhor alocação dos postos de cobrança e avaliar os riscos de créditos.

O sistema permitirá avaliar o grau de acertividade x efetividade. A acertividade se refere ao total de contatos telefônicos que foram válidos, ou seja, o telefone informado no momento da compra do serviço ou produto era um número válido e, sendo assim, foi possível contatar o devedor. Efetividade se refere ao total de acordos realizados pelo operador de telecobrança.

A análise do grau de risco permitirá mensurar se a estratégia adotada pela assessoria de cobrança foi adequada.

1.3 CONTRIBUIÇÕES

A técnica de apoio à tomada de decisão desenvolvida para o sistema constitui uma abordagem que envolve planejamento estratégico, visto que o Sistema de Informação Geográfica para Apoio à Tomada de Decisão no Setor de Cobrança (SIGATDSC) de maneira indireta indica se a estratégia adotada pelo setor de cobrança, para a recuperação de crédito de uma determinada empresa, está obtendo bons resultados. O trabalho aqui descrito foi publicado em SALMASO *et alii* [8] que até então trata de uma ferramenta inédita acessada via *web*.

1.4 ORGANIZAÇÃO

Esta dissertação [9] está dividida em seis capítulos. No próximo capítulo, realiza-se um estudo sobre todos os conceitos e tecnologias necessárias para o perfeito entendimento do escopo deste trabalho, definindo-se, por exemplo, o que são os sistemas de informação geográfica (SIG), Sistema de Gerenciamento de Banco de Dados, o funcionamento do processo de cobrança, Lógica Nebulosa abordando os conceitos e inferências e a ferramenta MapServer necessária para o desenvolvimento do SIG SIGATDSC.

O terceiro capítulo apresenta a modelagem do SIGATDSC realizada em *Unified Modeling Language* (UML), com descrição de seus módulos e suas funcionalidades. A ferramenta utilizada para modelagem foi o Java and UML Developer Environment (JUDE).

O quarto capítulo é sobre a implementação do sistema e apresenta sua interface, arquitetura, base de dados e o algoritmo de inferência nebulosa.

O quinto capítulo trata dos testes e resultados obtidos pelo SIG SIGATDSC

Finalmente, o sexto capítulo diz respeito às Conclusões, onde serão descritos todos os objetivos alcançados, como também, possíveis avanços ou seqüências para melhorias futuras.

CAPÍTULO 2

Fundamentos Teóricos

2.1 INTRODUÇÃO

Este capítulo tem por objetivo proporcionar uma visão geral de todos os conceitos necessários para o perfeito entendimento do escopo deste trabalho. Inicialmente, apresenta-se o conceito de sistemas de informação geográfica, sua evolução no tempo, implantação e aplicações. Abordam-se também os conceitos de sistema de gerenciamento de banco de dados, o processo de cobrança, a evolução da ferramenta MapServer e os conceitos da teoria dos conjuntos nebulosos, além do funcionamento de um Sistema de Inferência Nebuloso.

2.2 O PROCESSO DE COBRANÇA

Venda e cobrança são palavras que possuem significados diferentes, mas que estão correlacionadas no processo de qualquer negócio. Ao realizar vendas a prazo é necessário possuir uma estrutura de cobrança ou contratar uma empresa que realize esse serviço para trabalhar os vencimentos das vendas a prazo.

[10]: “A política de cobrança é um método para lidar com contas vencidas; a primeira etapa é analisar o período médio de recebimento e preparar um quadro de idade que relacionem tempo de existência das contas à proporção que representam todas as contas a receber; a etapa seguinte é decidir quanto ao método de cobrança e avaliar a possibilidade de recorrer a serviços de factoring, ou seja, à venda de contas vencidas”.

Muitas vezes, empresas que obtiveram sucesso na área comercial e tiveram um bom nível de venda, suspendem seus pagamentos por falta de liquidez, ou ainda pior, cessam suas atividades empresariais por falência. Esse inesperado fracasso pode ocorrer em virtude da empresa ter concentrado seus esforços em vender, negligenciando a cobrança. A inadimplência dos clientes pode levar uma empresa à bancarrota. Para minimizar os efeitos negativos da inadimplência, é necessário agilizar o processo de cobrança de notas fiscais para garantir a competitividade da empresa, qualquer atraso na hora de cobrar aumenta o descompasso temporal entre os fluxos de saída de caixa e entrada, o que gera uma deterioração da tesouraria, aumentando assim seus custos financeiros.

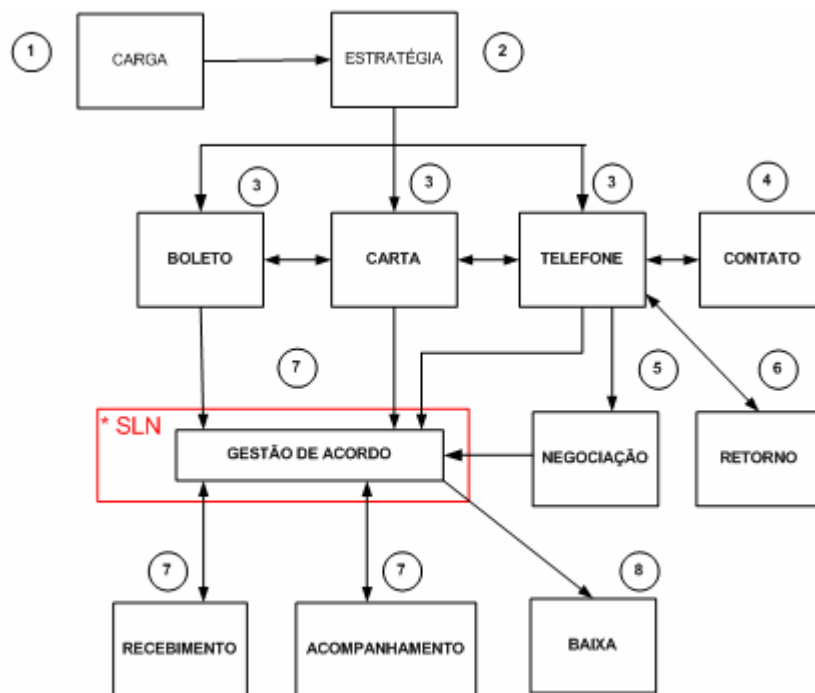
A inadimplência pode ser classificada em três fases distintas:

Inicial – Comercial: É a identificação do não cumprimento do acordo anteriormente firmado, buscando as causas do ocorrido, desenvolvendo o contato direto com o cliente, seja através de uma carta ao cliente informando que se encontra em débito ou através de um telefonema.

Intermediária – Financeira: É a não possibilidade de o cliente saldar a sua dívida, nesse momento a empresa procura oferecer condições para que o cliente regularize sua situação.

Final – Contábil: Normalmente desenvolvido por empresas especializadas na recuperação de crédito (assessorias de cobrança), as quais oferecerão uma condição de prazo de pagamento bem mais flexível que a empresa que efetuou a venda com o objetivo de recuperar o aspecto financeiro. Nesse caso, o processo de cobrança cumpre várias etapas para tentar resgatar a dívida.

Para descrever as várias etapas foi elaborado um diagrama do fluxo de cobrança para facilitar a visão, Figura 2.1:



* SLN - Sistema de Lógica Nebulosa

Figura 2.1: Diagrama do fluxo de cobrança.

Os blocos da figura 2.1 são comentados a seguir:

Bloco 1 - Carga

A carga contendo as informações dos devedores é enviada à assessoria de cobrança pelo credor, para que a assessoria possa importar a carga para o sistema de cobrança. No capítulo 4, na seção 4.2 será descrito esse procedimento de forma detalhada.

Bloco 2 - Estratégia

Antes de iniciar o processo de cobrança é necessário montar uma estratégia. Por exemplo, a empresa A possui devedores com parcelas em atraso no seguinte perfil:

- Menos de 15 dias de atraso
- Entre 15 e 30 dias de atraso
- Entre 30 e 60 dias de atraso
- Entre 60 e 90 dias de atraso
- Mais de 90 dias de atraso

A política de cobrança deve definir as ações e procedimentos que serão adotados para cada período de atraso, como faixa de negociação e forma de contato a qual pode ser por boleto, envio de carta de cobrança ou ação de telecobrança.

Bloco 3 – Boleto, Carta ou Telecobrança

Realizada a importação da carga e definida a estratégia, inicia-se o processo de cobrança. De acordo com o perfil do devedor ou durante a negociação pode ser enviada uma carta informando que o mesmo encontra-se em débito, um reenvio do boleto para pagamento ou telecobrança.

Bloco 4 – Contato

O bloco 3 busca estabelecer um contato com o devedor para iniciar uma negociação e posterior saldo da dívida. O cobrador é o profissional encarregado por esse contato, isto é, pela cobrança do cliente. Cabe ao cobrador obedecer às regras previstas na política de

cobrança da empresa, assim como a legislação vigente com relação aos direitos do consumidor. Sua atuação é fundamental na determinação da rapidez e eficácia com que o crédito atrasado é recuperado.

Nem sempre no 1º acionamento o cobrador obtém sucesso. Um exemplo típico seria: O cobrador não consegue encontrar o devedor em casa, o telefone dá ocupado, deixa recado com algum parente etc. Logo, ele re-agenda aquele contato para ligar em outro horário.

Bloco 5 – Negociação

Estabelecido o contato com o devedor, o cobrador inicia o processo de negociação, respeitando as faixas de negociação para realizar o acordo com o devedor. Caso o cobrador obtenha sucesso na negociação, passa para a fase de gestão de acordo.

Bloco 6 – Retorno

O retorno ocorre quando o cobrador realiza o acordo com o devedor e o mesmo não cumpre. Nesse caso, o cobrador entra em negociação novamente, para tentar resgatar a dívida.

Bloco 7 – Gestão de Acordo

O devedor mantendo o acordo passa para a fase de gestão de acordo. Essa fase verifica o acompanhamento do acordo e o recebimento do pagamento. Na fase do acordo, é possível mensurar o grau de risco da cobrança através do SIGATDSC, ou seja, nesta fase do fluxo de cobrança se aplica o sistema de inferência nebulosa implementado no SIG.

Bloco 8 – Baixa

Após o devedor liquidar sua dívida, ele é retirado do processo de cobrança.

2.3 LÓGICA NEBULOSA

A Lógica Nebulosa [11,12,13,14,15,16,17,18,19,20,21,22] (*Fuzzy Logic*) surgiu por volta de 1965 em estudos desenvolvidos por L. Zadeh [23,24,25,26] sobre os conjuntos nebulosos (*Fuzzy Sets*), ela se preocupa com os princípios formais do raciocínio aproximado e

procura modelar o modo impreciso do raciocínio que tem um papel fundamental na habilidade humana de tomar decisões [12]. Do inglês, “*fuzzy*” significa “vago”, ou “nebuloso” ou “não totalmente delineado”.

Uma aplicação que use a Lógica Nebulosa deve utilizar os conceitos da teoria dos conjuntos nebulosos combinados com o uso de um sistema de inferência nebuloso, as seções a seguir abordam esses conceitos.

2.3.1 Teoria dos Conjuntos Nebulosos

Para a teoria clássica dos conjuntos, o conceito de pertinência (*membership*) de um elemento a um conjunto é definido de forma simples: um elemento pertence ou não pertence a um determinado conjunto.

Formalmente, se quisermos definir um conjunto A de um universo de discurso U , e qualquer elemento x pertencente a U , ter-se-ia a função:

$$f_A(x): U \rightarrow A, \quad (2.1)$$

onde: f_A é chamada de **função característica**, pois define os elementos de A , tal que:

$$f_A(x) = 1 \text{ se e somente se } x \in A$$

$$f_A(x) = 0 \text{ se e somente se } x \notin A$$

Nesse caso, a pertinência é indicada pelo valor 1 (um) e a não pertinência é indicada pelo valor 0 (zero). A figura 2.2 mostra a função de pertinência para o conjunto ordinário (*crisp*) de Temperaturas Altas:

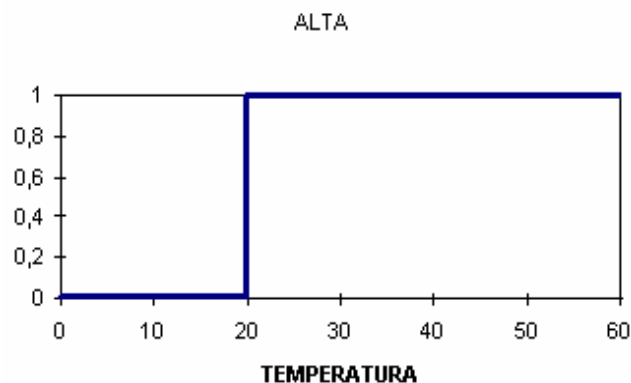


Figura 2.2: Conjunto crisp de Temperaturas Altas

Se generalizarmos essa função f_A para qualquer **valor de saída** dentro de um intervalo fechado $[0, 1]$, teremos uma função nebulosa (*fuzzy*), definida como:

$$\mu_A(x): U \rightarrow [0, 1], \quad (2.2)$$

onde o grau de pertinência (*membership*) dessa função pode assumir um número infinito de valores dentro desse intervalo. A figura 2.3 mostra a função de pertinência para o conjunto nebuloso (*fuzzy*) de Temperaturas Altas:

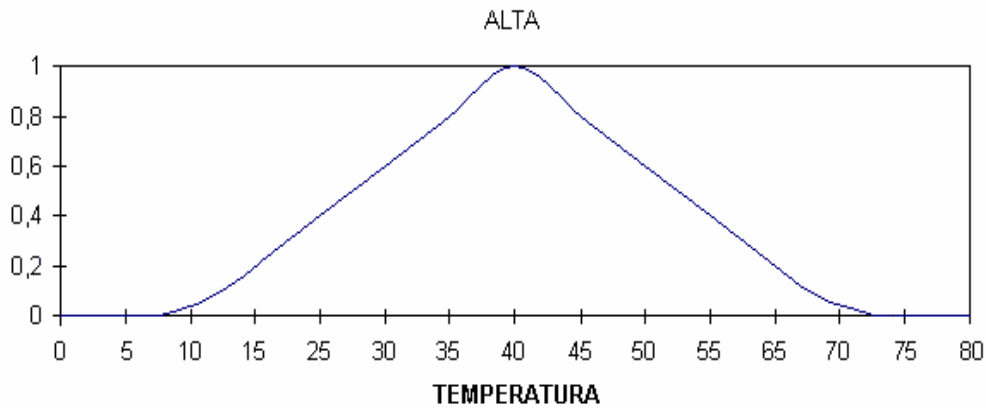


Figura 2.3: Conjunto fuzzy de Temperaturas Altas

Para a teoria clássica, basta uma lista contendo seus itens para definir um conjunto, ou seja:

$$A = \{ x_1, x_2, x_3, \dots, x_n \} \text{ ou } A = \sum_{i=1}^n x_i, \quad (2.3)$$

onde o operador “+” denota união. Caso o conjunto represente um intervalo contínuo, ter-se-ia:

$$A = \int_U x \, dx \quad (2.4)$$

Para definirmos um conjunto nebuloso, é necessária a lista de seus elementos e, além disso, o grau de pertinência de cada elemento. Assim, teríamos:

$$A = \{ \mu_1/x_1, \mu_2/x_2, \mu_3/x_3, \dots, \mu_n/x_n \} \text{ ou } A = \sum_{i=1}^n \mu_i(x_i) / x_i \quad (2.5)$$

Da mesma forma, para um intervalo contínuo de valores teríamos:

$$A = \int_U \mu_A(x) / x \, dx \quad (2.6)$$

Um outro exemplo de conjunto nebuloso seria o conjunto de **Homens de Meia-Idade**, dado por:

$$V = \{0 / 5 ; 0 / 10 ; 0,2 / 15 ; 0,3 / 20 ; 0,4 / 25 ; 0,6 / 30 ; 0,8 / 35 ; 1 / 40 ; 0,8 / 45 ; 0,6 / 50 ; 0,4 / 55 ; 0,3 / 60 ; 0,2 / 65 ; 0 / 70 ; 0 / 75 \} \quad (2.7)$$

A Figura 2.4 mostra a representação gráfica desse conjunto:

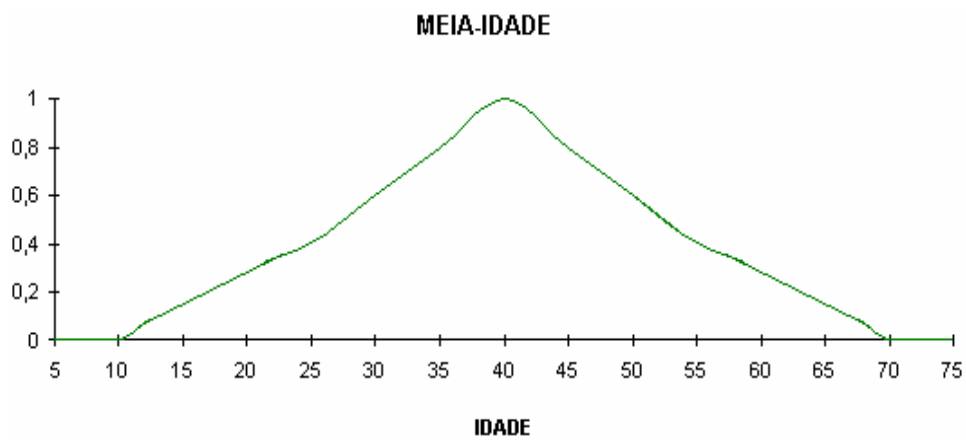


Figura 2.4: Função de pertinência para o conjunto nebuloso de Homens de Meia-Idade

A Lógica Nebulosa é uma aplicação da teoria dos conjuntos nebulosos. Observe que muitos dos conhecimentos que encontramos no nosso dia-a-dia são imprecisos, ou seja, possuem um certo grau de nebulosidade (*fuzziness*) na descrição de sua natureza, como pode ser observado nos exemplos demonstrados anteriormente. Essa imprecisão pode estar associada à sua forma, posição, cor, textura, ou até mesmo na situação em que esses acontecimentos estão incluídos. Em muitos casos, um mesmo conceito terá diferentes graus de imprecisão em diferentes situações e/ou momentos. Um dia quente no inverno não é exatamente o mesmo que um dia quente no verão. A definição exata de quando a temperatura muda de morna para quente é imprecisa, ou seja, não podemos identificar um único ponto em que a temperatura é morna, e quando aumentamos em 1 (um) grau, a temperatura passa a ser considerada quente.

Os Conjuntos Nebulosos fornecem uma base matemática que permite o manuseio dessas imprecisões através das chamadas **variáveis linguísticas**, usadas na implementação dos Sistemas de Lógica Nebulosa, também conhecidos como Sistemas de Inferência Nebulosos.

A idéia de se utilizar variáveis lingüísticas com valores de palavras ou frases de linguagens naturais ou artificiais está relacionada ao fato desses valores não serem tão específicos como números. Nesse sentido, nas ocasiões em que o tratamento de um valor numérico preciso é irrelevante para o resultado final da operação, como em controle de sistemas não lineares, por exemplo, a manipulação dos valores lingüísticos é mais fácil, pois esses estão em menor quantidade e são mnemônicos.

Uma variável lingüística é definida por uma quintupla $(x, T(x), U, G, M)$ onde:

- x é o nome da variável;
- $T(x)$ denota o conjunto de termos de x , isto é, o conjunto de nomes dos **valores lingüísticos** de x com cada valor sendo um conjunto nebuloso;
- U é o universo de discurso dos conjuntos nebulosos que formam os termos de $T(x)$;
- G é a regra sintática que usualmente tem a forma de uma gramática, para gerar os nomes dos valores lingüísticos;
- $M(X)$ é a regra semântica que atribui um significado ao termo X do conjunto $T(x)$, ou seja, $M(X)$ é um subconjunto nebuloso de U .

Como exemplo, podemos citar a variável lingüística com rótulo x =velocidade, com conjunto de termos $T(\text{velocidade})=\{\text{muito_lenta, lenta, rápida, muito_rápida}\}$, universo de discurso $U=[10\text{Km/h}, 120\text{Km/h}]$ e um dos valores $M(X)$ como mostrado a seguir.

$$M(\text{lenta})=\{(u, \mu_{\text{lenta}}(u)) \mid u \in [10 \text{ Km/h}, 120\text{Km/h}]\} \quad (2.8)$$

Uma variável lingüística é dita estruturada se a regra semântica $M(X)$ e o conjunto de termos $T(x)$ podem ser gerados algoritmicamente, usando para tanto a regra sintática de construção dos termos G e os chamados modificadores (ou *hedges*) que consiste de uma operação para alterar o significado de um termo mais geral, obtendo assim, um conjunto nebuloso a partir de outro.

A Lógica Nebulosa [27] normalmente é empregada na construção dos chamados Sistemas de Lógica Nebulosa (SLN) [28], representados pela figura 2.5. Nesses sistemas, são fornecidas entradas precisas para um módulo codificador que, por sua vez, fornece valores nebulosos para um motor de inferência, o qual processa a aplicação de uma regra do tipo SE-ENTÃO, constituída de proposições, envolvendo termos de variáveis lingüísticas. Após o processamento de uma regra, o valor nebuloso obtido como resposta da inferência é decodificado, obtendo-se, dessa forma, a saída precisa do sistema.

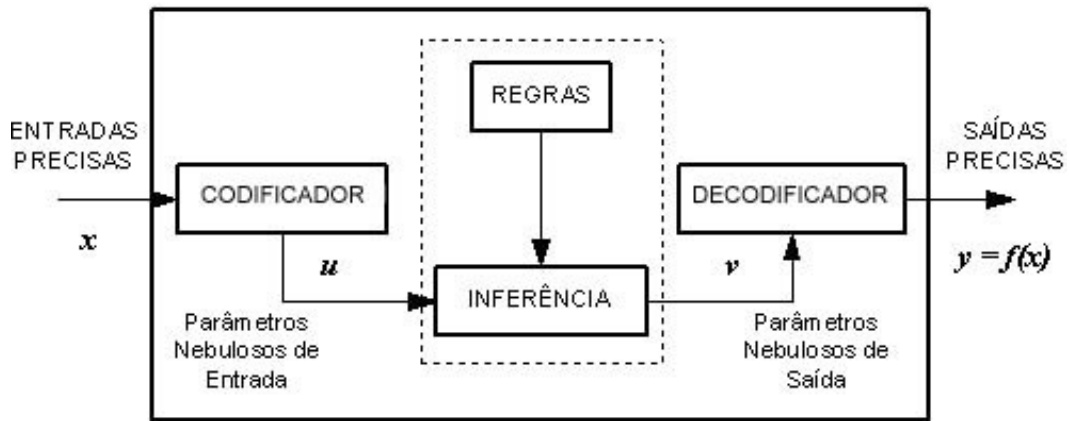


Figura 2.5: Sistema de Lógica Nebulosa.

O *codificador* mapeia um valor preciso (*crisp*) $x \in X$ para um conjunto nebuloso \tilde{A} em X . Um valor preciso pode ser codificado seguindo um método simples, trapezoidal, triangular ou gaussiano. O método mais usado é o simples que nada mais é do que criar o conjunto nebuloso \tilde{A} com sua função de pertinência assumindo valor 1 (um) no ponto de X em que o dado parâmetro tem a sua definição e, nos demais pontos, o valor da pertinência é 0 (zero).

O método simples nem sempre é adequado, principalmente quando os dados podem ser corrompidos por ruídos em processos de medidas, por essa razão foram propostos os outros três métodos mostrados nas figuras 2.6, 2.7 e 2.8. Nos casos do triângulo e da curva gaussiana, o valor do parâmetro a ser codificado é colocado no ponto de máximo da função de pertinência (1.0), enquanto os outros pontos que se afastam desse máximo possuem valores decrescentes de pertinência até chegar a zero. No caso do trapézio, o valor a ser codificado fica no ponto médio dos máximos da função, como mostra a figura 2.8. A largura do triângulo, do trapézio e da área da gaussiana é escolhida com base no problema considerado.

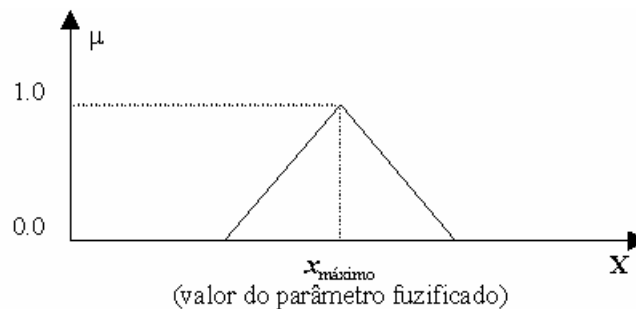


Figura 2.6: Formato da função de pertinência de um conjunto nebuloso obtido pela *codificação* de um parâmetro $x_{\text{máximo}}$ pelo método do triângulo.

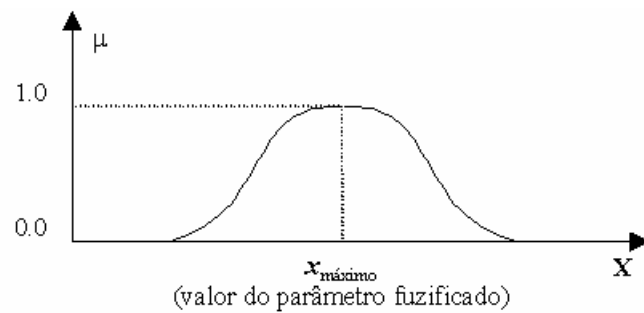


Figura 2.7: Formato da função de pertinência de um conjunto nebuloso obtido pela *codificação* de um parâmetro $x_{\text{máximo}}$ pelo método da gaussiana.



Figura 2.8: Formato da função de pertinência de um conjunto nebuloso obtido pela *codificação* de um parâmetro $x_{\text{máximo}}$ pelo método do trapézio.

A *decodificação* pode ser feita através do valor máximo, da média de valores máximos ou por meio do cálculo do centróide. O método do máximo consiste em escolher como valor decodificado, o ponto do universo de discurso em que a função de pertinência é máxima. Esse método provoca alguma confusão nos casos em que a função de pertinência possui vários valores máximos, logo, foi proposto em seguida, a utilização da média dos máximos para achar o valor a ser decodificado. Todavia, numa situação como a mostrada pela figura 2.9, a média dos máximos conduziria a um valor onde a função de pertinência é zero, o que não faria muito sentido.

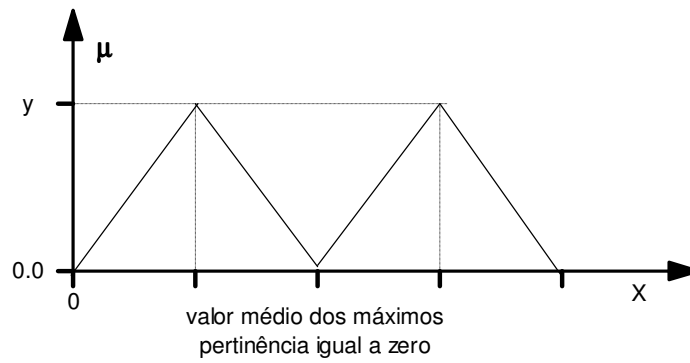


Figura 2.9: Exemplo de conjunto nebuloso em que a decodificação pela média dos máximos conduziria a um valor de pertinência igual a zero.

Diante dos inconvenientes dos métodos do máximo e da média dos máximos, foi proposto em seguida, o método do centróide, que se baseia no cálculo do centro de gravidade conforme a expressão mostrada abaixo. O único inconveniente desse último método seria apenas a complexidade do cálculo que, no caso geral, é obtido de forma numérica.

$$\bar{x} = \frac{\int_S x \mu_{\tilde{A}}(x) dx}{\int_S \mu_{\tilde{A}}(x) dx} \quad (2.9)$$

Na fórmula (2.9), \bar{x} é o valor decodificado de um dado parâmetro que corresponde ao centro de gravidade (centróide) do conjunto nebuloso com função de pertinência $\mu_{\tilde{A}(x)}$ e suporte S.

2.3.2 Sistema de Inferência

Um sistema de lógica nebulosa típico será composto de várias regras de inferência, manipulando termos de algumas variáveis lingüísticas como no exemplo exposto a seguir:

- SE Largura (ε) É estreita E Trânsito (ε) É intenso ENTÃO Velocidade (ε) é muito_lenta
- SE Largura (ε) É estreita E Trânsito (ε) É livre ENTÃO Velocidade (ε) é rápida
- SE Largura (ε) É estreita E Trânsito (ε) É moderado ENTÃO Velocidade (ε) é lenta
- SE Largura (ε) É média E Trânsito (ε) É intenso ENTÃO Velocidade (ε) é lenta

- SE Largura (ε) É ampla E Trânsito (ε) É intenso ENTÃO Velocidade (ε) é rápida
- SE Largura (ε) É ampla E Trânsito (ε) É livre ENTÃO Velocidade (ε) é muito_rápida

As regras apresentadas anteriormente modelam um raciocínio nebuloso com o intuito de estabelecer o limite de velocidade em uma dada estrada ε. As variáveis lingüísticas empregadas nesse exemplo seriam:

- $T(velocidade) = \{muito_lenta, lenta, rápida, muito_rápida\}$
- $T(largura) = \{ampla, média, estreita\}$
- $T(trânsito) = \{livre, moderado, intenso\}$

Cada um dos termos dessas variáveis lingüísticas devem ter suas funções de pertinência definidas de acordo com os seus respectivos universos de discurso. A variável velocidade teria como universo de discurso, um intervalo de valores reais de 10Km/h até 120Km/h. Já a variável largura poderia ter como universo de discurso valores reais de 3m até 30m, enquanto a variável trânsito poderia estar definida em um universo de discurso fazendo referência à quantidade de cruzamentos e/ou semáforos existentes, logo, esse universo seria um intervalo inteiro de 0 até 20, por exemplo.

2.4 SISTEMA DE GERENCIAMENTO DE BANCO DE DADOS (SGBD)

Para desenvolver um SIG determinados componentes básicos são necessários, tais como: interface com usuário, entrada e integração de dados, consulta e manipulação, saída de dados e sistema de gerenciamento de banco de dados. Cada sistema de informações geográficas em função de seus objetivos e necessidades implementa esses componentes de forma distinta, mas todos estão presentes em um SIG [29].

Um sistema de gerenciamento de banco de dados (SGBD) consiste de uma coleção de dados inter-relacionados e em um conjunto de programas para acessá-los. Um conjunto de dados, normalmente referenciado como *banco de dados*, contém informações sobre uma empresa particular, por exemplo. O principal objetivo de um SGBD é prover um ambiente que seja adequado e eficiente para recuperar e armazenar informações de banco de dados.

