

It -
208

UNIVERSIDADE EDUARDO MONDLANE



FACULDADE DE CIÊNCIAS

DEPARTAMENTO DE MATEMÁTICA E INFORMÁTICA

**SISTEMA AUTOMATIZADO DE GESTÃO DE
ARRENDAMENTOS DE IMOBILIÁRIO:**

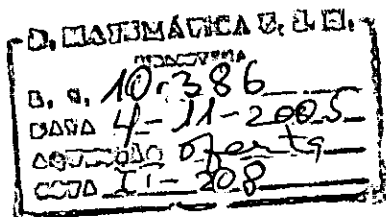
O CASO DA IMOVISA, Lda

TRABALHO DE LICENCIATURA

ESTUDANTE:
RODRIGUES ZICAI FAZENDA

SUPERVISOR:
Dr. JOÃO DIAS

Maputo, aos 17 de Julho de 2001



AGRADECIMENTOS

Um velho adágio do Shakespeare diz que "O mundo inteiro é um palco, onde todas mulheres e todos homens são meros participantes: Eles tem suas saídas e suas entradas, e cada um por sua vez desempenha muitos papéis".

Várias pessoas como Dr. Augusto Tomo Psico e o Eng^o Jorge Damasceno Correia, desempenharam vários papéis durante a minha fase de estudante rumo à licenciatura

Fora as citadas, outras pessoas foram úteis até que a ideia deste trabalho fosse concretizada. É o caso do Dr. João Dias, na qualidade de meu Supervisor, que me orientou no necessário enquadramento metodológico deste trabalho, iniciando-me na actividade de reflexão científica.

Das coisas mais importantes e que me aponquentaram na vida, não fica alheia a memória dos meus pais, Zicai Fazenda e Helena Chuva, os verdadeiros mestres, ao gerarem-me e educarem-me desde a fase de miúdo até ao meio da minha carreira universitária. A minha memória nunca os abandonará.

À minha família, aos amigos e a todos que, directa ou indirectamente, me ajudaram nas discussões inevitáveis à volta deste trabalho, vai o meu grande louvor e agradecimento

Rodrigues Zicai Fazenda

DECLARAÇÃO DE HONRA

Declaro por minha honra, que este trabalho é resultado da minha investigação e que não foi submetido para outro grau que não seja o indicado – **Licenciatura em Informática** – da Universidade Eduardo Mondlane.

Maputo, aos 17 de Julho de 2001

O Estudante

(Rodrigues Zicai Fazenda)



ESTRUTURA DO RELATÓRIO DO PRESENTE TRABALHO DE LICENCIATURA

O presente trabalho referente ao estudo do caso de arrendamento de Imóvel com base na Metodologia de Edward Yourdon é subdividido em 11 (onze) capítulos, dos quais alguns reúnem subcapítulos consoante a sua complexidade.

O capítulo 1, referente a introdução, resume-se a actividade fundamental sobre os arrendamentos, tomando o caso concreto da Imovisa.

O capítulo 2 define os objectivos principais a atingir com este projecto em estudo, enquanto que a descrição detalhada sobre o sistema que se pretende modelar pode ser visto no capítulo 3.

As formas usadas para a recolha dos dados para o presente estudo são descritas no capítulo 4 e a análise essencial (modelo ambiental e comportamental) no capítulo 5.

A fronteira de automatização, desenho dos interfaces, relatórios feitos pelo utilizador e os comentários dessas exigências encontram-se no capítulo 6.

O capítulo 7 fala do desenho (Lógico e físico) da base de dados, o diagrama de estrutura, a especificação geral dos módulos e o esquema definitivo dos interfaces.

O capítulo 8 comporta a criação e implementação da base de dados, detalhando a rede a implementar no futuro dentro da Imovisa, auditoria do sistema, programação e testes da base de dados e a segurança lógica que poderá ser implementada.

O capítulo 9 mostra as conclusões e recomendações, capítulo 10 a Bibliografia e por fim o capítulo 11 que contém os anexos.

CONTEÚDO

	Página
1. INTRODUÇÃO.....	4
2. OBJECTIVOS.....	6
3. DESCRIÇÃO DETALHADA DO SISTEMA A MODELAR.....	7
4. METODOLOGIA DE TRABALHO APLICADA.....	10
5. A ANÁLISE ESSENCIAL DO SISTEMA.....	11
DE ARRENDAMENTO DA IMOVISA.....	11
O Modelo Essencial segundo Yourdon.....	11
5.1 O MODELO AMBIENTAL.....	11
O Diagrama de Contexto.....	13
Eventos.....	16
Tabela de Eventos do sistema em estudo.....	19
5.2 O MODELO COMPORTAMENTAL.....	21
Modelação das funções do sistema: o diagrama de fluxo de dados (DFD).....	22
O Dicionário de Dados.....	26
A modelação dos dados a reter pelo sistema: o Diagrama de Entidades – Relacionamentos (DER).....	26
Especificação geral de Processos Terminais.....	28
Regras de Actividade.....	32
6. O MODELO DE IMPLEMENTAÇÃO DO UTILIZADOR.....	34
Fronteira de Automatização escolhida pelo utilizador.....	35
Desenho dos interfaces feitos pelo utilizador.....	36
Desenho de relatórios feitos pelo utilizador.....	37
Comentários das exigências do utilizador.....	38
7. DESENHO DO SISTEMA.....	38
7.1 DESENHO LÓGICO DA BASE DE DADOS.....	38
Esquema de Relações.....	38
Esquema de Tabelas Normalizadas.....	39
Restrições de Integridade.....	39
7.2 DESENHO FÍSICO DA BASE DE DADOS.....	41
VISÕES (VIEWS).....	41
Indexação.....	43
7.3 A VISÃO ESTRUTURAL DO SISTEMA :.....	45
O DIAGRAMA DE ESTRUTURA.....	45
7.4 ESPECIFICAÇÃO GERAL DOS MÓDULOS.....	47
7.5 ESQUEMA DE INTERFACES DEFINITIVO.....	50
8. CRIAÇÃO E IMPLEMENTAÇÃO DA BASE DE DADOS.....	56
8.1 REDE A IMPLEMENTAR NA IMOVISA.....	56
8.2 AUDITORIA DO SISTEMA.....	57
8.3 PROCEDIMENTOS DE AUDITORIA DO SISTEMA.....	59
IMOBILIÁRIO DA IMOVISA.....	59
8.4 TESTE DA BASE DE DADOS.....	62
PROGRAMAÇÃO E TESTES.....	62

8.5 SEGURANÇA DO SISTEMA A SER IMPLEMENTADO	63
9. CONCLUSÕES E RECOMENDAÇÕES	64
9.1- CONCLUSÕES	64
9.2- RECOMENDAÇÕES	65
10. BIBLIOGRAFIA	67
REFERÊNCIAS BILIOGRÁFICAS	67
BILIOGRAFIA CONSULTADA	68
11 - ANEXOS	70
Anexo I	71
Anexo II	72
DIAGRAMA DE FLUXO DE DADOS PRELIMINAR	72
Anexo III	81
PROCESSOS DE BAIXO NÍVEL	81
Anexo IV: DICIONÁRIO DE DADOS	83
DICIONÁRIO DE DADOS DE ALGUNS FLUXOS	83
DICIONÁRIO DE DADOS DE PROCESSOS	84
DICIONÁRIO DE DADOS DE ARQUIVOS	86
Anexo V: O DIAGRAMA ENTIDADE RELACIONAMENTO	88
Anexo VI: ESQUEMA RELACIONAL	89
Anexo VIII: CRIAÇÃO DE TABELAS USANDO O SQL	91
Anexo IX: CRIAÇÃO DE ALGUMAS CONSULTAS USANDO O SQL	97
Anexo X: DIAGRAMA ESTRUTURAL	100
Anexo XI: GLOSSÁRIO DE TERMOS USADOS	103
Anexo XII: MATERIAL USADO	118
PARTE TEÓRICA	118
PARTE PRÁTICA	118

1. INTRODUÇÃO

Chegado ao ano de 1987, Moçambique implementa uma estratégia de economia do mercado que visa potenciar o empresariado nacional rumo ao desenvolvimento. É neste âmbito que a criação de empresas privadas tenha sido a parte mais notável nos últimos anos, o que determina sobremaneira a rápida industrialização e desenvolvimento. Assim posiciona-se o grupo Visabeira – Moçambique, que possui várias empresas como são os casos da Digitel, Televisa e Imovisa.

O presente trabalho, tem como base a Imovisa, uma componente do grupo Visabeira – Moçambique, cuja função essencial é de Gestão imobiliária desse mesmo Grupo Visabeira, gestão imobiliária de pessoas colectivas e gestão imobiliária de terceiros.

Para a Imovisa, as imobiliárias que estão sob a sua gerência pertencente a pessoas colectivas são do Banco de Moçambique e TDM

As actividades diárias da Imovisa, desde a inscrição para obtenção dum arrendamento até ao pagamento das facturas mensais pela parte do inquilino (Locatário), são feitas num monoposto, que funciona na Sede desta.

Uma pessoa singular ou colectiva, na necessidade de arrendar um Imóvel (Vivenda, Armazém, Edifício ou Apartamento), realiza todas operações na sede da Imovisa, desde a sua candidatura a inquilino (Locatário), onde preenche uma série de documentos que correspondem a realização de todas as formalidades, Locatário (que passa a ser residente num dos Imóveis sob tutela da Imovisa), tem a obrigação de pagar os valores de arrendamentos mensais tanto como a hipótese de solicitação de rescisão de contratos.

Para tal, o maior constrangimento que se coloca é referente às distâncias que estes clientes percorrem. É o exemplo dum residente na Matola (cerca de 19

Km), que tem que sair mensalmente, da casa onde se encontra a residir para exercer a obrigação de pagar a factura da renda correspondente.

O cliente da Imovisa nem sempre é satisfeito, podendo, em certas ocasiões, ser obrigado a levar várias horas a espera do fim do atendimento, pois todo o processo é manual.

Existiriam duas alternativas para melhor servir os clientes :

- Instalar postos em vários Distritos de Maputo-Cidade e Matola mantendo o processamento manual;
- Desenvolver um sistema de Gestão Imobiliária, como forma de rentabilizar o atendimento dos clientes.

Outra questão que se levanta é o de difícil controle do cliente com mau currículo, i.e., que tenha sido expulso dum Imóvel da Imovisa e queira tomar a voltar depois de vários anos com intuito de obter novo arrendamento, o que pelas normas da empresa não é permitido.

Com as situações descritas, pretende-se criar uma base de dados para o melhor controle do cliente, atendimento rápido e oferta de melhores serviços, contribuindo sobremaneira na satisfação e progressão da empresa.

O propósito do tema consiste em delinear uma estratégia de implementação do processo de descentralização do trabalho de atendimento, no sentido da satisfação do cliente, implementação dum sistema, no intuito de resolver pontualmente situações não previstas, e sobretudo, garantir a qualidade da informação processada sem deturpar a segurança do sistema, e também a apresentação de mecanismos que regulem o funcionamento dos serviços para evitar a violação das normas dentro da empresa. A estrutura orgânica da empresa é apresentada no Anexo I.

2. OBJECTIVOS

2.1 OBJECTIVO GERAL

- Conceber, desenhar e Implementar um sistema automatizado de Gestão de Arrendamentos de Imóveis, para o caso concreto da Imovisa;
- Estudar a metodologia Yourdon e aplicá-la à concepção de uma base de dados.

2.2 OBJECTIVOS ESPECÍFICOS

- Melhorar a situação actual de processamento imobiliário da Imovisa no que concerne a atribuição de Imóvel e ao sistema de facturação de rendas na forma de:
 - a) Cadastro dos Imóveis;
 - b) Cadastro dos Locatários;
 - c) Registo dos pedidos de arrendamento de casa.
- Propor um modelo novo de processamento de dados para:
 - a) Tratar da cobrança de rendas;
 - b) Tratar da cobrança do valor de caução
- Delinear uma estratégia única e modelos para todos processos relativos à gestão imobiliária, optimização do atendimento e adopção de melhores mecanismos de controle de devedores através de :
 - a) Criação e emissão de recibos para pagamentos simples e acumulados
 - b) Listagem de todos devedores
 - c) Listagem da situação de cada casa (ocupada, desocupada ou degradada)
- Propôr mecanismos de Auditoria de Sistemas para :
 - a) A avaliação da confiabilidade de dados processados por computador;
 - b) Criação de Log Files e de outros dispositivos de segurança para verificar os intrusos no sistema

3. DESCRIÇÃO DETALHADA DO SISTEMA A MODELAR

Os serviços de arrendamento de Imóveis dentro da Imovisa são tratados como principal tarefa por ser este o trabalho essencial que a empresa executa.

Tomemos como exemplo uma pessoa que se desloca à Imovisa (Recepção) com o intuito de arrendar um Imóvel (Apartamento, Edifício ou Estabelecimento Comercial) e que lhe são entregues todas condições necessárias para ser inquilino, ao que passarei a designar candidato a Locatário. Este candidato, depois de realizar uma apreciação exaustiva das condições de arrendamento e manifesta seguidamente, vontade de se tornar cliente, é lhe entregue uma ficha de inscrição pelo sector de Gestão de Propostas, que depois de preenchida é adicionada ao contrato de arrendamento após a assinatura do candidato, e posteriormente entregue ao Sector de Serviços do Cliente, como forma de se criar o seu cadastro. No formulário com cláusulas de arrendamento estão descritas todas as premissas para o efeito, incluindo as condições a oferecer à entidades individuais (pessoa singular) ou colectivas (empresas).

Um pedido de arrendamento possui uma descrição exaustiva do proponente (candidato à Locatário), onde não se esquecem o nome do cônjuge e o agregado familiar deste, como forma de aclarar a situação dos futuros viventes do Imóvel e também servir-se de barreira para criar zelo, pois não se poderia admitir que a uma casa do tipo 2, por exemplo, viessem a habitar mais de seis pessoas. Sem descurar tudo isto, adiciona-se a componente ligada com o assunto que leva o cliente a pretender o arrendamento. O assunto caracteriza-se pelo tipo de actividade a desempenhar dentro da casa (moradia, actividade comercial ou escritórios).

Como atrás descrito, o pedido de arrendamento pode ser individual ou colectivo. Para o caso do pedido ser colectivo, importará reter, para além da maior parte

da informação acerca do caso individual, informações sobre o legítimo representante da empresa que pretende o apartamento de aluguer.

A Imovisa, para além de gerir imóveis já existentes, também executa operações de compra ou mesmo de recepção de imóveis de terceiros (Locadores), como forma de aumentar o parque imobiliário que possui, num claro intuito de satisfação imediata da demanda.

Adquirido o novo imóvel através dos Sectores de Gestão e Tesouraria, este passará por uma fase de cadastro (que é retirar todas referências que importam à empresa para fácil identificação).