Os sistemas de banco de dados são projetados para gerenciar grandes grupos de informações. O gerenciamento de dados envolve a definição de estruturas para

armazenamento de informação e o fornecimento de mecanismos para manipulá-las. Além disso, o sistema de banco de dados precisa fornecer segurança das informações armazenadas, caso o sistema dê problema, ou contra tentativas de acesso não-autorizado. Se os dados devem ser divididos entre diversos usuários, o sistema precisa evitar possíveis resultados anômalos. Um SGBD busca eliminar ou reduzir os seguintes problemas que ocorrem quando se faz uso de arquivos para o armazenamento de dados:

Inconsistência e Redundância dos Dados - A criação e manutenção de arquivos e programas geralmente é realizada por pessoas diferentes, os arquivos podem ter formatos diferentes e os programas podem ser desenvolvidos em diferentes linguagens. Uma mesma informação pode estar presente em mais de um arquivo, isso contribui para uma maior solicitação de espaço em disco para o seu armazenamento e tornando o seu acesso mais lento. O programa pode realizar alterações apenas em um arquivo que contenha os dados, deixando os demais arquivos desatualizados gerando uma inconsistência nos dados. Ao utilizar um SGBD problemas como inconsistência e redundância de dados são resolvidos.

Dificuldade de Acesso aos Dados - Os SGBD's possuem uma linguagem para manipulação dos dados (*Structured Query Language* – SQL), que permite diversas formas de consulta, sem a necessidade do uso de programação específica, possibilitando uma recuperação rápida das informações.

Sem utilizar SGBD cada vez que há a necessidade de recuperar dados de uma forma específica, o programador é obrigado a fazer um novo programa, consumindo tempo para atender as necessidades do usuário. Por exemplo, pode ser importante saber os inadimplentes que estão devendo acima de um certo valor ou em outro momento saber os devedores em uma determinada região.

Isolamento de Dados – Ao utilizar o arquivo para armazenar dados, uma determinada consulta pode solicitar dados que estejam em diferentes arquivos e esses podem estar em diferentes formatos. Como os dados podem estar dispersos torna-se difícil criar uma aplicação que recupere os dados de forma adequada. Por exemplo, para atender uma solicitação que deseja saber os devedores de uma determinada região nos últimos dois anos de um determinado credor, seria necessário uma nova aplicação para a resolução desse problema. Ao utilizar um SGBD, os dados estão sempre armazenados em um mesmo formato e em tabelas específicas, a solução desse problema consistiria em montar uma consulta na sua linguagem de consulta (SQL) para recuperar os dados.

Problemas de Integridade dos Dados – Antes de armazenar dados em um arquivo, seus valores devem satisfazer certas restrições para a manutenção de sua consistência, para isso um programa deve criticar e restringir os valores de um determinado dado, mas é necessário que

todos os outros programas que lidam com os outros formatos de arquivos, por exemplo, mantenham o mesmo padrão de validação para que nenhum valor indevido seja armazenado. No SGBD, os dados são armazenados em tabelas específicas e ao criar os campos que compõem a tabela, as particularidades de cada campo já são definidas de forma que antes de armazenar um dado em uma tabela, as restrições de validação são verificadas, evitando assim a inconsistência dos dados.

Problemas de Atomicidade – Um computador é uma máquina composta por partes mecânicas, elétricas e eletrônicas e, portanto, está sujeito à falhas. Um sistema de arquivos pode realizar parte de uma transação, mas não terminá-la. Um exemplo de problema de atomicidade [30] seria um caixa eletrônico debitar da conta corrente de um usuário e não lhe entregar o dinheiro. Em casos de falhas como esse, é necessário um mecanismo capaz de desfazer as operações realizadas antes do término da transação e voltar ao estado inicial, ou seja, uma transação deve ser única, deve ser atômica, de tal forma que a mesma ocorra por completo ou simplesmente não ocorra e gere uma mensagem de erro. Um SGBD deve ser capaz de resolver os problemas de falta da atomicidade.

Anomalias nos Acessos Concorrentes – Para um sistema que controla as vendas de passagens aéreas, por exemplo, é comum ocorrer solicitações simultâneas, caso não exista um mecanismo para lidar com os acessos concorrentes pode resultar na inconsistência dos dados. Nesse cenário exemplo, uma inconsistência seria a venda de passagens de um mesmo lugar para um mesmo voo para dois passageiros diferentes. Um SGBD trata os acessos concorrentes garantindo assim a consistência dos dados.

Problemas de Segurança – De acordo com o grupo a que um usuário pertence, esse pode ter permissão apenas para consultar os dados e não para efetuar inclusão, alteração ou exclusão. Em um sistema *web* de um laboratório, os pacientes só podem consultar os resultados de seus exames, já o médico pode lançar os dados do exame e alterá-los. Um SGBD deve prover segurança aos dados, através de restrições de acesso, impedindo que um usuário não autorizado acesse dados restritos para ele ou realize operações restritas.

Ao usar um SGBD, busca-se compartilhar informações de forma segura, garantir a integridade e a consistência dos dados, permitir acesso rápido e eficiente, oferecer uma forma de afastar do usuário comum a necessidade de compreender as estruturas físicas dos dados. Dentre os SGBD's disponíveis no mercado os mais utilizados são: Oracle, SQL Server, DB2, PostgreSQL, MySQL, entre outros.

2.5 SISTEMA DE INFORMAÇÃO GEOGRÁFICA (SIG)

Um Sistema de Informação Geográfica (SIG) é um caso especial de sistema de informação, pois esse é um sistema de computador capaz de montar, armazenar e manipular informações referenciadas geograficamente, ou seja, dados que são identificados de acordo com suas localizações.

O termo **Sistema** deve-se ao fato do SIG ser composto por vários componentes inter-relacionados. O termo **Informação** deve-se ao fato do SIG permitir a conversão de dados em informações, a partir de manipulações e consultas interativas sobre os dados armazenados. O termo **Geográfica** implica que os dados possuem localizações conhecidas ou podem ser calculadas em termos de coordenadas geográficas.

Várias são as definições para SIG, porém as mais comumente encontradas serão citadas a seguir:

[31]: “É um banco de dados contendo uma discreta representação da realidade geográfica na forma estática de objetos geométricos em duas dimensões, com seus atributos ou dados não espaciais associados, com uma funcionalidade grandemente limitada pelas operações geométricas primitivas para criar novos objetos ou para computar as relações entre objetos, ou para simples interrogáveis e descrições sumárias”.

[32]: “Um sistema que garante decisões envolvendo a integração de dados referenciados espacialmente em um ambiente específico.”

[33]: “Um sistema que contém dados espacialmente referenciados que possam ser analisados e convertidos em informações para uso em um conjunto específico de finalidades. A característica principal de um SIG é analisar dados para gerar novas informações”.

[34]: “Um SIG é qualquer sistema de gerenciamento de informação capaz de: coletar, armazenar e recuperar informações baseadas nas suas localizações espaciais; identificar locais dentro de um ambiente que tenha sido selecionado a partir de determinados critérios; explorar relações entre os dados de um certo ambiente; analisar os dados espaciais para subsidiar os critérios de formulação de decisões; facilitar a exportação de modelos analíticos capazes de avaliar alternativas de impactos no meio ambiente; exibir e selecionar áreas tanto graficamente como numericamente e /ou depois das análises”.

[35]: “Qualquer conjunto de procedimento manual ou computacional usado para armazenar e manipular dados geograficamente referenciados”

[36]: “Um elenco de funções automáticas que fornece aos profissionais, com avançada capacidade, o armazenamento, recuperação, manipulação e exibição de dados geograficamente localizados”.

Os SIG's podem ser utilizados em diversos segmentos como, por exemplo:

- Empresas prestadoras de serviços especializados em dados espaciais, como de Topografia, Aerofotogrametria, Cartografia e Sensoriamento Remoto, que vêem a tecnologia SIG como repositório dos dados que geram.
- Empresas de consultoria em Engenharia e áreas afins que vêem a tecnologia SIG como um aliado poderoso na solução dos problemas de seus clientes.
- Empresas e órgãos estatais que em suas atividades utilizam SIG no planejamento, administração, monitoramento e gerência do meio físico regional e federal.
- Prefeituras municipais que utilizam SIG para solução de cadastros técnicos voltados a atividades de planejamento e gestão urbana, tributação, controle de tráfego, meio ambiente, saneamento e outros.
- Concessionárias de serviços públicos que planejam, projetam, implantam, operam e gerenciam redes de água, esgoto, eletricidade, gás, telefone e TV a cabo.
- Empresas públicas responsáveis por atendimentos a emergências e gerenciamento de riscos ambientais.
- Empresas nas áreas de negócios e propaganda que planejam, projetam e implantam atividades econômicas baseadas em variáveis de mercados distribuídas espacialmente.
- Órgãos públicos que prestam informações turísticas e de hotelaria.
- Empresas públicas e privadas que trabalham com logística, distribuição e transporte de bens e serviços.
- Empresas agrícolas e florestais, em suas atividades de planejamento, projeto, implantação, cultivo, colheita e transporte de produtos.
- Empresas de mineração, em suas atividades de planejamento, projeto, implantação, prospecção e lavra de recursos naturais.
- Universidades e institutos de pesquisa em suas atividades de ensino, pesquisa e desenvolvimento científico e tecnológico.

Mas para representar as entidades do mundo real em um SIG, essas devem se enquadrar em um dos seguintes atributos: espacial, temporal e temáticos.

Atributos espaciais guardam informações sobre localização, topologia e geometria das entidades. A localização é registrada em coordenadas geográficas, coordenadas de projeção ou coordenadas retangulares com uma origem local. A topologia contém informações sobre vizinhança, distância; a geometria contém informações sobre área,

perímetro e forma. A tecnologia atual de Sistemas de Informação Geográfica permite a geração de topologia e geometria a partir dos dados de localização.

Atributos temporais referem-se à idade do objeto de estudo, à data ou a frequência de aquisição.

Atributos temáticos referem-se a outras propriedades das entidades, que não são de localização nem temporais, tais como tipos de rochas, índice pluviométrico anual, tipos de solos, presença de minerais. Os atributos temporais e os temáticos são também conhecidos como atributos não-espaciais ou atributos descritivos.

Para a elaboração de um SIG, de uma forma geral, é preciso lidar com diversos quesitos dentre os quais se destacam:

- Relacionamento entre informações de fontes diferentes
- Aquisição de dados
- Integração de dados
- Projeções e registros
- Estruturas de dados
- Modelagem de dados

A figura 2.10 mostra o ciclo de extração e utilização de informações em um SIG, mostrando os estágios citados anteriormente.

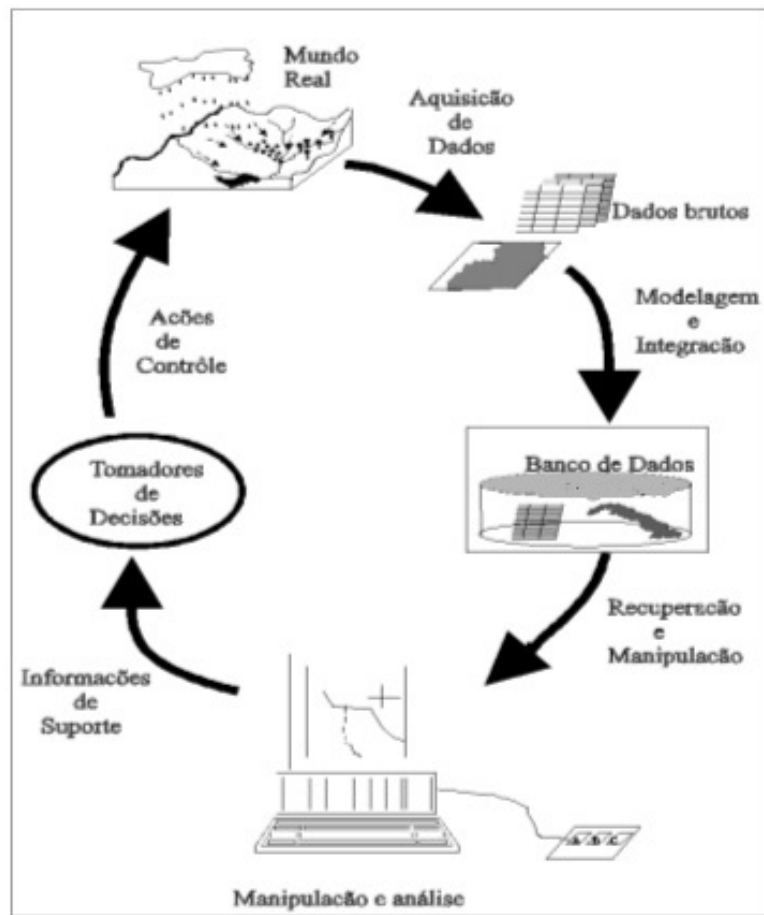


Figura 2.10: Ciclo de extração e utilização de informações em um SIG

FONTE: adaptada de Aronoff (1989), pg 34

A seguir, a descrição das etapas para elaboração de um SIG.

2.5.1 Relacionamento entre informações de fontes diferentes

Se fosse possível relacionar informação sobre a chuva de um estado com fotografias aéreas de um município desse estado, seria possível contar quais áreas irrigadas secam em certas épocas do ano. Um SIG que pode usar informação de muitas fontes diferentes de muitas formas diferentes que pode ajudar com tais análises. A exigência primária para os dados da fonte é que as localizações para as variáveis sejam conhecidas. Localização pode ser descrita através de coordenadas (x, y e z) de longitude, latitude e elevação, ou por sistemas como códigos de endereçamento postal ou ainda por marcadores de quilômetros das estradas.

Qualquer variável, que pode ser localizada espacialmente, pode ser inserida em um SIG. Várias bases de dados computadorizadas, que podem ser adicionadas diretamente em um SIG, estão sendo produzidas por agências federais e empresas privadas. Diferentes tipos de dados, que se encontram em forma de um mapa, podem ser adicionados a um SIG.

Um SIG pode também converter informação existente na forma digital, mas que ainda não se encontra em forma de mapa. Tais informações podem ser reconhecidas e utilizadas (inseridas) em um mapa. Por exemplo, podem ser analisadas imagens de satélite digitais para produzir um mapa como uma camada de informação digital sobre coberturas de vegetação.

2.5.2 Aquisição de dados

Os Sistemas de Informação Geográfica podem capturar a informação na forma de mapa de diversas maneiras. O mapa capturado deve ser colocado em uma forma que o computador possa reconhecer, pois só assim o mesmo poderá armazenar e processar as informações de um mapa. Mapas podem ser digitalizados ou feitos diretamente no computador, ou ainda podem ser armazenadas as coordenadas cartesianas que descrevem o mapa.

Um SIG [37] pode ser usado para enfatizar as relações de espaço entre os objetos que são traçados. Enquanto um traçado feito por uma ferramenta de desenho (CAD) pode representar uma estrada simplesmente como uma linha, um SIG também pode reconhecer aquela estrada como a fronteira entre uma área rural para plantio e uma outra área para o desenvolvimento urbano.

A aquisição de dados é a etapa mais demorada do trabalho de desenvolvimento de um SIG. Devem ser especificadas as identidades dos objetos no mapa, como também as relações de espaço entre eles. A edição de informação que é capturada automaticamente também pode ser difícil. “Scanners” eletrônicos registram marcas em um mapa da mesma maneira fiel como eles registram as características do mesmo mapa. Por exemplo, uma mancha de sujeira poderia conectar duas linhas que não deveriam ser conectadas. Os dados estranhos devem ser editados ou removidos do arquivo de dados digital.

2.5.3 Integração de dados

Um SIG torna a integração de dados possível, ou seja, ele une ou integra informação que seria difícil de associar por qualquer outro meio. Assim, um SIG pode usar combinações de variáveis traçadas para construir e analisar variáveis novas.

A tecnologia de SIG sendo usada por uma companhia de água que possui a informação sobre o seu faturamento, poderia simular a descarga de materiais rio acima nos sistemas sépticos em um bairro de um município. As contas mostram quanta água está sendo usada em cada endereço. A quantidade de água que um dado cliente usa irá predizer a quantidade de

material que será descarregado nos sistemas sépticos, de forma que áreas de descarga séptica pesadas podem ser localizadas usando um SIG.

2.5.4 Projeções e registros

O mapa de uma propriedade pode estar em uma escala diferente de um mapa de terras. As informações de mapas em um SIG devem ser manipuladas para que elas registrem ou se ajustem às informações oriundas de outros mapas. Antes dos dados digitais serem analisados, eles podem ter que sofrer outras manipulações como, por exemplo, conversões de projeção. Isso os integra em um Sistema de Informações Geográficas. Projeção é um componente fundamental da elaboração de um mapa. Uma projeção é um meio matemático de transferir informação da superfície curva tridimensional da Terra para uma área bidimensional como papel ou uma tela de computador. Projeções diferentes são usadas para tipos diferentes de mapas porque cada projeção é particularmente apropriada a certas aplicações. Por exemplo, uma projeção que representa com precisão as formas dos continentes irá distorcer os seus tamanhos relativos.

Considerando que muito da informação em um SIG vem de mapas existentes, um SIG usa o poder de processamento do computador para transformar, em informação digital, uma projeção comum oriunda de várias fontes com projeções diferentes.

2.5.5 Estruturas de dados

O mapa de uma propriedade relacionado a uma imagem de satélite pode ser um indicador oportuno de usos da terra? Sim, mas desde que os dados digitais são adquiridos e armazenados de vários modos, as duas fontes de dados podem não ser completamente compatíveis. Assim um SIG deve poder converter dados de uma estrutura para outra.

Dados de um SIG podem ser lidos da imagem de um satélite que foi interpretado por um computador para produzir um mapa de uso da terra em um formato conhecido como “raster”.

Estruturas Matriciais ou *Raster*

Este tipo de estrutura tem seus valores associados a uma matriz de células. Cada célula é um endereço identificado por coordenadas de linhas e colunas representando o mundo real, como está sendo exemplificado na figura 2.11 abaixo.



Figura 2.11: Representação de um mapa do mundo real através de estrutura matricial ou *raster* (Fonte: Silva, 2004)

Os arquivos de dados *raster* podem ser manipulados rapidamente pelo computador, mas eles são freqüentemente menos detalhados e visualmente menos agradáveis do que os dados de arquivos vetorizados que podem aproximar a aparência de mapas tradicionais traçados manualmente. Um exemplo seria a classificação da cobertura da terra.

Estruturas Vetoriais

Estruturas vetoriais representam mapas através de coordenadas X e Y, ou longitude e latitude, onde os símbolos do mundo real são localizados por pontos, linhas e polígonos, como exemplificado na figura 2.12 abaixo.



Figura 2.12: Representação do mundo real na estrutura vetorial

Um exemplo de dados tipicamente mantidos como um arquivo de vetores seriam as fronteiras de uma propriedade para subdivisão de moradia.

Reestruturação de dados pode ser executada por um SIG para converter dados em formatos diferentes. Por exemplo, um SIG pode ser usado para converter um mapa de uma imagem de satélite para uma estrutura de vetores, gerando linhas ao redor de todas as células

com a mesma classificação, enquanto determina as relações espaciais das células, tais como adjacência ou inclusão.

2.5.6 Modelagem de dados

É difícil relacionar mapas de terras com quantidade de chuva registrada em pontos diferentes como aeroportos, estações de televisão e escolas. Porém, um SIG pode ser usado para descrever características de duas e três dimensões da superfície, do subsolo e da atmosfera da Terra, oriundas de diversos pontos de informação. Por exemplo, um SIG pode gerar um mapa rapidamente com linhas que indicam quantidades de chuva. Tal mapa pode ser pensado como um mapa de contorno de chuva. Muitos métodos sofisticados podem calcular as características de superfícies de um número limitado de medidas de ponto. Um mapa de contorno bidimensional criado da superfície que modela as medidas de pontos de chuva pode ser sobreposto com qualquer outro mapa em um SIG que cobre a mesma área.

2.6 MAPSERVER

O Mapserver [38,39,40] é uma ferramenta para o desenvolvimento de aplicações que utilizam dados geográficos, ou seja, dados que descrevem os aspectos terrestres através do referenciamento a um sistema de coordenadas através de aplicações *web* (Internet/Intranet). A Internet é um conglomerado de redes onde dois ou mais computadores e outros dispositivos ligados entre si compartilham dados, impressoras, trocam mensagens (*e-mails*) etc., em escala mundial. Já a Intranet é uma rede de computadores privativa que utiliza as mesmas tecnologias que são utilizadas na Internet.

MapServer é um servidor de mapas, ele é um ambiente de desenvolvimento de código aberto para construção de aplicativos espaciais na Internet. MapServer não é um SIG, ao invés disso, MapServer se sobressai na apresentação de dados espaciais (mapas, imagens e dados vetoriais) na *web*. Além de permitir visualizar dados de SIGs, MapServer permite que se crie imagens de mapas geográficos, permite desenvolver soluções WebGIS corporativas, integrando diversos repositórios de dados geográficos com simplicidade e alta performance. Implementa algumas especificações do OGC (*Open Geospatial Consortium*) — Consórcio Geoespacial Livre — que é uma organização voluntária internacional de padrões de consenso. No OGC, mais de 280 organizações comerciais, governamentais, não-lucrativas e instituições de pesquisa do mundo todo colaboram em um processo de consenso aberto encorajando o desenvolvimento e a implementação de padrões para conteúdo e serviços geoespaciais, SIG,

processamento de dados e troca. Anteriormente era conhecido por Open GIS Consortium ou Consórcio OpenGIS.

2.6.1 História

O MapServer [41] foi originalmente desenvolvido pelo projeto ForNet da Universidade de Minnesota (UMN) em cooperação com a NASA e o Departamento de Recursos Naturais de Minnesota (MNDNR). Atualmente o projeto MapServer é abrigado pelo projeto TerraSIP, um projeto patrocinado conjuntamente pela NASA, UMN e um consórcio de interesses de gerenciamento da terra. O software é mantido por um número crescente de desenvolvedores (cerca de 20) de vários lugares do mundo e é patrocinado por um grupo de organizações que custeia melhorias e a manutenção.

A seguir uma descrição resumida da evolução do MapServer:

1996 -1997

O MapServer foi originalmente desenvolvido pelo projeto Fornet, parceria entre a UMN e o Minnesota DNR, sob patrocínio da NASA, o objetivo era prover equipes da área florestal com produtos obtidos a partir de imagens de sensoriamento remoto. Dentre os requisitos do projeto, havia a necessidade de se entregar dados geográficos via Internet. A primeira versão do MapServer foi o MapServer 1.0: desenvolvido em C, por Stephen Lime, para uso no projeto ForNet.

1998

Surge a versão MapServer 2.0, na qual o MapServer dá suporte a projeção cartográfica de dados e permite que ele possa ser utilizado fora do projeto ForNet.

2000

MapServer 3.0

TerraSIP: neste outro projeto financiado pela NASA, o desenvolvimento do MapServer continua. Daniel Morissette (da DM Solutions) adere ao desenvolvimento do MapServer. O mapserver passa a ter suporte para dados matriciais (através da biblioteca LibTIFF) e fontes TrueType (biblioteca FreeType) são adicionados, surge o *website* oficial do MapServer que torna-se então um projeto Open Source de conhecimento público.

2001 - 2002

Surge o MapScript através de uma iniciativa da empresa DM Solutions que disponibiliza a API do MapServer para a linguagem PHP com a extensão PHP/MapScript. Os mecanismos internos do MapServer são todos re-escritos para permitirem o uso de dados oriundos de bancos de dados espaciais (Oracle Spatial, PostGIS, suporte ArcSDE).

2003 - 2004

A versão MapServer 4.0 dá suporte a formatos matriciais de 24bits (entrada/saída), suporte a saída em formatos PDF e SWF (flash), a equipe de desenvolvimento já conta com mais de 10 integrantes, 800 usuários inscritos na lista de discussão oficial, ocorre o primeiro encontro de usuários MapServer, em St.Paul/Minnesota/E.U.A. que reuniu 110 visitantes, é inaugurado o Fórum MapServer com o objetivo de promover o uso do MapServer entre falantes da língua portuguesa, surge a lista de discussão Yahoo MapServer Brasil. Em 2004 ocorre o segundo Mapserver Users Meeting que reuniu 220 pessoas e ampliou o foco para outras tecnologias Open Source para SIG.

2005

Ocorre o terceiro MapServer users Meeting, 3 livros sobre MapServer chegam ao Mercado e são os seguintes: Web mapping Illustrated — Tyler Mitchell, MapServer: Open Source Gis Development— Bill Kropla, Mapping Hacks — Shuyler Erle.

2.6.2 Formatos de entrada e Saída

O MapServer pode ler dados nos formatos vetoriais e matriciais e pode gerar mapas em diversos formatos de saída. A tabela 2.1 lista os formatos de entrada e saída suportados pelo MapServer.

Tabela 2.1: Formatos de entrada e Saída do MapServer

Formatos de Entrada		Formatos de Saída
Dados Vetoriais	Dados Matriciais	PNG
ShapeFiles	TIFF/GeoTIFF	JPEG
ArcSDE	JPEG	GIF
Oracle Spatial	GIF	WBMP (para WAP)
PostGis	PNG	SWF (Flash)
MySql	EPPL7	PDF
Formatos OGR: a biblioteca OGR fornece acesso a diversos formatos vetoriais	Formatos GDAL: a biblioteca GDAL fornece acesso a diversos formatos matriciais	SVG
ODBC (somente tabelas de pontos)	ECW	DXF
MapInfo	ENVI	formatos GDAL
DGN	HDF	PNG
CSV	ERDAS	JPEG
GML	PCRaster	GIF

2.6.3 Funcionalidades

O Mapserver possui diversas funcionalidades e dentre elas podemos destacar:

- indexação espacial para ShapeFiles
- customização através de arquivos "template" ou MapScript
- seleção de objetos por ponto, área, valor ou item
- suporte a fontes TrueType
- legenda, barra de escala, mapa de referência e controles de navegação
- desenho de objetos de acordo com a escala
- mapas temáticos a partir de expressões lógicas, regulares ou constantes string
- sistema anti-colisão para rótulos (*labels*)
- reprojeção de dados cartográficos em tempo de execução (PROJ.4)
- configuração de parâmetros via URLs
- compatibilidade OGC

- WMS - *Web Map Service*
- WFS - *Web Feature Service*
- WCS - *Web Coverage Service*

2.6.4 Linguagens e Plataformas

O Mapserver foi desenvolvido para suportar as seguintes linguagens de programação: PHP, PYTHON, PERL, RUBY, JAVA e C#. Essa ferramenta foi projetada para as seguintes plataformas: LINUX, Windows [42], MAC OS, FREEBSD e SOLARIS.

2.6.5 Funcionamento

O MapServer [43] se sobressai na apresentação de dados espaciais (mapas, imagens e dados vetoriais) na *web*, permitindo dessa forma, a visualização de dados de um SIG, por exemplo. Para acessar os dados espaciais é necessário a execução de algumas etapas. A seguir uma descrição detalhada do funcionamento desse ciclo, ou seja, desde a requisição do usuário para visualizar um dado espacial, passando pelo processamento realizado pelo Mapserver para converter um dado espacial em uma imagem até a informação final disponibilizada para o usuário.

1. Navegador *web*: esse é o software de interface de usuário. Por meio de um navegador como FireFox, Konqueror, Internet Explorer, Netscape Navigator ou Opera, o usuário acessa um endereço que contém a aplicação *web*;
2. O servidor *web*, como o Apache ou *Internet Information Services* (IIS) é o software responsável pela publicação de textos, arquivos HTML, imagens e *hyperlinks* em uma conexão *web*;
3. O servidor *web* comunica-se com o MapServer, que é o componente responsável por realizar a leitura parametrizada de arquivos de dados geográficos (mapas), efetuar uma operação específica (aproximar, afastar, deslocar, classificar, localizar etc.) e converter o resultado de uma consulta em uma imagem (GIF, PNG, JPG);
4. Caso a aplicação necessite de um grau de customização além do padrão fornecido pelo software MapServer, existe a possibilidade de se efetuar uma customização mais aprimorada usando-se linguagens de programação;

5. Os dados geográficos devem ser armazenados em formato padrão suportado pelo Mapserver ou diretamente no banco de dados, caso esse suporte dados espaciais;
6. O banco de dados compreende as informações de interesse dos usuários e deve estar estruturado de modo que seus dados possam ser relacionados aos dados geográficos e conseqüentemente possibilite o usuário a obter respostas coesas relacionadas ao seu negócio.

A figura 2.13 a seguir exemplifica o funcionamento do MapServer descrito anteriormente.

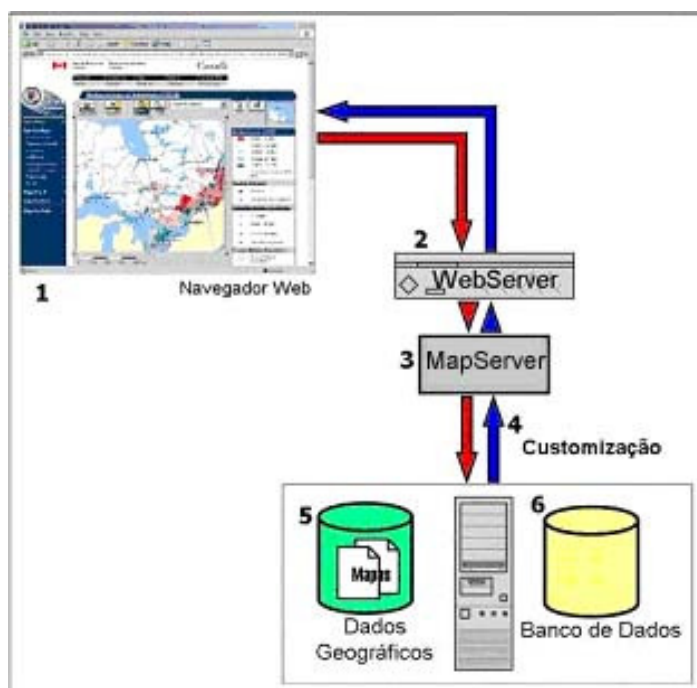


Figura 2.13: Funcionamento

A lógica do MapServer consiste em criar um objeto do tipo Mapa, calcular a extensão corrente do mapa e de acordo com o tipo de ferramenta solicitada, executar os seguintes passos: caso a ferramenta selecionada for para aproximar, afastar, mover ou identificar será criado um novo projeto a partir do clique do *mouse*, calcula-se a nova extensão e em seguida o mapa é redesenhado. Caso a ferramenta escolhida for para enquadrar, será realizado um novo cálculo para determinar a nova extensão do mapa e o mesmo será redesenhado. Caso a ferramenta for para atualizar, ocorre apenas o processo de redesenhar o mapa e por último caso a ferramenta escolhida for para consultar, será necessário ler o banco de dados e se o número de registros retornados for maior do que um, calcula-se a nova extensão do mapa para que esse seja desenhado. A figura 2.14 descreve a lógica de uma aplicação típica MapServer.