Um terceiro (Locador) é a pessoa que, tendo casa própria a alugar não o consegue o fazer porque talvez não encontre um cliente para o efeito, contacta o Sector de Intermediação da Imovisa, pedindo a esta para fazê-lo em troca de uma comissão mensal.

Se o candidato a Locatário é aceite, deverá pagar o primeiro mês de renda em regime de adiantamento e um valor de caução que também corresponde a uma renda mensal, junto ao Serviço de Tesouraria; só assim é que passará de Locatário para cliente. Seguindo-se a este processo, segue-se o período de atribuição do Imóvel escolhido pelo Serviço de Cliente, ao que o contrato de aluguer deverá ser assinado entre este e a empresa. Depois da assinatura do contrato e entregues as chaves, passa a ser Locatário da Imovisa.

Todo o Locatário, em datas que vão entre 01 à 10 de cada mês, procede o pagamento mensal de renda, junto à Tesouraria. Esta renda é referente ao mês seguinte. Em casos de não efectivação do pagamento, é considerado devedor que até 15 dias depois terá de ser notificado à expulsão da casa, salvo em casos excepcionais que possam ser de inteiro reconhecimento do Administrador da empresa.

Cada pagamento possui seus factores identificadores que são: O tipo de pedido e o número do contrato. Quando se efectua um pagamento e para efeitos de confirmação, um recibo é emitido e, só assim, é que se considera cliente não devedor.

Por cada mês, depois dos pagamentos efectuados, interessará à empresa saber o valor das rendas cobradas, rendas em dívida, rendas em dívidas acumuladas, meses em dívida, conhecer os apartamentos pela sua localização ou pela topologia.

Todos os imóveis sob tutela da Imovisa garantem excelentes condições de habitabilidade porque está prevista a sua inspecção constante, para evitar que um candidato à cliente o arrende degradado. Caso o Imóvel esteja arrendado por uma entidade e note-se um estado de degradação, o inquilino é notificado para reparar os danos causados. Se porventura continua a situação anterior, uma notificação de expulsão do Imóvel é passada e entregue, a este inquilino.

Conferida a hipótese de sua expulsão, avalia-se o dano. Se o dano não for extremamente elevado, isto é, não supera o valor depositado no dia da assinatura do contrato de arrendamento (caução), o cliente não pagará pelo dano porque se usa este valor da caução para estes propósitos e a parte restante e entregue ao ex-cliente.

Pode salientar-se ainda, que um contrato possui uma data limite de sua vigência. Ultrapassada a data sem que o Locatário tenha manifestado a vontade de abandonar a casa, o contrato é considerado renovado automaticamente para mais um período igual ao anterior. Caso o cliente não possa cumprir com o contrato e queira abandonar antes do prazo afixado, poderá fazê-lo, bastando um pré aviso. Se o pedido de rescisão do contrato ocorrer na sua data fim e as condições da casa permanecerem óptimas, o valor da caução (pago inicialmente pelo cliente) é lhe devolvido na totalidade.

4. METODOLOGIA DE TRABALHO APLICADA

A captação dos requisitos do sistema da Imovisa foi realizada através de :

- Entrevistas às pessoas mais informadas sobre os problemas e operações da empresa, registando-se as necessidades do sistema a conceber e implementar;
- Entrevistas a grupos de funcionários típicos de uma área de actividade de modo a encontrar contrastes entre os requisitos;
- Estudo da documentação envolvida de modo a seleccionar os dados necessários, estudar exemplos de uso de dados, detectar regras de actividade e procedimentos.
- Observação do trabalho dos funcionários a horas seleccionadas de modo a reter como os dados são tratados e que informação deve ser produzida.

No que diz respeito à Metodologia de Análise de Sistemas foi escolhida a abordagem de Yourdon pelo seu carácter reactivo e por ser mais rápida na ultimização do Modelo lógico definitivo. Uma vez que o autor deste trabalho conhecia outras metodologias estruturadas como o SSADM isto não constituiu grande problema, apesar do necessário estudo do Modelo Essencial de Yourdon descrito mais à frente.

Na fase de Desenho seguiu-se o material actualmente apresentado nas disciplinas de Análise de Sistemas e Base de Dados .

Como meio de confrontação do sistema concebido foi construído um protótipo antes da implementação definitiva. Este protótipo foi apresentado a alguns utilizadores e ao supervisor. O protótipo e a pré implementação final do sistema foi feita em Access.

5. A ANÁLISE ESSENCIAL DO SISTEMA DE ARRENDAMENTO DA IMOVISA

O Modelo Essencial segundo Yourdon

Importa realçar, que a parte mais interessante são os requisitos para satisfazer o usuário, mencionando o mínimo possível sobre como o sistema será implementado. Então, categoricamente posso afirmar, que o modelo de sistema deve pressupor que se dispõe de perfeita tecnologia e que pode ser obtida a custo zero.

De uma forma específica, significa que é necessário a interação com o usuário sobre os requisitos do sistema, procurando evitar no máximo descrever as implementações específicas dos processos do sistema, isto é, não se pode mostrar as funções dum protótipo. O modelo essencial deve descrever o conteúdo dos fluxos e arquivos de dados, sem se preocupar com a sua organização física ou os suportes onde serão guardados no acto da gravação.

Este modelo surge como um dos avanços na análise em que se introduz a utilização da *lista de eventos*, que segundo Shiller (1990), ajuda a orientar o desenvolvimento do próprio sistema, como forma de determinar os seus procedimentos. Permite investigar os dados que o sistema precisa manter, como os dados interactivam, os processamentos que o software precisa de executar e a ordem pela qual serão feitos.

O modelo é composto pelos modelos ambiental e comportamental que a seguir são descritos.

5.1 O MODELO AMBIENTAL

A construção do modelo ambiental é a primeira e mais importante parte da construção de um modelo completo dos requisitos do usuário para um novo

sistema. A essa altura, isso poderia parecer uma tarefa fácil; afinal de contas, o diagrama de contexto consiste apenas em uma e única bolha, e a lista de eventos parece-se com uma simples lista de transações. Mas num grande projecto, pode estar envolvido um grande volume de trabalho: A única bolha do Diagrama de Contexto(DC) pode interagir com dezenas de terminadores externos e podem ter mais de uma centena de fluxo de dados chegando e partindo. A lista de eventos é também um importante esforço em grandes sistemas; podendo existir facilmente mais de uma centena de eventos com os quais o sistema terá de lidar, e todos deverão ser identificados. Além disso, pode ser difícil conseguir uma declaração simples consensual acerca do motivo da existência do sistema.

Uma vez construído o modelo ambiental, ele deve ser minuciosamente revisto por todos os representantes dos usuários principais, bem como pela equipe do projecto. Depois disso pode-se iniciar a construção do modelo comportamental, o modelo do interior do sistema, observando-se o seguinte nos dois modelos:

- A lista de eventos parece-se com uma simples lista de transações enquanto que o DC só tem uma única bolha.
- Cada fluxo de entrada no DC seria necessário ao sistema para reconhecer que um evento ocorreu ou para a produção de uma resposta a um evento
- Cada fluxo de saída deve ser uma resposta a um evento
- Cada evento não temporal da lista de eventos deve ter entradas a partir das quais o sistema pode detectar que o evento ocorreu
- Cada evento deve produzir uma saída imediata com resposta, ou armazenar dados para serem posteriormente emitidas como saída (como resposta ou parte de uma resposta para algum outro evento), ou então ele deve fazer com que o sistema mude de estado (como indicado no

respectivo diagrama de eventos). "Veja-se em Edward Yourdon (1991), Análise Estruturada Moderna, tradução da terceira edição Americana. pp 407 e 397. Prentice-Hall, Inc, New Jersey, USA".

Os produtos finais que compõe este modelo são :

- Uma Declaração dos Objectivos do Sistema;
- Uma Lista/Tabela de Eventos
- Um Diagrama de Contexto
- Um Dicionário de Dados preliminar.

→ Uma Declaração dos Objectivos do Sistema não é necessária na documentação do sistema em estudo neste lugar, pois está inserida nos objectivos deste trabalho de licenciatura (atrás apresentados) .

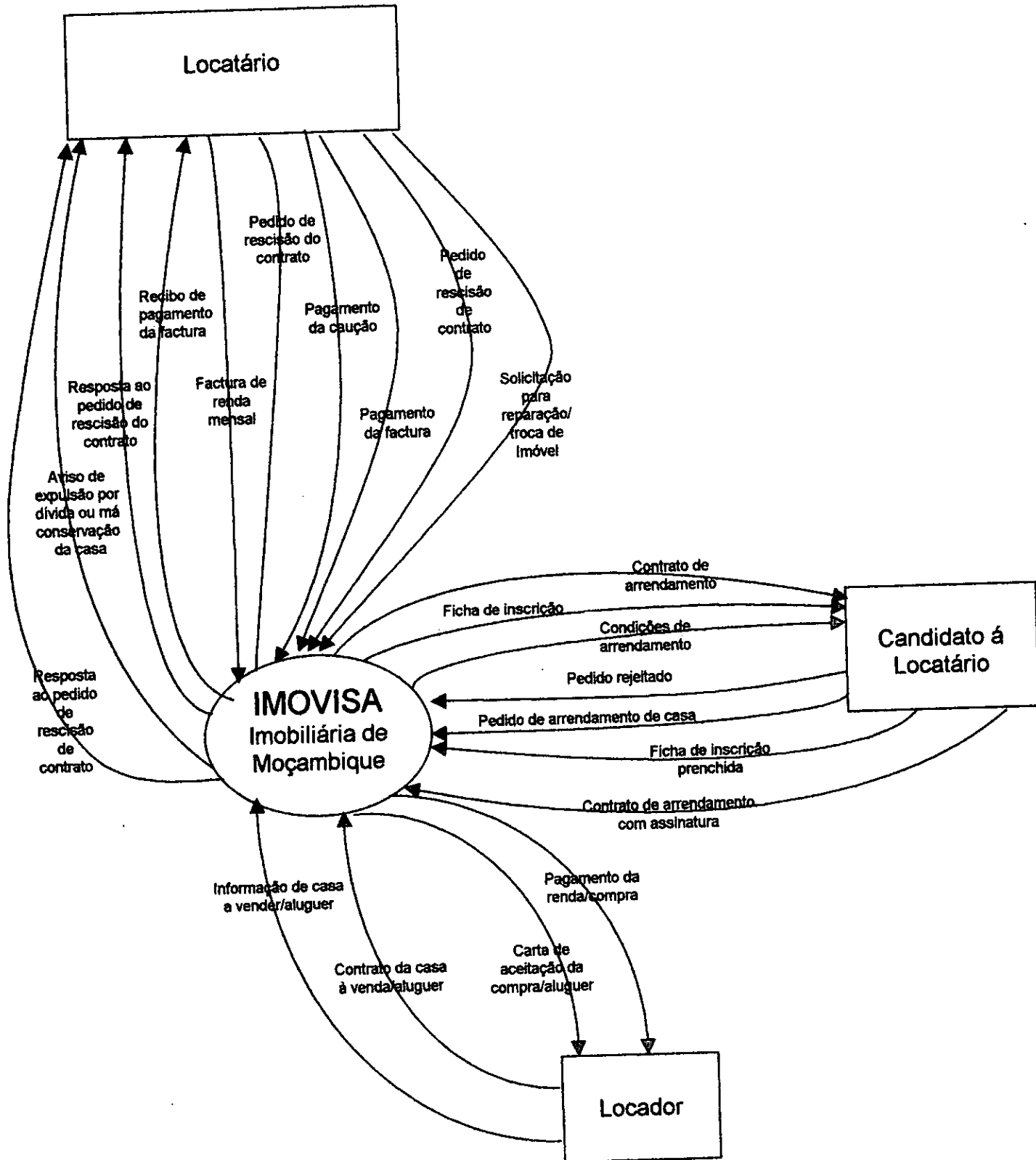
O Diagrama de Contexto

Yourdon (1990), define o Diagrama de Contexto (DC) como sendo uma ferramenta que realça diversas características importantes do sistema no futuro contexto onde ele vai viver. Ele mostra a interacção entre o sistema e o ambiente externo constituído pelas Entidades Externas ou Terminadores, nomeadamente, Candidato à Locatário, Locatário e Locador. Todas as entidades externas que queiram algo do sistema ou recebem alguma peça de dados do mesmo devem ser ilustradas no DC. Caso o sistema em estudo receba ou envie dados para um outro sistema (informatizado) este também deve ser pictorizado como um terminador.

Nele estão representados todos os fluxos de e para o exterior do sistema enviados ou recebidos pelos terminadores..

Como se pode ver no DC a seguir representado, a Imovisa (Imobiliária de Moçambique), constitui um núcleo onde vai toda a informação das entidades externas e onde as mesmas colhem informação após ser processada.

Identificam-se no DC as entidades Candidato à Locatário e Locador como sendo a principal fonte das entradas que são geradas para a entidade Locatário.



Eventos

Segundo Yourdon, uma condição é um evento no ambiente externo que o sistema seja capaz de detectar, podendo ser um sinal, uma interrupção ou a chegada de um pacote de dados⁽¹⁾. Isso fará o sistema passar do estado de aguardando X para o estado de aguardando Y ou de executando a actividade X para executando a actividade Y.

Como parte da mudança de estado, o sistema ao receber um estímulo (normalmente um fluxo de dados) empreenderá uma ou mais acções (efectuará um cálculo, por exemplo) e poderá produzir uma saída como a apresentação de uma mensagem no terminal do usuário,

O padrão para cada evento deverá ser

- O evento acontece numa entidade externa (por exemplo, ' Candidato à Locatário pretende arrendar Imóvel');
- Um estímulo é enviado pela entidade externa através de um fluxo de dados (por exemplo, "Contrato de arrendamento assinado") ou de controlo (há casos em que é possível não haver estímulo como mais tarde se explicará) ;
- Resposta(s) interna(s) do sistema através de um ou mais processos (por exemplo, 'Tratar Serviços Cliente');
- Um ou mais outputs para alguma(s) entidade(s) externa(s) (por exemplo, ' Recibo de Pagamento de caução') ..

Estes eventos podem ocorrer num determinado intervalo de tempo previsto, como são os casos da Tesouraria emitir facturas mensalmente⁽²⁾.

Em certos momentos, a nível do sistema, pode ocorrer um evento não previsto, mas com um tempo curto de duração. Aí estaria a introduzir-se um controlo para o sistema⁽³⁾.

Os eventos que aparecem referenciados são classificados em:

- (1) Evento orientado ao fluxo (EvOF)
- (2) Evento temporal (EvT) (não é necessário um estímulo!)
- (3) Evento de controle (EvC)

Os eventos orientados a fluxos são os eventos que enviam fluxos de dados como estímulo ao sistema.