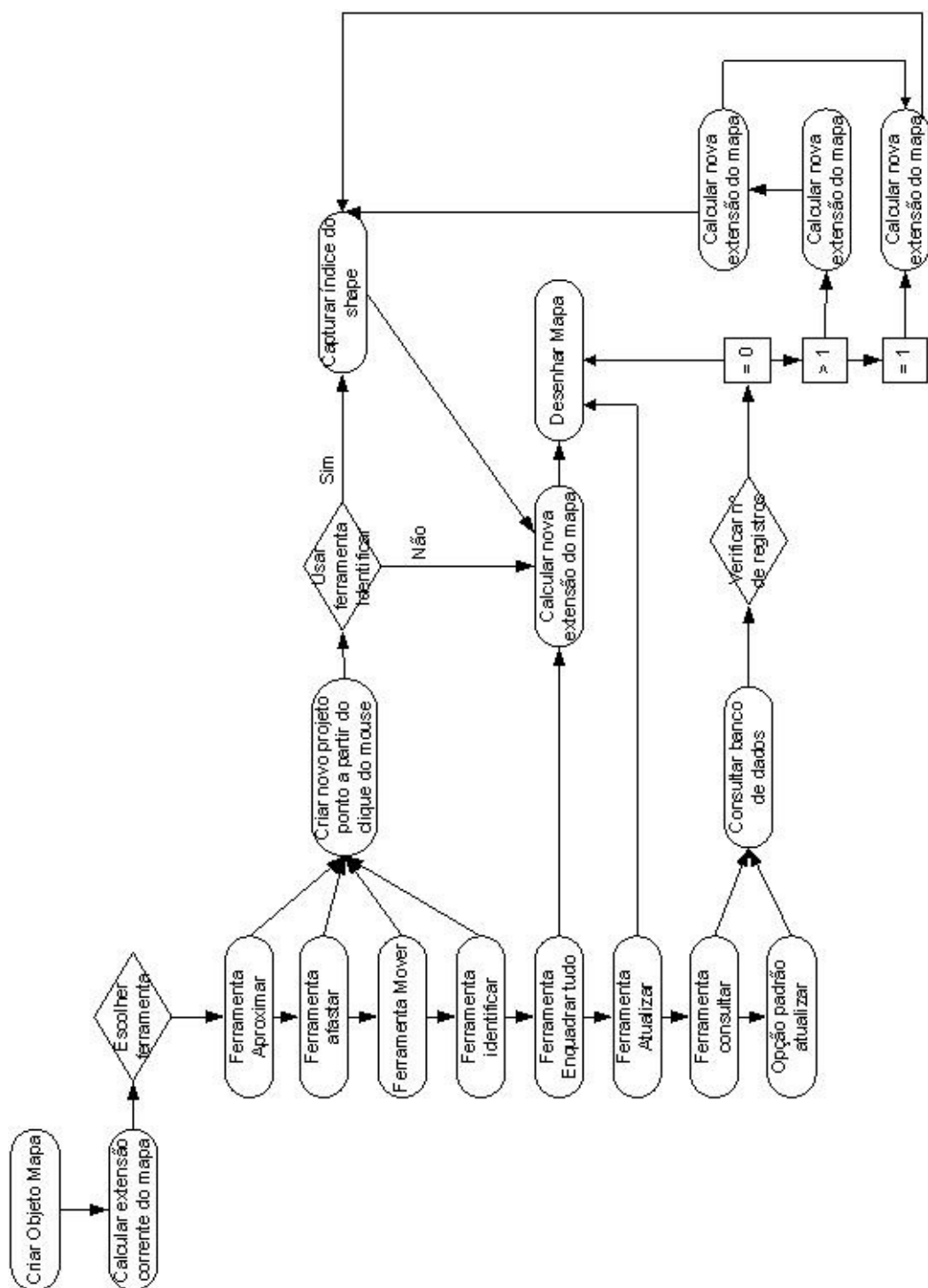


Figura 2.14: Lógica de uma aplicação típica MapServer

2.6.6 Modos de Utilização do MapServer

Pode-se trabalhar com o MapServer de três formas: CGI, MapScript e WebServices.

CGI

Esta é a maneira mais simples de se trabalhar com o MapServer. Quando se utiliza o MapServer em modo CGI, o seu arquivo executável deve ser colocado em diretório apropriado do servidor *web*. Esse executável irá receber parâmetros de iniciação da aplicação, processar as requisições solicitadas e retornar ao aplicativo cliente (navegador) o resultado esperado (imagens do mapa, legenda, barra de escala, mapa de referência, ou mesmo códigos HTML).

MapScript

Historicamente o conceito do MapScript foi introduzido em 2001 quando a canadense DM Solutions disponibilizou a API do MapServer para a linguagem de programação PHP, numa extensão chamada de PHP/MapScript. De maneira sucinta, o MapScript é a disponibilização dos recursos do MapServer para linguagens de programação. Dessa forma, pode-se combinar os recursos do MapServer com recursos da sua linguagem de programação preferida, visando a criação de aplicações com um grau de customização maior, eventualmente não atingido com aplicações do MapServer em modo CGI.

WebServices

Um WebService é um conjunto de funções chamáveis através da Internet. Um WebService permite que dois programas se comuniquem de uma maneira tecnicamente muito semelhante à invocação de páginas *web*.

Estrutura de uma Aplicação MapServer:

1. Mapas

Antes de mais nada, é preciso ter em mãos os mapas que se deseja publicar com uma aplicação MapServer. Os mapas são os dados de entrada de sua aplicação e devem estar em um formato que possa ser lido pelo MapServer.

2. MapFile

O MapFile é um arquivo de extensão .map, em formato texto puro, que faz todas as definições e configurações iniciais necessárias para execução de uma aplicação MapServer. Esse arquivo é lido pelo MapServer em cada interação do usuário com a aplicação e define diversas características da aplicação como: quais mapas serão disponibilizados, como esses mapas serão apresentados, com qual cor, com qual símbolo, até qual escala o usuário poderá aproximar-se, ou seja, o MapFile define como os MAPAS (dados) serão apresentados ao usuário.

3. Formulário de Iniciação

Em aplicações MapServer no modo CGI, é necessário a presença de um “formulário de iniciação” da aplicação. Esse formulário é uma declaração em HTML que enviará ao executável do MapServer parâmetros básicos para a iniciação da aplicação, tais como o caminho do MapFile e endereço (URL) do MapServer CGI.

4. Arquivos *Template*

Os arquivos *template* definem a interface ou *design* da aplicação. Em outras palavras, definem como os componentes gerados pelo MapServer (mapa, legenda, barra de escala etc.) serão apresentados para o usuário e de que forma o usuário poderá interagir com a aplicação.

2.7 COMENTÁRIOS

Neste capítulo, procurou-se expor, de maneira sintética, o processo de cobrança, os principais conceitos da lógica nebulosa, que foram empregados no desenvolvimento do sistema, os conceitos e o uso dos Sistemas Gerenciadores de Banco de Dados (SGBD), o conceito necessário para o desenvolvimento de um Sistema de Informação Geográfica (SIG) e o entendimento da ferramenta MapServer, que foi escolhida para o desenvolvimento do SIGATDSC.

CAPÍTULO 3

Modelagem do SIGATDSC

3.1 INTRODUÇÃO

Foi utilizada a *Unified Modeling Language* (UML) [44,45] para modelar o sistema e a ferramenta *Java and UML Developer Environment* (JUDE) [46] para elaborar os diagramas. Na seção 3.2, são apresentados o diagrama de casos de uso, os diagramas de classe, os diagramas de seqüência e o diagrama de estado utilizados para a modelagem do SIGATDSC. As seções 3.3 e 3.4 mostram como foi implementada a técnica da lógica nebulosa para o sistema. Na seção 3.3, as variáveis lingüísticas são apresentadas e descritas, assim como os termos dessas variáveis e suas respectivas funções de pertinência. Na seção 3.4, são apresentadas e definidas as regras de inferência que foram usadas pelo sistema de inferência criado.

3.2 MODELOS DE ANÁLISE

3.2.1 Modelo de Casos de Uso

A figura 3.1 a seguir corresponde ao diagrama de casos de uso do SIGATDSC:

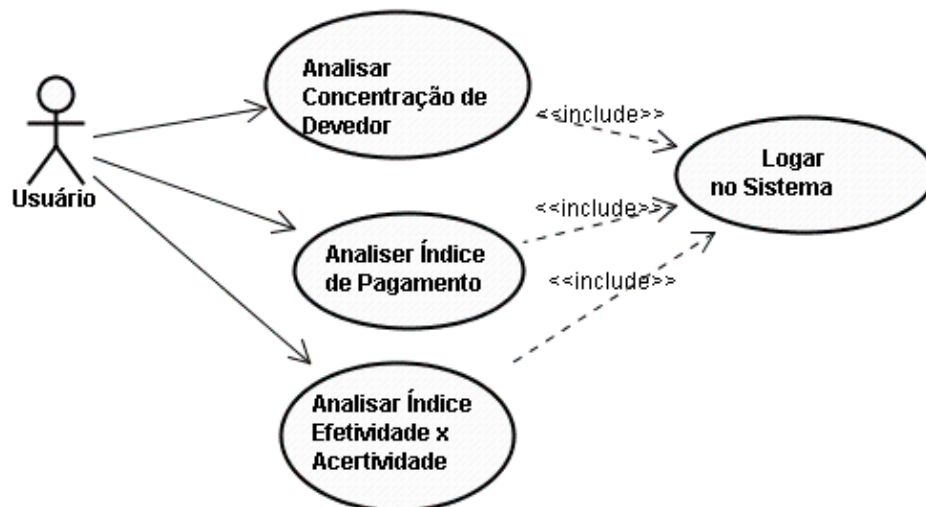


Figura 3.1: Diagrama de Casos de Uso do SIGATDSC.

A análise de concentração de devedor pode ser realizada por cidade, estado ou bairro, para não deixar o diagrama de casos de uso complexo, fez-se um caso de uso “Analisar concentração de devedor”. Na especificação dos casos de uso a seguir, é apresentada a descrição detalhada dos casos de uso analisar concentração de devedor por cidade, estado e bairro.

Descrição dos Casos de Uso

Nome do caso de uso: *Logar* no sistema

Pré-Condição: Nenhuma

Objetivo: Validar os usuários que acessam o sistema.

Atores: Supervisor e Administrador da Empresa.

Cenário Principal

1. Usuário informa *login* e senha.
2. Sistema verifica *login* e senha.
3. Sistema exibe a página inicial do SIGATDSC.

Cenário Alternativo 1

Login Inválido – Usuário informa um *login* inválido para acessar o sistema.

1. Usuário informa *login* e senha.
2. Sistema verifica *login* e senha.
3. Sistema emite mensagem “Usuário e/ou senha inválidos. Verifique se você digitou corretamente”.

Cenário Alternativo 2

Senha Inválida – Usuário informa uma senha inválida para acessar o sistema.

1. Usuário informa *login* e senha.
2. Sistema verifica *login* e senha.
3. Sistema emite mensagem “Usuário e/ou senha inválidos. Verifique se você digitou corretamente”.

Nome do caso de uso: Analisar Concentração de Devedor por Cidade

Pré-Condição: Nenhuma

Objetivo: Permitir analisar a concentração de devedor por cidade

Atores: Supervisor e Administrador da Empresa

Cenário Principal

1. Sistema carrega as empresas da assessoria de cobrança
2. Usuário seleciona a empresa
3. Usuário informa o tipo de Consulta: “Concentração por devedor”
4. Usuário seleciona a opção “cidade”
5. Usuário seleciona o tipo de exibição
 - 5.1. Se tipo de exibição for “exibir todas as concentrações de devedores”
 - 5.1.1. Usuário clica em “Gerar Mapa”
 - 5.1.2. Sistema mostra a concentração de devedores para cada cidade
 - 5.2. Senão Se tipo de exibição for “exibir a maior concentração de devedor”
 - 5.2.1. Usuário clica em “Gerar Mapa”
 - 5.2.2. Sistema mostra a cidade com maior concentração de devedor
 - 5.3. Senão Se tipo de exibição for “exibir a menor concentração de devedor”
 - 5.3.1. Usuário clica em “Gerar Mapa”
 - 5.3.2. Sistema mostra a cidade com menor concentração de devedor
 - 5.4. Senão Se tipo de exibição for “Personalizar”
 - 5.4.1. Sistema carrega as cidades dos devedores da empresa selecionada
 - 5.4.2. Usuário seleciona a cidade para exibir a concentração de devedor
 - 5.4.3. Usuário clica em “Gerar Mapa”
 - 5.4.4. Sistema exibe a concentração de devedor da cidade selecionada

Nome do caso de uso: Analisar Concentração de Devedor por Estado

Pré-Condição: Nenhuma

Objetivo: Permitir analisar a concentração de devedor por estado

Atores: Supervisor e Administrador da Empresa.

Cenário Principal

1. Sistema Carrega as empresas da assessoria de cobrança
2. Usuário seleciona a empresa
3. Usuário informa o tipo de Consulta: “Concentração por devedor”
4. Usuário seleciona a opção “estado”
5. Usuário seleciona o tipo de exibição
 - 5.1. Se tipo de exibição for “exibir todas as concentrações de devedores”
 - 5.1.1. Usuário clica em “Gerar Mapa”
 - 5.1.2. Sistema mostra a concentração de devedores para cada estado
 - 5.2. Senão Se tipo de exibição for “exibir a maior concentração de devedor”
 - 5.2.1. Usuário clica em “Gerar Mapa”
 - 5.2.2. Sistema mostra o estado com maior concentração de devedores
 - 5.3. Senão Se tipo de exibição for “exibir a menor contração de devedor”
 - 5.3.1. Usuário clica em “Gerar Mapa”
 - 5.3.2. Sistema mostra o estado com menor concentração de devedores
 - 5.4. Senão Se tipo de exibição for “personalizar”
 - 5.4.1. Sistema carrega os estados dos devedores da empresa selecionada
 - 5.4.2. Usuário seleciona o estado
 - 5.4.3. Usuário clica em “Gerar Mapa”
 - 5.4.4. Sistema exibe a concentração de devedor do estado selecionado

Nome do caso de uso: Analisar Concentração de Devedor por Bairro

Pré-Condição: Nenhuma

Objetivo: Permitir analisar a concentração de devedor por bairro

Atores: Supervisor e Administrador da Empresa.

Cenário Principal

1. Sistema Carrega as empresas da assessoria de cobrança
2. Usuário seleciona a empresa
3. Usuário informa o tipo de Consulta: “Concentração por devedor”
4. Usuário seleciona a opção Bairro
5. Usuário seleciona o tipo de exibição
 - 5.1. Se tipo de exibição for “exibir a maior concentração de devedor”

- 5.1.1. Usuário clica em “Gerar Mapa”
- 5.1.2. Sistema mostra o bairro com maior concentração de devedores
- 5.2. Senão Se tipo de exibição for “exibir a menor concentração de devedor”
 - 5.2.1. Usuário clica em “Gerar Mapa”
 - 5.2.2. Sistema mostra o bairro com menor concentração de devedores
- 5.3. Senão Se tipo de exibição for “Personalizar”
 - 5.3.1. Sistema carrega as cidades dos devedores da empresa selecionada
 - 5.3.2. Usuário seleciona a cidade
 - 5.3.3. Sistema carrega os bairros da cidade selecionada
 - 5.3.4. Usuário seleciona o bairro para exibir a concentração de devedor
 - 5.3.5. Usuário clica em “Gerar Mapa”
 - 5.3.6. Sistema exibe a concentração de devedor do bairro selecionado

Analisar o Índice de Pagamento

A análise do índice de pagamento pode ser realizada por cidade, estado e bairro. Essa análise tem como objetivo localizar geograficamente as regiões onde ocorre maior e menor índice de pagamento, por exemplo. Para não deixar o diagrama de casos de uso complexo fez-se um diagrama de casos de uso Analisar Índice de Pagamento. A seguir a descrição detalhada dos casos de uso analisar o índice de pagamento por cidade, estado e bairro.

Nome do caso de uso: Analisar o Índice de Pagamento por Cidade

Pré-Condição: Nenhuma

Objetivo: Permitir analisar o índice de pagamento por cidade

Atores: Supervisor e Administrador da Empresa.

Cenário Principal

1. Sistema Carrega as empresas da assessoria de cobrança
2. Usuário seleciona a empresa
3. Usuário informa o tipo de Consulta: “Índice de Pagamento”
4. Usuário seleciona a opção “cidade”
5. Usuário seleciona o tipo de exibição
 - 5.1. Se tipo de exibição for “exibir todos os índices de pagamento”
 - 5.1.1. Usuário clica em “Gerar Mapa”

- 5.1.2. Sistema mostra os índices de pagamento para cada cidade
- 5.2. Senão Se tipo de exibição for “exibir o maior índice de pagamento”
 - 5.2.1. Usuário clica em “Gerar Mapa”
 - 5.2.2. Sistema mostra a cidade com maior índice de pagamento
- 5.3. Senão Se tipo de exibição for “exibir o menor índice de pagamento”
 - 5.3.1. Usuário clica em “Gerar Mapa”
 - 5.3.2. Sistema mostra a cidade com menor índice de pagamento
- 5.4. Senão Se tipo de exibição for “Personalizar”
 - 5.4.1. Sistema carrega as cidades dos devedores da empresa selecionada
 - 5.4.2. Usuário seleciona a cidade para exibir o índice de pagamento
 - 5.4.3. Usuário clica em gerar mapa
 - 5.4.4. Sistema exibe o índice de pagamento da cidade selecionada

Nome do caso de uso: Analisar o Índice de Pagamento por Estado

Pré-Condição: Nenhuma

Objetivo: Permitir analisar os índices de pagamento de devedor por estado

Atores: Supervisor e Administrador da Empresa.

Cenário Principal

1. Sistema Carrega as empresas da assessoria de cobrança
2. Usuário seleciona a empresa
3. Usuário informa o tipo de Consulta: “Índice de Pagamento”
4. Usuário seleciona a opção “estado”
5. Usuário seleciona o tipo de exibição
 - 5.1. Se tipo de exibição for “exibir todos os índices de pagamento”
 - 5.1.1. Usuário clica em “Gerar Mapa”
 - 5.1.2. Sistema mostra os índices de pagamento para cada estado
 - 5.2. Senão Se tipo de exibição for “exibir o maior índice de pagamento”
 - 5.2.1. Usuário clica em “Gerar Mapa”
 - 5.2.2. Sistema mostra o estado com maior índice de pagamento
 - 5.3. Senão Se tipo de exibição for “exibir o menor índice de pagamento”
 - 5.3.1. Usuário clica em “Gerar Mapa”
 - 5.3.2. Sistema mostra o estado com menor índice de pagamento
 - 5.4. Senão Se tipo de exibição for “Personalizar”

- 5.4.1. Sistema carrega os estados dos devedores da empresa selecionada
- 5.4.2. Usuário seleciona o estado para exibir o índice de pagamento
- 5.4.3. Usuário clica em “Gerar Mapa”
- 5.4.4. Sistema exibe o índice de pagamento do estado selecionado

Nome do caso de uso: Analisar o Índice de Pagamento por Bairro

Pré-Condição: Nenhuma

Objetivo: Permitir analisar os índices de pagamento por bairro

Atores: Supervisor e Administrador da Empresa.

Cenário Principal

- 1. Sistema Carrega as empresas da assessoria de cobrança
- 2. Usuário seleciona a empresa
- 3. Usuário informa o tipo de Consulta: “Índice de Pagamento”
- 4. Usuário seleciona a opção “bairro”
- 5. Usuário seleciona o tipo de exibição
 - 5.1. Se tipo de exibição for “exibir o maior índice de pagamento”
 - 5.1.1. Usuário clica em “Gerar Mapa”
 - 5.1.2. Sistema mostra o bairro com maior índice de pagamento
 - 5.2. Senão Se tipo de exibição for “exibir o menor índice de pagamento”
 - 5.2.1. Usuário clica em “Gerar Mapa”
 - 5.2.2. Sistema mostra o estado com menor índice de pagamento
 - 5.3. Senão Se tipo de exibição for “Personalizar”
 - 5.3.1. Sistema carrega as cidades dos devedores da empresa selecionada
 - 5.3.2. Usuário seleciona a cidade
 - 5.3.3. Sistema carrega os bairros da cidade selecionada
 - 5.3.4. Usuário seleciona o bairro para exibir o índice de pagamento
 - 5.3.5. Usuário clica em “Gerar Mapa”
 - 5.3.6. Sistema exibe o índice de pagamento do bairro selecionado

Analisar Índice de Efetividade x Acertividade

A análise do Índice de Efetividade x Acertividade pode ser realizada apenas por bairro. Essa análise permite avaliar o grau de risco em um determinado bairro levando em

consideração o percentual de acertividade e efetividade. A seguir a descrição detalhada do caso de uso analisar Índice de Efetividade x Acertividade por bairro.

Nome do caso de uso: Analisar Índice Efetividade x Acertividade por Bairro

Pré-Condição: Nenhuma

Objetivo: Permitir analisar os índices efetividade x acertividade por bairro

Atores: Supervisor e Administrador da Empresa.

Cenário Principal

1. Sistema Carrega as empresas da assessoria de cobrança
2. Usuário seleciona a empresa
3. Usuário informa o tipo de Consulta: “Índice efetividade x acertividade”
4. Usuário seleciona a opção “bairro”
5. Usuário seleciona o tipo de exibição “Personalizar”
6. Sistema carrega as cidades dos devedores da empresa selecionada
7. Usuário seleciona a cidade
8. Sistema carrega os bairros da cidade selecionada
9. Usuário seleciona o bairro para exibir o índice efetividade x acertividade
10. Usuário clica em “Gerar Análise”
11. Sistema exibe o Grau de Risco do bairro selecionado

3.2.2 Diagrama de Classe do SIGATDSC

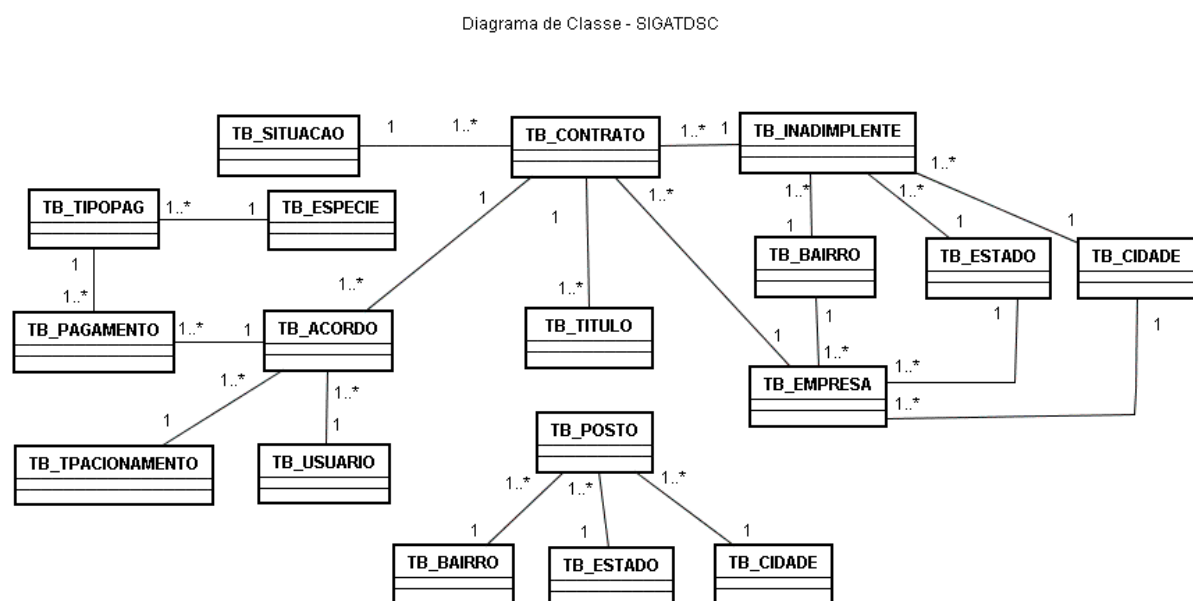


Figura 3.2: Diagrama de Negócio do SIGSATDSC

3.2.3 Diagrama de Classe Detalhado

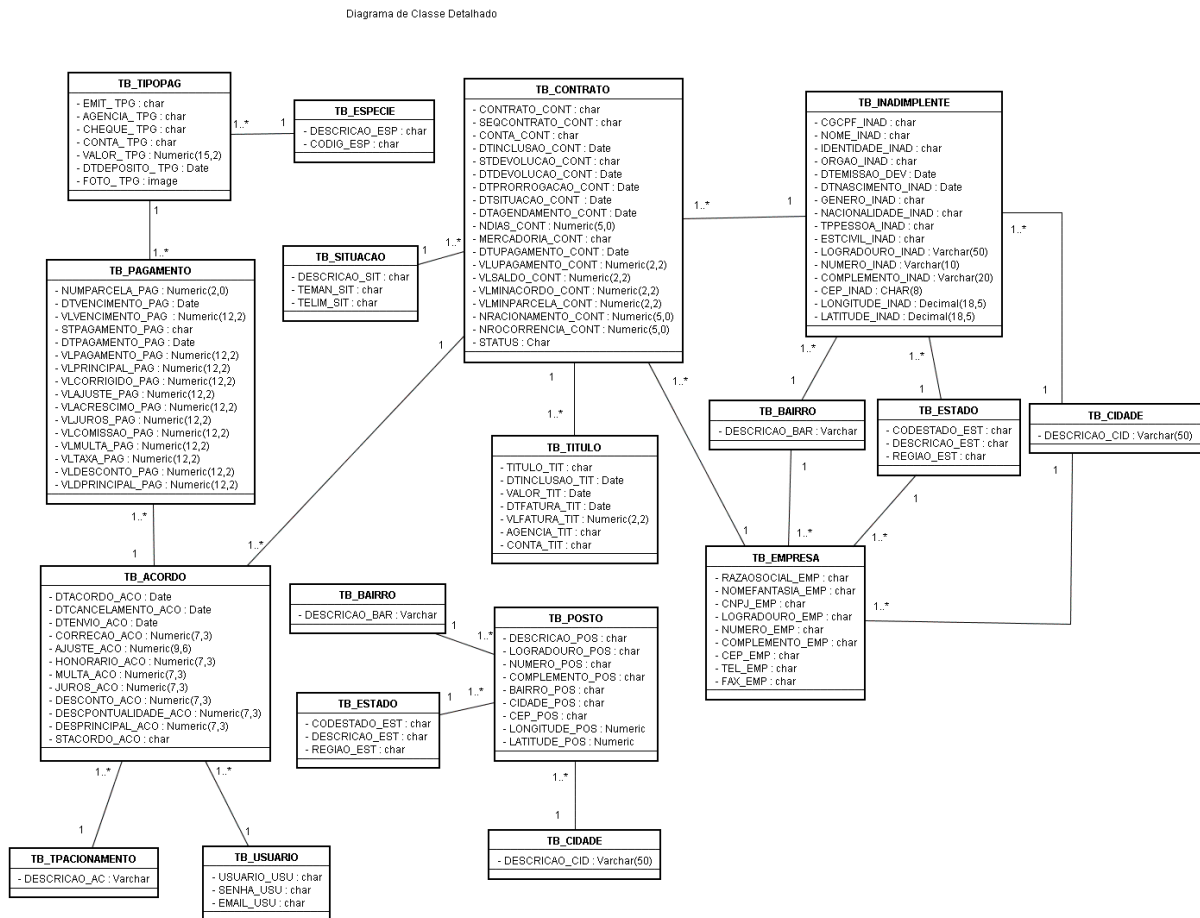


Figura 3.3: Diagrama de Classe Detalhado do SIGSATDSC

3.2.4 Diagrama de Seqüência

Diagrama de seqüência para Logar no Sistema

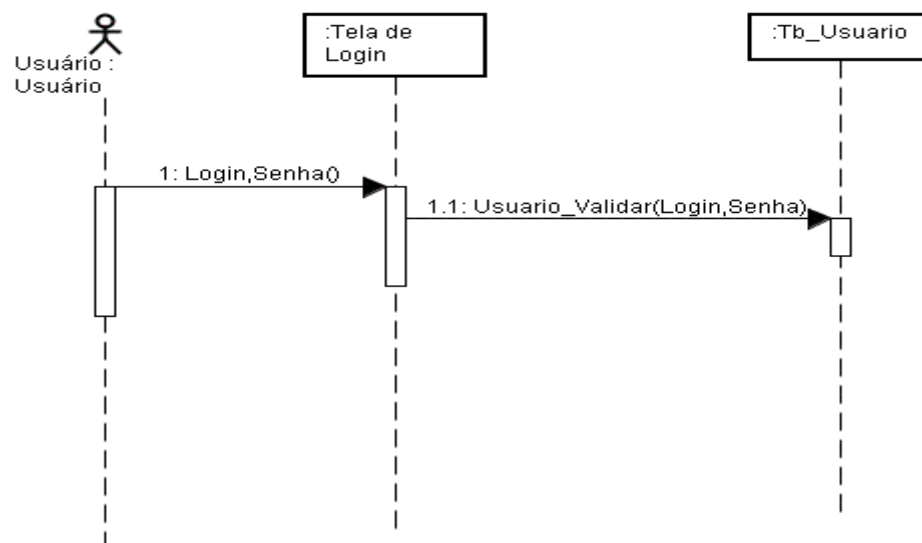


Figura 3.4: Diagrama de Seqüência – Logar no Sistema

Com a finalidade de evitar diagramas de seqüências semelhantes, mudando apenas o tipo de consulta e a forma de exibição, foi elaborado um diagrama de seqüência padrão para esses casos. A figura 3.5 a seguir atende às seguintes formas de consultas:

- Concentração de devedor por cidade e estado para os seguintes modos de exibição: exibir concentração de devedor, exibir a maior concentração de devedor ou a menor concentração de devedor.
- Concentração de devedor por Bairro para os seguintes modos de exibição: exibir a maior concentração de devedor e a menor concentração de devedor.
- Índice de pagamento por cidade e estado para os seguintes modos de exibição: exibir concentração de devedor, exibir a maior concentração de devedor e a menor concentração de devedor.
- Índice de pagamento por bairro para os seguintes modos de exibição: exibir a maior concentração de devedor e a menor concentração de devedor.