Os eventos temporais são os que acontecem sempre que termina um ciclo de tempo determinado. Assim, estes não necessitam de estímulo e actuam como se o sistema possuísse um relógio interno, e soubesse que numa determinada altura deverá desencadeá-lo e executar as acções (processos) previstos. Por exemplo, o evento 'É tempo de enviar o Relatório Semanal de contas à Direcção Geral', não necessita de um estímulo, porque o sistema todas as Segundas-feiras activa-o logo mal que se entre no sistema.

Os eventos de controlo ocorrem nos sistemas de Tempo-Real ou noutros quando uma entidade externa envia/recebe um sinal (On/Off, Sim/Não, Activar/Desactivar) que conceptualmente se pode representar por 1 bit . Como resposta ao sinal o sistema deverá possuir pelo menos um processo de controlo.

Os eventos e as respostas do sistema deverão ser detectados duma forma assíncrona, i.e., não deve ser feito o esforço de tentar sincronizá-los logo desde o início. Existem ferramentas posteriores para esta tarefa.

Para o caso do sistema em estudo a lista de eventos e a respectiva classificação seria

- 1- Candidato á Locatário realiza pedido de arrendamento: (EvOF)
- 2- Candidato á Locatário é solicitado pelo sector de controlo de propostas para assinatura do contrato de arrendamento: (EvOF)
- 3- Pedido é encaminhado á Direcção para efeitos de aprovação: (EvOF)

- 4- Departamento Comercial rejeita Pedido do Candidato à Locatário: (EvOF)
- 5- Direcção analisa pedido do Candidato à Locatário: (EvOF)
- 6- Direcção rejeita o pedido do Candidato à Locatário: (EvOF)
- 7- Candidato à Locatário indaga a situação de pedido de arrendamento: (EvC, EcOF)
- 8- Direcção devolve pedido do candidato à locatário para efeitos de regularização: (EvC, EvOF)
- 9- Locatário faz o pagamento da caução: (EvOF)
- 10- Direcção considera aceitável, o Candidato à Locatário ser Locatário: (EvC, EvOF)
- 11- Direcção aceita ter mais um Locatário: (EvOF)
- 12- Locatário solicita à renovação do contrato: (EvC, EvOF)
- 13- Locatário solicita a revisão do contrato: (EvC, EvOF)
- 14- Locatário solicita os Serviços de Cliente para reparação do imóvel: (EvC, EvOF)
- 15- Locatário recebe aviso de expulsão do Imóvel por acumulação de dívidas: (EvT)
- 16- Locatário solicita troca do Imóvel: (EvC, EvOF)
- 17- Factura precisa de ser remetida ao Locatário: (EvT)
- 18- Locatário indaga sobre a situação de uma factura: (EvC, EvOF)
- 19- Locatário envia pagamento da factura: (EvC, EvOF)
- 20- Locatário realiza o pagamento da dívida de rendas atrasadas à Tesouraria: (EvT)
- 21- Locador informa ao serviço de intermediação do aluguer/Venda de Imóvel: (EvC, EvOF)
- 22- O Locador aceita o aluguer do seu Imóvel: (EvOF)
- 23- Direcção pede informação de Imóvel a aluguer/comprar à Gestão: (EvOF)
- 24- Tesouraria propõem a compra do Imóvel ao Locador: (EvOF)
- 25- Direcção aceita proposta do Locador de aluguer/venda de Imóvel: (EvOF)
- 26- Direcção autoriza a tesouraria para pagamento do aluguer/compra do Imóvel: (EvT)
- 27- Tesouraria efectua o pagamento da compra de Imóvel: (EvOF)
- 28- Serviço do Cliente pede o processamento de facturas: (EvT)
- 29- Direcção pretende saber a situação de pagamentos de rendas: (EvC, EvOF)
- 30- Direcção exige listagem de devedores de cada mês: (EvC, EvOF)

31- Direcção solicita listagem de Imóveis pela Topologia: (EvC, EvOF)

32- Direcção precisa (mensalmente) do relatório de receitas: (EvT)

Para que ocorra um determinado evento e se enviem todos os dados necessários, para além do seu estímulo, poderá ser necessário que haja um outro fluxo de dados partindo, por exemplo, de uma outra entidade externa. Vejamos a tabela seguinte que mostra alguns eventos, estímulos, respostas e saídas do processamento.

Tem sido comum a apresentação de uma Tabela de Eventos onde se completam 4 colunas que descrevem, em cada linha, o evento, o estímulo (se houver) que forçará o sistema a dar uma resposta, a(s) resposta(s) do sistema e os outputs (se houverem). De realçar que, para muitos praticantes, é muito mais cómodo iniciar a construção do Modelo Essencial a partir da Tabela de Eventos, pois, desde o início, se obtém um visão macro das funcionalidades do sistema.

Tabela de Eventos do sistema em estudo

Evento	Estímulo (Entrada)	Resposta	Saída(Output)
Candidato á Locatário realiza pedido de arrendamento	Pedido de arrendamento preenchido	Condições de arrendamento	
Candidato á Locatário é solicitado pelo sector de controlo de propostas para assinatura do contrato de arrendamento	Recepção do contrato de arrendamento	Contrato de arrendamento assinado	Contrato + valor de caução a pagar
Pedido é encaminhado á Direcção Geral para efeitos de aprovação	Ficha de inscrição preenchida	Informação de recepção de ficha de inscrição	Parecer da direcção sobre o pedido
Departamento Comercial rejeita Pedido do Candidato á Locatário	Pedido de arrendamento preenchido	Informação sobre a rejeição do pedido de arrendamento	Pedido rejeitado
Direcção analisa pedido do Candidato á Locatário	Pedido + Ficha de inscrição preenchida	Resposta do pedido analisado	
Direcção rejeita o pedido do Candidato á Locatário	Pedido + Ficha de inscrição preenchida	Informação sobre rejeição do pedido á Dep. Comercial	Pedido rejeitado

Evento	Estímulo (Entrada)	Resposta	Saída(Output)
Candidato á Locatário indaga a situação de pedido de arrendamento	Existência do anterior contrato com mau curriculum	Informação de rejeição do pedido	Lista de pedidos rejeitados
Direcção devolve pedido do candidato á locatário para efeitos de regularização	Pedido de arrendamento mal preenchido	Carta de devolução do pedido de arrendamento	-----
Locatário faz o pagamento da caução	Contrato + valor de caução	Recibo	Recibo
Direcção considera aceitável, o Candidato á Locatário ser Locatário	Contrato + Recibo de caução	Carta de admissão à ocupação do Imóvel	Relatório actualizado de novos Locatários
Direcção aceita ter mais um Locatário	Contrato assinado + valor da caução	Pagamento da caução	Recibo
Locatário solicita à renovação do contrato	Anterior Contrato + carta de prorrogação de contrato	Resposta da solicitação	Relatório de Locatário
Locatário solicita a revisão do contrato	Carta solicitando revisão do contrato	Parecer da Direcção	Relatório de contrato revistos
Locatário solicita os Serviços de Cliente para reparação do imóvel	Informação de danos do Imóvel	Resposta á solicitação	Lista de material gasto por Imóvel
Locatário recebe aviso de expulsão do Imóvel por acumulação de dívidas	Lista de Devedores	Aviso de expulsão	Notificação
Locatário solicita troca do Imóvel	Pedido de troca de Imóvel	Carta de parecer sobre o pedido	Relatório de situação do Imóvel
Factura precisa de ser remetida ao Locatário	Contrato	Emissão de factura	Factura
Locatário indaga sobre a situação de uma factura	Factura com irregularidades	Factura correcta	Nova factura
Locatário envia pagamento da factura	Factura	Pagamento da factura	Recibo
Locatário realiza o pagamento da dívida de rendas atrasadas á Tesouraria	Carta ultimato ao locatário para liquidação da dívida	Pagamento da factura em dívida	Recibo de factura
Locador informa ao serviço de intermediação do aluguer/Venda de Imóvel	Informação da casaá vender/alugar	Carta de recepção de imfomação	-----
O Locador aceita o aluguer do seu Imóvel	Contrato de propriedade mais condições de alugar	Contrato de serviços	Novo Imóvel na lista de Imóveis cadastrados
Direcção pede informação de Imóvel a aluguer/comprar á Gestão	Proposta de compra de Imóvel	Id_Locador + Id_casa carta solicitando referência do Imóvel	-----

Evento	Estímulo (Entrada)	Resposta	Saída(Output)
Tesouraria propõem a compra do Imóvel ao Locador	Termo de pagamento do Imóvel	Termo de pagamento do Imóvel assinado + Contrato do Imóvel	
Direcção aceita proposta do Locador de aluguer/venda de Imóvel	Contrato mais valor de aluguer/venda	Carta ao Locador de aceitação do valor	Lista de Imóveis à comprar
Direcção autoriza a tesouraria para pagamento do aluguer/compra do Imóvel	Carta de pedido de elaboração de contrato de compra/aluguer	Contrato de compra/aluguer com assinatura	Contrato de Imóvel
Tesouraria efectua o pagamento da compra de Imóvel	Factura do Imóvel	Pagamento de factura	Recibo
Serviço do Cliente pede o processamento de facturas	Contrato + data para alocação da factura	Factura processada	Factura
Direcção pretende saber a situação de pagamentos de rendas	Carta à Tesouraria solicitando listagem da situação das rendas	Valor total de rendas + lista de devedores	Relatório de rendas
Direcção exige listagem de devedores de cada mês	Carta pedindo lista de devedores	Lista de devedores	Listagem de devedores
Direcção solicita listagem de Imóveis pela Topologia	Carta à Gestão, solicitando a listagem de Imóveis	Listagem de Imóveis por topologia	Lista de Imóveis por Topologia
Direcção precisa (mensalmente) do relatório de receitas	Solicitação de receitas à tesouraria	Relatório de receitas	Relatório de receitas

A descrição de alguns eventos na lista da eventos encontra-se no anexo II

5.2 O MODELO COMPORTAMENTAL

O modelo comportamental é geralmente denominado por "Modelo do interior do sistema", porque ele ilustra a resposta interna do sistema aos estímulos externos, detectando-se as funções necessárias (processos), o conteúdo dos arquivos de dados necessários e as associações entre eles, o comportamento do sistema no tempo (estados e transições) e uma especificação muito geral de alguns processos para aumentar o grau real da análise.

Os habituais produtos deste modelo são :

- O Diagrama de Fluxos de Dados (nivelado)

- O Diagrama de Entidades e Relacionamentos
- O Diagrama de Estados e Transições
- A Especificação dos processos terminais
- O Dicionário de Dados actualizado.

Modelação das funções do sistema:

o diagrama de fluxo de dados (DFD)

Um antigo ditado da actividade de desenvolvimento de sistemas enfatiza que um sistema de processamento de dados envolve dados e processamento, e que não se pode construir um sistema com êxito sem a participação de ambos os componentes. O processamento de um sistema é, certamente, um aspecto importante para ser modelado e examinado com o usuário, pode-se vêr no "Theory, T. J. (1999). Database Modeling & Design, Third Edition, California, Morgan Kaufman Publishers".

A modelação de que se está a referenciar pode ser descrita de várias maneiras:

- a) Que funções deve o sistema executar? Quais são as interações entre as funções?
- b) Que transformações deve executar o sistema? Que entradas são transformadas e em que saídas?
- c) Que espécie de trabalho faz o sistema? Onde é que ele obtém a informação para para o seu consumo? Para onde é que ele remete os resultados do trabalho?

A ferramenta de modelação usada para descrever a transformação de entradas em saídas é o diagrama de fluxo de dados.

Pode-se observar o seguinte: Os diagramas de fluxo de dados consistem em processos, arquivos (depósito de dados) e fluxos de dados (ou de controlo).

Processos são mostrados como círculos no diagrama. Eles representam as diversas funções individuais que o sistema executa como resposta aos estímulos do ambiente externo. Funções (processos) transformam entradas em saídas.

Fluxos, são representados por setas direccionais. Elas são as conexões entre os processos (funções do sistema), e representam a informação que os processos exigem como entrada e/ou as informações que eles geram como saída.

Depósito de dados, são limitados por dois seguimentos de recta equidistantes. Eles mostram colecções (agregados) de dados que o sistema deve manter na memória por um período (tempo). Quando os projectistas e programadores terminam a construção do sistema, os depósitos existirão, tipicamente, como arquivos ou banco de dados.

Terminadores mostram as entidades externas com as quais o sistema se comunica. Os terminadores são, tipicamente, indivíduos, grupos de pessoas ou divisão da organização, sistemas externos de computador e organizações externas.

Embora o DFD, na página seguinte, ofereçam na prática uma visão geral dos principais componentes funcionais do sistema, não fornece qualquer detalhe sobre os mesmos.

Como construir DFD nivelados ? Construindo DFDs resposta.

A abordagem para a modelação dos DFDs resposta envolve as quatro etapas seguintes ;

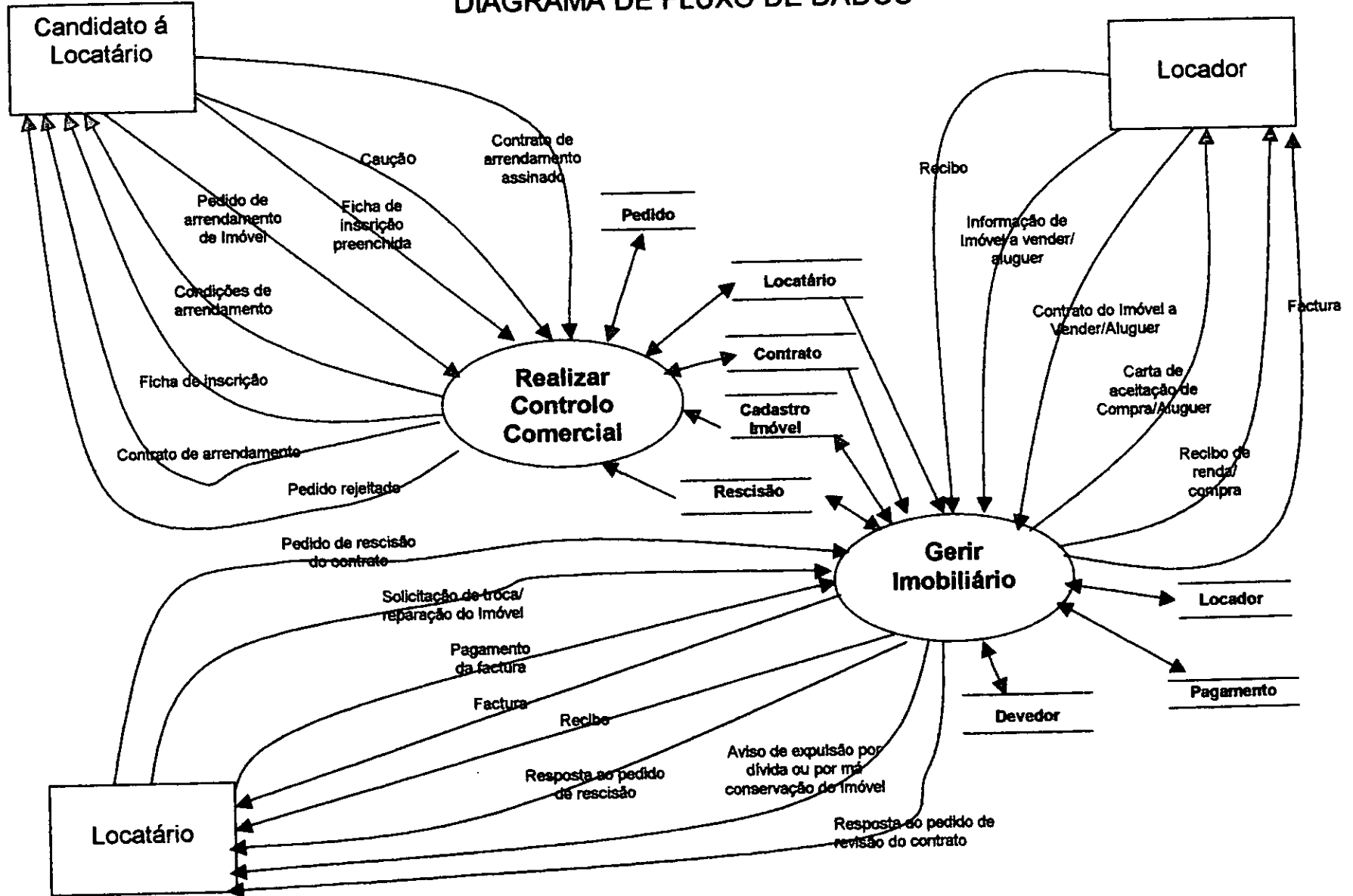
- a) Desenha-se um processo, para cada evento da lista de eventos
- b) O proceso recebe um estímulo de acordo com a resposta que o sistema deve dar ao evento associado
- c) Desenham-se entradas e saídas apropriadas de modo a que o processo seja capaz de emitir resposta necessária e desenham-se depósitos, como forma

mais adequada, para comunicação entre processos ou para arquivo de dados

d) O resultante DFD inicial é verificado em relação ao DC e à lista de eventos para que se confirme se está completo, e se é consistente

Os DFD nivelados podem ser vistos em anexo II e os expandidos em anexo III

DIAGRAMA DE FLUXO DE DADOS



Para mostrar os detalhes de qual informação é transformada e como é transformada, precisamos de duas ferramentas de suporte textual de modelagem: O Dicionário de dados e a especificação de processos

O Dicionário de Dados

O Dicionário de Dados (DD) consiste numa listagem de todos os elementos de dado importantes para o sistema, incluindo definições precisas e rigorosas para que o usuário e os analistas de sistemas possam ter informação sobre as entradas, saídas, os componentes de arquivos e cálculos intermediários.

Para o presente relatório o modelo de DD usado não é o típico de Yourdon. Optou-se, portanto, por fazer uma descrição de fácil consulta ao DD.

Para o sistema em estudo o DD pode ser visto no anexo IV.

A modelação dos dados a reter pelo sistema: o Diagrama de Entidades – Relacionamentos (DER)

Nota: Se no DFD temos arquivos de dados e reconhecimento de eventos, este transforma-se em entidades, associações e tabelas no DER.

Embora o DFD seja de facto uma ferramenta útil para a modelação de sistemas, ele somente enfatiza um aspecto principal de um sistema. As Suas Funções. A notação de depósito de dados no DFD nos apresenta a existência de um ou mais grupos de depósito de dados, mas, deliberadamente, diz muito pouco acerca de detalhes de dados.

Não é necessário saber-se, em detalhes, que informação está contida em cada depósito de dados. Esse aspecto do sistema não está realçado pelo diagrama de fluxo de dados, mas está activamente retratado por uma ferramenta de modelação, o Diagrama de entidade – relacionamento. Este diagrama possui dois importantes componentes:

- a) *Tipos de objectos*: são apresentados por um quadro rectangular no diagrama de entidades – relacionamentos. Um tipo de objecto representa uma colecção, um conjunto, ou objectos (coisas) do mundo real cujos membros desempenham um papel no sistema que está sendo desenvolvido. Eles podem ser identificados de forma única, podendo serem descritos por um ou mais factos (atributos).
- b) *Relacionamentos*: Não são representados, ou quando o são é na forma de losangos no diagrama. Um relacionamento representa um conjunto de conexões ou associações, entre os tipos de objectos interligados por setas ao relacionamento.
- c) *As participações*: As participações representadas no presente estudo não são próprias da notação de Yourdon. Elas foram usadas para fácil compreensão, onde > (Representa vários) e ● que mostra associações mutuamente exclusivas

Um DER do sistema novo é apresentado no anexo V

Observação

O desenvolvimento do modelo de dados inicial: Sempre que se pretende um sistema, começa-se a trabalhar na versão inicial do DER como actividade independente, em paralelo com o desenvolvimento do DFD inicial. O DER e o DFD são desenvolvidos em paralelo, eles podem ser usados para verificações cruzadas entre eles. Dessa forma, os depósitos que tenham sido definidos experimentalmente no DFD preliminar, podem ser utilizados para sugerirem objectos no DER preliminar e os objectos que tenham sido experimentalmente identificados no DER preliminar podem ser usados para auxiliar a escolha dos depósitos adequados no DFD preliminar, leia-se no " Penny A. Kendall, (1992)

Introduction to Systems Analysis and Design – A Structured Approach, second edition USA”. Nenhum dos modelos deve ser considerado como sendo o modelo dominante, todos estão no mesmo pé de igualdade e podem dar um inestimável auxílio aos outros. A lista de eventos é tão útil na elaboração do DER inicial como na do DFD inicial. Os substantivos na lista de eventos muitas vezes se mostrarão como objectos no DER. Por exemplo, se um evento for “Candidato á Locatário paga factura”, identificamos imediatamente Candidato á Locatário e Factura como objectos experimentais. Podemos usar do mesmo modo, a lista de eventos como meio de verificação cruzada do DER inicial. Todos os tipos de objectos do DER devem corresponder a substantivos na lista de eventos.

Especificação geral de Processos Terminais

Exploremos a especificação de processos, a descrição do que ocorre dentro de cada bolha (processo) terminal, num diagrama de fluxo de dados. Vários escritos incluindo os de DeMarco 1978, Gane e Sarson 1977 e Weinberg 1978, usam o termo “Miniature specification” como alternativa para especificação de processos. Qualquer que seja o seu nome, o propósito de uma especificação de processos é totalmente directo: Ela define o que deve ser feito para transformar entradas em saídas. Ela é uma detalhada descrição da orientação empresarial do usuário executada pelas bolhas.

São diversas as ferramentas que podem ser usadas para produzir uma especificação de processos: Tabelas de Decisão, Linguagem estruturada, condição pré/pós, fluxogramas, Diagramas de Nassi-Shneiderman e outras. Embora a maioria dos Analistas de Sistemas prefiram a linguagem estruturada, não podemos nos esquecer que podemos usar qualquer método, desde que ele satisfaça dois essenciais requisitos, a saber:

1º) A especificação de processos deve ser expressa de uma forma que possa ser verificada pelo usuário e pelo analista de sistemas: É precisamente por esta razão que evitamos a linguagem comum como ferramenta de especificação: Ela é notoriamente ambígua, principalmente para descrever acções (decisões) alternativas e repetitivas (laços). Por sua natureza, ela também tende a causar grande confusão ao expressar condições booleanas compostas (isto é, combinações dos operadores booleanos AND, OR e NOT).

2º) A especificação de processos deve ser expressa de uma forma que possa ser efectivamente comunicada às diversas audiências envolvidas: Embora geralmente seja o analista de sistemas quem redige a especificação de processos, existe uma diversificada audiência de usuários, gerentes, auditores, controladores de qualidade e outros, que irão ler a especificação. A especificação de processos talvez possa ser expressa em cálculo de predicados, em Pascal, em Access ou numa abordagem de diagrama formal. Porém, se a comunidade usuária recusar-se a ler tais especificações, elas serão inúteis. O mesmo pode ser dito das tabelas de decisão, linguagem estruturada ou sobre outras ferramentas de especificação, trata-se mais de uma função da personalidade, retrospecto e da atitude dos usuários com os quais lidamos.

Talvez seja mais importante afirmar que a maioria dos analistas de sistemas e das empresas usa uma ferramenta para elaborar todas as especificações. Isso, na minha opinião, é um grande erro: Deve-se ter a liberdade de se empregar uma combinação de ferramentas de especificação, dependendo de:

- a) Preferência do usuário
- b) Preferências do analista
- c) Natureza dos diversos projectos

A boa ferramenta de especificação de processos deve possuir, também, uma terceira característica, não deve impor (ou implicar) decisões arbitrárias do projecto e de implementação. Isso muitas vezes é difícil, porque o usuário, do qual dependemos para estabelecer a política seguida em cada bolha do DFD,

está disposto a descrever a orientação em termos de como ela é hoje executada. É tarefa do analista, de "destilar" dessa apresentação a essência do que seja a orientação e não como ela é actualmente executada.

Nota: Para explorar muito bem a especificação de processo, devemos enfatizar um ponto: As especificações de processos só são desenvolvidas para os processos do nível mais baixo dos diagramas de fluxos de dados em níveis. O propósito desta parte foi para mostrar que existem muitas maneiras diferentes de descrever a orientação detalhada do usuário no interior de cada bolha (processo) primitiva(o) de um diagrama de fluxo de dados. Embora a linguagem estruturada seja a técnica mais comumente utilizada nos dias de hoje, pode-se também considerar o uso de tabelas de decisão, fluxogramas, pseudocódigos e outras abordagens que podem ser verificadas e transmitidas facilmente aos seus usuários. Basta ter em conta lembrar-se que a especificação de processos representa o maior volume do trabalho minucioso na construção do modelo de um sistema, podem existir centenas e até milhares de especificações de processos, e cada um pode ocupar uma página (neste caso geralmente é mais usual aplicar a abordagem TOP DOWN).

A título exemplificativo apresentam-se algumas especificações em linguagem estruturada. As especificações dos módulos definitivos são apresentados na fase final de Desenho depois de pictorizado o Diagrama de Estrutura.

1- Gerir Propostas

Begin

Pedido=0

Do while <existência de pedidos>

If <Ficha de inscrição bem preenchida> then

Addnew pedido, "Pedido = Pedido + 1" assim se assume que há mais um pedido novo

Else

<Mensagem: Ficha de inscrição com insuficiência de dados, queira rectificá-la>

Endif

<Gravar todos pedidos inscritos>

Loop, que simboliza o fim do Do
Display pedido
end

2- Tratar Serviços Cliente

Begin
If <Contrato de arrendamento assinado e o pagamento da caução feito junto á tesouraria por intermédio dos serviços do cliente> then
Addnew Locatário
Else
<Mensagem. Este candidato não possui número de contrato>
Endif
<Gravar todos Locatários inscritos>
end

3- Executar Intermediação

Begin
If <Locador realiza proposta de venda/aluguer do seu imóvel + Título de propriedade> then
<Negociar o valor do imóvel ou comissões>
<Encaminhar a proposta à Gestão >
Else
<Rejeitar proposta do Locador>
endif
end

4- Realizar Gestão Imóvel

Begin
<Após recepção da proposta do Locador>
if <Proposta negociada com intermediação é útil> then
<Enviar a proposta à Direcção com parecer favorável>
Addnew Locador <Mais um novo Locador>
<Enviar dados do locador à tesouraria para pagamento da compra ou pagamento de rendas mensais>
<Informar a intermediação da aceitação da proposta feita>

```
Else  
<Informar a intermediação da rejeição da proposta>  
endif  
end
```

5- Gerir Tesouraria

```
<verificar se é Locatário ou Locador>  
If <Locatário> and <Data entre 1 à 10> then  
<Emitir Factura>  
  If <Factura paga> then  
    <Emitir Recibo>  
  Else  
    <Considerar devedor e emitir notificação de expulsão caso passem 30 dias sem regularizar  
    a situação>  
  Endif  
Else  
  If <Imóvel alugado à Locador> and <Data entre 1 à 10 > then  
    <Pagar o valor da comissão mensal>  
  else  
    If <Imóvel é para comprar ao Locador> then  
      <Entrgar o valor de compra>  
    Endif  
  Endif  
Endif  
Endif
```

Regras de Actividade

A modelação conceptual é um processo paulatino/progressivo da documentação de requisitos, quer sobre a estrutura de dados, quer sobre as regras de integridade dos mesmos.

As regras de actividade são especificações que preservam a integridade do modelo de dados, elas preservam a integridade do Modelo conceptual ou lógico de dados (Hoffer, George e Valacich ,1999).

Existem quatro tipos básicos de regras de actividade

- Integridade das entidades – Cada instância de uma entidade deve possuir um único identificador que não é null;
- Integridade referencial – é uma regra que diz respeito à relação existente entre tipos de entidade (entender-se-á melhor na fase de desenho);
- Domínios – Diz respeito a valores válidos para os atributos de cada entidade;
- Constrangimentos Operacionais – Outras regras de actividade que protegem a validade dos valores dos atributos.

Para o sistema em estudo temos definidas as seguintes regras básicas de actividade, para além das necessárias devido ao modelo de dados relacional escolhido :

1. Um Imóvel não deve possuir mais que um registo;

Se existir no arquivo dos Imóveis, um Imóvel que tenha os mesmos dados de identificação (Nome, Localização, ImóvelId, Topologia, etc.), com mais de uma inscrição, dever-se cancelar todas as restantes e ficar com apenas uma.
⇒ Integridade das entidades

2. O Proponente não deve ter mais que um cônjuge. Só regista-se cônjuge se houver cadastro do Proponente. Se existir, deve-se reparar no número total do agregado familiar, colocando como proporção máxima duas pessoas por quarto. ⇒ Integridade referencial

3. Um mau Locatário (o que tenha saído por expulsão ou durante a saída tenha provocado danos ao Imóvel) não deverá ser readmitido no futuro para algum

Imóvel da Imovisa, nem que seja por intermédio duma pessoa colectiva. ⇒
Constrangimentos Operacionais

4. Se um Locatário tiver vários Imóveis a arrendar, terá mensalmente, de ser facturado um de cada vez e não constar todos arrendamentos numa única factura. ⇒ Constrangimentos Operacionais

6. O MODELO DE IMPLEMENTAÇÃO DO UTILIZADOR

Aquí importa realçar que o projectista deve ter já preparado dois níveis de alocação de processos e armazenamento de dados. Em tarefa individual, o computador funciona na modalidade assíncrona, i.e, só pode ocorrer uma actividade de cada vez. O modelo mais utilizado para organizar a actividade com uma só unidade sincronizada é o diagrama estrutural, que mostra a organização hierárquica dos módulos de uma tarefa.