Diagrama de seqüência para Consulta por Cidade, Estado ou Bairro

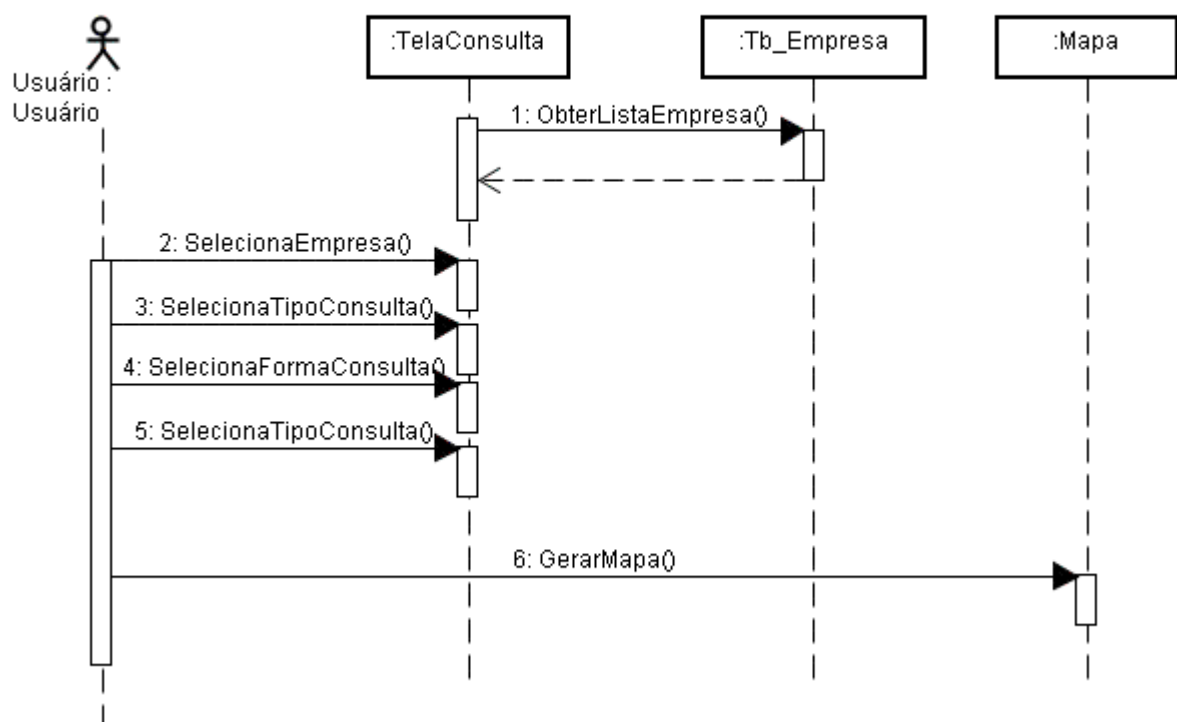


Figura 3.5: Diagrama de Seqüência – Consulta por Cidade, Estado ou Bairro

Com a finalidade de evitar diagramas de seqüências semelhantes para a forma de exibição “personalizada” e para o tipo de consulta por cidade, foi elaborado um diagrama de seqüência padrão para esse caso. A figura 3.6 a seguir atende às seguintes formas de

consultas:

- Concentração de devedor por cidade
- Índice de pagamento por cidade

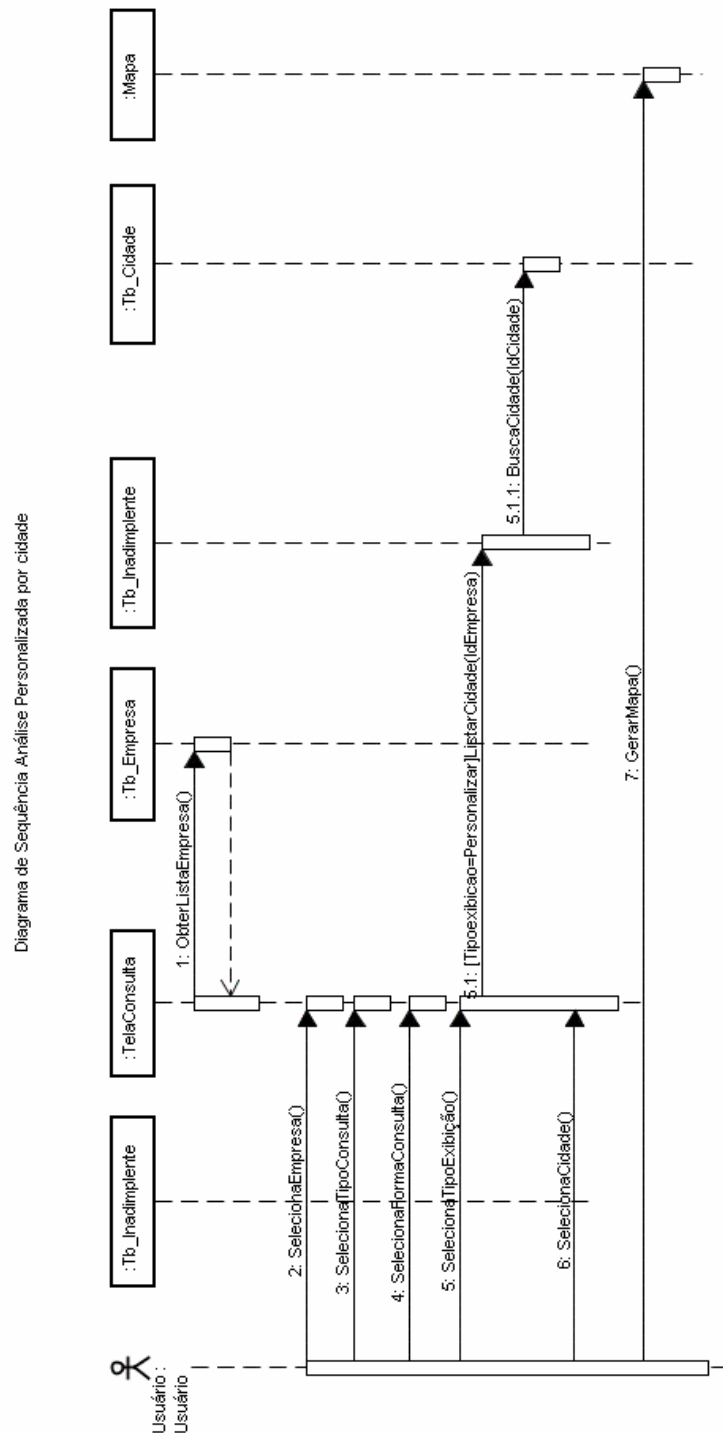


Figura 3.6: Diagrama de Seqüência – Análise personalizada por Cidade

Com a finalidade de evitar diagramas de seqüências semelhantes para a forma de exibição “personalizada” e para o tipo de consulta por estado, foi elaborado um diagrama de

seqüência padrão para esse caso. A figura 3.7 a seguir atende às seguintes formas de consultas:

- Concentração de devedor por estado
- Índice de pagamento por estado

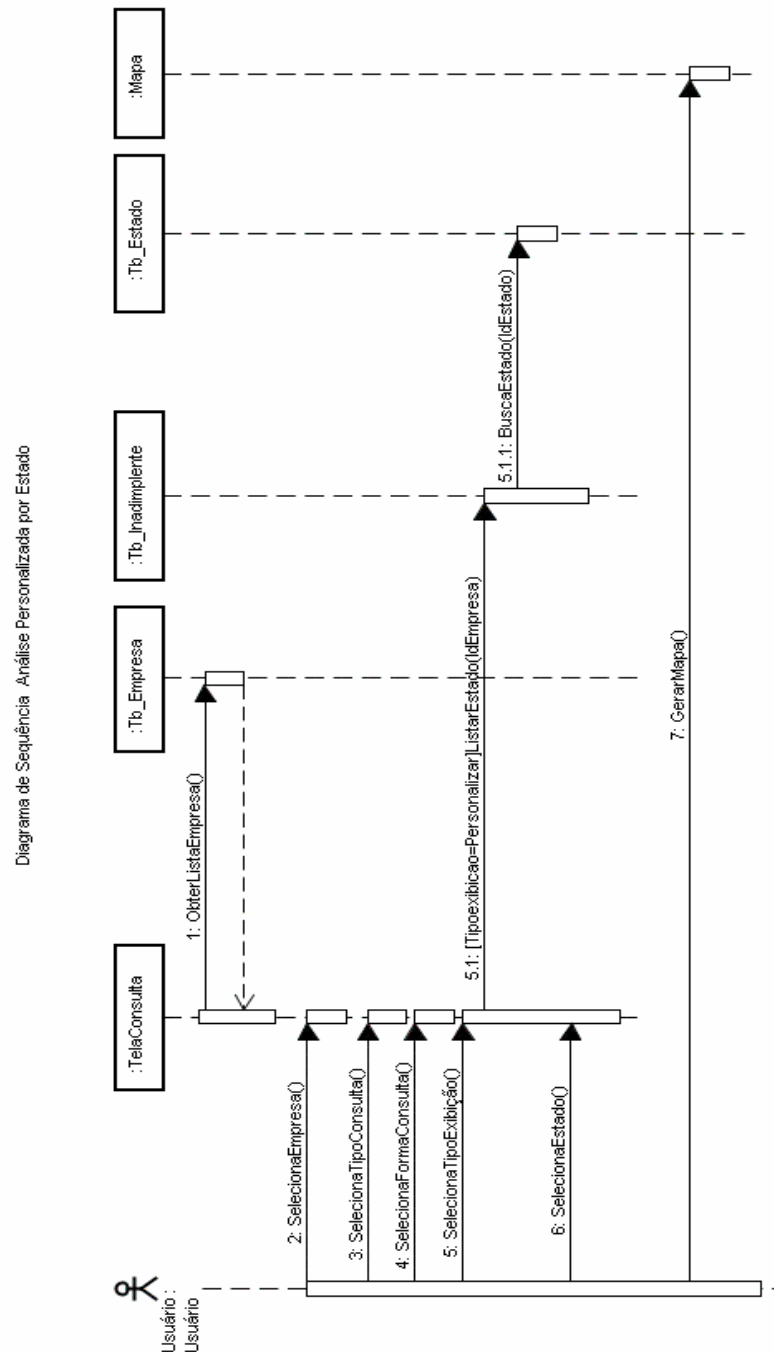


Figura 3.7: Diagrama de Seqüência – Análise personalizada por estado

Com a finalidade de evitar diagramas de seqüências semelhantes para a forma de exibição “personalizada” e para o tipo de consulta por bairro, foi elaborado um diagrama de seqüência padrão para esse caso. A figura 3.8 a seguir atende às seguintes formas de consultas:

- Concentração de devedor por bairro
- Índice de pagamento por bairro
- Índice de efetividade x acertividade por bairro

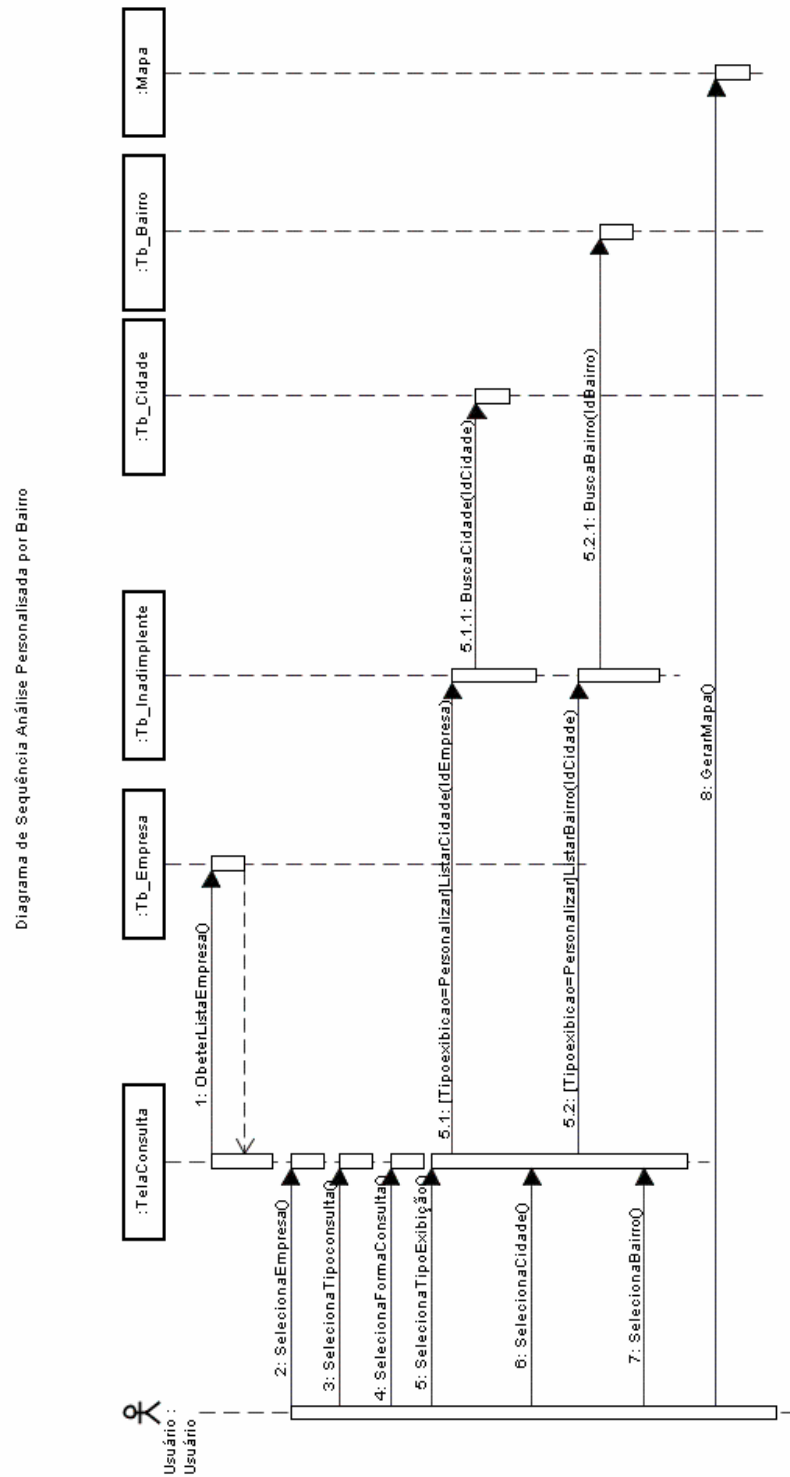


Figura 3.8: Diagrama de Sequência – Análise personalizada por Bairro

3.2.5 Diagrama de Estado

No diagrama de classe, tem-se a classe Tb_Contrato que possui o atributo situação do contrato. O contrato de um devedor está em uma determinada situação ao longo do tempo, por esse motivo foi elaborado o diagrama de estado a seguir para apresentar as possíveis situações de um contrato.

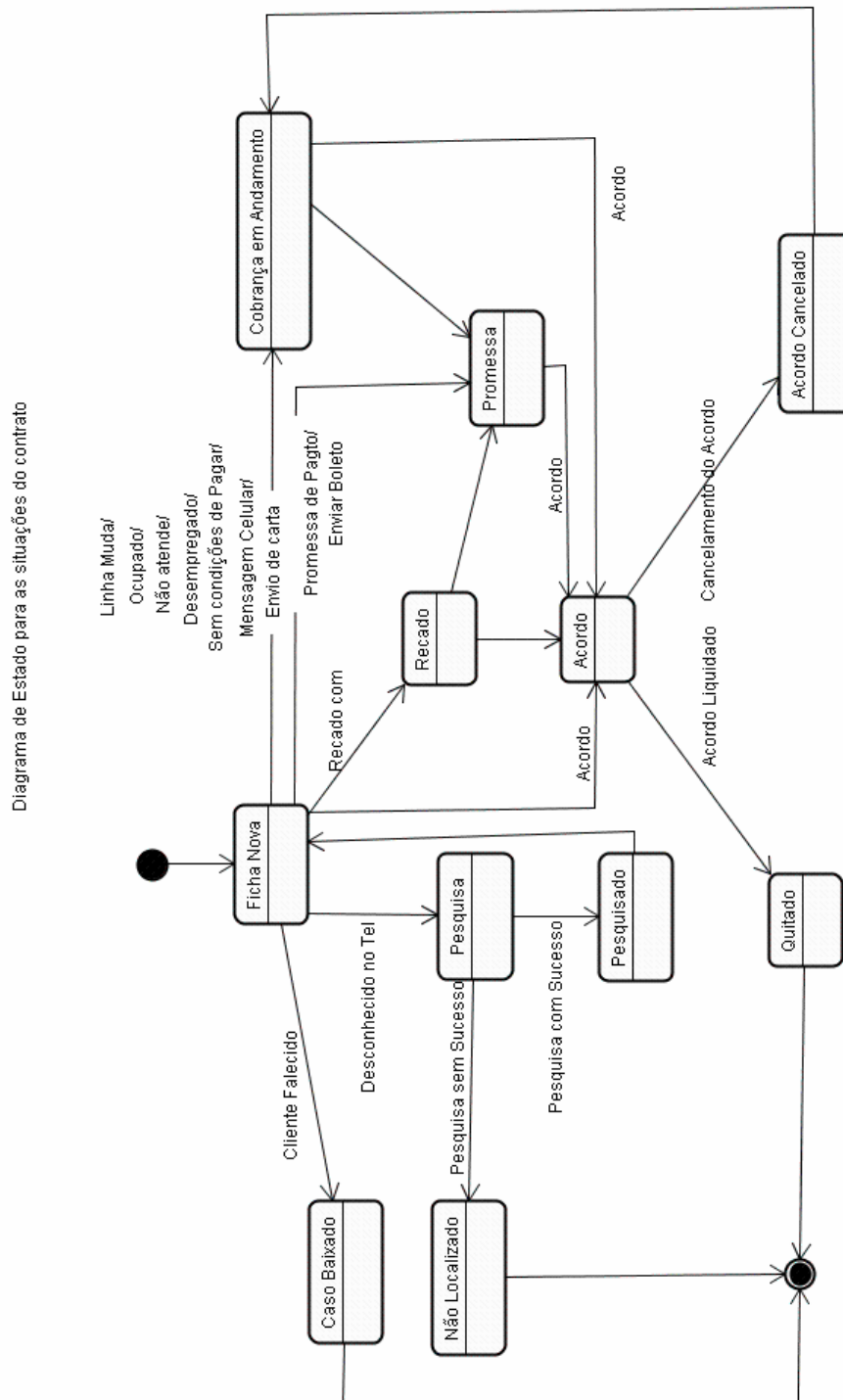


Figura 3.9: Diagrama de Estado

3.3 MODELO DO SISTEMA DE INFERÊNCIA NEBULOSA

A implementação do motor de inferência do SIGATDSC teve início com a escolha das variáveis lingüísticas do Sistema de Lógica Nebulosa (SLN) e com a definição dos seus respectivos termos. O SLN, responsável por indicar o grau de risco após adotar uma estratégia de cobrança em um determinado bairro, possui 25 regras de inferência as quais por sua vez possuem dois antecedentes e um conseqüente.

Todos os termos de todas as variáveis lingüísticas do SLN desenvolvido têm suas funções de pertinência montadas com segmentos de reta, definindo funções lineares, na maioria das vezes apresentando perfis trapezoidais. Assim, deve-se calcular o coeficiente linear e o coeficiente angular para cada parte de cada função de pertinência. Como cada função de reta é definida através da fórmula:

$$y = Mx + Q \quad (3.1)$$

Onde:

$M \rightarrow$ é o coeficiente angular;

$Q \rightarrow$ é o coeficiente linear.

Dessa forma, como podemos obter os valores de y , x e Q diretamente sobre o gráfico, basta calcularmos o coeficiente angular (M), através da fórmula:

$$M = \frac{y - Q}{x} \quad (3.2)$$

Uma vez que os coeficientes lineares e os coeficientes angulares foram calculados para todas as funções de pertinência, todas as partes das funções de pertinência encontram-se codificadas em um banco de dados, de modo que basta atualizar o banco para se alterar, incluir ou excluir uma parte de uma função de pertinência, sem que haja a necessidade de se modificar o código interno do programa. Dessa forma, podemos facilmente alterar o número de partes de uma função de pertinência ou mesmo os intervalos iniciais e finais, assim como o universo de discurso U de uma variável lingüística.

A seguir são apresentadas as variáveis lingüísticas conforme a sua definição pela quintupla $(x, T(x), U, G, M(X))$, no entanto, a gramática G não será usada neste caso. As

funções de pertinência dos termos das variáveis lingüísticas definidas como antecedentes e como conseqüente encontram-se no apêndice A1.

- Para o primeiro antecedente. Variável lingüística: acertividade

Tabela 3.1: Descrição da variável lingüística: acertividade

X	Acertividade
T(x)	{MuitoBaixa, Baixa, Média, Alta, MuitoAlta}
U	[0%,100%]
M:	$M(\text{MuitoBaixa}) = \{(u, \mu_{\text{MuitoBaixa}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{Baixa}) = \{(u, \mu_{\text{Baixa}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{Média}) = \{(u, \mu_{\text{Média}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{Alta}) = \{(u, \mu_{\text{Alta}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{MuitoAlta}) = \{(u, \mu_{\text{MuitoAlta}}(u)) \mid u \in [0\%, 100\%]\}$

- Para o segundo antecedente. Variável lingüística: efetividade

Tabela 3.2: Descrição da variável lingüística: efetividade

X	Efetividade
T(x)	{MuitoBaixa, Baixa, Média, Alta, MuitoAlta}
U	[0%,100%]
M:	$M(\text{MuitoBaixa}) = \{(u, \mu_{\text{MuitoBaixa}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{Baixa}) = \{(u, \mu_{\text{Baixa}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{Média}) = \{(u, \mu_{\text{Média}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{Alta}) = \{(u, \mu_{\text{Alta}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{MuitoAlta}) = \{(u, \mu_{\text{MuitoAlta}}(u)) \mid u \in [0\%, 100\%]\}$

A variável lingüística do conseqüente caracteriza a resposta sobre a consulta da efetividade x acertividade de uma dada região, logo, essa variável recebe o nome de Grau de Risco e seus termos são apresentados a seguir:

Para o conseqüente. Variável lingüística: Grau de Risco

Tabela 3.3: Descrição da variável lingüística: Grau de Risco

X	Grau de Risco
T(x)	{MuitoBaixo, Baixo, Médio, Alto, MuitoAlto}
U	[0%,100%]
M:	$M(\text{MuitoBaixo}) = \{(u, \mu_{\text{MuitoBaixo}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{Baixo}) = \{(u, \mu_{\text{Baixo}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{Médio}) = \{(u, \mu_{\text{Médio}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{Alto}) = \{(u, \mu_{\text{Alto}}(u)) \mid u \in [0\%, 100\%]\}$ $M(\text{MuitoAlto}) = \{(u, \mu_{\text{MuitoAlto}}(u)) \mid u \in [0\%, 100\%]\}$

3.4 DEFINIÇÃO DAS REGRAS DE INFERÊNCIA NEBULOSA

O SIGATDSC utiliza a combinação de todas as regras nebulosas para inferir e gerar uma resposta com a melhor precisão possível. Podemos obter o total de regras do sistema, através da multiplicação da quantidade de termos encontrados em cada uma das variáveis lingüísticas, através de um arranjo. Através de um arranjo foram então geradas todas as regras necessitando então adicionar os termos da variável lingüística Grau de Risco para cada regra. Posteriormente todas foram incluídas na tabela de Regras do SIGATDSC.

Tabela 3.4: Variáveis e quantidade de termos lingüísticos

Variável Lingüística	Numero de Termos Lingüísticos
Acertividade	5
Efetividade	5
Número Total de Regras	25

A tabela 3.5 mostra as 25 regras de inferência do SLN no formato FAM (*Fuzzy Associative Memory*), sendo que o detalhamento de cada regra encontra-se no apêndice A2. Tais regras encontram-se codificadas em um banco de dados de modo que basta atualizar o banco para se alterar, incluir ou excluir uma regra, sem que haja a necessidade de se modificar o código interno do programa.

Tabela 3.5: Regras definidas para o SLN

		Efetividade				
Acertividade	GRAU DE RISCO	MB	B	M	A	MA
	MB	MA	MA	A	A	M
	B	MA	A	A	M	M
	M	A	M	M	M	B
	A	M	M	B	B	MB
	MA	M	M	B	MB	MB

Legenda: MA-Muito Alto, A – Alto, M- Médio, B –Baixo e MB –Muito Baixo

3.5 COMENTÁRIOS

Neste capítulo, observou-se a modelagem do sistema seguindo a metodologia da UML. Foram apresentadas as partes das funções de pertinência e as regras utilizadas na modelagem do Sistema de Inferência Nebulosa. No próximo capítulo, será apresentada a implementação do SIGATDSC.

CAPÍTULO 4

Implementação do SIGATDSC

4.1 INTRODUÇÃO

Este projeto foi desenvolvido utilizando em conjunto as facilidades da linguagem HTML para a criação de interfaces de entradas e de apresentação da resposta gerada pelo usuário e a tecnologia PHP [47,48,49] utilizando a API do MapServer disponível para esta linguagem de programação, ou seja, PHP/Mapscript. Neste capítulo, utilizam-se os conceitos e técnicas mostradas nos capítulos anteriores para o desenvolvimento do trabalho. Inicialmente, na seção 4.2, é mostrado o processo de importação dos dados que se trata de um dos pré-requisitos para o funcionamento do SIGATDSC.

Na seção 4.3, será apresentado o funcionamento lógico do SIGATDSC e serão definidas as interfaces do usuário para a manipulação do sistema. São ainda apresentadas as interfaces do usuário, para entrada de dados e o recebimento da resposta gerada pelo sistema. Na seção 4.4, será apresentada a arquitetura do sistema, a descrição de algumas classes e como foram calculados os percentuais de acurácia e efetividade, que são os antecedentes do sistema de inferência nebulosa. Na seção 4.5, será definida a base de dados e serão descritas as tabelas de dados para o uso do sistema. Na seção 4.6, apresenta-se de forma detalhada as classes que foram criadas para o sistema de inferência nebulosa.

4.2 PROCESSO DE IMPORTAÇÃO DOS DADOS

Esta seção tem por objetivo explicar o processo de importação das informações dos inadimplentes para a base de dados MYSQL, essa base de dados é utilizada tanto pelo SIG quanto pelo sistema de cobrança.

O credor (empresa contratante) envia os dados dos devedores através de arquivos para a assessoria de cobrança. Esses arquivos podem estar no formato txt, csv, dbf etc. O arquivo possui um *layout* específico que é fornecido pela empresa contratante. A figura 4.1 é um exemplo de um arquivo txt enviado pelo credor CobrRj para assessoria de cobrança ASCobranças contendo as informações de seus devedores. Para realizar a cobrança o credor envia o *layout* do arquivo que está descrito na tabela 4.1.

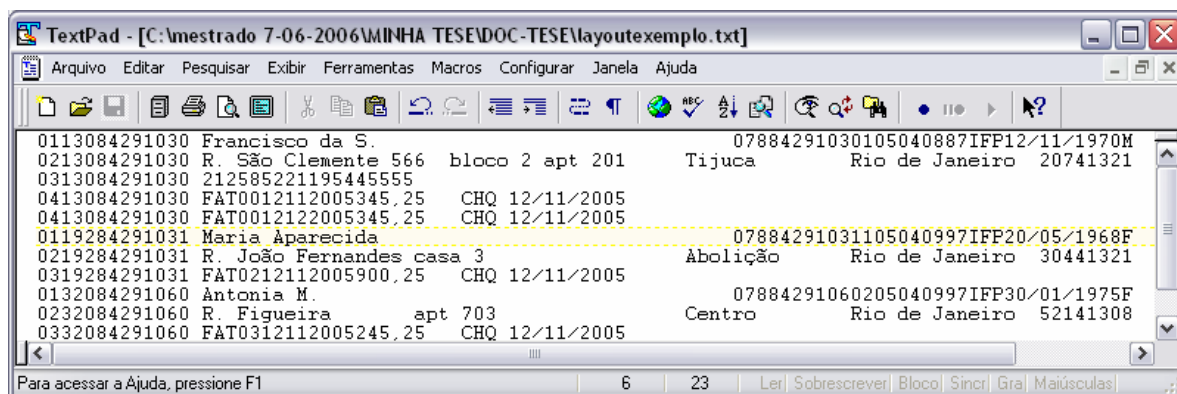


Figura 4.1: Arquivo com dados dos inadimplentes

Tabela 4.1: Layout do arquivo com dados dos inadimplentes

01- Dados do Devedor			
Posição Inicial	Posição Final	Tamanho	Descrição
1	3	2	Detalhe do Devedor
3	14	11	Número do cartão
15	60	45	Nome
60	71	11	Cpf
71	80	9	Identidade
80	83	3	Órgão Expedidor
83	93	10	Data da emissão – formato d/mm/aaaa
93	94	1	Sexo
02 – Dados do Endereço			
Posição Inicial	Posição Final	Tamanho	Descrição
1	3	2	Detalhe do Endereço
3	14	11	Número do cartão
15	56	41	Endereço
56	70	14	Bairro
70	86	16	Cidade
86	94	8	Cep
03 – Dados do Telefone			
Posição Inicial	Posição Final	Tamanho	Descrição
1	3	2	Detalhe do Telefone
3	14	11	Número do cartão
15	17	2	DDD
17	25	8	Tel. Residencial
25	33	8	Tel. Celular
04 – Dados da Fatura			
Posição Inicial	Posição Final	Tamanho	Descrição
1	3	2	Detalhe da Fatura
3	14	11	Número do cartão
15	28	13	Fatura
28	37	9	Valor da fatura
37	41	4	Tipo do Documento
41	51	10	Data da Fatura– formato dd/mm/aaaa

Para realizar o processo de importação, foi desenvolvido um sistema de importação de arquivos denominado SIID (Sistema de Importação de Informação dos Devedores). Com base no *layout* enviado pelo credor, um profissional na área de informática realiza uma análise e utiliza a linguagem estruturada de consulta (SQL) para montar um *script* de banco para inserir as informações dos devedores no banco de dados. Esse *script* é inserido no SIID no cadastro de *layout*, figura 4.2.

O sistema suporta a importação das seguintes informações dos devedores: dados pessoais, contratos, avalista, títulos, endereços, telefones, parente, *e-mail*, acordos, dados de baixas, pagamento e ajustes a serem feitos. Uma vez efetuado o cadastro do *layout* no sistema, modificações só são necessárias se houver alterações do *layout*. A seguir o exemplo de um *layout* cadastrado no sistema com base no *layout* descrito na tabela 4.1.

Cadastro de Layout

SIID - Cadastro de Layout

Layout's Existentes:

Empresa: Arquivo: Tipo:

SQL
 Tamanho da Linha:

Devedor | Contrato | Avalista | Título | Parente | Endereço | Telefone | Email

```

INSERT INTO IM_DEVEDOR
(CGCPF_DEV,NOME_DEV,IDENT_DEV,ORGAO_DEV,EMISS_DEV,
DTNASC_DEV,GENERO_DEV,NACIO_DEV,TPES_DEV,ESTCI_DEV,
CONTR_DEV)
SELECT DISTINCT
SUBSTRING(TEXT0,60,11)AS CGCPF_DEV,
SUBSTRING(TEXT0,15,60)AS NOME_DEV,
  
```

Sql Padrão
 Limpar
 Separador

Novo Salvar Excluir Sair

Figura 4.2: Tela de cadastro do *layout*

Para realizar a importação, o usuário deve salvar os arquivos da empresa numa estrutura de diretório montada pelo sistema, por exemplo, a COBRRJ, possui a seguinte estrutura de diretório: "C:\SIID\COBRRJ". Já a empresa XYZ teria o seguinte diretório: C:\SIID\XYZ. Esse procedimento visa à organização das informações. A figura 4.3 a seguir mostra o conteúdo completo dessa estrutura de diretório.