Existe uma óbvia pergunta neste ponto: Como é que o projectista do sistema transforma um modelo de processos em rede de um DFD no modelo sincronizado representado por um diagrama estrutural? Algumas obras sobre projectistas de sistemas incluindo (Pages Jones, 1998) e (Yourdon e Constatine, 1989), discutem detalhadamente essa questão. Existe uma estratégia para transformar o modelo de rede de fluxo de dados num modelo de diagrama estrutural sincronizado; na realidade, a estratégia é, conhecida como projecto centralizado na transformação, apenas uma das diversas estratégias para converter um modelo de rede de fluxo de dados num modelo hierárquico sincronizado. Pages Jones - 1998, Yourdon e Constatine -1989 e Ward e Mellor - 1985 discutem várias dessas estratégias. Observe-se que cada processo no DFD torna-se um módulo no diagrama estrutural derivado. Essa situação é realista se os processos forem relativamente pequenos e simples. Além do módulo que implementa os processos dos fluxos de dados, é evidente que o

diagrama estrutural também contém módulos para coordenar e gerenciar a actividade geral, bem como módulos dedicados a trazer entradas para o sistema e levar saídas para fora do sistema.

Outras estratégias de projecto utilizam o diagrama de entidade – relacionamentos ou outras formas de diagramas de estrutura de dados como ponto de partida para a derivação ao adequado diagrama estrutural (Jackson, 1975) e (Orr, 1977)

O modelo de implementação do utilizador abrange os quatro aspectos seguintes:

- A alocação do modelo essencial a pessoas versus máquinas.
- Detalhes de interação homem/máquina.
- Suplementares actividades manuais que podem vir a ser necessárias.
- Restrições operacionais que o usuário deseja impor ao sistema.

Fronteira de Automatização escolhida pelo utilizador

O presente trabalho só irá centrar-se nos serviços do Departamento Comercial (Sector de Controlo de Propostas e Sector de Serviço do Cliente) e Departamento Imobiliário (Sector de Gestão, Sector de Tesouraria e Sector de Intermediação). Com isso, serão estudadas as melhores formas de control de propostas dos Clientes e dos Locadores, tratamento de todos serviços do cliente (controle em relação á caução e facturação mensal), cadastragem dos Imóveis e/ou alocá-los ao cliente, cobrança de rendas e a intermediação de todos outros serviços.

A automação dos escritórios da Imovisa para este trabalho só irá incorporar o que se referencia no parágrafo anterior, mantendo as anteriores tecnologias existentes casos de E-mail, Faxes e Telefones.

Também incorpora a componente de Auditoria de Sistemas em particular de gestão imobiliária

Desenho dos interfaces feitos pelo utilizador

Esta é a actividade que mais consome tempo, na qual o utilizador demonstra mais interesse. Entretanto, constitui a especificação da interface humana . veja-se (Yourdon, 1990).

O utilizador apresentou a forma na qual desejaria ter os seus formulários de entrada de dados no sistema. Foi realizado o ' blueprint' dos formulários seguintes :

- 1- Formulário para registo de dados do proponente individual
- 2- Formulário para registo de dados do cônjuge
- 3- Formulário para registo de dados do Agregado Familiar
- 4- Formulário para registo de dados do Assunto a tratar no Imóvel
- 5- Formulário para registo de dados do Proponente Colectivo
- 6- Formulário para registo de dados do representante do Proponente Colectivo
- 7- Formulário para cadastrar Imóvel, cuja topologia é Flat
- 8- Formulário para cadastrar Imóvel, cuja topologia é vivenda
- 9- Formulário para atribuição de Imóvel
- 10-Formulário para assinatura de contrato como novo Locatário
- 11-Formulário para facturar o Locatário
- 12 - Formulário para atribuição de privilégios

As versões definitivas dos interfaces são apresentadas como resultado da fase de Desenho.

Desenho de relatórios feitos pelo utilizador

Esta é a fase em que o utilizador especifica o formato de relatórios e o tipo por ele desejado. No caso em estudo, foram identificados alguns tipos e formatos de relatórios básicos exigidos pelo utilizador:

1- Recibo processado pelo computador

Imovisa – Imobiliária de Moçambique Recibo Nº 1	
Nº de Contrato:	:
Localização :	:
Tipo de Imóvel:	:
Topologia :	:
Inquilino :	:
Valor Pago :	:
Último Mês :	:
Ano :	:
Recibo processado por computador	Maputo, aos de de

2- Lista de control de uso de objectos no sistema

Logon List para control de uso de objectos								
Tempo de entrada Sistema	Tempo de saída sistema	Nome usuário	Pedidos	Imóveis	Pagamentos	Relatórios	Administração	
12/04/2001 12:35:14	12/04/2001 12:38:04	Marcos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
A data de impressão é: / /								Página de

Comentários das exigências do utilizador

Estas exigências do usuário são todas justas e que são cumpridas na fase do modelo físico novo. Contudo requereram uma atenção especial do analista de sistemas, porque existem restrições de integridade a considerar e que serão vistas na fase do desenho lógico da base de dados.

O modelo de implementação do sistema foi feito usando o Sistema de Gestão de Base de Dados Relacional Access 2000. Esta opção justifica-se pelo facto de permitir ter SQL Server como hospedeiro para diversas consultas no futuro.

7. DESENHO DO SISTEMA

7.1 DESENHO LÓGICO DA BASE DE DADOS

Neste ponto, será feita a descrição do desenho lógico com especial ênfase ao modelo relacional.

O desenho lógico é o processo de transformação do modelo conceptual de dados no modelo lógico que pode ser implementado em qualquer sistema de gestão de base de dados (McFadden, Hoffer e Prescott, 1999).

Esquema de Relações

O esquema de relações descreve todas as entidades que foram identificadas incluindo as suas chaves primárias e estrangeiras. Para o presente estudo o esquema pode ser visto no Anexo VI.

Esquema de Tabelas Normalizadas

Um esquema de tabelas é a indicação real da relação lógica do modelo físico após a implementação. Quais dados se relacionam? Esta é a questão essencial que o esquema de tabelas responde do ponto de vista conceptual.

Fala-se de esquema de tabelas porque o padrão base para esta explicativa é o uso de tabelas como entidades que se relacionam, baseando-se no pressuposto de se referir ao Modelo relacional.

O esquema de tabelas do presente trabalho é apresentado no anexo VII

Restrições de Integridade

Na fase de construção do modelo conceptual, independentemente da metodologia adoptada, urge detectar e registar algumas regras

O modelo relacional inclui vários tipos de integridade cujo propósito é facilitar a precisão e a integridade dos dados na base de dados. Os maiores tipos de restrições de integridade são:

- Restrições de domínios;
- Integridade das entidades;
- Integridade referencial;
- Constrangimentos Operacionais
- Triggers (triggering operations)

Restrição de domínio - Conjunto de valores que devem ser associados ao mesmo atributo.

Na programação preveu-se, por exemplo, para este caso, que todas datas teriam o formato dd/mm/yyyy, e não permitindo que haja um dia superior a 31,

um mês superior a 12 e um ano superior a 9999. São casos de, data do Início do Contrato, data fim Contrato, etc.

Integridade das entidades - Esta é uma regra desenhada para garantir todas as relações que tenham chave primária e que os valores para estas chaves são válidas. Isto garante que todas as chaves primárias não sejam nulas.

Para o trabalho, muitas das chaves tomadas são do tipo Numeração automática (autonumber), o que pressupõe não existir para mesmo objecto, duas ocorrências iguais. É o caso do AF_Id (a chave da entidade Agregado Familiar)

Integridade referencial - É uma regra que mantém a consistência entre linhas de duas relações. Refere em muito dos casos as chaves estrangeiras. Para o trabalho pode-se vêr, por exemplo através do esquema relacional em anexo VI, que Loc_Id na entidade Imóvel é chave estrangeira em relação ao Locid da entidade Localização. Se o Locid na Entidade Localização ocorrer, então o Loc_id da Imóvel ocorrerá com o mesmo valor ou então é null.

Constrangimentos Operacionais - Esta regra está associada às regras de actividade.

Triggers (triggering operations) - Outras regras de actividades que protegem a validade dos valores dos atributos. São asserções ou regras que governam a validade das operações de manipulação de dados. O exemplo mais nítido é do caso da Data_Fim do contrato não ter de aceitar ser inferior que a Data_Inicio. Para isso foi desenhada uma linha do código em access com seguinte declaração:

```
If Contrato![Data_fim]< Contrato![Data_Inicio] then  
Contrato![Data_fim]=Null  
Contrato![Data_fim].Setfocus
```

Isto significa que para o caso da data fim ser inferior à data de início, essa data será apagada e o cursor mantido sem deslocar para outro local.

Para o presente trabalho, as regras acima foram aplicadas na fase de programação do sistema para garantir que a visão do utilizador fosse tomada em consideração.

7.2 DESENHO FÍSICO DA BASE DE DADOS

Esta é a fase em que o desenhador da base de dados toma a decisão sobre a organização física da própria base de dados no computador e esta estrutura (tipo de objectos e tipo de dados para os atributos) é definida no sistema de gestão da base de dados.

O desenho físico é feito numa coordenação fechada com todos outros aspectos físicos do sistema de informação: Programas, Hardware do computador, sistema operativo e rede de comunicação de dados.

VISÕES (VIEWS)

Serve para resolver a necessidade de diferentes usuários em verem a mesma tabela ou tabelas de forma distinta (Para o Access substitui views do SQL Server como Queries normais por uma questão de compatibilidade). Uma View ou uma Query funciona como uma janela, dando uma determinada perspectiva da Base de Dados. Em termos práticos, uma View é uma forma diferente ter acesso a uma tabela ou a um conjunto de tabelas. Na realidade, uma View consiste num comando SELECT realizado sobre tabelas ou outras Views. No entanto, a View não existe fisicamente, é apenas uma forma de

visualizar os dados que existem fisicamente, é uma forma de visualizar os dados que existem noutras tabelas ou Views. O uso de Views pode ocorrer de várias razões: Por questões de segurança, evitando que alguns campos ou linhas estejam acessíveis a alguns tipos de utilizadores; Por questões de confidencialidade, evitando que os utilizadores possam consultar dados de acesso reservado; Por questões de simplicidade; Reforçam a independência dos dados das aplicações a que se destinam; Um Select simples realizado a uma View pode ser equivalente a um comando select bastante complexo; Podemos condensar numa única View o acesso a várias tabelas e que o acesso a uma View é exactamente igual ao acesso a uma tabela. Depois de criada uma View, os dados que ela apresenta não lhe pertencem, mas às tabelas onde foram introduzidos. Por isso, qualquer alteração nos dados de uma tabela reflecte-se nos dados das Views a ela associadas, leia-se no "Tecnologias de Informação, SQL, Luís Damas, Editora de Informática, 2ª edição, 2000".

A declaração duma view é seguinte:

```
Create View <Nome da view> as  
<select normal como as do anexo IX>
```

Exemplo:

```
Create View Flats_dum_Andar as  
  Select Flat_Id, Num_Flat, Num_Andar, Localização  
  From Flat F, Imovel I, Localização L, Assunto A  
  Where Localização = (Select I.I_localização  
                      From I, A  
                      Where I.I_Id = A.Loc_Id or I.Edificio_Id = A.Edi_Id)  
  Group by Localização
```

Indexação

A indexação é um método de ordenação de uma tabela. Ela é feita através de campos não nulos da tabela (chave primária) e facilita a junção entre tabelas.

Para o desenho físico da base de dados em access usou-se a indexação oferecida pelo próprio Access como ferramenta e outra feita pelo desenhador do sistema e que é apresentada mais adiante.

Reparemos numa explicação mais concreta. Pensemos, por momentos, no serviço de Gestão Imobiliária da Imovisa sobre o Id do cliente que permite obter o registo por nome ou morada, nome e moradas pelo Id, etc. Já agora, pensemos na aplicação que acompanha este trabalho e o que faz esta gestão.

Este serviço tem uma lista (uma tabela), de todos inquilinos que deverão no mínimo ser perto de 600 registos.

Será que essa lista está ordenada?

Talvez sim, talvez não. Vejamos então. Em princípio, a lista terá que estar ordenada, pois assim é possível evitar a pesquisa de um registo sem ter que percorrer toda ela. Por outro lado, pode-se perguntar: Se a lista está ordenada, por que campos está ordenada? Pelo nome do Locatário? Pela morada?

Se a lista estiver ordenada, só poderá estar segundo um critério. Se estiver ordenada pelo nome, não pode estar ordenada por moradas, e vice-versa. Portanto, temos aqui um problema, pois se a lista estivesse ordenada, cada novo registo introduzido obrigava a uma enorme remodelação dela. Então qual é a solução?

Quando adicionados os registos ao ficheiro ou à Base de Dados, são sempre colocados no fim do mesmo. Evita-se, desta forma, pesadas reorganizações sistemáticas do ficheiro sempre que um novo registo é introduzido. Imaginemos só o processamento necessário para introduzir um registo com nome 'Helena'

no sistema quando tiver perto de 10 milhões de registos. Sendo os registos colocados sempre no final do ficheiro ou tabela, outra forma qualquer terá que existir para simular uma ordenação sobre os dados.

A Indexação é, pois, a estrutura que permite simular a ordenação dos dados. Na realidade, consiste normalmente numa árvore de termos que revertem para uma determinada posição no ficheiro ou tabela, veja em "A Visual introduction to SQL, J. Harvey Trimble and Jr. David Chappell, Great Falls, Virginia, February 1989"

Ao indexar a Base de Dados durante a implementação, permitimos que não se possa percorrer a lista de dados ou de tabelas de dados, excepto a metade da região que o dado se encontra.

Pode-se observar que os índices existem fisicamente no disco, o que é contrário às visões que são consideradas tabelas virtuais

Outra forma de Indexação para além daquela que ocorre normalmente por defeito no Access, o desenhador da implementação física que aqui está a ser retratada, realizou um código que a seguir se descreve

```
Private Sub Reindexar_Click()
Dim dbs As Database
Dim rst, rst1 As Recordset
    Dim strCriteria As String
Set dbs = CurrentDb
Set rst = dbs.OpenRecordset("Pedido Individual", dbOpenSnapshot)
rst.MoveFirst
'MsgBox (rst![P_Ficha_inc])
Do While Not rst.EOF
    If IsNull(rst![P_Ficha_inc]) And IsNull(rst![P_Nome]) Then
        rst.Delete
    Else
        rst.MoveNext
    End If
Loop
```

```

Set rst1 = dbs.OpenRecordset("Pedido_Colectivo", dbOpenDynaset)
rst1.MoveFirst
Do While Not rst1.EOF
    If IsNull(rst1![Pc_Ficha_Insc]) And IsNull(rst1![PC_Denominacao]) Then
        rst1.Delete
    Else
        rst1.MoveNext
    End If
Loop
Beep
MsgBox ("    Reindexação completa    ")
End Sub

```

7.3 A VISÃO ESTRUTURAL DO SISTEMA : O DIAGRAMA DE ESTRUTURA

Uma ferramenta de modelagem gráfica usada para representar a hierarquia do software é o diagrama de estrutura. No diagrama, um rectângulo representa um módulo. As setas que interligam os quadros representam a chamada de módulos. O Diagrama também apresenta parâmetros de entrada passados para cada módulo chamado, e os parâmetros de saída retornados pelo módulo, quando ele termina a sua tarefa e restitui o controle a quem o chamou.

No relacionamento entre módulos pode-se concluir o seguinte: o DFD mostra as funções, o DER realça os relacionamentos de dados e o Diagrama de eventos enfatiza a captura de requisitos. Esses aspectos isoladamente num sistema típico como o da Imovisa, mostram que existe muita complexidade, por outro lado, essas três visões do sistema devem ser consistentes e compatíveis entre si. No entanto, pode-se ver que cada depósito no DFD deve corresponder a um objecto ou a um relacionamento do DER.

O Diagrama estrutural típico para além de mostrar a hierarquia funcional, mostra também as interfaces de dados entre componentes, (vide anexo X).

O Diagrama Estrutural não tem uso na área de análise de sistemas. Porquê? Porque ele é utilizado como uma ferramenta de projecto para modelar a hierarquia sincronizada de módulos de um sistema, isto quer dizer, um módulo é executado a cada momento, o que é acurada representação da maneira como as coisas funcionam na maioria dos processamentos actuais.

O Analista de sistemas, por seu lado, necessita de uma ferramenta de modelagem que lhe permita mostrar uma hierarquia de redes assíncronas de processos. Isto é efectivamente realizado com o conjunto de diagramas de fluxo de dados em vários níveis, vide "Theory, T. J. (1999). Database Modeling & Design, Third Edition, 365 pp. California, Morgan Kaufman Publishers" e "Lucas, J.H.C. (1992) The Analysis, Design and Implementation of Information Systems, Fourth Edition McGraw Hill – International Editions".

De modo diverso de muitos dos diagramas anteriores, o quadro rectangular num diagrama estrutural não representa uma única declaração computacional ou um grupo contíguo de declarações; em vez disso, ele representa um módulo (exemplos comuns de módulos de procedimentos em Access). As setas que interligam os módulos não representam comandos goto mas sim, chamadas de procedimentos, a notação implica em que o procedimento sairá ou retornará para o ponto de onde foi chamado, ao terminar de executar sua função.

Existe uma óbvia pergunta neste ponto: Como é que o projectista de sistemas transforma um modelo de processos em rede de um diagrama de fluxo de dados no modelo sincronizado representado por um diagrama estrutural?

Alguns livros sobre projecto de sistemas, incluindo (Pages-Jones, 1988) e (Yourdon e Constantine, 1989), discutem detalhadamente essa questão. Existe uma estratégia para transformar o modelo de rede de fluxo de dados num modelo de diagrama estrutural sincronizado. Na realidade, a estratégia é, em

geral, conhecida como projecto centralizado na transformação. Este projecto é apenas uma das diversas estratégias para converter um modelo de rede de fluxo de dados em um modelo hierárquico sincronizado, existem várias dessas estratégias. Observe que cada bolha de processo no diagrama de fluxo de dados, torna-se um módulo no diagrama estrutural derivado. Essa situação é realista se os processos forem relativamente pequenos e simples (Ex: Se a especificação do processo for menor que uma página de linguagem estruturada). Além do módulo que implementa os processos de fluxo de dados, é evidente que o diagrama estrutural também contém módulos para coordenar e gerenciar a actividade geral, bem como módulos dedicados a trazer entradas para o sistema e levar saídas para fora do sistema.

Outras estratégias de projecto utilizam o diagrama de entidade relacionamentos ou outras formas de diagramas de estruturas de dados como ponto de partida da derivação do adequado diagrama estrutural

7.4 ESPECIFICAÇÃO GERAL DOS MÓDULOS

São diversas as ferramentas que podem ser usadas para produzir uma especificação de módulos: Linguagem estruturada, condição pré/pós, fluxogramas, ...

1º) A especificação de módulos deve ser expressa de uma forma que possa ser verificada pelo Programador: É precisamente por esta razão que evitamos a linguagem comum como ferramenta de especificação de módulos:

Atente-se para o facto de que, os módulos produzidos na especificação devem ser todos implementados e vice-versa. A especificação dos módulos é usada como a mais alta ferramenta para facilitar o programador durante a fase de implementação do sistema .

Neste trabalho, a especificação de módulos será baseada em pseudocódigos, próximos da linguagem orientada a objectos, caso concreto do Microsoft Access 2000.

1- Tratar Pedido: Vejamos o que ocorre nos processos de baixo nível deste processo

Dim Pedido as Database

Processo Receber Pedido

Begin

If Pedido preenchido then

Do While <Antes da tabela de pedidos ser percorrida até ao fim>

If <Um pedido com o mesmo número de BI existe> **Then**

<Questionar o Candidato a Inquilino>

Else

Pedido.AddNew

<Atribuir número a ficha de inscrição que será número do contrato>

Processo Analisar Pedido

If <Pedido é Individual> **Then**

<Indicar o Proponente. Cadastrar o candidato>

<Descrever o assunto que lhe leva a precisar do aluguer>

<Registar Cônjuge, neste caso pode ser esposa ou marido>

<Registar o Agregado Familiar>

<Submeter o pedido à Direcção>

Else

Então o pedido é colectivo

< Registrar o Proponente. Cadastrar o candidato >

< Registrar o Representante>

<Submeter o pedido à Direcção>

Processo Gerir Pedido

If <Direcção Aceita este Pedido> **Then**

<Cobrar valor da caução e da primeira renda>

<Entregar ao Locatário todos condimentos para ocupar o Imóvel Pretendido>

Pedido.Update

Else

Goto Line1

End if

End if

Loop

Line1:

End if

2- Gerir Imóvel: Vejamos o que ocorre nos processos de baixo nível deste processo

2.1 Cadastrar Imóvel (Analisar contrato do Locador)

```
If <Novo Imóvel cedido ou comprado> Then
  'Cadastrar esse imóvel
  <Verificar se esse Imóvel não é da empresa>
  <registar todos dados do título de propriedade e a topologia>
  If <Topologia> = Edifício then
    <Registar os dados da Flat>
  Else
    'Então é uma Vivenda
    <Registar se tem Garagem e se também tem piscina>
  End if
```

Realizar Compra

```
If <Imóvel tem Id Contrato de Título de Propriedade> and <Dados do Locador conferem com as
do título de propriedade> Then
  <Realizar a compra do Imóvel> or <Alugar o Imóvel>
end if
```

Atribuir Imóvel

```
Begin
  <Indicar o número da Ficha de inscrição>
  If <Número da Ficha de inscrição correcto> Then
    <Atribuir o Imóvel registando todos seus dados do cadastro>
    <Assinatura do Contrato>
  Else
    MsgBox(Introduzir o número de ficha de inscrição correcto)
  End if
```

3. Efectuar Pagamento: Vejamos o que ocorre nos processos de baixo nível deste processo

'Processo Emitir Factura

<Indicar o número da Ficha de inscrição>

<Indicar o Tipo de Pedido>

If <Número da Ficha de inscrição e Número do Contrato correctos> Then

<Indicar a moeda de pagamento se é USD ou Metical>

<Escrever o valor a pagar, não se aceitam pagamentos parciais>

<Aceitar o Pagamento>

<Emitir recibo>

Else

Msgbox(Esta pessoa não tem contrato assinado)

End if

4. Produzir Relatórios

Exemplo de pseudocódigo dum relatório

If <Data do mês for 11> Then

<Seleccionar todos devedores>

<Imprimir a Lista dos Devedores>

end if

7.5 ESQUEMA DE INTERFACES DEFINITIVO

Com base no Modelo de Implementação do Utilizador e nas opções feitas na fase de Desenho construiriam-se os interfaces definitivos .

1- Formulário para registo de dados do proponente individual

Imovisa
Emprego Individual

Picão de identificação	<input type="checkbox"/>	Tempo de Permanência	<input type="text"/>
Nome	Marcos Santos Maria	Idade	<input type="text"/>
Data Nascimento	<input type="text"/>	Sexo	<input type="text"/>
Estado Civil	<input type="text"/>	Emprego em	<input type="text"/>
Profissão	<input type="text"/>	Emprego Identificado	<input type="text"/>
Endereço Postal	<input type="text"/>	Residência Atual	<input type="text"/>
Telefone	<input type="text"/>	Telefone do emprego	<input type="text"/>
Nome e endereço	<input type="text"/>	Telefone Residência	<input type="text"/>

2- Formulário para registo de dados do cônjuge

Imovisa
Emprego Indivíduo

Nome	<input type="text"/>	Tempo de Permanência	<input type="text"/>
Data Nascimento	<input type="text"/>	Idade	<input type="text"/>
Estado Civil	<input type="text"/>	Sexo	<input type="text"/>
Profissão	<input type="text"/>	Emprego em	<input type="text"/>
Endereço Postal	<input type="text"/>	Emprego Identificado	<input type="text"/>
Nome e endereço	<input type="text"/>	Residência Atual	<input type="text"/>
Nome e endereço	<input type="text"/>	Telefone do emprego	<input type="text"/>
Nome e endereço	<input type="text"/>	Telefone Residência	<input type="text"/>

3- Formulário para registo de dados do Agregado Familiar

Imoviso-Imobiliária de Moçambique

Agregado Familiar

Nome	Idade	Grupos de Parentesco	Profissão
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

4- Formulário para registo de dados do Assunto a tratar no Imóvel

Imoviso-Imobiliária de Moçambique

Critérios Alugar	Tipo	Área	Artificialidade e Exatidão	Localização	Edifício pretendido	Taxa
<input type="text"/>	3	<input type="text"/>	<input type="text"/>	Benfica	Torres Vermelhas	tipo 2
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

5- Formulário para registo de dados do Proponente Colectivo

Pedido Colectivo

Proponente Proponente Colectivo

Nome de inscrição AF	<input type="text"/>	Sob o nº	12
Denominação do fim	CFM	Beiro Fiscal	2 Beiro Fiscal
Tipo de actividade	Vivenda	Inscrição nº	Beira
Sede Localização	Maputo	Actividade Principal	Comerciante
Data de constituição	<input type="text"/>	Taxação	435502
Região Comercial	Beira		

6- Formulário para registo de dados do representante do Proponente Colectivo

Representante
Representante

Nome	<input type="text"/>	Endereço	<input type="text"/>
Função	<input type="text"/>	Acto de Identificação	<input type="text"/>
Data de Inscrição	<input type="text"/>	Código Atribuído	<input type="text"/>
Estado Civil	<input type="text"/>	Documento Nacional de Identificação	<input type="text"/>
Nacionalidade	<input type="text"/>	Sexo	<input type="text"/>
Número de Identificação	<input type="text"/>	Residência	<input type="text"/>
B.I.M.	<input type="text"/>	Telefone	<input type="text"/>
Código I.T.R.	<input type="text"/>		

7- Formulário para cadastrar Imóvel, cuja topologia é Flat

Cadastrar Imóvel

Nome Imóvel	<input type="text"/>
Número do Imóvel	<input type="text"/>
Nome do Proprietário	<input type="text"/>
Localização	<input type="text"/>
Número	<input type="text"/>
Topologia	Edifício
Área	<input type="text"/>

Flat	Flat Nº	Área Nº	Tipo de Casa	Área	Parque
>					

8- Formulário para cadastrar Imóvel, cuja topologia é vivenda

Cadastro Imóvel

ID do Imóvel:
 Nome do Imóvel:
 Nome do Proprietário:
 Localização:
 Número:
 Topologia:
 Área:

Garagem:
 Com Garagem
 Sem Garagem

Piscina:
 Com Piscina
 Sem Piscina

9- Formulário para atribuição de Imóvel

Atribuição de Casa

Id	Nome	Endereço	Topologia	Área	Valor	Quantidade
1	Tomas Vermelhas	Berfica	3	tipo 2	2	1
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

10- Formulário para assinatura de contrato como novo Locatário

Contrato

Número Contrato:
 Data de Início do Contrato: 11-Dec-00
 Data do Fim do Contrato: 11-Nov-01
 Valor da Caução: \$300.00
 Renda Mensal: \$300.00

11- Formulário para facturar o Locatário

Pagamentos

2001
Abri

12- Formulário para atribuição de privilégios

Manuel

<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>

1

8. CRIAÇÃO E IMPLEMENTAÇÃO DA BASE DE DADOS

Neste capítulo, é apresentado o código usado na criação das tabelas do sistema em ACCESS 2000, bem como a estrutura das tabelas. A declaração de tabelas pode ser vista no anexo X.

8.1 REDE A IMPLEMENTAR NA IMOVISA

Do processamento manual que a Imovisa está ainda condenada a ter, antes da automação dos seus escritórios, pensamos que o contrário só ocorrerá a partir do instante em que uma rede for instalada.

Neste caso, irá colocar-se em vários pontos, como são os casos da Sede na Av. 25 de Setembro, da Sucursal, sita num dos edifícios das Torres Vermelhas, terminais com acesso remoto. Para isso, é realizado um acordo de nível de serviço entre a Administração da Rede e do Sistema com vários usuários, para que a aderência tenha fundamento dentro da instituição.

Os Administradores da Rede de comunicação e do Sistema irão gerenciar o sistema numa arquitectura Cliente – Servidor (arranjo utilizado em *redes* locais que trata tanto o servidor como as *workstations* individuais como dispositivos programáveis inteligentes, explorando toda a *capacidade* computacional de cada um), sem descurar de colocar o Host na sede a ser escolhida pela Gerência da Instituição. Os Administradores da Rede e do Sistema garantirão a integridade de dados através de atribuição de níveis de acesso a cada usuário, o que conferirá maior Segurança aos dados, assim como a Segurança da rede, leia-se no "PC Magazine "2001"(Intranets Globais e a informática no novo milénio), edição de Agosto de 1998". O acesso ao Mainframe será feito através de Logon atribuídos aos usuários.

Todas as salvaguardas tecnológicas e administrativas estabelecidas e aplicadas para protecção de *hardware*, *software* e *dados* contra alterações acidentais, propositais ou destruição, serão tomadas em conta.

Pressupõem-se o uso de terminais com recursos próprios de memória, processador e *software*, capazes de executar certas funções independentes do *host*.