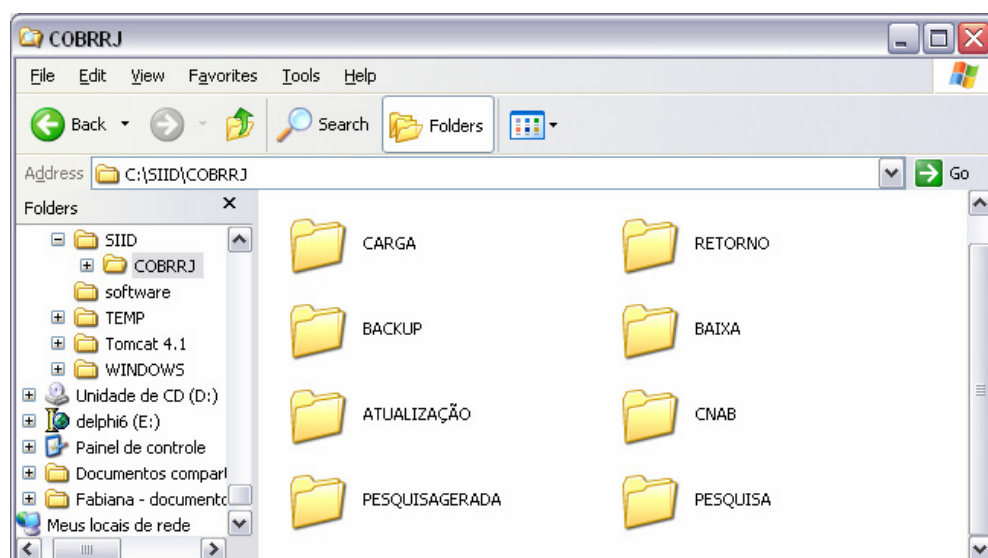


Figura 4.3: Estrutura de diretório

Após o usuário salvar o(s) arquivo(s) no diretório específico da empresa, ele deve acessar a opção Importar carga do sistema SIID e selecionar o *layout* correspondente aos arquivos a serem importados. A figura 4.4 mostra o exemplo da tela de importação de uma empresa fictícia chamada COBRRJ.

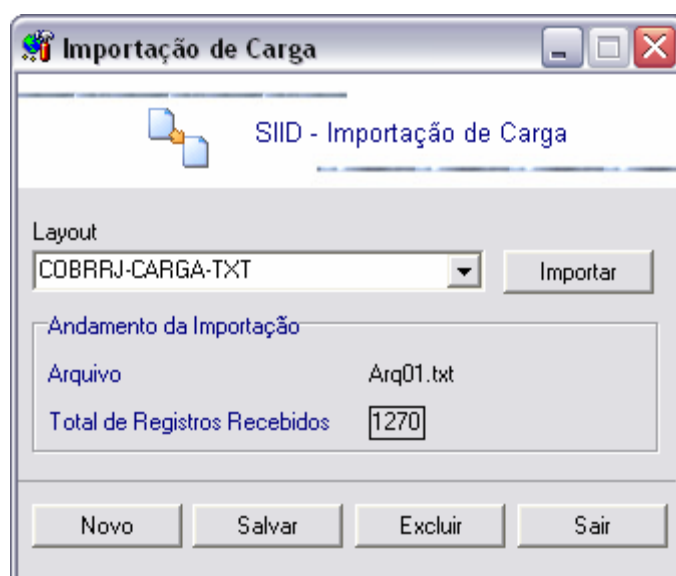


Figura 4.4: Tela de importação

Realizada a importação dos dados, é feito um processo de higienização, ou seja, são retirados caracteres inválidos de informações como nome do devedor, logradouro, cpf, cnpj etc. Por exemplo, um cpf com valor 06gh844490635 após o processo de higienização passa a ter o seguinte valor 06844490635. Feitas as etapas descritas anteriormente, os dados já estão disponíveis para o uso da assessoria de cobrança ASCobrança.

4.3 INTERFACE DO SIGATDSC

O sistema possui apenas quatro interfaces com o usuário, sendo duas para a entrada e duas para saída. A primeira é usada para acesso ao sistema, onde o usuário deve se identificar para que seja validado e possa utilizar o SIGATDSC e a segunda interface permite realizar a parametrização da consulta para obter informações sobre os devedores e a cobrança.

4.3.1 Funcionamento Básico do Sistema

O funcionamento básico do SIGATDSC é descrito pela figura 4.5. Nela, observa-se que a primeira página, Interface 1, serve de acesso ao sistema, sendo restrita ao *login* do usuário. Com o usuário validado, passamos à pagina principal, Interface 2, responsável pelas consultas. Nesta tela o usuário pode optar por analisar como os seus devedores e postos de cobranças estão dispostos geograficamente, analisar o índice de pagamento ou analisar o grau de risco da cobrança em um determinado bairro, dependendo da consulta realizada será exibida a página de resposta 3 (Interface 3) ou a página de resposta 4 (Interface 4).

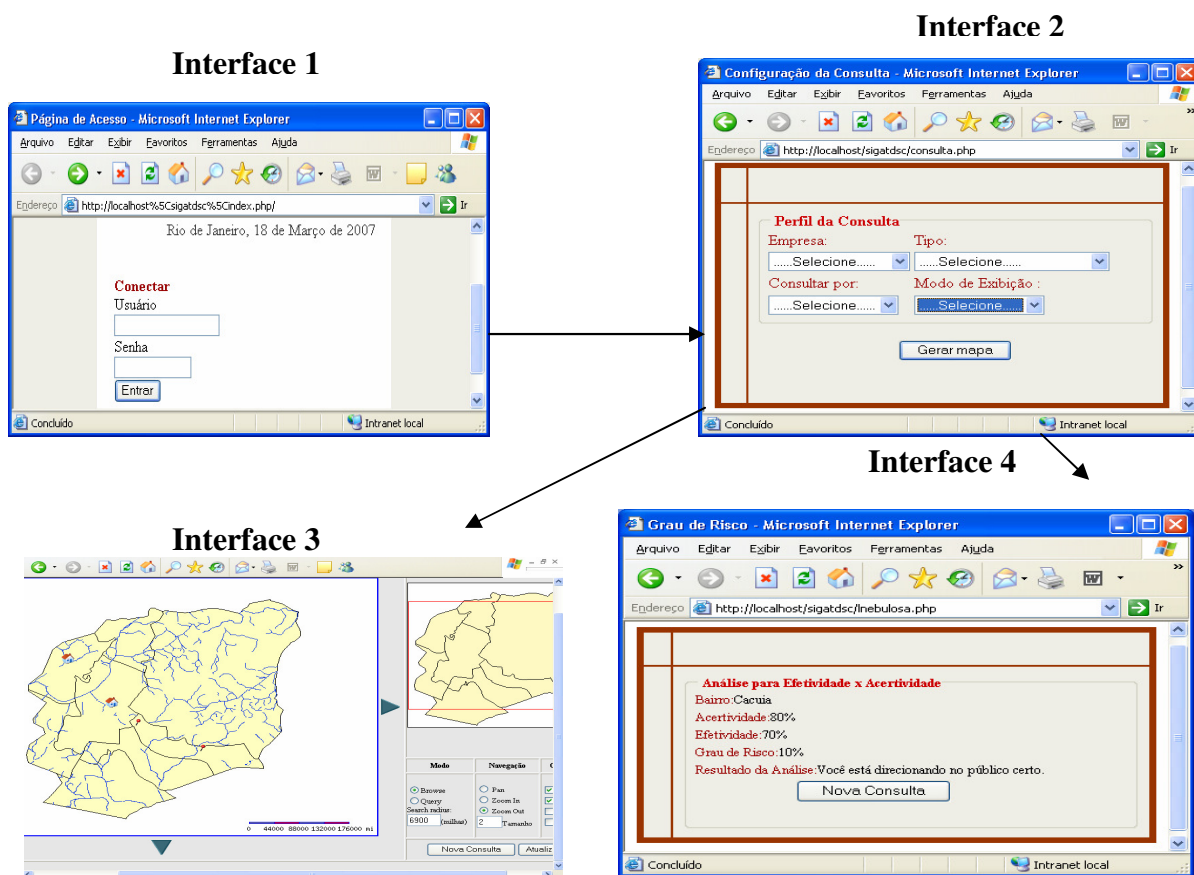


Figura 4.5: Funcionamento Lógico do Sistema

4.3.2 Definição das Interfaces com o Usuário

Como visto antes, o sistema possui apenas quatro interfaces com o usuário, sendo elas:

4.3.2.1 A página Index.php

A primeira página do sistema é responsável pelo acesso do usuário (index.php). Nessa página, o usuário deve informar seu nome de *login* e senha, para que o sistema possa então validar seus dados e liberar ou não o seu uso. Os dados inseridos nos campos *login* e senha são então passados para a classe *usuario_validar.php* que será responsável por validar o *login* e a senha informados pelo usuário. Caso o usuário informe um *login* ou uma senha inválida à página de aviso, *usuario_invalido.php*, será exibida conforme a figura 4.6. Caso contrário a página principal do sistema, *consulta.php*, será exibida conforme a figura 4.7.

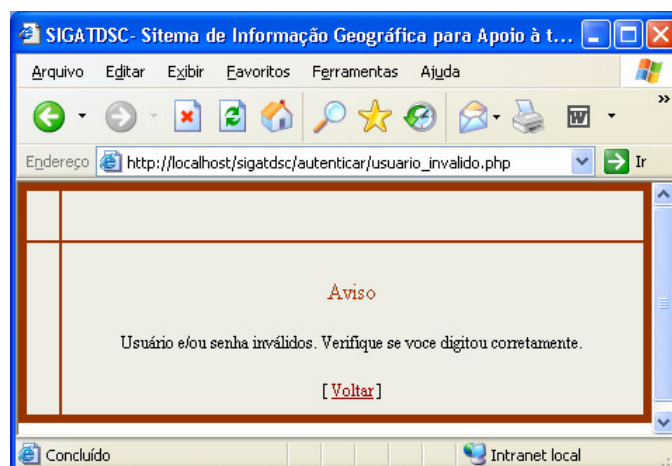


Figura 4.6: Tela de Aviso

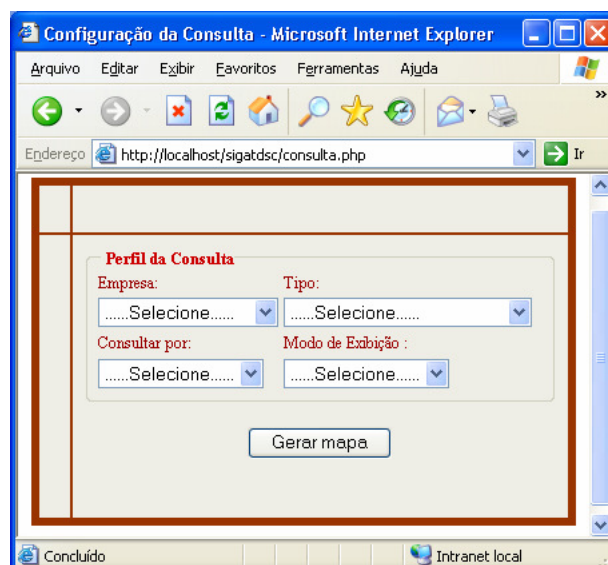


Figura 4.7: Tela de Consulta

4.3.2.2 A página consulta.php

Após a validação do usuário, o sistema exibe a página de consulta, como se pode observar na figura 4.7. Nessa página, são feitas as parametrizações da consulta através dos seguintes filtros:

Filtro Empresa – Empresas cadastradas no banco de dados, para as quais, a assessoria de cobrança presta serviço.

Filtro Tipo – Informa os tipos de consultas, que podem ser realizadas:

Acertividade x Efetividade, Concentração de Devedor e Índice de Pagamento.

Para melhor entendimento, será apresentado um detalhamento de cada um dos tipos citados anteriormente:

Acertividade x Efetividade – A acertividade se refere ao total de contatos telefônicos que foram válidos, ou seja, o telefone informado no momento da compra do serviço ou produto era um número válido e, sendo assim, foi possível contatar o devedor. Efetividade se refere ao total de acordos realizados pelo operador de telecobrança. Ao selecionar esse tipo de filtro, o supervisor de cobrança poderá realizar a análise do grau de risco da cobrança em um bairro, para isso, será utilizada a lógica nebulosa.

Concentração de Devedor – Ao selecionar esse tipo de filtro, o supervisor de cobrança poderá analisar a disposição geográfica de seus devedores, bem como a alocação de seus postos de cobrança. Através dessa análise, pode-se definir uma estratégia de cobrança mais adequada.

Índice de Pagamento – Ao selecionar esse tipo de filtro, o supervisor poderá analisar como está o índice de pagamento dos devedores e informações sobre os meios de cobranças utilizados.

Filtro Consulta por Bairro, Cidade ou Estado – Este filtro permite que a consulta seja analisada por bairro, cidade ou estado.

Filtro Modo de Exibição – Este filtro permite exibir a maior, menor, exibir todas ou personalizar as concentrações de devedores. Caso a opção personalizar seja escolhida, o usuário pode personalizar por cidade, estado ou bairro.

4.3.2.3 A Página LNebulosa.php

Uma das páginas de resultados do sistema é LNebulosa.php, figura 4.8, que é exibida após o usuário selecionar os filtros de Empresa, Tipo de consulta “efetividade x acertividade” e o Bairro. Esta página exibe a análise do grau de risco da cobrança e para esse cálculo foi utilizado um sistema de inferência nebuloso modelado para o SIGATDSC.

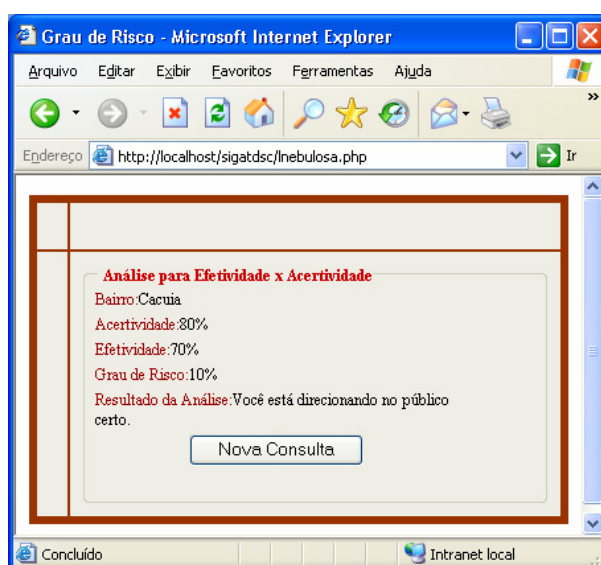


Figura 4.8: Tela de resposta do Grau de Risco

Essa página de resposta mostra a empresa e o bairro escolhidos para realizar a análise, calcula o percentual de acertividade e efetividade dos devedores da empresa que a assessoria de cobrança presta serviço para o bairro definido. Após esse cálculo, o sistema de inferência nebulosa calcula o percentual do Grau de Risco e com base nesse percentual é gerado o resultado da análise.

Descrição do funcionamento da página

LNebulosa.php

Calcular o percentual da acertividade da empresa e bairro informados

Calcular o percentual da efetividade da empresa e bairro informados

Instanciar um objeto da classe DeFuzzyBean.php passando o percentual de acertividade e efetividade calculados

A classe DeFuzzyBean.php instancia um objeto da classe DeFuzzy.php

O método GetResultadoFinal da classe DeFuzzyBean.php retorna o percentual do Grau de Risco calculado

Interpretar o percentual do grau de risco calculado

Exibir resposta

Fim LNebulosa.php

Descrição do funcionamento da classe DeFuzzy.php

INÍCIO DA CLASSE INFERÊNCIA

somatorioCentroide = 0, somatorioArea = 0, centroideRegra = 0, areaRegra = 0

SELECIONAR as Regras

PARA cada Regra FAZER

centroideRegra = 0, areaRegra = 0;

Obter o 1º Termo da Regra

mi1 = Avaliar o mi do 1º Termo

Obter o 2º Termo da Regra

mi2 = Avaliar o mi do 2º Termo

Obter menorMi entre 1 e 2

SE menorMi > 0 ENTÃO

Obter termoConsequente da Regra

CalcularCentroideRegra (termoConsequente, centroideRegra, areaRegra)

somatorioCentroide = somatorioCentroide + (centroideRegra * areaRegra);

somatorioArea = somatorioArea + areaRegra;

FIM SE

LOOP

ValorFinal = somatorioCentroide / somatorioArea;

FIM INFERÊNCIA

4.3.2.4 A Página Posto_Mapa.php

Uma das páginas de resultados do sistema é a Posto_Mapa.php, que é exibida após o usuário selecionar os filtros de Empresa, Tipo de consulta Concentração de devedor ou Índice de Pagamento. Caso o usuário selecione o tipo de consulta concentração de devedor, será exibida a disposição geográfica dos devedores e seus postos de cobranças conforme a figura 4.9 e 4.10. Essa página necessita de um *scroll* na horizontal devido à imagem do mapa e as ferramentas para manipular o mapa.

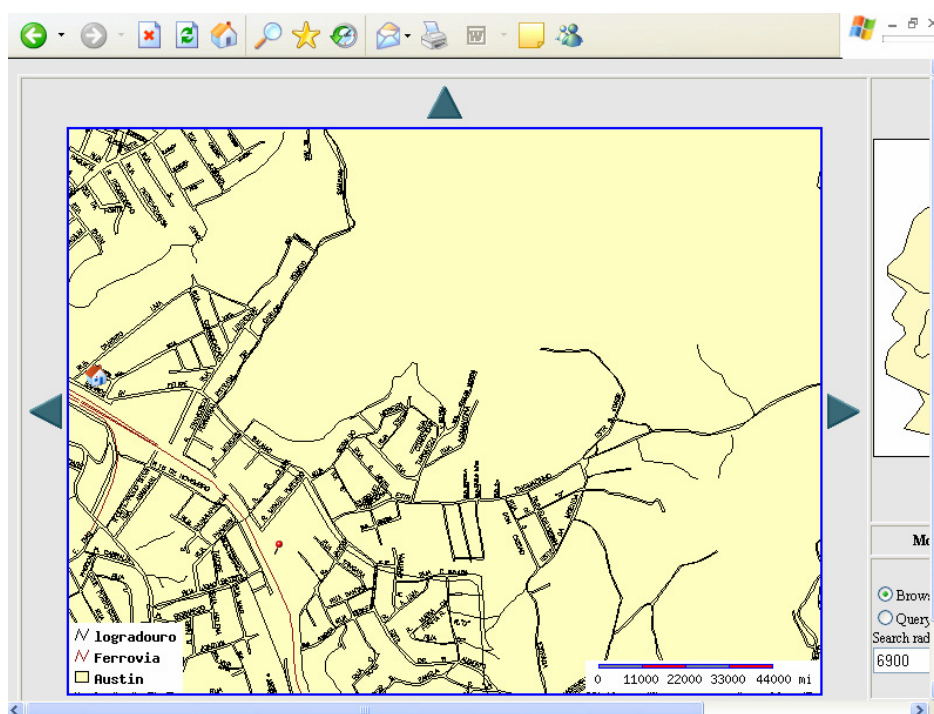


Figura 4.9: Tela de resposta de concentração de devedor

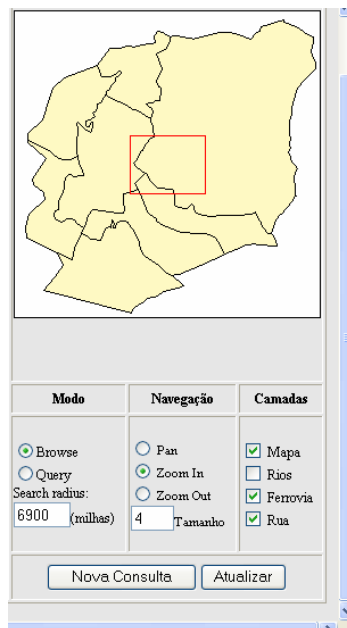


Figura 4.10: Tela de resposta ferramentas

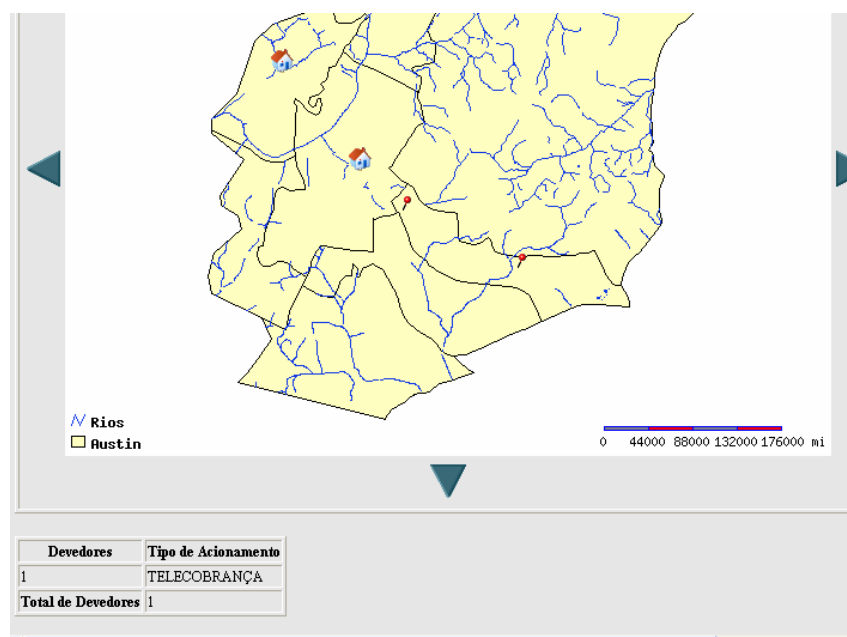


Figura 4.11: Tela de resposta índice de pagamento

Caso o usuário selecione a consulta de índice de pagamento, a página de Posto_Mapa.php será exibida, figura 4.11. Essa página informa a disposição geográfica dos devedores que realizaram pagamento e uma tabela para visualizar a quantidade de devedores que foram contatados por um determinado meio, ou seja, permite visualizar os meios de acionamento que estão sendo mais eficazes em um determinado bairro por exemplo.

4.4 FUNCIONAMENTO LÓGICO DO SIGATDSC

O funcionamento lógico do sistema é demonstrado na figura 4.12. As classes mais relevantes são: `usuario_validar.php`, `DefuzzBean.php` e `Defuzzy.php`. A seguir uma descrição de cada classe:

4.4.1 A classe `usuario_validar.php`

A classe `usuario_validar.php` recebe o nome de *login* do usuário e sua senha, através da Interface de Acesso (`Index.php`). É feita então uma pesquisa na tabela de usuário (`tb_usuario`), através do módulo de verificação do mesmo, que averigua se esse é válido e se sua senha confere.

Após essa verificação, caso o usuário ou senha estejam incorretos, será exibida uma mensagem de aviso, caso contrário será exibida a página de consulta (`consulta.php`).

4.4.2 A Classe `DefuzzBean` e `Defuzzy.php`

As classes `defuzzybean.php` e `defuzzy.php` são responsáveis pelo Módulo de Inferência Nebulosa. A classe `defuzzybean.php` recebe os percentuais de entrada do sistema de inferência nebulosa que são: *acertividade* e *efetividade*. Após receber as entradas, a classe `defuzzybean` instancia um objeto da classe `defuzzy` que será responsável por realizar a inferência e retornar a resposta.

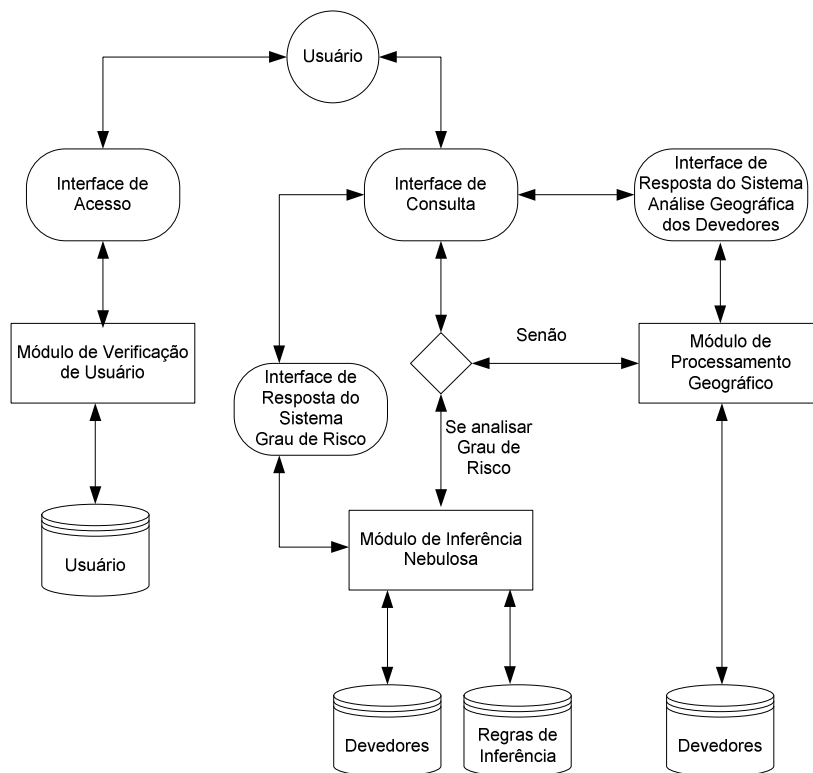


Figura 4.12: Arquitetura do sistema

Para realizar o processo de inferência, é necessário obter os dados de entrada a serem codificados. Os dados de entrada são acertividade e efetividade e são em percentuais. Com base na empresa e no bairro escolhidos pelo usuário, calculamos os percentuais das entradas da seguinte forma:

$$A = \frac{Acerto * 100\%}{TDEVA} \quad (4.1)$$

Onde:

A – corresponde ao percentual de acertividade;

Acerto – corresponde ao total de devedores que possuem telefones válidos de acordo com a empresa e bairro informados;

TDEVA – corresponde ao total de devedores que foram acionados por telecobrança de acordo com a empresa e bairro informados.

$$E = \frac{Acordo * 100\%}{TDEVA} \quad (4.2)$$

Onde:

E – corresponde ao percentual de efetividade;

Acordo – corresponde ao total de devedores que possuem telefones válidos de acordo com a empresa e bairro informados e que realizaram acordos;

TDEVA – corresponde ao total de devedores que foram acionados por telecobrança de acordo com a empresa e bairro informados.

De posse dos dados de entrada, inicia-se o processo de inferência, que irá carregar os antecedentes e o conseqüente, codificar os dados de entrada, carregar as regras e fazer o cálculo da inferência para cada regra. Após essa etapa, deve-se decodificar a saída obtida pela inferência e montar a resposta para o usuário.

4.5 BASE DE DADOS

O SIGATDSC foi criado para auxiliar a tomada de decisão no setor de cobrança, as informações são obtidas através de consultas realizadas na base de dados de um sistema de cobrança, portanto as informações dos devedores, empresa e regras de cobranças são de responsabilidade do sistema de cobrança. Nesta seção, são listadas somente as tabelas do sistema de cobranças que são necessárias para o SIGATDSC.

O SGBD escolhido foi o MYSQL Server versão 5.0. Qualquer outro gerenciador poderia ter sido usado, tais como Oracle[®] ou SqlServer. Foram utilizados dois banco de dados implementados em MySQL, um com os dados dos devedores e o outro para armazenar os dados do sistema de inferência nebulosa modelado para o SIGTDSC. O SIGATDSC utiliza 21 tabelas sendo 5 referentes à lógica nebulosa. Não foram criados módulos de inclusão, alteração ou exclusão de registros, pois, seus dados são obtidos de diversas fontes apenas para consulta, não necessitando nenhum tipo de manutenção pelo sistema. As alterações na base de dados são feitas externamente ao sistema, pois o mesmo importa as informações já atualizadas para utilização em seu processamento.

4.5.1 Descrição das tabelas utilizadas pelo SIGTDSC

As tabelas envolvidas no uso do SIGATDSC são:

- TB_USUARIO
- TB_INADIMPLENTE
- TB_CONTRATO
- TB_EMPRESA
- TB_TITULO
- TB_SITUACAO
- TB_ESTADO
- TB_ACORDO
- TB_PAGAMENTO
- TB_TIPOPAG
- TB_ESPECIE
- TB_POSTO
- TB_CIDADE
- TB_BAIRRO
- TB_TPACIONAMENTO

Dicionário de dados

Tabela 4.2: Entidades

TABELAS	
TB_USUARIO	Tabela que contém informações sobre os usuários do sistema
TB_INADIMPLENTE	Tabela que contém informações dos inadimplentes (devedores)
TB_CONTRATO	Tabela que contém informações sobre os contratos dos devedores
TB_EMPRESA	Tabela que contém informações das empresas trabalhadas pela assessoria
TB_TITULO	Tabela que contém informações sobre o valor da dívida do devedor
TB_SITUACAO	Tabela que contém informações sobre a situação do devedor
TB_ESTADO	Tabela que contém informações dos estados
TB_ACORDO	Tabela que contém informações sobre os acordos realizados com o devedor para quitar sua dívida
TB_PAGAMENTO	Tabela que contém informações sobre os pagamentos realizados pelo devedor
TB_TIPOPAG	Tabela que contém informações sobre a forma de pagamento
TB_ESPECIE	Tabela que contém informações sobre o tipo de pagamento
TB_POSTO	Tabela contendo informações sobre os postos de cobrança da assessoria de cobrança
TB_CIDADE	Tabela de cidades
TB_BAIRRO	Tabelas de bairros
TPACIONAMENTO	Tabela que contém informações sobre os tipos de acionamento

Estrutura das Tabelas

A tabela 4.3 contém os dados dos usuários autorizados a usar o sistema. Tal tabela possui como campos idusuario_usu, usuario_usu, senha_usu e email_usu. A seguir, pode-se acompanhar a descrição desses campos na tabela.

Tabela 4.3: Tabela de usuário

TB_USUARIO			
Campo	Tipo Dado	Opção Nula	Comentário
IDUSUARIO_USU	Int	Não	Identificação
USUARIO_USU	Varchar(30)	Não	Nome do usuário
SENHA_USU	Varchar(60)	Não	Senha do usuário
EMAIL_USU	Varchar(50)	Sim	Email do usuário

A tabela 4.4 contém os dados dos inadimplentes. Destacam-se nessa tabela os campos latitude_inad e longitude_inad que fornecem a localização geográfica dos inadimplentes para o SIGATDSC.