8.2 AUDITORIA DO SISTEMA

Para se garantir Auditorias, deverá ter-se em conta, que para além das especificações e documentação do sistema, a Imovisa criará mecanismos legais de uso do sistema, como forma de garantir a repreensão dos usuários desonestos e intrusos no sistema.

Qualquer manutenção das especificações irá constar num formulário conhecido como solicitação de alteração de sistema, o que facilita a Auditoria em virtude de se poder garantir que há sempre uma informação de qualquer alteração feita.

Os obstáculos mais comuns à realização de auditorias de sistemas de informação são:

- a) Medo de auditar por ser uma área relativamente nova;
- b) Em função das mudanças tecnológicas constantes, o nível técnico do auditor quase sempre é insuficiente;
- c) Ausência de normas, metodologia e procedimentos padronizados.

Um dos objectivos deste Procedimento de Auditoria é tentar minimizar algumas destas barreiras, provendo ao auditor objectivos de controle, procedimentos e falhas mais comuns em auditoria de sistemas.

Pretende-se ainda, descrever as principais áreas de controle nas auditorias em sistemas de processamento de dados, e fornecer orientações básicas ao

auditor. Para cada área serão apresentados os objectivos de controle mais usuais e as falhas correspondentes mais comuns. Para verificar a ocorrência destas, o auditor poderá utilizar entrevistas, visitas, análise de documentos, programas, dados, uso do sistema, extração de dados de teste, entre outros.

Vejamos os ambientes de Auditoria

- Ambiente de Informática

Este tipo de auditoria abrange as áreas de segurança física e lógica, planeamento de contingências e operação do centro de processamento de dados.

- Organização

Este tipo de auditoria engloba aspectos gerais da organização, tais como : políticas, padrões e procedimentos da organização, responsabilidades organizacionais do departamento de gestão, gerência de pessoal e planeamento de capacidade.

- Aplicativos

Ao auditar um aplicativo, deve-se levar em consideração aspectos técnicos de informática (entrada, processamento e saída de dados) e aspectos legais relacionados à área a qual o sistema aplicativo atende (orçamento, contabilidade, pessoal etc...).

- Banco de Dados

Este tipo de auditoria refere-se a todas as actividades envolvidas na administração de dados (controles, disponibilidade, integridade, recuperação etc...).

- **Redes de Comunicação e Microcomputadores**

Esta é a área de informática que mais se desenvolveu nos últimos anos. Para realizar um bom trabalho, deve-se identificar, inicialmente, as tecnologias utilizadas pela unidade a ser auditada e capacitar-se antes de iniciar a auditoria. Neste tipo de auditoria serão verificadas as actividades de gerência, operação e segurança da rede de comunicações e de microcomputadores. Também serão analisados os controles sobre softwares de rede e de micro.

- **Desenvolvimento de Sistemas**

Este tipo de auditoria abrange metodologias de desenvolvimento de sistemas, projecto, estudo de viabilidade, design, implementação, operação e manutenção de sistemas de informática.

8.3 PROCEDIMENTOS DE AUDITORIA DO SISTEMA IMOBILIÁRIO DA IMOVISA

Auditar que significa garantir o cumprimento de normas, procedimentos, ordens de serviço, circulares e comunicações internas é sempre feita para verificar:

1- CONTROLES GERAIS

- Políticas, Padrões e Procedimentos
- Responsabilidades Organizacionais do Sector de Control de Propostas
- Planeamento no departamento Imobiliário

2- SEGURANÇA FÍSICA E LÓGICA

- Responsabilidade por Segurança Física e Lógica
- Acesso ao Computador Central
- Acompanhamento ao Visitante

- Administração de Password
- Relatórios de Violação de Segurança
- Restrições ao Acesso Lógico
- Segurança de Acesso On-line aos Dados
- Protecção contra Incêndios e Dispositivos de Segurança
- Treinamento em Procedimentos de Segurança e Manuseamento de Informação

3- APLICATIVO - ENTRADA DE DADOS

- Documentos ou Telas de Entrada de Dados (Formulários)
- Rotinas de Preparação dos Dados
- Autorização para Entrada de Dados
- Retenção de Documentos de Entrada
- Entrada de Dados On-line
- Validação dos Dados de Entrada
- Tratamento de Erros

4- APLICATIVO - PROCESSAMENTO E SAÍDA DE DADOS

- Integridade do Processamento
- Validação do Processamento
- Tratamento de Erros do Processamento
- Revisão dos Dados de Saída
- Distribuição dos Dados de Saída
- Tratamento de Erros dos Dados de Saída
- Segurança dos Dados de Saída

5- BANCO DE DADOS

- Administração de Dados
- Controles de Mudanças e Descrição de Dados
- Controles de Concorrência e Acesso a Dados
- Disponibilidade e Recuperação do Banco de Dados
- Integridade do Banco de Dados

6- OPERAÇÃO DA DIRECÇÃO OPERACIONAL

- 1- Distribuição da Carga de Trabalho Computacional**
- 2- Distribuição de Pessoal**
- 3- Manutenção Preventiva de Hardware**
- 4- Gerência de Problemas**
- 5- Gerência de Mudanças**
- 6- Controle e Identificação Externa dos Meios Magnéticos**
- 7- Procedimentos de Operação**

7- REDES DE COMUNICAÇÃO E MICROCOMPUTADORES

- 1- Gerência de Rede**
- 2- Dados e Segurança da Rede**
- 3- Operação da Rede**
- 4- Software de Rede**
- 5- Gerência de Microinformática**
- 6- Software de Micro**
- 7- Segurança de Microinformática**
- 8- Controle sobre o Processamento**

8- DESENVOLVIMENTO DE SISTEMAS

- 1- Metodologias e Responsabilidades**
- 2- Projecto e Estudo de Viabilidade**
- 3- Design**
- 4- Desenvolvimento e Implementação**
- 5- Operação e Manutenção**
- 6- Revisão de Pós-Implementação**

8.4 TESTE DA BASE DE DADOS PROGRAMAÇÃO E TESTES

A programação e teste normalmente começam, como era de esperar, quando termina a actividade de projecto. A programação, ou fase de implementação de um projecto típico (vide anexos X e XI) envolve a escrita de instruções em Access 2000, para implementar o que foi especificado e o que foi organizado em módulos. Os testes envolvem a experimentação do sistema para ver se ele produz as saídas correctas e apresenta o comportamento correcto para um grande número de entradas.

As perguntas seguintes são colocadas: Por que é que isso deve interessar ao analista de sistemas? Não é verdade que já não deveria estar interessado por neste ponto? Não, não necessariamente. Por vários motivos, o trabalho que os programadores e os testadores fazem podem influir no seu trabalho, na maneira como eles realizam o deles.

Geralmente quando a programação inicia, o analista pode ter-se transferido para um outro projecto. Mas alguns motivos podem fazer com que o analista permaneça no projecto, tais como:

- a) Ele é o líder do projecto, é o responsável dos programadores
- b) É analista de sistemas júnior, e seu cargo é programador/analista ou analista/programador
- c) Faz parte do grupo que prepara os testes que serão usados para testar os programas pelos programadores. É só recordar o que *Tom de Marco* diz, A especificação é o teste do sistema
- d) Pode estar envolvido no desenvolvimento dos manuais, treinando os usuários, planeando a instalação do novo sistema e a conversão dos dados do sistema antigo.
- e) Os programadores podem não entender certas especificações, ou podem estar incompletas, inconsistentes ou contraditórias

- f) Os usuários podem começar a mudar de opinião a respeito dos requisitos quando os programadores já estiverem a implementar os requisitos que disseram desejar

8.5 SEGURANÇA DO SISTEMA A SER IMPLEMENTADO

Em termos de segurança para este sistema a ser implementado, pode-se considerar os seguintes mecanismos:

- Administração do Sistema através de senhas vulgos passwords: A implementação prevê a criação de usuários para acesso à base de dados (veja-se no formulário 12 de "Níveis de Acesso") o que pressupõem que só terão acesso ao sistema somente os usuários criados. Se não for suficiente, o sistema mostra o nome do usuário que está a trabalhar naquele momento (Relatório 2 "Logon List")

- O uso de Visões

- O uso de restrições como cláusula where e having nas visões e consultas normais da Base de Dados. São úteis quando se pretende restringir os registos a considerar na selecção. A cláusula Having serve para restringir os grupos que foram formados depois de aplicada a restrição da cláusula Where

9. CONCLUSÕES E RECOMENDAÇÕES

9.1- CONCLUSÕES

- No presente trabalho pode-se vêr, facilmente, que os pressupostos da Metodologia de Yourdon funcionam plenamente pela maneira de detecção da informação através de eventos que suportam os DFD e DE e DER, para criar a estrutura do software em linguagens da 5ª geração baseadas nas metodologias orientadas à objectos.
- Desde já, garante-se que a situação actual de processamento imobiliário (no que concerne a atribuição de casa e no sistema de facturação de rendas,...), que ainda é manual, tem sua alternativa credível, o sistema físico actual proposto para alguns sectores.
- A rede proposta neste trabalho (numa arquitectura cliente-servidor), para ser implementada na Imovisa, garante a maior integridade de dados, confere maior Segurança dos mesmos, bem como a Segurança da própria rede. O uso de terminais com recursos próprios de memória, processador e software, será importante para se acessar às funções numa forma independente no host.
- As Auditorias propostas terão eco se houver mecanismos que regulem o funcionamento dos sistemas ali instalados. A criação de normas documentadas e colocadas à disposição do usuário será essencial.
- Os mecanismos de Auditoria propostos neste trabalho provam ser melhores no mundo informático moderno por englobar todas áreas tocantes à gestão e, em particular, para sistemas de gestão imobiliárias.

- Todos processamentos e control da gestão imobiliária, optimização do atendimento e adopção de melhores mecanismos de controle de devedores, são previstos com enfoque, e o sistema funcional (protótipo) proposto é visto como a real e única maior solução.

- A simulação num modelo relacional vê o seu principal eco quando se fala de indexação, por ser a criação de uma árvore abstrata com nós onde as entidades residem e que existem fisicamente no disco, ao contrário das visões.

- O mecanismo de segurança proposto através da criação de usuários é de extrema importância na protecção da informação, e os mecanismos de auditoria consagrados neste trabalho garantem a interligação constante entre os usuários e os Administradores do Sistema assim como a fiabilidade da informação.

- A segurança de dados usando visões prova ser a melhor implementada em sistemas de bases de dados relacionais por ser a melhor maneira de particionamento da Base de Dados.

9.2- RECOMENDAÇÕES

- 1) O novo modelo físico que acompanha este trabalho deverá ser instalado simplesmente em computadores que tenham o Windows NT, porque ele permite o acesso remoto, tem um excelente nível de segurança e pode incorporar várias estações de serviço inteligentes. O Windows NT é melhor que outro produto da Microsoft, falando nomeadamente do Netware, porque este contém uma configuração muito mais fechada.

- 2) O Access, embora contenha ferramentas excelentes para gestão de informação desta base de dados relacional como hospedeiro o SQL Server, contém algumas lacunas de segurança de informação, podendo ser quebradas a partir do momento em que o usuário tenha acesso à região da organização física dos dados (Tabelas). Por isso, terá de se aproveitar das vantagens tecnológicas do Windows NT.
- 3) A modelagem gráfica usando as metodologias orientadas a objectos tem um papel importante na construção de modelos, pela excelente facilidade, mas, a maior parte dos objectos que foram apresentados no modelo físico novo possuem um código de programação para efeitos de optimização, consistência da Base de Dados e protecção dos Dados, não tomando novos objectos sem reparar neste pormenor
- 4) No futuro, terá de ser acrescentada mais uma tabela assunto para separar os colectivos e individuais, de modo a termos uma para os colectivos e outra para os individuais, como forma de facilitar a busca e pesquisa
- 5) Qualquer manutenção de especificações deve ser documentada e verificada com o usuário sem que se esqueça o preenchimento de um formulário conhecido como solicitação de alteração do sistema.

10. BIBLIOGRAFIA

REFERÊNCIAS BIBLIOGRÁFICAS

- A Visual introduction to SQL, J. Harvey Trimble and Jr. David Chappell, Great Falls, Virginia, February 1989.
- Lucas, J.H.C. (1992) The Analysis, Design and Implementation of Information Systems, Fourth Edition McGraw Hill – International Editions.
- MOELLER, Robert R.. Computer audit, control and security. New York, John Wiley & Sons, 1989.
- PC Magazine "2001"(Intranets Globais e a informática no novo milênio), edição de Agosto de 1998
- Penny A. Kendall, (1992) Introduction to Systems Analysis and Design – A Structured Approach, second edition USA;
- Shiller, L. (1990). Software Excellence. Prentice Hall, 239 pp. New Jersey, Englewood Cliffs.
- TCU, Secretaria de Controle Externo, Portaria n. 24-GP, de 04 de Setembro de 1997, *Procedimentos de Auditoria*, Tribunal de contas
- TCU, Secretaria de Auditoria e Inspeções, Portaria n. 455-GP, de 30 de Setembro de 1998, *Manual de Auditoria de Sistemas*, Tribunal de contas

- Tecnologias de Informação, SQL, *Luís Damas*, Editora de Informática, 2ª edição, 2000
- Theory, T. J. (1999). Database Modeling & Design, Third Edition, 365 pp. California, Morgan Kaufman Publishers.
- Yourdon, E. (1990). Análise Estruturada Moderna. 3ª Edição, 836 pp. Rio de Janeiro , Editora Campus.

BILIOGRAFIA CONSULTADA

- A vida da Empresa. Projecto Informático, P. Morange, Clássica editora 1988
- C.J. Date e Hugh Darwen, A Guide to The SQL Standard Addison Wesley, 4ª Ed.
- Cood, P. and E. Yourdon (1991) Object Oriented Analysis, Second Edition, Yourdon Press
- DeMarco, T. (1989). Análise Estruturada e Especificação de Sistema. Campus, 352 pp. , Rio de Janeiro.
- DOUGLAS, Ian J.. Computer audit and control handbook. Oxford, Butterworth Heinemann Ltd, 1995.
- GAO - United States General Accounting Office. The System Assessment Framework, Version 1.0 (Glossary). Washington, DC, GAO, Accounting and Information Management Division, 1996.

- <http://www.fca.pt> e <http://www.lidel.pt>

- Jeff Perkins e Bryan Morgan, Teach Yourself SQL in 14 days SAMS

- Microprocessadores: Dutos de sistema, técnicas de interface e sistemas de comunicação de dados (série microprocessadores) de João António Zuffo.

- Satir, R. M. (1994) Princípios de Sistemas de Informação, 2ª Edição, São Paulo Brasil, Editora McGraw Hill

- SERPRO - Serviço Federal de Processamento de Dados. Glossário de Termos Técnicos em Processamento de Dados. Brasília, SERPRO, Centro de Comunicação Social, 1985.