Tabela 4.4: Tabela de inadimplente

TB_INADIMPLENTE			
Campo	Tipo Dado	Opção Nula	Comentário
IDINADIMPLENTE_INAD	Int	Não	Chave
CGCPF_INAD	Char(14)	Não	Cgc ou cpf
NOME_INAD	Char(50)	Sim	Nome
IDENTIDADE_INAD	Char(20)	Sim	Nº da identidade
ORGAO_INAD	Char(10)	Sim	Órgão expedidor
DTEMISSAO_INAD	Datetime	Sim	Data da emissão
DTNASCIMENTO_INAD	Datetime	Sim	Data de nascimento
GENERO_INAD	Char(1)	Sim	Gênero
NACIONALIDADE_INAD	Char(35)	Sim	Nacionalidade
TPPESSOA_INAD	Char(1)	Não	Tipo de pessoa (física ou jurídica)
ESTCIVIL_INAD	Char(1)	Sim	Estado civil
LOGRADOURO_INAD	VARCHAR(50)	Não	Endereço do devedor
NUMERO_INAD	VARCHAR(10)	Não	Número
COMPLEMENTO_INAD	VARCHAR(20)	Não	Complemento
BAIRRO_INAD	INTEGER	Não	Bairro
CIDADE_INAD	INTEGER	Não	Chave estrangeira da cidade
IDESTADO_INAD	INTEGER	Não	Chave estrangeira do estado
CEP_INAD	CHAR(8)	Não	Cep
LONGITUDE_INAD	DECIMAL(18,5)	Não	Longitude
LATITUDE_INAD	DECIMAL(18,5)	Não	Latitude

A tabela 4.5 indica os contratos dos inadimplentes de uma empresa, a tabela contém informações de como o devedor será cobrado e algumas informações sobre a dívida.

Tabela 4.5: Tabela de contrato

TB_CONTRATO			
Campo	Tipo Dado	Opção Nula	Comentário
IDCONTRATO_CONT	Int	Não	Chave
IDINADIMPLENTE_CONT	Int	Não	Identificador do inadimplente - Chave estrangeira
IDEMPRESA_CONT	Int	Não	Identificador da empresa - Chave estrangeira
IDCARTEIRA_CONT	Int	Não	Identificador da carteira - Chave estrangeira
IDASSESSORIA_CONT	Int	Não	Identificador da assessoria - Chave estrangeira
IDBORDERO_CONT	Int	Não	Identificador do bordero - Chave estrangeira
IDCLASSE_CONT	Int	Sim	Identificador da classe - Chave estrangeira
IDMOTIVO_CONT	Int	SIM	Identificador do motivo - Chave estrangeira
IDLOJA_CONT	Int	SIM	Identificador da loja - Chave estrangeira
IDBANDEIRA_CONT	Int	SIM	Identificador da bandeira - Chave estrangeira
IDSEGMENTACAO_CONT	Int	Sim	Identificador da segmentação - Chave estrangeira
IDCAMPANHA_CONT	Int	Não	Identificador da campanha - Chave estrangeira
IDEQUIPE_CONT	Int	Não	Identificador da equipe - Chave estrangeira
IDSITUACAO_CONT	Int	Não	Identificador da situação – Chave estrangeira
IDOCORRENCIA_CONT	Int	Sim	Identificador da ocorrência – Chave estrangeira
CONTRATO_CONT	Char(19)	Não	Número do contrato

SEQCONTRATO_CONT	Char(2)	Não	Número sequencial do contrato
CONTA_CONT	Char(11)	Sim	Número da conta
DTINCLUSAO_CONT	Datetime	Não	Data da inclusão do contrato
STDEVOLUCAO_CONT	Char(1)	Não	Status da devolução
DTDEVOLUCAO_CONT	Datetime	Sim	Data da devolução do contrato
DTPRORROGACAO_CONT	Datetime	Sim	Data da prorrogação
DTSITUACAO_CONT	Datetime	Sim	Data da mudança de situação
DTAGENDAMENTO_CONT	Datetime	Sim	Data do agendamento
NDIAS_CONT	Numeric(5, 0)	SIM	Número de dias de atraso do contrato
MERCADORIA_CONT	Char(50)	Sim	Mercadoria
DTUPAGAMENTO_CONT	Datetime	Sim	Data do último pagamento
VLUPAGAMENTO_CONT	Numeric(2,2)	Sim	Valor do último pagamento
VLSALDO_CONT	Numeric(2,2)	Sim	Saldo do contrato
VLMINACORDO_CONT	Numeric(2,2)	Sim	Valor mínimo do acordo
VLMINPARCELA_CONT	Numeric(2,2)	Sim	Valor mínimo da parcela
NRACIONAMENTO_CONT	Numeric(5,0)	Sim	Número de acionamento
NROCORRENCIA_CONT	Numeric(5,0)	Sim	Número de ocorrência
STATUS	Char(1)	Sim	Flag para identificar se o contrato está ativo ou não, ou seja se o devedor ainda deve ser cobrado

A tabela 4.6 contém informações das empresas para as quais a assessoria de cobrança presta serviço.

Tabela 4.6: Tabela de empresa

TB_EMPRESA			
Campo	Tipo Dado	Opção Nula	Comentário
IDEMPRESA_EMP	Int	Não	Chave primária
IDESTADO_EMP	Int	Não	Identificador do estado - Chave estrangeira
RAZAOSOCIAL_EMP	Char(50)	Não	Razão social
NOMEFANTASIA_EMP	Char(50)	Não	Nome fantasia
CNPJ_EMP	Char(14)	Não	Cnpj
LOGRADOURO_EMP	Char(50)	Não	Logradouro
NUMERO_EMP	Char(10)	Não	Número
COMPLEMENTO_EMP	Char(20)	Não	Complemento
BAIRRO_EMP	Integer	Não	Chave estrangeira do Bairro
CIDADE_EMP	Integer	Não	Chave estrangeira da Cidade
CEP_EMP	Char(8)	Não	Cep
TEL_EMP	Char(10)	Sim	Telefone
FAX_EMP	Char(10)	Sim	Número de fax

A tabela 4.7 contém informações da dívida do inadimplente.

Tabela 4.7: Tabela de Título

TB_TITULO			
Campo	Tipo Dado	Opção Nula	Comentário
IDTITULO_TIT	Int	Não	Chave primária
IDCONTRATO_TIT	Int	Não	Identificador do contrato – Chave estrangeira
IDDOCUMENTO_TIT	Int	Não	Identificador do documento – Chave estrangeira
IDALINEA_TIT	Int	Sim	Identificador da alínea – Chave estrangeira
IDBANCO_TIT	Int	Sim	Identificador do Banco – Chave estrangeira
TITULO_TIT	Char(25)	Sim	Número do Título
DTINCLUSAO_TIT	Datetime	Sim	Data da Inclusão
VALOR_TIT	Numeric(2,2)	Sim	Valor Original do Título
DTFATURA_TIT	Datetime	Sim	Data de Vencimento do Título
VLFATURA_TIT	Numeric(2,2)	Sim	Valor da Fatura do Título
AGENCIA_TIT	Char(6)	Sim	Código da Agência
CONTA_TIT	Char(10)	Sim	Conta do Título

A tabela 4.8 contém informações sobre os estados. Trata-se de uma tabela de cadastro:

Tabela 4.8: Tabela de Estado

TB_ESTADO			
Campo	Tipo Dado	Opção Nula	Comentário
IDESTADO_EST	Int	Não	Chave
CODESTADO_EST	Char(2)	Sim	Código do Estado
DESCRICAO_EST	Char(50)	Sim	Descrição
REGIAO_EST	Char(1)	Sim	Região que o estado pertence

A tabela 4.9 contém informações sobre o *status* de um contrato. O diagrama de estado do capítulo de modelagem do SIGATDSC exemplifica as situações que um contrato pode ter.

Tabela 4.9: Tabela de situação

TB_SITUACAO			
Campo	Tipo Dado	Opção Nula	Comentário
IDSITUACAO_SIT	Int	Não	Chave primária
DESCRICAO_SIT	Char(50)	Sim	Descrição
TEMAN_SIT	Char(8)	Sim	Tempo médio de ligação
TELM_SIT	Char(8)	Sim	Tempo limite de ligação

Tabela 4.10: Tabela de Espécie

TB_ESPECIE			
Campo	Tipo Dado	Opção Nula	Comentário
IDESPECIE_ESP	Int	Não	Chave primário
IDEMPRESA_ESP	Int	Não	Identificador da Empresa
DESCRICAO_ESP	Char(50)	Sim	Descrição
CODIG_ESP	Char(10)	Sim	Código da espécie

Tabela 4.11: Tabela de Tipo de pagamento

TB_TIOPAG			
Campo	Tipo Dado	Opção Nula	Comentário
IDFORMPAG_TPG	Int	Não	Chave primária
IDPAGAMENTO_TPG	Int	Não	Identificador do pagamento – Chave estrangeira
IDESPECIE_TPG	Int	Não	Identificador da espécie – Chave estrangeira
EMIT_TPG	Char(50)	Sim	Nome ou razão social do emitente
IDBANCO_TPG	Int	Não	Identificador do banco – Chave estrangeira
AGENCIA_TPG	Char(6)	Sim	Código da Agência
CHEQUE_TPG	Char(15)	Sim	Número da boleta ou cheque
CONTA_TPG	Char(15)	Sim	Número da conta
VALOR_TPG	Numeric(15,2)	Sim	Valor
DTDEPOSITO_TPG	Datetime	Sim	Data do depósito
FOTO_TPG	Blob	Sim	Foto do cheque

A tabela 4.12 contém informações sobre os pagamentos dos inadimplentes.

Tabela 4.12: Tabela de pagamento

TB_PAGAMENTO			
Campo	Tipo Dado	Opção Nula	Comentário
IDPAGAMENTO_PAG	Int	Não	Chave primária
IDACORDO_PAG	Int	Não	Identificador do acordo – Chave estrangeira
IDPRESTACAO_PAG	INT	Não	Identificador da prestação – Chave estrangeira
NUMPARCELA_PAG	Numeric(2,0)	Sim	Número da parcela que corresponde o pagamento
DTVENCIMENTO_PAG	Datetime	Sim	Data do vencimento
VLVENCIMENTO_PAG	Numeric(12,2)	Sim	Valor do vencimento
STPAGAMENTO_PAG	Char(1)	Sim	Status do pagamento
DTPAGAMENTO_PAG	Datetime	Sim	Data do pagamento
VLPAGAMENTO_PAG	Numeric(12,2)	Sim	Valor do pagamento
VLPRINCIPAL_PAG	Numeric(12,2)	Sim	Valor do principal
VLCORRIGIDO_PAG	Numeric(12,2)	Sim	Valor corrigido
VLAJUSTE_PAG	Numeric(12,2)	Sim	Valor do ajuste
VLACRESCIMO_PAG	Numeric(12,2)	Sim	Valor do acréscimo
VLJUROS_PAG	Numeric(12,2)	Sim	Valor de juros
VLCOMISSAO_PAG	Numeric(12,2)	Sim	Valor da comissão
VLMULTA_PAG	Numeric(12,2)	Sim	Valor da multa
VLTAAXA_PAG	Numeric(12,2)	Sim	Valor da taxa
VLDESCONTO_PAG	Numeric(12,2)	Sim	Valor do desconto
VLDPRINCIPAL_PAG	Numeric(12,2)	Sim	Valor do desconto no principal

A tabela 4.13 contém informações sobre o acordo firmado com o inadimplente.

Tabela 4.13: Tabela de acordo

TB_ACORDO			
Campo	Tipo Dado	Opção Nula	Comentário
IDACORDO_ACO	Int	não	Chave Primária
IDCONTRATO_ACO	Int	Não	Identificador do Contrato – Chave estrangeira
IDPARCELA_ACO	Int	não	Identificador da Parcela – Chave estrangeira
IDUSUARIO_ACO	Int	não	Identificador do usuário – Chave estrangeira
IDASSESSORIA_ACO	Int	Não	Identificador da assessoria – Chave estrangeira
DTACORDO_ACO	Datetime	Sim	Data do acordo
DTCANCELAMENTO_ACO	Datetime	Sim	Data do cancelamento do acordo
DTENVIO_ACO	Datetime	Sim	Data do envio do acordo
CORRECAO_ACO	Numeric(7,3)	Sim	Percentual de correção
AJUSTE_ACO	Numeric(9,6)	Sim	Percentual de ajuste
HONORARIO_ACO	Numeric(7,3)	Sim	Percentual de honorário a ser pago pelo cliente
JUROS_ACO	Numeric(7,3)	Sim	Percentual de juros
MULTA_ACO	Numeric(7,3)	Sim	Valor da Multa
DESCONTO_ACO	Numeric(7,3)	Sim	Desconto máximo de acordo com a tabela de negociação do cliente
DESCPONTUALIDADE_ACO	Numeric(7,3)	Sim	Percentual de desconto de pontualidade de pagamento
DESPRINCIPAL_ACO	Numeric(7,3)	Sim	Percentual de desconto no principal
STACORDO_ACO	Char(1)	Sim	Status do acordo
IDTP_ACO	INTEGER	Não	Chave estrangeira do tipo do acordo

A tabela 4.14 contém informações sobre os postos de cobrança da assessoria de cobrança. Trata-se de uma tabela de cadastro. Destacam-se dessa tabela os campos longitude_pos e latitude_pos que informam ao SIGATDSC a localização geográfica dos postos de cobrança.

Tabela 4.14: Tabela de posto

TB_POSTO			
Campo	Tipo Dado	Opção Nula	Comentário
IDPOSTO_POS	Int	Não	Chave primária
IDESTADO_POS	Int	Não	Identificador do estado – Chave estrangeira
DESCRICAO_POS	Nvarchar(50)	Sim	Nome do posto
LOGRADOURO_POS	nvarchar(50)	Sim	Endereço
NUMERO_POS	Nvarchar(50)	Sim	Número
COMPLEMENTO_POS	Nvarchar(20)	Não	Complemento
BAIRRO_POS	Integer	Sim	Chave estrangeira do Bairro
CIDADE_POS	Integer	Sim	Chave estrangeira da Cidade
CEP_POS	Nvarchar(8)	Sim	Cep
LONGITUDE_POS	Numeric(18,5)	Não	Coordenada x de localização
LATITUDE_POS	Numeric(18,5)	Não	Coordenada y de localização

A tabela 4.15 contém informações sobre as cidades. Trata-se de uma tabela de cadastro.

Tabela 4.15: Tabela de cidade

TB_CIDADE			
Campo	Tipo Dado	Opção Nula	Comentário
IDCIDADE_CID	INTEGER	Não	Chave primária da tabela
DESCRICAO_CID	VARCHAR(50)	Não	Nome da cidade

A tabela 4.16 contém informações sobre o bairro. Trata-se de uma tabela de cadastro.

Tabela 4.16: Tabela de bairro

TB_BAIRRO			
Campo	Tipo Dado	Opção Nula	Comentário
IDBAIRRO_BAR	INTEGER	Não	Chave primária da tabela
DESCRICAO_BAR	VARCHAR(30)	Não	Nome do bairro

A tabela 4.17 contém informações sobre os tipos de acionamento. Trata-se de uma tabela de cadastro.

Tabela 4.17: Tabela de tipo de acionamento

TB_TPACIONAMENTO			
Campo	Tipo Dado	Opção Nula	Comentário
IDTPAC_AC	INTEGER	Não	Chave estrangeira da tabela
DESCRICAO_AC	VARCHAR(45)	Não	Tipo de acionamento

4.5.2 Descrição das tabelas do Sistema de Inferência Nebuloso - SIN

Tabelas Envolvidas

- ANTECEDENTE
- REGRA
- SEGMENTO
- TERMO
- VARIABEL_LINGUISTICA

Dicionário de dados

Tabela 4.18: Tabela de Entidades do SIN

TABELAS	
ANTECEDENTE	Tabela que contém informações dos termos antecedentes associados às suas respectivas regras do sistema nebuloso.
REGRA	Tabela que contém informações sobre as regras.
SEGMENTO	Tabela que contém informações dos segmentos.
TERMO	Tabela que contém informações dos termos antecedentes e conseqüentes do sistema nebuloso.
VARIABEL_LINGUISTICA	Tabela que contém informações das variáveis lingüísticas

Estrutura das tabelas

Tabela 4.19: Tabela Antecedente

ANTECEDENTE			
Campo	Tipo Dado	Opção Nula	Comentário
COD_REGRA	INTEGER	Não	Chave estrangeira da regra
COD_TERMO	INTEGER	Não	Chave estrangeira do termo

Tabela 4.20: Tabela de Regra

REGRA			
Campo	Tipo Dado	Opção Nula	Comentário
COD_REGRA	INTEGER	Não	Chave estrangeira da regra
COD_TERMO_CONS	INTEGER	Não	Chave estrangeira do termo conseqüente

Tabela 4.21: Tabela de Segmento

SEGMENTO			
Campo	Tipo Dado	Opção Nula	Comentário
COD_TERMOS	INTEGER	Não	Chave estrangeira do termo
COD_SEGMENTO	INTEGER	Não	Chave primária da tabela
INTERVALO_ESQUERDO	INTEGER	Não	Intervalo esquerdo
INTERVALO_DIREITO	INTEGER	Não	Intervalo direito
COEF_ANGULAR	DECIMAL(18,4)	Não	Coefficiente angular
COEF_LINEAR	DECIMAL(18,4)	Não	Coefficiente linear
MI_ESQUERDO	INTEGER	Não	Mi esquerdo
MI_DIREITO	INTEGER	Não	Mi direito

Tabela 4.22: Tabela de Termo

TERMO			
Campo	Tipo Dado	Opção Nula	Comentário
COD_TERMOS	INTEGER	Não	Chave estrangeira do termo
COD_VARIAVEL	INTEGER	Não	Chave estrangeira da variável lingüística
NOM_TERMOS	VARCHAR(50)	Não	Nome do termo

Tabela 4.23: Tabela de Variável Lingüística

VARIABEL_LINGUISTICA			
Campo	Tipo Dado	Opção Nula	Comentário
COD_VARIAVEL	INTEGER	Não	Chave primária da tabela
NOM_VARIAVEL	VARCHAR(50)	Não	Nome da variável lingüística
INI_UN4ERSO	INTEGER	Não	Início do universo de discurso
FIM_UN4ERSO	INTEGER	Não	Fim do universo de discurso
FLAG_ANTECEDENTE	CHAR(1)	Não	Flag que identifica se a variável é um antecedente 1- antecedente 0- conseqüente

4.6 ALGORITMO DE INFERÊNCIA NEBULOSA

4.6.1 Classe DeFuzzyBean

A classe DefuzzyBean é responsável por fazer a comunicação entre a página Lnebulosa.php e a classe Defuzzy. Esta classe possui 4 métodos para atribuir valores às

variáveis privadas da classe, um método para calcular o Grau de Risco e um método para ler o resultado do Grau de Risco calculado pelo sistema de inferência nebuloso. A seguir, expõe-se a explicação dos métodos da classe DeFuzzy e nos Apêndices A1 e A2, pode-se observar o seu código em PHP.

4.6.1.1 Método setResultadoFinal

Este método é responsável por instanciar um objeto da classe DeFuzzy e iniciar o processo do cálculo do Grau de Risco.

Início do Método setResultadoFinal

Des ← novo objeto DeFuzzy (cód_variavel1, valo_var1, cód_variavel2, valor_var2)

resFinal ← des.percorrerRegras

ResultadoFinal ← resFinal

Fim do Método setResultadoFinal

4.6.2 Classe DeFuzzy

Esta classe é responsável pelo cálculo do conseqüente do sistema de inferência nebuloso e possui a declaração de variáveis globais que são necessárias aos métodos. O processo se inicia com o método PercorrerRegras que acionará todos os outros métodos da classe.

4.6.2.1 Método PercorrerRegras

Este método é responsável por acessar a tabela de regras e transferi-las para o vetor de regras (rs). Cada regra é montada a partir da tabela de regras no vetor rs. Este método é importante para aumentar a performance do sistema e permitir que novas regras sejam adicionadas, retiradas ou alteradas, bastando que para isso seja feita uma manutenção na tabelas de regras, pois essas são sempre carregadas quando do momento do processamento. Além de buscar as regras na tabela de regras, este método é responsável por calcular o conseqüente (Grau de Risco) do sistema de inferência nebulosa, mas para isso, é necessário chamar seis métodos.

Descrição do método PercorrerRegras

```
Rs ← regras da tabela de Regras
ENQUANTO (existe registro em rs)
    Centroide ← 0
    DenominadorRegra ← 0
    Cod_Regra ← rs(cód_regra)
    Cod_Termo1 ← VerificarTermoRegra(cod_regra, cod_variavel1)
    Alfa1 ← AvaliarMi(cod_termo1, valor_var1)
    cod_termo2 ← VerificarTermoRegra(cod_regra, cod_variavel2)
    Alfa2 ← AvaliarMi(cod_termo2, valor_var2)
    SE Alfa1 <= Alfa2 ENTÃO
        Menor_Alfa ← Alfa1
    SENÃO
        Menor_Alfa ← Alfa2
    FIM DO SE
    cod_termo_cons ← VerificarTermoConsRegra( cod_regra)
    centroideRegra ← CalculaCentroideRegra(cod_termo_cons, MenorAlfa)
    NumeradorGeral ← NumeradorGeral + (centroideRegra * DenominadorRegra)
    DenominadorGeral ← DenominadorGeral + DenominadorRegra
FIM ENQUANTO
ValorFinal ← NumeradorGeral / DenominadorGeral
RETORNA ValorFinal
```

4.6.2.1.1 Método VerificarTermoRegra

Este método recebe como parâmetros as variáveis de entrada e as regras do sistema de inferência nebulosa e retorna o código do termo que atende a cada regra e a cada variável de entrada.

Descrição do método VerificarTermoRegra(cod_regra, cód_variavel)

```
Cod_termo ← 0
Rs ← cod_termo da tabela de Termo, regra, antecedente ONDE
termo.cod_termo = antecedente.cod_termo OU termo.cod_termo = Regra.cod_termo_cons) E
```

```

Regra.cod_regra=antecedente.cod_regra E Regra.cod_regra=cod_regra E
termo.cod_variavel=cod_variavel
SE rs possui registro ENTÃO
    Cod_termo ← rs.cod_termo
FIM DO SE
RETORNA cod_termo

```

4.6.2.1.2 Método AvaliarMi

Este método retorna o grau de pertinência do valor de entrada do antecedente referente ao conjunto do termo encontrado no método “VerificarTermoRegra”.

Descrição do método AvaliarMi(cod_termo,valor_var)

```

Alfa ← 0
Rs ← cod_segmento,coef_angular,coef_linear da tabela de Segmento ONDE
segmento.cod_termo=cod_termo E (intervalo_esquerdo ≤ valor_var) E
(intervalo_direito ≥ valor_var) E (mi_esquerdo <> 0 OU mi_direito <> 0)
SE Rs possui registro ENTÃO
    Codsegmentos_termo=rs.cod_segmento
    Coef_linear=rs.coef_linear
    Coef_angular=rs.coef_angular
    Alfa=CalculaAlfa(Coef_Linear, coef_Angular,Valor_var)
RETORNA Alfa
FIM DO SE

```

4.6.2.1.3 Método CalculaAlfa

Este método calcula o grau de pertinência.

Descrição do Método CalculaAlfa(coef_Linear,Coef_Angular,Valor_var)

```

Alfa ← (coef_angular * valor_var) + coef_linear
RETORNA Alfa

```

4.6.2.1.4 Método VerificarTermoConsRegra

Descrição do Método VerificartermoConsRegra(cod_regra)

Cód_termo \leftarrow 0

Rs \leftarrow cod_termo_cons da tabela de regra ONDE regra.cod_regra=cod_regra

RETORNA rs. cod_termo_cons

4.6.2.1.5 Método CalculaCentroideRegra

Este método transforma a saída nebulosa em um valor determinístico, ou seja, ocorre a decodificação. A decodificação pode ser feita através de vários métodos, entre eles pode-se citar: a **Média do Conjunto Suporte**; o **Centróide**; a **Média dos Centróides** e a **Média dos Máximos**. Foi utilizado o método da Média dos Centróides, onde se obtém a saída determinística calculando-se a média do **centro de gravidade** da função de pertinência de cada conseqüente atenuado pelo grau de ativação de cada regra e, em seguida, faz uma média ponderada desses centróides, empregando a área dessas funções de pertinência como peso.

Descrição do Método calculaCentroideRegra(cód_termo_cons, MenorAlfa)

CentroideRegra \leftarrow 0

NumeradorRegra \leftarrow 0

Denominadorregra \leftarrow 0

Rs \leftarrow cod_segmento, intervalo_esquerdo, intervalo_direito, mi_esquerdo, mi_direito ONDE
cód_termo=cód_termo_cons E (mi_esquerdo \neq 0 OU mi_direito \neq 0)

ENQUANTO existe registro em Rs

 Cód_segmento \leftarrow rs.cod_segmento

 Intesq \leftarrow rs.intervalo_esquerdo

 Intdir \leftarrow rs.intervalo_direito

 Miesq \leftarrow rs.mi_esquerdo

 Midir \leftarrow rs.mi_direito

 Numeradorsegmento \leftarrow 0

 Denominadorsegmento \leftarrow 0

 SE miEsq=1 ENTÃO

 miEsq=MenorAlfa

```

FIM DO SE
SE miDir=1 ENTÃO
    miDir=MenorAlfa
FIM DO SE
CoefAng=(miEsq-miDir) / (intEsq-IntDir)
CoefLinear=miEsq – (intEsq * coefAng)
NumeradorSegmento=calculacentroide(intesq,IntDir,coefAng,CoefLinear)
DenominadorSegmento=calculaarea(intesq,intDir,coefAng,coefLinear)
NumeradorRegra=NumeradorRegra+numeradorSegmento
DenominadorRegra=DenominadorRegra+denominadorSegmento
FIM ENQUANTO
CentroideRegra=Numeradorregra / DenominadorRegra
RETORNA CentroideRegra

```

4.6.2.1.6 Método CalculaCentroide

Descrição do Método CalculaCentroide(intesq,IntDir,coefAng,CoefLinear)

```

Potencia1 ← (intDir)3
Potencia2 ← (intDir)2
Potencia3 ← (intEsq)3
Potencia4 ← (intEsq)2
Aux1 ← (potencia1 * coefang) / 3
Aux2 ← (potencia2 * coeflinear) / 2
Aux3 ← (potencia3 * coefang) / 3
Aux4 ← (potencia4 * coeflinear) / 2
Centroide ← (aux1+aux2-aux3-aux4)
RETORNA Centroide

```

4.6.2.1.7 Método CalculaArea

Descrição do Método CalculaArea(intesq,IntDir,coefAng,CoefLinear)

```

Potencia1 ← (intDir)2
Potencia3 ← (intEsq)2

```

```

Aux1 ← (potencia1 * coefang) / 2
Aux2 ← (intDir * coeflinear)
Aux3 ← (potencia3 * coefang) / 2
Aux4 ← (intEsq * coeflinear)
Area ← (aux1+aux2-aux3-aux4)
RETORNA Area

```

4.7 COMENTÁRIOS

Neste capítulo, observou-se a implementação do sistema. Viu-se o funcionamento lógico do sistema e como foi definida a estrutura da Base de Dados do SIGATDSC. Foram apresentadas também as soluções para a codificação do sistema de inferência nebulosa e como foram calculados os percentuais dos antecedentes.

CAPÍTULO 5

Testes e Resultados

5.1 INTRODUÇÃO

Este capítulo trata dos testes realizados no SIGATDSC. Foram realizados testes para analisar a Concentração de Devedores, Índice de Pagamento e Grau de Risco. As seções 5.2, 5.3 e 5.4 descrevem os resultados dos testes realizados.

5.2 TESTE PARA CALCULAR O GRAU DE RISCO

O procedimento de testes foi feito utilizando o sistema pronto. O primeiro teste calcula o Grau de Risco da cobrança para o bairro Tinguazinho do Município de Nova Iguaçu. Este teste teve como objetivo analisar o Grau de Risco da cobrança em um determinado bairro.

Filtros escolhidos:

Empresa escolhida: “Lojas Sobral”

Tipo de consulta: Acertividade x Efetividade

Consulta por Bairro

Modo de exibição: Personalizada

Cidade: Rio de Janeiro

Bairro: Tinguazinho

Com base nos filtros informados o SIGATDSC apresentou como resultado o Grau de Risco para a empresa “Lojas Sobral” no Bairro Tinguazinho. Essa análise é possível após realizar uma semana de trabalho na assessoria de cobrança, por exemplo, e verifica se a sua estratégia adotada está obtendo resultados positivos ou negativos. A seguir, têm-se os detalhes do processo realizado pelo sistema de inferência nebulosa para calcular o Grau de Risco.

Tabela 5.1: Valores de entrada no sistema de inferência nebulosa

Variável Lingüística	Valor de Entrada
Acertividade (Ent1)	80%
Efetividade (Ent2)	70%

Termos Lingüísticos gerados (com pertinência acima de 0 (zero)):

Tabela 5.2: Termos Lingüísticos

Variável Lingüística	Termos Lingüísticos
Acertividade	Alta/Muito Alta
Efetividade	Alta/Muito Alta

Regras Sensibilizadas:

De acordo com os termos lingüísticos sensibilizados, as regras acionadas foram:

REGRAS: 19, 20, 24 e 25

Sendo que a regra 24 possui um maior grau de ativação. Essa regra cobre a faixa de 0 a 21%. Como se pode observar, a regra mais sensibilizada foi a “24”, gerando como resposta o Grau de Risco com 21%. A resposta apresentada pelo sistema pode ser vista na figura 5.1 a seguir, onde pode ser constatado que de fato o Grau de Risco ficou com 21%.



Análise para Efetividade x Acertividade
Empresa: Lojas Sobral
Bairro: TINGUAZINHO
Acertividade: 80%
Efetividade: 70%
Grau de Risco: 21%
Resultado da Análise: Grau de Risco BAIXO. Manter Estratégia.
Nova Consulta

Figura 5.1: Grau de Risco

5.3 TESTE PARA ANALISAR A CONCENTRAÇÃO DE DEVEDORES

O segundo teste analisa a concentração de devedores da empresa "Lojas Sobral".

Filtros escolhidos:

Empresa escolhida: “Lojas Sobral”

Tipo de consulta: Concentração de devedores

Consulta por Cidade

Modo de exibição: Exibir todas as concentrações de devedores da empresa selecionada

Com base nos filtros informados o SIGATDSC apresenta como resultados a concentração de todos os devedores da empresa “Lojas Sobral” e a alocação dos postos de cobrança da assessoria de cobrança. Conforme a figura 5.2, uma primeira análise seria: o posto de cobrança localizado no bairro Inconfidência está alocado inadequadamente, visto que não possui nenhuma concentração de devedor nesse bairro, uma tomada de decisão seria remanejá-lo para um dos bairros onde possui uma concentração de devedor crítica. Uma segunda análise seria estudar uma estratégia de cobrança com base nos bairros onde possui concentração de devedores com bases em experiências passadas.

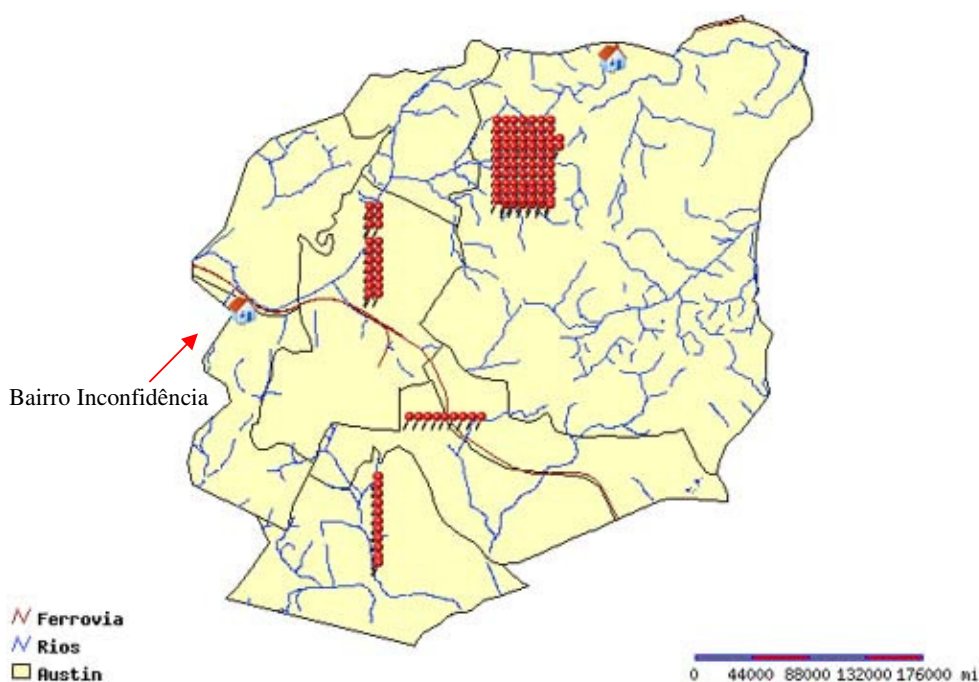


Figura 5.2: Concentração de Devedor

5.4 TESTE PARA ANALISAR O ÍNDICE DE PAGAMENTO

O terceiro teste analisa o Índice de Pagamento do bairro "Tinguazinho" da empresa “Lojas Sobral”.

Filtros escolhidos:

Empresa escolhida: “Lojas Sobral”

Tipo de consulta: Índice de Pagamento

Consulta por Bairro

Modo de exibição: Personalizada

Cidade: Rio de Janeiro

Bairro: Tinguazinho

Com base nos filtros informados o SIGATDSC apresenta como resultados a concentração dos devedores que realizaram pagamentos da empresa “Lojas Sobral” e que estão localizados no Bairro Tinguazinho e a totalização dos meios de acionamento utilizados para contatar os inadimplentes. Conforme a figura 5.3 uma primeira análise seria verificar o tipo de acionamento que está sendo mais eficaz e o menos eficaz.

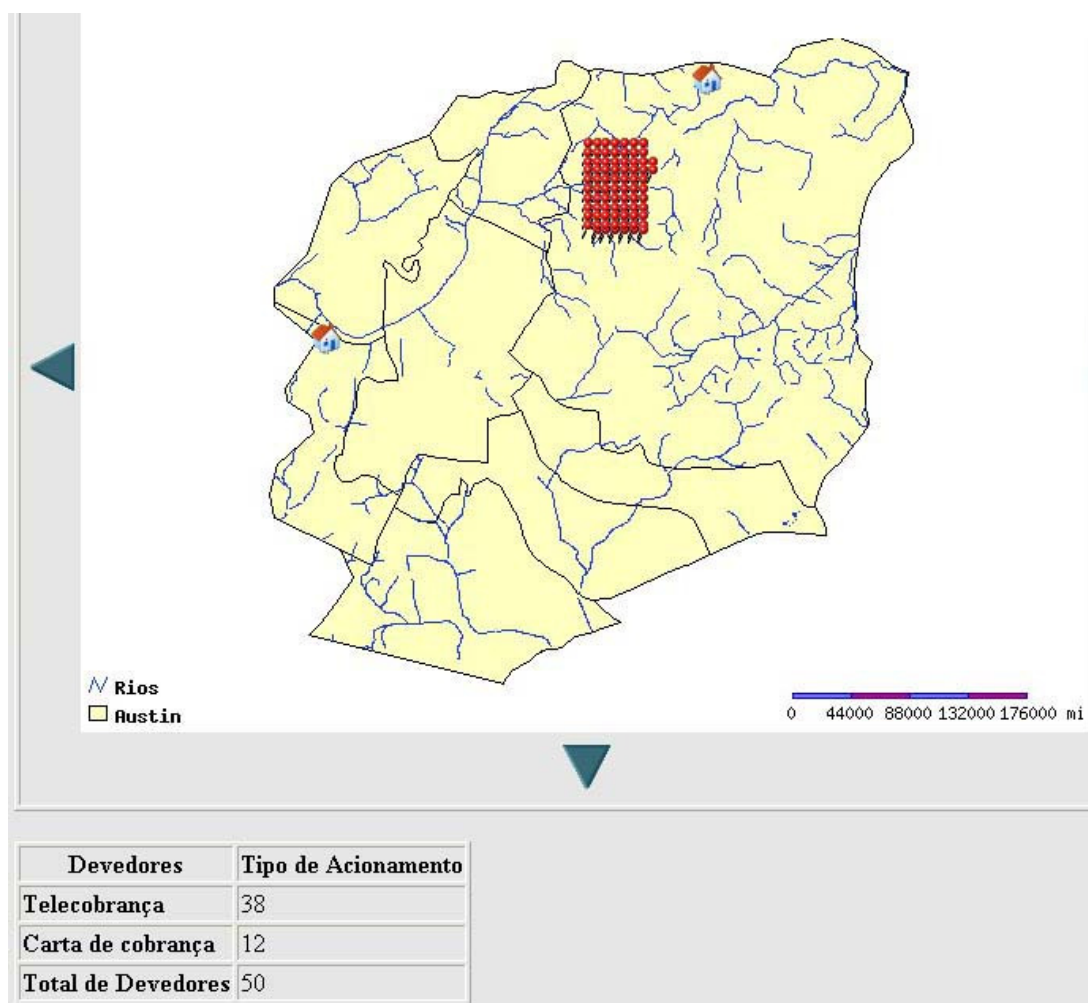


Figura 5.3: Índice de Pagamento

5.5 COMENTÁRIOS

Neste capítulo observou-se que os testes realizados comprovaram o funcionamento correto do SIGATDSC e que as respostas geradas estão corretas e de acordo com o sistema implementado.

CAPÍTULO 6

Conclusões

Com o uso cada vez mais freqüente de ferramentas automatizadas no processo de tomada de decisão, o geoprocessamento vem surgindo como um dos principais instrumentos para o desenvolvimento desse processo, uma vez que ao utilizar as suas ferramentas é possível realizar análises complexas através da integração de dados geográficos com dados comuns (não geográficos), permitindo com isso um tratamento mais eficiente da informação.

O escopo deste trabalho foi desenvolver um Sistema de Informação Geográfica para Apoio à Tomada de Decisão no Setor de Cobrança com o objetivo de analisar as informações sobre os inadimplentes e avaliar a estratégia de cobrança adotada pela assessoria de cobrança.

Foi realizado um estudo sobre a ferramenta MapServer e as metodologias empregadas para o desenvolvimento do protótipo. Também foi apresentada uma modelagem para o SIGATDSC.

A principal contribuição deste trabalho foi o desenvolvimento de um SIG para o setor de cobrança, pois atualmente no Brasil não se tem conhecimento da criação de um SIG nacional para auxiliar o setor de cobrança no processo de tomada de decisão. O presente trabalho mostrou que a aplicação de um algoritmo que auxilie no processo de tomada de decisão é viável e seria uma ferramenta importante para aumentar a eficácia no setor de cobrança.

O protótipo desenvolvido (SIGATDSC) teve seus resultados apresentados no capítulo anterior. Como pôde ser observado, os objetivos esperados foram alcançados, uma vez que o sistema mostrou-se eficaz dentro do escopo em que foi especificado. Todas as técnicas e tecnologias necessárias para o seu desenvolvimento foram aplicadas e atenderam ao que era esperado.

O sistema mostrou-se capaz de auxiliar a tomada de decisão no processo de alocação de postos de cobrança ao analisar a concentração de devedor, já no processo de análise de índice de pagamento foi possível verificar o meio de comunicação mais eficaz em um determinado bairro e por último o sistema permitiu também auxiliar a tomada de decisão após adotar uma estratégia de cobrança através da análise do Grau de Risco da cobrança em um determinado bairro.

Os testes provaram a eficácia do sistema e do motor de inferência criado. As respostas geradas pelo sistema foram conferidas passo a passo e verificou-se que as mesmas encontram-

se corretas de acordo com as especificações do sistema e os resultados obtidos eram os resultados esperados.

O SIGATDSC, da maneira como foi idealizado, possui uma boa funcionalidade, no entanto, o mesmo poderá contar com mais temas para compor as informações geográficas, de modo a obter uma base de dados mais sólida, mais próxima ainda da realidade do mercado de cobrança, o que irá favorecer uma análise mais criteriosa e confiável.

A partir desta dissertação são apresentadas algumas sugestões para trabalhos futuros:

1. Expandir a análise da cobrança para outros estados, visto que este trabalho teve como escopo o Município de Nova Iguaçu do Estado do Rio de Janeiro.
2. Acrescentar as informações geográficas das lojas da empresa que contratou os serviços da assessoria de cobrança, pois o inadimplente poderia realizar o pagamento tanto nos postos da assessoria de cobrança ou diretamente nas lojas da empresa, assim a assessoria de cobrança poderia avaliar se realmente seria necessário manter um posto de cobrança uma vez que nas proximidades teria uma filial da empresa contratante, minimizando os seus gastos.

Referências Bibliográficas

- [1] FIORENTINI, B. S. R., Cuidado com a Inadimplência: Saiba Evitar e Resolver, disponível em <http://www.sebraesp.com.br/principal/abrindo%20seu%20neg%C3%B3cio/produtos%20sebrae/artigos/listadeartigos/cuidado_inadimplencia.aspx>, acessado em 20/03/2007
- [2] RUIZ, M., Globalização, disponível em <<http://www.sociedadedigital.com.br/artigo.php?artigo=123>> , acessado 10/01/2007
- [3] AMADEO, E., Mercado Globalização, disponível em <<http://www.geocities.com/projetopiracema/ArtigosGlobo/MercadoGlobalizacao.html>>, acessado em 10/01/2007
- [4] GOMIERO, F., Redução de custos no mundo globalizado, disponível em <<http://www.economiabr.net/colunas/gomiero/01custos.html>>, acessado em 10/1/2007
- [5] LIMA, M., A Arte da recuperação de clientes inadimplentes, disponível em <<http://www.workshop.com.br/paginas/artigos/arteInadim.htm>>, acessado em 10/01/2007
- [6] BRACHFIELD, J. P., Espanha: A venda e a cobrança: 2 irmãs siamesas, disponível em <http://www.reportercc.com.br/portada/41/espanha_a_venda_e_a_cobran_a_2_irmas_siamesas/>, acessado em 10/01/2007
- [7] (Câmara Neto, 1995)
- [8] SALMASO, F. V.; BERNARDO FILHO, O. e RIBEIRO, J. A., Sistema de Informação Geográfica para Apoio à Tomada de Decisão no Setor de Cobrança – SIGATDSC, Anais do XXIII Congresso Brasileiro de Cartografia, 2007.
- [9] SÁ, E.S, GAUDIE-LEY, M. D. L M., DAVELLI, A. L. F., SOUZA, M. M., FIGUEIREDO, M. G. G., SODRÉ, S. F. e BARROS, V. L., *Manual de normalização de trabalhos técnicos, científicos e culturais*, Editora Vozes, 7^a. edição, 2001.
- [10] Para Ross, Westerfield e Jaffe (ROSS et. al., 1995):

- [11] FILHO, O. B, PEDROZA, A. C. e LEÃO J. L. S. Uma ferramenta para verificação de sistemas distribuídos com lógica nebulosa: Implementação e experiências, Artigo do Departamento de Enga. De Sistemas e Computação/FEN-UERJ, 2001.
- [12] TANSCHKEIT, Ricardo. Controle Nebuloso, 9o CBA, UFES, Vitória, ES, Brasil, pp. 82-95.
- [13] SILVA, A, P. da, Escalamento Heurístico Baseado em Lógica Difusa, Tese de Mestrado, Programa de Engenharia Elétrica, COPPE/UFRJ, 1995
- [14] ZIMMERMANN, H.J., (1985). *Fuzzy MATHematical Tecniques with Applications*, Addison-Wesley Publishing Co, 1985.
- [15] ZIMMERMANN, H. J., (1985). *Fuzzy Set Theory - and Its Applications*, Kluwer-Nijhoff Publishing, 1985.
- [16] KANDEL, A., *Fuzzy Mathematical Techniques with Applications*, Addison-Wesley Publishing Co., 1986.
- [17] TANSCHKEIT, R., *Controle Nebuloso*, Anais do 9º Congresso Brasileiro de Automática, setembro, 1992.
- [18] DUBOIS, D. e PRADE, H., *Fuzzy Sets and Systems, Theory and Applications*, Academic Press, Orlando, 1980.
- [19] TERANO, T., ASAI, K. e SUGENO, M., *Fuzzy Theory and its Applications*, Academic Press, Boston, 1987.
- [20] LEE, C. C., *Fuzzy Logic in Control Systems: Fuzzy Logic Controller*, parts I & II, IEEE Transactions on Systems, Man and Cybernetics, vol. 20, nº 2, 1990.
- [21] SILVA, A. P. da, *Escalamento Heurístico Baseado em Lógica Difusa*, Tese de Mestrado, Programa de Engenharia Elétrica, COPPE/UFRJ, 1995.
- [22] AGUIAR, H e OLIVEIRA, JR., *Lógica Difusa Aspectos práticos e aplicações*, 1999

- [23] ZADEH, L. A., *Fuzzy Sets*, Information and Control, 1965.
- [24] ZADEH, L. A., *Fuzzy Sets as a Basis for a Theory of Possibility*, Fuzzy Sets and Systems 1, 1978.
- [25] ZADEH, L. A., *The Role of Fuzzy Logic in the Management of Uncertainty in Expert Systems*, Fuzzy Sets and Systems 11, 1983.
- [26] ZADEH, L. A., *Fuzzy Logic*, Computer, 1988.
- [27] COX, E., *The fuzzy systems handbook: a practitioner's guide to building, using, and maintaining fuzzy systems*, 1994
- [28] MENDEL, J. M., *Fuzzy Logic Systems for Engineering: A Tutorial*, Proceedings of the IEEE, vol. 83, nº 3, março, 1995.
- [29] WIKIPÉDIA, Sistema de informação geográfica, disponível em <http://pt.wikipedia.org/wiki/Sistema_de_informa%C3%A7%C3%A3o_geogr%C3%A1fica>, acessado em 10/03/2007, 2007
- [30] GALHARDO, R., Transações com Hibernate, disponível em <<http://www.jeebrasil.com.br/mostrar/3>>, acessado em 10/03/2007
- [31] Goodchild, M.F., *Spatial Analysis with GIS: Problems and Prospects* GIS/LIS, The Inform Atlanta, Georgia, pp. 40-48, 1991
- [32] Cowen, D.J., *GIS versus CAD versus DBMS: What are the differences.*, Photog. Eng. and Rem. Sen. 54: 1551-4, 1988
- [33] Parent, P.J., *Geographic Information System: Evolution Academic Involvement and Issus Arising from proliferation of information*, Master Teses University of California, 1988
- [34] Hanigan, F., *GIS by any other name is still...* The GIS Forum 1:6, 1988

- [35] Aronoff,S., Geographic Information Systems: a Management Perspective, WDL Publications, Ottawa, Canadá, 1989
- [36] Ozemoy, V. M., EMITH, D. R e SICHERMAN, A., Evaluating Computerized Geographic Information Systems Using Decision Analysis., Interfaces 11, pp. 92-98, 1981
- [37] DUEKER, K.J., Land Resource Information Systems: A Review of Fifteen Years of Experience., Geoprocessing 1, pp. 105-108, 1979
Opershaw S., Guest Editorial: An Automated Geographical Analysis System., Env. and Plan. A, vol 19, pp 431-436, 1987
- [38] MAPSERVER, disponível em <<http://old-mapserver.gis.umn.edu/home.html>>, acessado em 10/03/2007
- [39] REGENTS OF THE UNIVERSITY OF MINNESOTA, Mapserver, disponível em <<http://old-mapserver.gis.umn.edu/home.html>>, acessado em 15/03/2007
- [40] Kropla, B, MapServer Open Source GIS Development
- [41] MAPSERVER, disponível em <http://mapserver.gis.umn.edu/bem-vindo-ao-mapserver?set_language=pt>, acessado em 10/03/2007
- [42] DM SOLUTIONS GROUP, MS4W – for Windows, disponível em <<http://www.maptools.org/ms4w/index.phtml?page=home.html>>, acessado em 10/01/2007
- [43] KANEGAE, P. E., Funcionamento do Mapserver, disponível em <<http://www.webmapit.com.br>>, acessado em 11/11/2006
- [44] MELLO, A.C., Desenvolvendo Aplicações com Uml, 2002
- [45] FOWLER, M. E KENDALL, S., Uml Essencial um breve guia para a linguagem – padrão de modelagem de objetos, 2000
- [46] Jude, disponível em <<http://www.portaljava.com.br/home/modules.php?name=News&file=article&sid=543>>, acessado em 10/11/2006

- [47] THE APACHE SOFTWARE FOUNDATION, Apache, disponível em <<http://www.apache.org/>>, acessado em 10/11/2006
- [48] THE PHP GROUP, PHP, disponível em <<http://br.php.net/tut.php>>, acessado em 10/11/2006
- [49] THE PHP GROUP, PHP, disponível em <<http://www.php.net/>>, acessado em 10/11/2006

APÊNDICES

APÊNDICE A1

Funções de Pertinência do SLN

As tabelas a seguir correspondem às funções de pertinência da variável lingüística do antecedente acertividade.

Tabela A1.1: Constantes das partes da função de pertinência do termo MuitoBaixa

Parte	Intervalo Inicial	Intervalo Final	Coeficiente Angular	Coeficiente Linear
1	0	20	0	1
2	20	25	-0,2	5
3	25	100	0	0

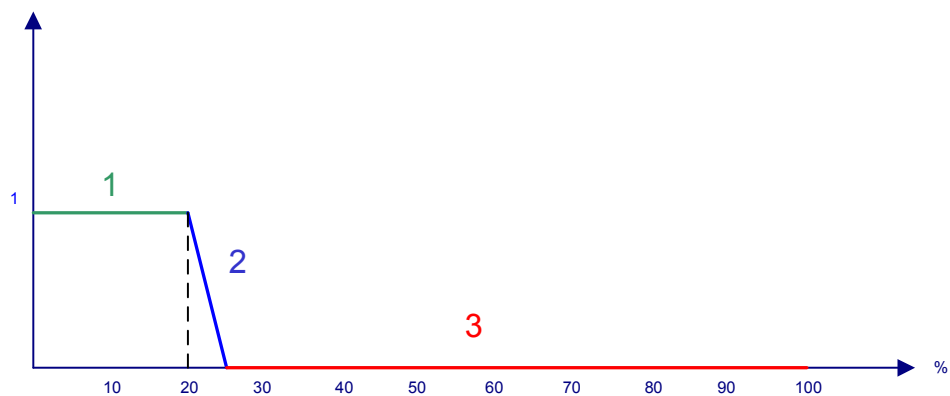


Figura A1.1: Função de pertinência do termo MuitoBaixa (variável Acertividade)

Tabela A1.2: Constante das partes da função de pertinência do termo Baixa

Parte	Intervalo Inicial	Intervalo Final	Coeficiente Angular	Coeficiente Linear
1	0	15	0	0
2	15	20	0,2	-3
3	20	40	0	1
4	40	45	-0,2	9
5	45	100	0	0

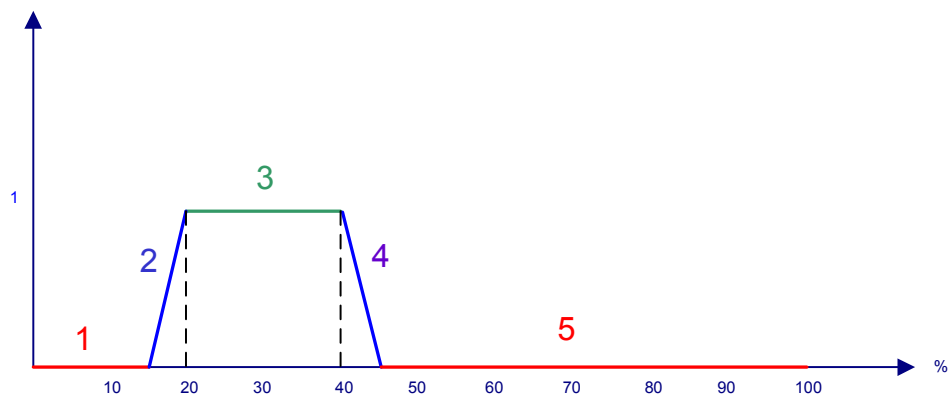


Figura A1.2: Função de pertinência do termo Baixa (variável Acertividade)

Tabela A1.3: Constantes das partes da função de pertinência do termo Média

Parte	Intervalo Inicial	Intervalo Final	Coefficiente Angular	Coefficiente Linear
1	0	35	0	0
2	35	40	0,2	-7
3	40	60	0	1
4	60	65	-0,2	13
5	65	100	0	0

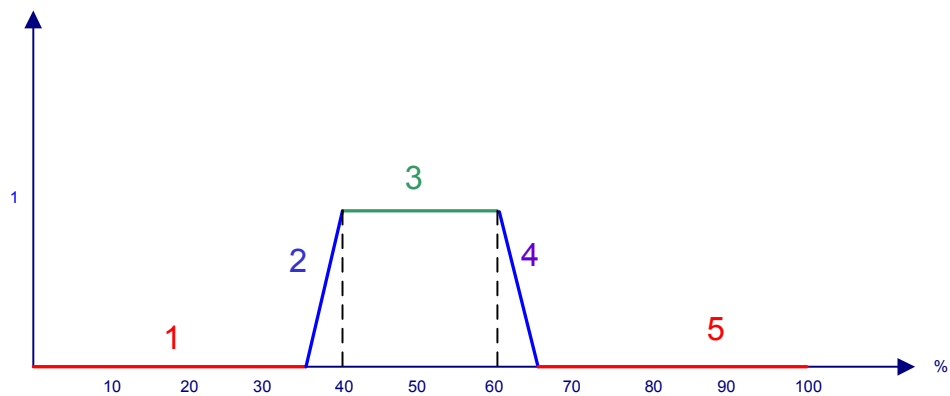


Figura A1.3: Função de pertinência do termo Média (variável Acertividade)

Tabela A1.4: Constantes das partes da função de pertinência do termo Alta

Parte	Intervalo Inicial	Intervalo Final	Coefficiente Angular	Coefficiente Linear
1	0	55	0	0
2	55	60	0,2	-11
3	60	80	0	1
4	80	85	-0,2	17
5	85	100	0	0

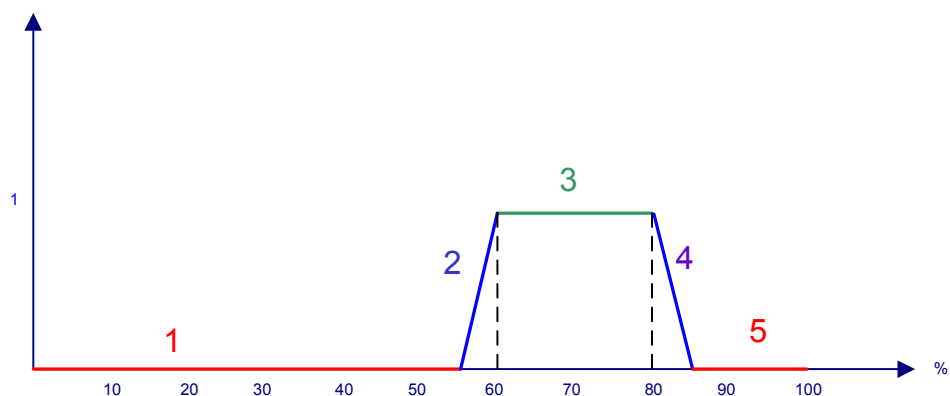


Figura A1.4: Função de pertinência do termo Alta (variável Acertividade)

Tabela A1.5: Constantes das partes da função de pertinência do termo MuitoAlta

Parte	Intervalo Inicial	Intervalo Final	Coefficiente Angular	Coefficiente Linear
1	0	75	0	0
2	75	80	0,2	-15
3	80	100	0	1

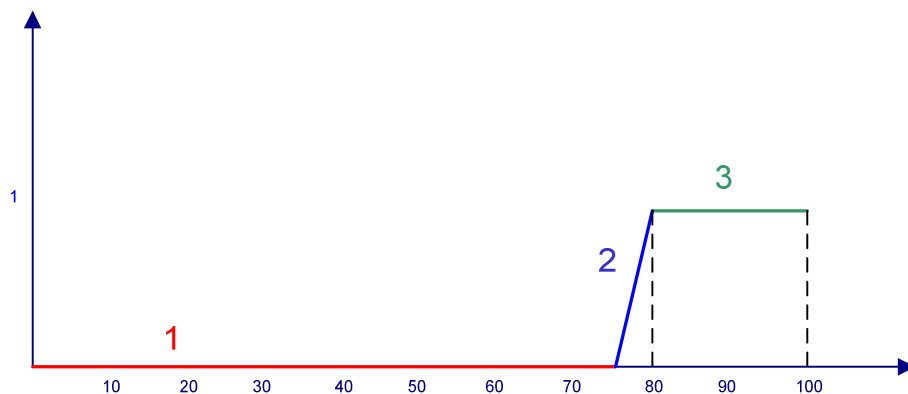


Figura A1.5: Função de pertinência do termo MuitoAlta (variável Acertividade)

As tabelas a seguir correspondem às funções de pertinência da variável lingüística do antecedente efetividade.

Tabela A1.6: Constantes das partes da função de pertinência do termo MuitoBaixa

Parte	Intervalo Inicial	Intervalo Final	Coefficiente Angular	Coefficiente Linear
1	0	20	0	1
2	20	25	-0,2	5
3	25	100	0	0

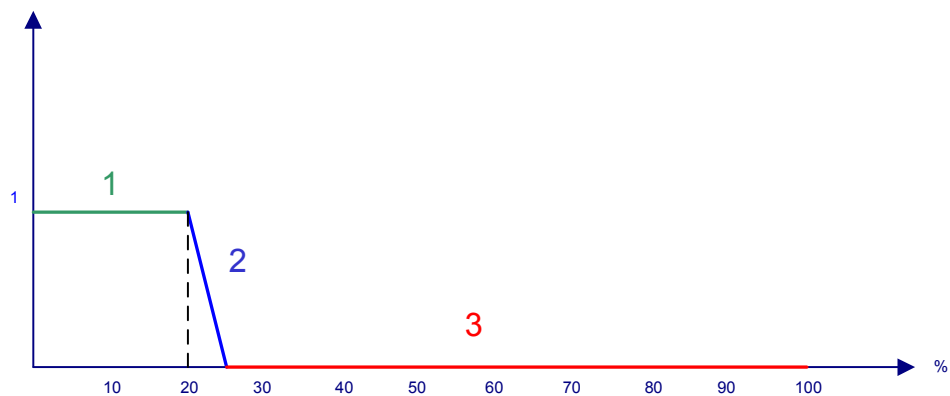


Figura A1.6: Função de pertinência do termo MuitoBaixa (variável efetividade)

Tabela A1.7: Constantes das partes da função de pertinência do termo Baixa

Parte	Intervalo Inicial	Intervalo Final	Coefficiente Angular	Coefficiente Linear
1	0	15	0	0
2	15	20	0,2	-3
3	20	40	0	1
4	40	45	-0,2	9
5	45	100	0	0

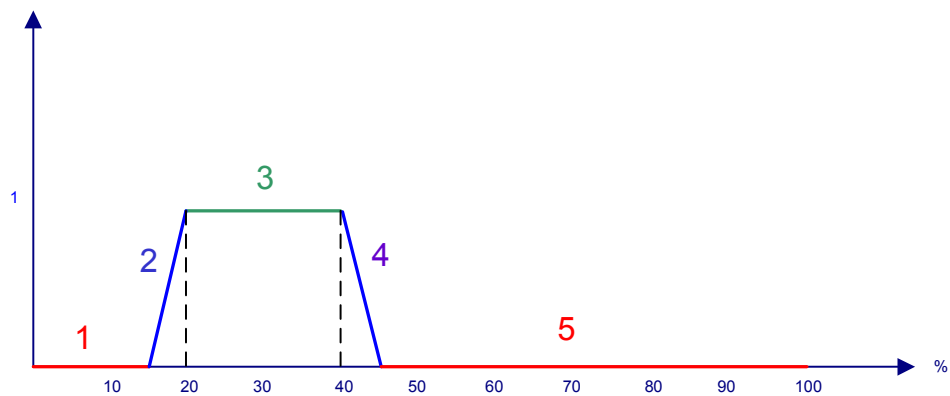


Figura A1.7: Função de pertinência do termo Baixa (variável Acertividade)

Tabela A1.8: Constantes das partes da função de pertinência do termo Média

Parte	Intervalo Inicial	Intervalo Final	Coefficiente Angular	Coefficiente Linear
1	0	35	0	0
2	35	40	0,2	-7
3	40	60	0	1
4	60	65	-0,2	13
5	65	100	0	0

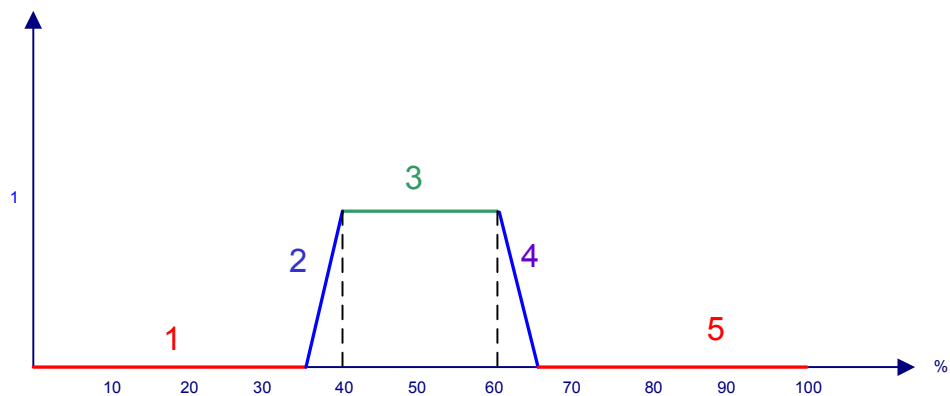


Figura A1.8: Função de pertinência do termo Média (variável Acertividade)

Tabela A1.9: Constantes das partes da função de pertinência do termo Alta

Parte	Intervalo Inicial	Intervalo Final	Coefficiente Angular	Coefficiente Linear
1	0	55	0	0
2	55	60	0,2	-11
3	60	80	0	1
4	80	85	-0,2	17
5	85	100	0	0

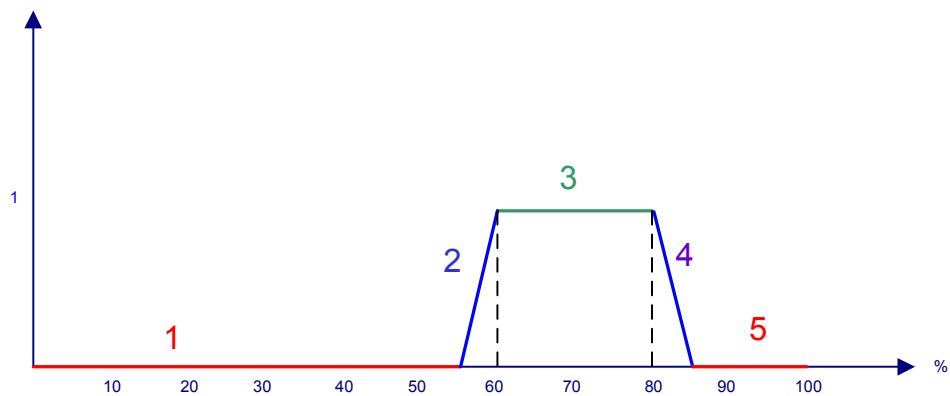


Figura A1.9: Função de pertinência do termo Alta (variável Acertividade)

Tabela A1.10: Constantes das partes da função de pertinência do termo MuitoAlta

Parte	Intervalo Inicial	Intervalo Final	Coefficiente Angular	Coefficiente Linear
1	0	75	0	0
2	75	80	0,2	-15
3	80	100	0	1

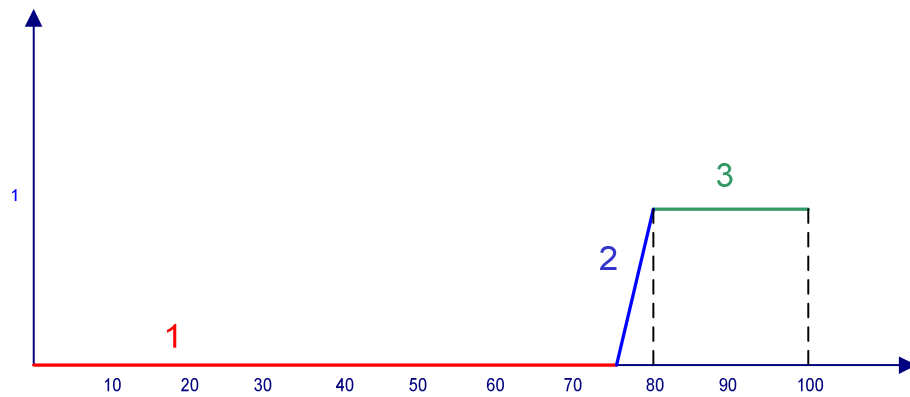


Figura A1.10: Função de pertinência do termo MuitoAlta (variável Acertividade)

As tabelas a seguir correspondem às funções de pertinência da variável lingüística do conseqüente denominado Grau de Risco.

Tabela A1.11: Constantes das partes da função de pertinência do termo MuitoBaixo

Parte	Intervalo Inicial	Intervalo Final	Coefficiente Angular	Coefficiente Linear
1	0	20	0	1
2	20	25	-0,2	5
3	25	100	0	0

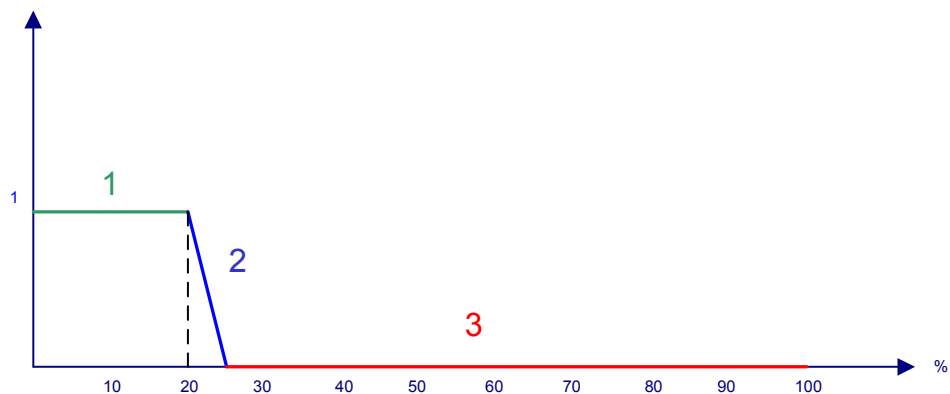


Figura A1.11: Função de pertinência do termo MuitoBaixo (variável Grau de Risco)

Tabela A1.12: Constantes das partes da função de pertinência do termo Baixo

Parte	Intervalo Inicial	Intervalo Final	Coeficiente Angular	Coeficiente Linear
1	0	15	0	0
2	15	20	0,2	-3
3	20	40	0	1
4	40	45	-0,2	9
5	45	100	0	0

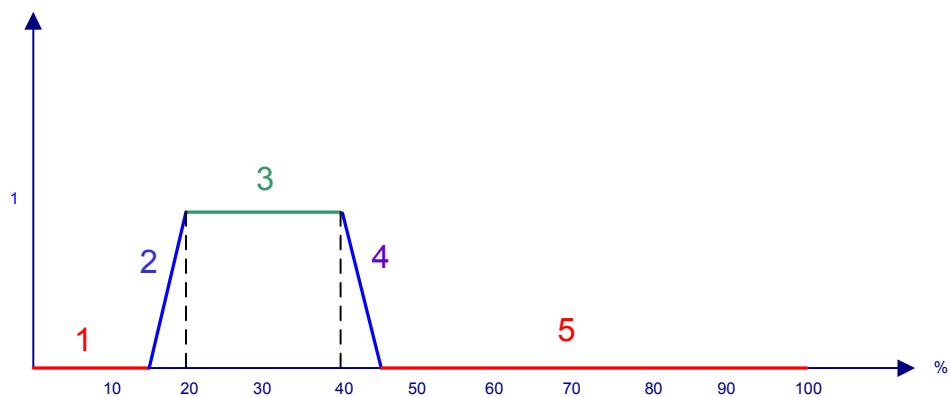


Figura A1.12: Função de pertinência do termo Baixo (variável Grau de Risco)

Tabela A1.13: Constantes das partes da função de pertinência do termo Médio

Parte	Intervalo Inicial	Intervalo Final	Coeficiente Angular	Coeficiente Linear
1	0	35	0	0
2	35	40	0,2	-7
3	40	60	0	1
4	60	65	-0,2	13
5	65	100	0	0

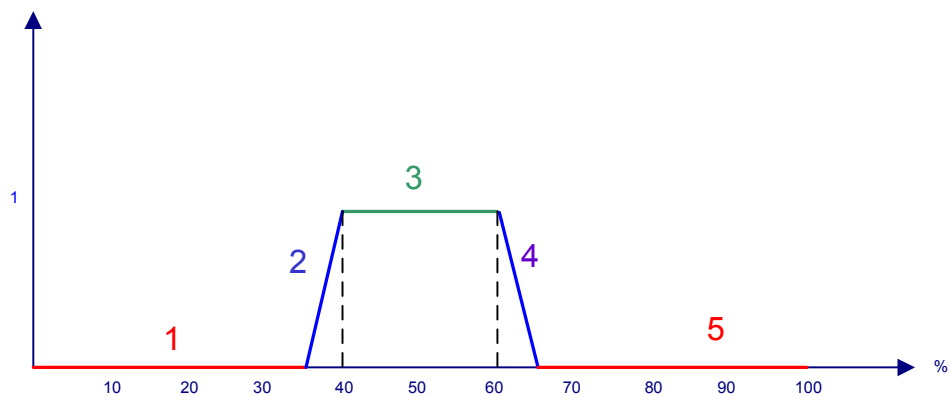


Figura A1.13: Função de pertinência do termo Médio (variável Grau de Risco)

Tabela A1.14: Constantes das partes da função de pertinência do termo Alto

Parte	Intervalo Inicial	Intervalo Final	Coeficiente Angular	Coeficiente Linear
1	0	55	0	0
2	55	60	0,2	-11
3	60	80	0	1
4	80	85	-0,2	17
5	85	100	0	0

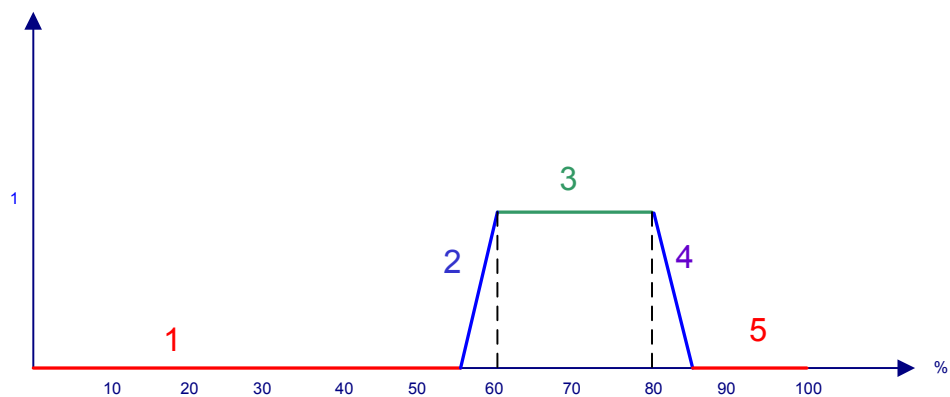


Figura A1.14: Função de pertinência do termo Alto (variável Grau de Risco)

Tabela A1.15: Constantes das partes da função de pertinência do termo MuitoAlto

Parte	Intervalo Inicial	Intervalo Final	Coeficiente Angular	Coeficiente Linear
1	0	75	0	0
2	75	80	0,2	-15
3	80	100	0	1

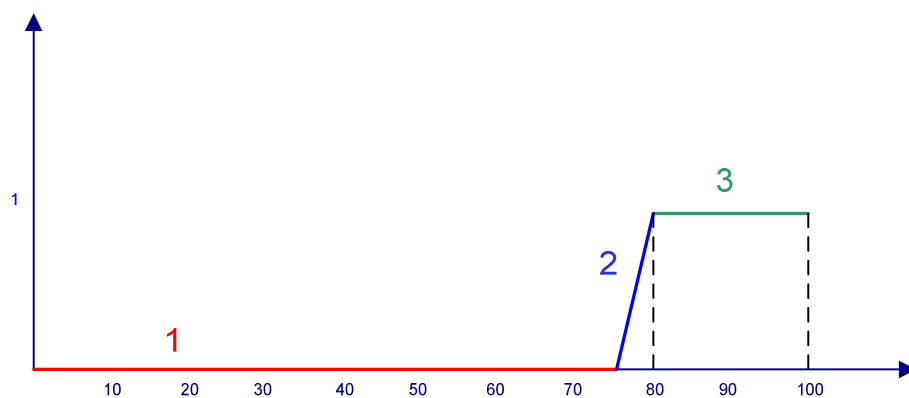


Figura A1.15: Função de pertinência do termo MuitoAlto (variável Grau de Risco)

APÊNDICE A2

Regras definidas para o SLN

A listagem a seguir corresponde às regras definidas para o sistema de lógica nebulosa para mensurar o Grau de risco de uma cobrança.

- Grau de Risco: M(MuitoAlto) - Total de Regras: 3

REGRA: 1

SE acertividade é MuitoBaixa E efetividade é MuitoBaixa

ENTÃO Grau de Risco é MuitoAlto

REGRA: 2

SE acertividade é MuitoBaixa E efetividade é Baixa

ENTÃO Grau de Risco é MuitoAlto

REGRA: 6

SE acertividade é Baixa E efetividade é MuitoBaixa

ENTÃO Grau de Risco é MuitoAlto

- Grau de Risco: M(Alto) - Total de Regras: 5

REGRA: 3

SE acertividade é MuitoBaixa E efetividade é Média

ENTÃO Grau de Risco é Alto

REGRA: 4

SE acertividade é MuitoBaixa E efetividade é Alta

ENTÃO Grau de Risco é Alto

REGRA: 7

SE acertividade é Baixa E efetividade é Baixa

ENTÃO Grau de Risco é Alto

REGRA: 8

SE acertividade é Baixa E efetividade é Média

ENTÃO Grau de Risco é Alto

REGRA: 11

SE acertividade é Média E efetividade é MuitoBaixa

ENTÃO Grau de Risco é Alto

- Grau de Risco: M(Médio) - Total de Regras: 10

REGRA: 5

SE acertividade é MuitoBaixa E efetividade é MuitoAlta

ENTÃO Grau de Risco é Médio

REGRA: 9

SE acertividade é Baixa E efetividade é Alta

ENTÃO Grau de Risco é Médio

REGRA: 10

SE acertividade é Baixa E efetividade é MuitoAlta

ENTÃO Grau de Risco é Médio

REGRA: 12

SE acertividade é Média E efetividade é Baixa

ENTÃO Grau de Risco é Médio

REGRA: 13

SE acertividade é Média E efetividade é Média

ENTÃO Grau de Risco é Médio

REGRA: 14

SE acertividade é Média E efetividade é Alta

ENTÃO Grau de Risco é Médio

REGRA: 16

SE acertividade é Alta E efetividade é MuitoBaixa

ENTÃO Grau de Risco é Médio

REGRA: 17

SE acertividade é Alta E efetividade é Baixa

ENTÃO Grau de Risco é Médio

REGRA: 21

SE acertividade é MuitoAlta E efetividade é MuitoBaixa

ENTÃO Grau de Risco é Médio

REGRA: 22

SE acertividade é MuitoAlta E efetividade é Baixa

ENTÃO Grau de Risco é Médio

- Grau de Risco: M(Baixo) - Total de Regras: 4

REGRA: 15

SE acertividade é Média E efetividade é MuitoAlta

ENTÃO Grau de Risco é Baixo

REGRA: 18

SE acertividade é Alta E efetividade é Média

ENTÃO Grau de Risco é Baixo

REGRA: 19

SE acertividade é Alta E efetividade é Alta

ENTÃO Grau de Risco é Baixo

REGRA: 23

SE acertividade é MuitoAlta E efetividade é Média

ENTÃO Grau de Risco é Baixo

- Grau de Risco: M(MuitoBaixo) - Total de Regras: 3

REGRA: 20

SE acertividade é Alta E efetividade é MuitoAlta

ENTÃO Grau de Risco é MuitoBaixo

REGRA: 24

SE acertividade é MuitoAlta E efetividade é Alta

ENTÃO Grau de Risco é MuitoBaixo

REGRA: 25

SE acertividade é MuitoAlta E efetividade é MuitoAlta

ENTÃO Grau de Risco é MuitoBaixo

APÊNDICE A3

Codificação: Classe DeFuzzyBean

```
<?php
include("DeFuzzy.php");
class DeFuzzyBean
{
    public $Cod_variavel1;
    public $Cod_variavel2;
    public $Valor_var1;
    public $Valor_var2;
    public $ResultadoFinal;

    Public function getResultadoFinal()
    {
        return $this->ResultadoFinal;
    }
    Public function setResultadoFinal()
    {
        $des = new DeFuzzy( $this->Cod_variavel1, $this->Valor_var1, $this->Cod_variavel2,
        $this->Valor_var2);
        $resFinal = $des->percorrerRegras();
        $this->ResultadoFinal = $resFinal;
    }
    Public function setCod_variavel1($cod_variavel1 /* int */)
    { // void
        $this->Cod_variavel1 = $cod_variavel1;
    }

    Public function setCod_variavel2($cod_variavel2/* int */)
    { // void
        $this->Cod_variavel2 = $cod_variavel2;
    }

    Public function setValor_var1($valor_var1/* int */)
    { // void
        $this->Valor_var1 = $valor_var1;
    }

    Public function setValor_var2($valor_var2/* int */)
    { // void
        $this->Valor_var2 = $valor_var2;
    }
}
?>
```

APÊNDICE A4

Codificação: Classe DeFuzzy

```
<?php
class DeFuzzy{
    private $Cod_regra;
    private $Cod_variavel1;
    private $Valor_var1;
    private $Cod_termo1;
    private $Alfa1; // double
    private $Cod_variavel2;
    private $Valor_var2;
    private $Cod_termo2;
    private $Alfa2; // double
    private $Menor_alfa; // double
    private $Cod_Termo_Cons;
    private $denominadorRegra = 0; // double
    private $numeradorRegra = 0; // double
    private $ValorFinal = 0; // double
    private $regras; // ArrayList
    private $novoAlfa; // ArrayList

    public function DeFuzzy($cod_variavel1, $valor_var1, $cod_variavel2, $valor_var2) {
        $this->Cod_variavel1 = $cod_variavel1;
        $this->Valor_var1 = $valor_var1;
        $this->Cod_variavel2 = $cod_variavel2;
        $this->Valor_var2 = $valor_var2;
    }

    public /*double*/ function calculaAlfa (/*double*/ $coef_linear, /*double*/ $coef_angular,
    $valor_var) {
        /*double*/ $alfa = ($coef_angular * $valor_var) + $coef_linear;
        return $alfa;
    }

    public function ConnectionBean(){
        $hostname_connWeb = "localhost";
        $database_connWeb = "lnebulosa";
        $username_connWeb = "root";
        $password_connWeb = "111";

        $con = mysql_connect($hostname_connWeb, $username_connWeb, $password_connWeb);

        if (!$con) {
            die('Não foi possível conectar: ' . mysql_error());
        };

        if ($con) {mysql_select_db($database_connWeb) or die("O banco de dados não foi
        selecionado!");}

        return $con;
    }

    public function verificarTermoRegra($cod_regra, $cod_variavel) {
        $cod_termo = 0;
        $sql = 'SELECT termo.cod_termo FROM termo, regra, antecedente '
        .'WHERE (termo.cod_termo = antecedente.cod_termo OR termo.cod_termo = regra.cod_termo_cons)
        AND '

        .' regra.cod_regra = antecedente.cod_regra AND '
        .' regra.cod_regra = ' . $cod_regra . ' AND '
        .' termo.Cod_Variavel = ' . $cod_variavel . ' ';

        try
        {
            $con = $this->ConnectionBean();
            $rs = mysql_query($sql);

            if ($row = mysql_fetch_array($rs, MYSQL_ASSOC)){
                $cod_termo = $row['cod_termo'];
            }

            mysql_close($con);
            return $cod_termo;
        }
        catch (Exception $e) {
```

```

        echo($e->getMessage());
        return -1;
    }
}

public function verificarTermoConsRegra($cod_regra){
    $cod_termo = 0;
    $sql = 'SELECT cod_termo_cons FROM regra ' . 'WHERE regra.cod_regra = ' . $cod_regra;

    try{
        $con = $this->ConnectionBean();
        $rs = mysql_query($sql);

        if ($row = mysql_fetch_array($rs, MYSQL_ASSOC)){
            $cod_termo = $row['cod_termo_cons'];
        }

        mysql_close($con);
        return $cod_termo;
    }
    catch (Exception $e) {
        echo($e->getMessage());
        return -1;
    }
}

public /*double*/ function avaliarMi($cod_termo, $valor_var) {
    /*double*/ $alfa = 0;
    $sql = 'SELECT cod_segmento, coef_angular, coef_linear FROM segmento WHERE cod_termo = ' . $cod_termo . ' AND (intervalo_esquerdo <= ' . $valor_var . ') AND (intervalo_direito >= ' . $valor_var . ') AND (mi_esquerdo <> 0 OR mi_direito <> 0 ) ';

    try{
        $con = $this->ConnectionBean();
        $rs = mysql_query($sql);

        if($row = mysql_fetch_array($rs, MYSQL_ASSOC)){
            $CodSegmentos_termo = $row['cod_segmento'];
            /*double*/ $coef_linear = $row['coef_linear'];
            /*double*/ $coef_angular = $row['coef_angular'];
            $alfa = $this->calculaAlfa($coef_linear, $coef_angular,
$valor_var);

            return $alfa;
        }

        mysql_close($con);
        return $alfa;
    }
    catch (Exception $e) {
        echo($e->getMessage());
        return -1;
    }
}

public /*double*/ function calculaCentroide(/*double*/ $intEsq, /*double*/ $intDir,
/*double*/ $coefAng, /*double*/ $coefLinear) {
    /*double*/ $potencial = pow($intDir, 3);
    /*double*/ $potencia2 = pow($intDir, 2);
    /*double*/ $potencia3 = pow($intEsq, 3);
    /*double*/ $potencia4 = pow($intEsq, 2);
    /*double*/ $aux1 = ($potencial * $coefAng) / 3;
    /*double*/ $aux2 = ($potencia2 * $coefLinear) / 2;
    /*double*/ $aux3 = ($potencia3 * $coefAng) / 3;
    /*double*/ $aux4 = ($potencia4 * $coefLinear) / 2;

    /*double*/ $centroide = ($aux1 + $aux2 - $aux3 - $aux4);
    return $centroide;
}

public /*double*/ function calculaArea(/*double*/ $intEsq, /*double*/ $intDir, /*double*/
$coefAng, /*double*/ $coefLinear) {
    /*double*/ $area;
    /*double*/ $potencial = pow($intDir, 2);
    /*double*/ $potencia3 = pow($intEsq, 2);
    /*double*/ $aux1 = ($potencial * $coefAng) / 2;
    /*double*/ $aux2 = ($intDir * $coefLinear);
    /*double*/ $aux3 = ($potencia3 * $coefAng) / 2;
    /*double*/ $aux4 = ($intEsq * $coefLinear);

```

```

        $area = ($aux1 + $aux2 - $aux3 - $aux4);
        return $area;
    }

    public /*double*/ function CalculaCentroideRegra($codTermoCons, $menor_alfa) {
        $this->numeradorRegra = 0;
        $this->denominadorRegra = 0;
        $sql = 'SELECT cod_segmento, intervalo_esquerdo, intervalo_direito, mi_esquerdo,
mi_direito FROM segmento WHERE cod_termo = ' . $codTermoCons . ' AND (mi_esquerdo <> 0 OR
mi_direito <> 0 ) ';

        try{
            $con = $this->ConnectionBean();
            $rs = mysql_query($sql);

            while($row = mysql_fetch_array($rs, MYSQL_ASSOC)){
                $codSegmento = $row['cod_segmento'];
                /*double*/ $intEsq = $row['intervalo_esquerdo'];
                /*double*/ $intDir = $row['intervalo_direito'];
                /*double*/ $miEsq = $row['mi_esquerdo'];
                /*double*/ $miDir = $row['mi_direito'];

                $numeradorSegmento = 0;
                $denominadorSegmento = 0;

                if ($miEsq == 1) {
                    $miEsq = $menor_alfa;
                }
                if ($miDir == 1) {
                    $miDir = $menor_alfa;
                }

                $coefAng = ($miEsq - $miDir) / ($intEsq - $intDir);
                $coefLinear = $miEsq - ($intEsq * $coefAng);

                $numeradorSegmento = $this->calculaCentroide($intEsq, $intDir, $coefAng,
$coefLinear);
                $denominadorSegmento = $this->calculaArea($intEsq, $intDir, $coefAng,
$coefLinear);

                $this->numeradorRegra = $this->numeradorRegra + ($numeradorSegmento);
                $this->denominadorRegra = $this->denominadorRegra +
($denominadorSegmento);
            }
            $CentroideRegra = $this->numeradorRegra / $this->denominadorRegra;
            mysql_close($con);
            return $CentroideRegra;
        }
        catch (Exception $e) {
            echo($e->getMessage());
            return -1;
        }
    }

    public /*double*/ function percorrerRegras(){
        /*double*/ $NumeradorGeral = 0;
        /*double*/ $DenominadorGeral = 0;
        /*double*/ $centroideRegra = 0;

        $sql = 'SELECT cod_regra, cod_termo_cons FROM regra';

        try{
            $con = $this->ConnectionBean();
            $rs = mysql_query($sql);

            while($row = mysql_fetch_array($rs, MYSQL_ASSOC)){
                $centroideRegra = 0;
                $this->denominadorRegra = 0;

                $this->Cod_regra = $row['cod_regra'];

                $this->Cod_termo1 = $this->verificarTermoRegra($this->Cod_regra, $this-
>Cod_variavell);
            }
        }
    }

```

```

        if ($this->Cod_termo1 > 0) {

            $this->Alfa1 = $this->avaliarMi($this->Cod_termo1, $this-
>Valor_var1);

            if ($this->Alfa1 < 0) {
                echo('ERRO DO ALFA 1');
                return -1;
            }
        }
        else{
            echo('ERRO DO TERMO 1');
            return -1;
        }

        $this->Cod_termo2 = $this->verificarTermoRegra($this->Cod_regra, $this-
>Cod_variavel2);

        if ($this->Cod_termo2 > 0) {
            $this->Alfa2 = $this->avaliarMi($this->Cod_termo2, $this-
>Valor_var2);

            if ($this->Alfa2 < 0) {
                echo('ERRO DO ALFA 2');
                return -1;
            }
        }
        else{
            echo('ERRO DO TERMO 2');
            return -1;
        }

        if ($this->Alfa1 <= $this->Alfa2){
            $this->Menor_alfa = $this->Alfa1;
        }
        else{
            $this->Menor_alfa = $this->Alfa2;
        }

        if ($this->Menor_alfa > 0){

            $this->Cod_Termo_Cons = $this->verificarTermoConsRegra($this-
>Cod_regra);

            $centroideRegra = $this->CalculaCentroideRegra($this->Cod_Termo_Cons, $this-
>Menor_alfa);

            $this->regras[] = $this->Cod_regra;
            $this->novoAlfa[] = $this->Menor_alfa;

            $NumeradorGeral = $NumeradorGeral + ($centroideRegra * $this-
>denominadorRegra);

            $DenominadorGeral = $DenominadorGeral + $this->denominadorRegra;
        }
    }
    $this->ValorFinal = $NumeradorGeral / $DenominadorGeral;
    mysql_close($con);

    return $this->ValorFinal;
}
catch (Exception $e) {
    echo($e->getMessage());
    return -1;
}
}
public /*Iterator*/ function getRegras() {
    /*Iterator*/ $Regras = $this->regras;
    return $Regras;
}
public /*Iterator*/ function getNovoAlfa() {
    /*Iterator*/ $Alfas = $this->novoAlfa;
    return $Alfas;
}
}
?>

```

APÊNDICE A5

MapFile: posto.map

```
NAME "posto"
UNITS dd
EXTENT 647518.88 7482948.23 658532.35 7490482.84
SIZE 640 480
IMAGECOLOR 255 255 255
IMAGETYPE PNG
FONTSET "FONTSET.txt"

WEB
  IMAGEPATH      "../tmp/"
  IMAGEURL       "/sigatdsc/tmp/"
END

SYMBOL
  NAME "Posto"
  TYPE pixmap
  IMAGE "postos.gif"
END

SYMBOL
  NAME "Devedor"
  TYPE pixmap
  IMAGE "alfinete.gif"
END

LEGEND
  IMAGECOLOR 255 255 255
  KEYSIZE 12 10
  STATUS embed
END

#####
# Basic layer
#
LAYER
  NAME "austin"
  DATA "AUSTIN"
  STATUS on
  TYPE polygon

  TOLERANCE 1          # must be within 1 tolerance unit
  TOLERANCEUNITS miles # units for tolerance values is miles

  CLASS
  TEMPLATE "dummy.html"
  NAME "Austin"
    OUTLINECOLOR 0 0 0
    COLOR 255 255 192
  END
END

#####
#Escala do mapa
SCALEBAR
  LABEL
    COLOR 0 0 0
    ANTIALIAS TRUE
    SIZE small
  END
  POSITION lr
  STYLE 0
  COLOR 128 128 128
  BACKGROUNDCOLOR 255 0 0
  IMAGECOLOR 255 255 255
  INTERVALS 4
  OUTLINECOLOR 0 0 255
  TRANSPARENT OFF
  STATUS embed
  UNITS MILES
END
```

```
#####
# Mapa de referência
#
REFERENCE
  IMAGE "C:/ms4w/Apache/htdocs/sigatdsc/mapas/fifth_wpgref.gif"
  SIZE 268 269
  EXTENT 647518.88 7482948.23 658532.35 7490482.84
  STATUS ON
  COLOR -1 -1 -1
  OUTLINECOLOR 255 0 0
END

#####
# Camada de Rios
#
LAYER
  NAME "rios"
  DATA "RIOS_INTER"
  STATUS on
  TYPE LINE
  CLASS
  NAME "Rios"
    OUTLINECOLOR 0 0 0
    COLOR 18 58 230
  END
END

#####
#Camada de Ferrovia
LAYER
  NAME "ferrovia"
  DATA "FERROVIA_INTER"
  STATUS OFF
  TYPE LINE
  CLASS
  NAME "Ferrovia"
    OUTLINECOLOR 136 17 17
    COLOR 136 17 17
  END
END

#####
#Camada de LOGRADOURO
LAYER
  NAME "LOGRADOURO"
  DATA "lograd_inter"
  STATUS OFF
  TYPE LINE
  CLASS
  NAME "logradouro"
    OUTLINECOLOR 0 0 0
    COLOR 0 0 0
  END
END

#####
#Camada de nomerua
LAYER
  NAME "ruas"
  DATA "lograd"
  STATUS OFF
  TYPE Line
  CLASS
    OUTLINECOLOR 0 0 0
    COLOR 0 0 0
  END
END
```

```
#####
# Camada de Posto de Cobrança - pontos de interesse
#
LAYER
  NAME "postos"
  STATUS default
  TYPE point
  LABELCACHE on
  TOLERANCEUNITS miles
  CLASS
    SYMBOL "Posto"
    SIZE 20
    STYLE
      COLOR 255 0 0
    END
    TEMPLATE "dummy.html"
    TEXT ""
    LABEL
      BUFFER 20
    END
  END
END

#####
# Camada de Devedores - pontos de interesse
#
LAYER
  NAME "poidev"
  STATUS default
  TYPE point
  LABELCACHE on
  TOLERANCEUNITS miles
  CLASS
    SYMBOL "devedor"
    SIZE 12
    STYLE
      COLOR 255 0 0
    END
    TEMPLATE "dummy2.html"
    TEXT ""
    LABEL
      BUFFER 20
    END
  END
END

END # mapfile
```