11 - ANEXOS

Anexo I

Estrutura Orgânica da Imovisa – Imobiliária de Moçambique, Lda

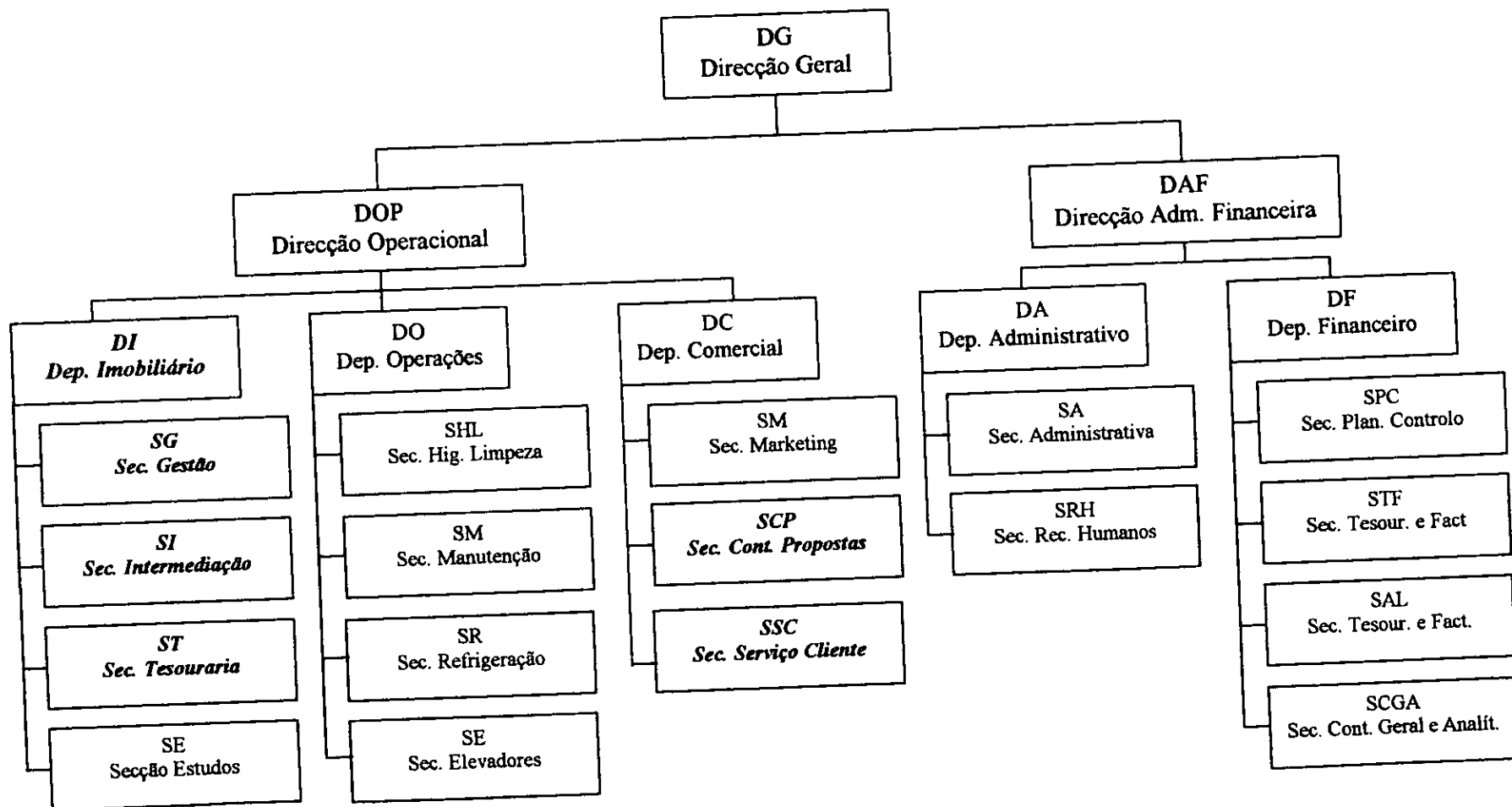
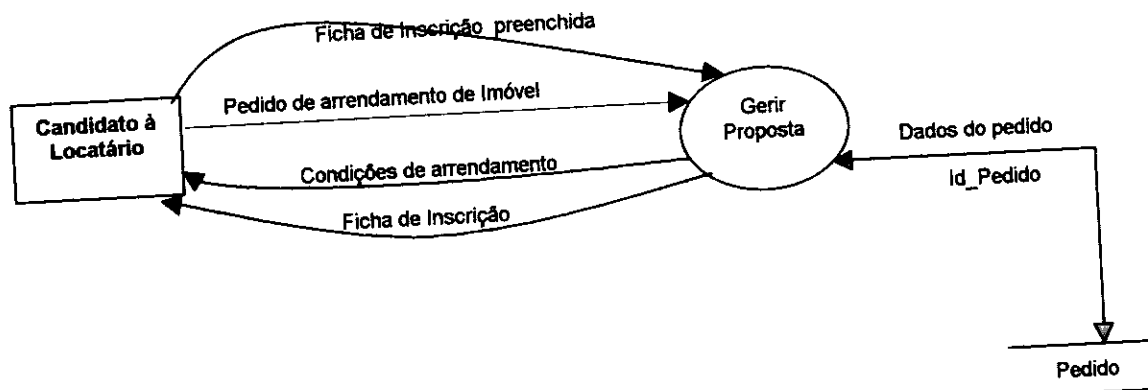


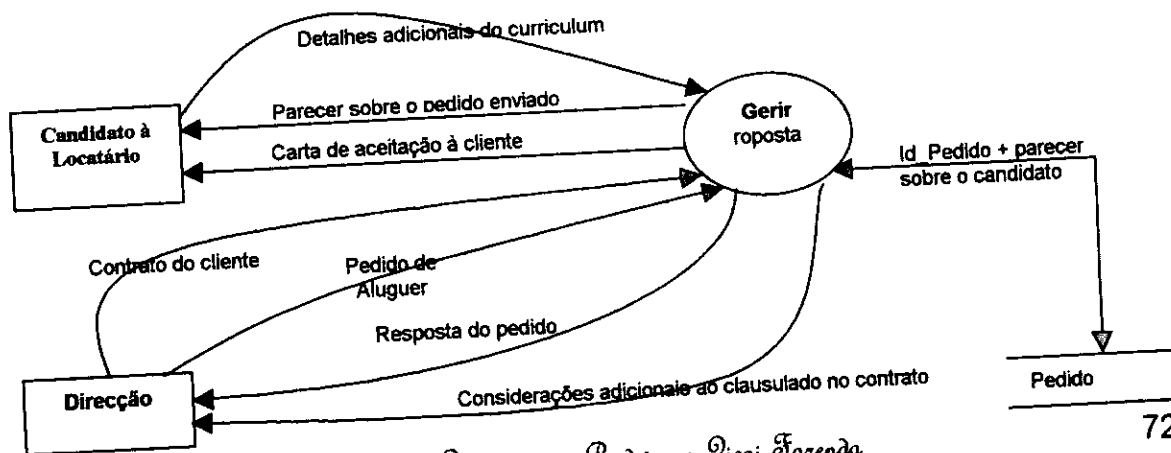
DIAGRAMA DE FLUXO DE DADOS PRELIMINAR

Evento 1: Candidato a Locatário realiza pedido de arrendamento



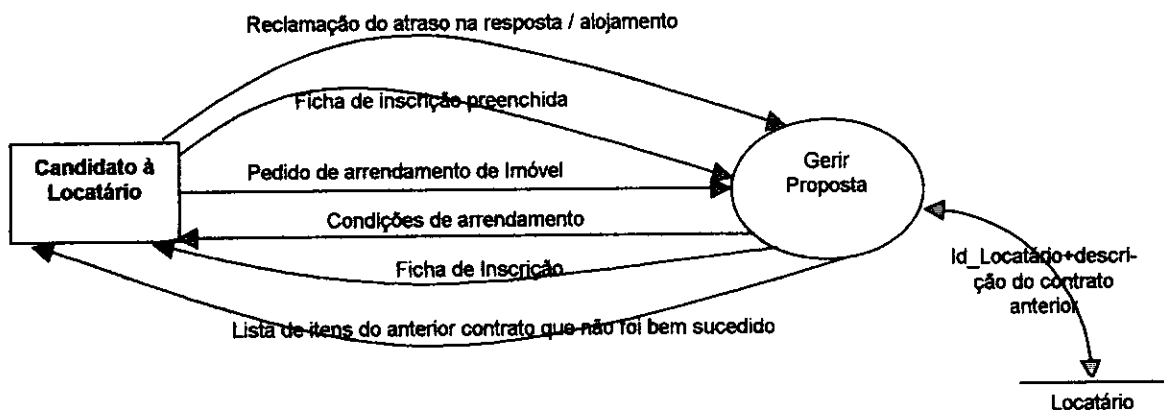
Quando um candidato em arrendar um Imóvel se desloca à Imovisa, são-lhe entregue as respectivas condições de arrendamento. Em caso de as aceitar, a Imovisa entrega acto esse que pode ser comunicado oralmente ou por escrito, a Imovisa entrega uma ficha de inscrição que após o seu preenchimento pelo candidato à Locatário, deve ser devolvida à empresa. Como forma de manter a actualização dos seus arquivos a Imovisa lança todos dados deste pedido no respectivo arquivo (pedido), para efeitos de posteriores consultas, porque poderia acontecer que o candidato tenha por lá passado noutras vezes como mau inquilino.

Evento 5: Direcção analisa pedido do Candidato à Locatário



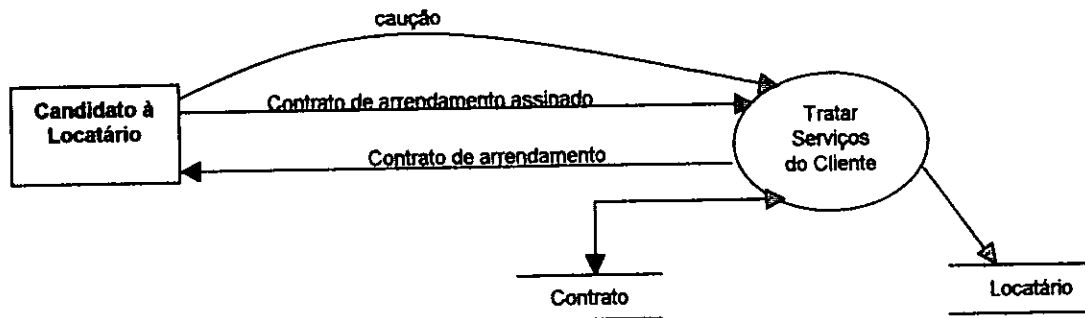
Quando um determinado pedido de arrendamento dá entrada nos escritórios da Imovisa, é enviado para a Direcção para efeitos de análise. Como os pedidos foram guardados num arquivo como descrito no evento 1, pode-se obter facilmente este pedido através do seu Id_pedido e o parecer que foi dado pela pessoa responsável da recepção dos documentos, o que facilitará a aprovação desse candidato à Locatário. Assim, se a Direcção o aceita como pessoa adequada a ser seu inquilino, é enviada uma carta de aceitação, no intuito de o ver a ocupar o Imóvel por ele solicitado.

Evento 7: Candidato à Locatário indaga a situação de pedido de arrendamento



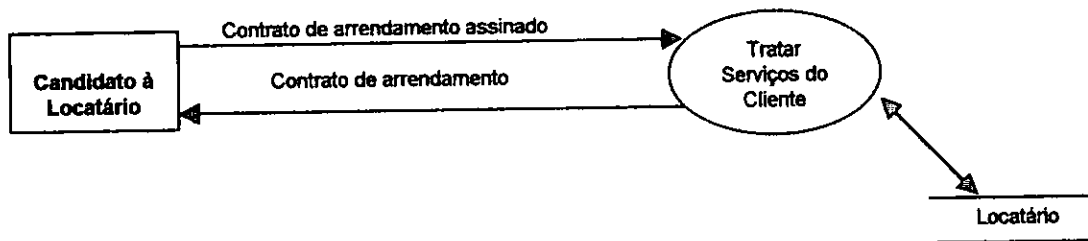
Caso o pedido de arrendamento dum determinado candidato não esteja ao fim da análise pela parte da Direcção, devido a vários constrangimentos, como são os casos de ter sido mau cliente nas vezes passadas e que estejam em debate diversas atenuantes, o candidato pode notar nisto um atraso, ao que inquire a Gestão sobre o seu pedido. Nisto, por estes estarem informados ou já com a decisão tomada, entregam-lhe uma lista de itens do anterior contrato que não foi bem sucedido adicionado aos argumentos que sustentam a direcção na tomada de qualquer decisão.

Evento 10: Direcção considera aceitável o Candidato à Locatário ser Locatário



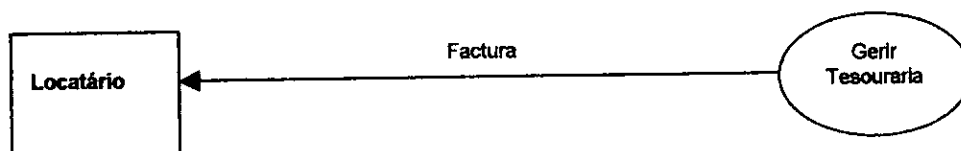
Se se aceitou como cliente segundo o evento 3, como primeira medida é obrigado a pagar um valor correspondente a uma renda mensal, o que é chamado **caução** (veja a descrição do sistema actual). Por um lado toma-se o contrato deste cliente para o arquivo *contrato*, porque se precisará, mensalmente, destes dados para processamento de facturas pela Tesouraria

Evento 11: Direcção aceita ter mais um Locatário



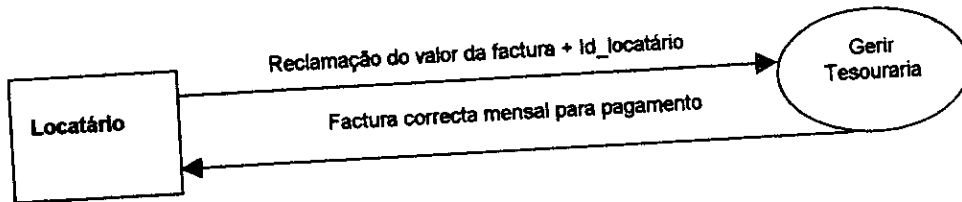
Depois do pedido analisado pela direcção e após a assinatura do contrato de arrendamento, este candidato, é declarado Locatário, pelo que a actualização do arquivo Locatário é pertinente, em virtude desta ser a adição de mais um cliente.

Evento 17: Factura precisa de ser remetida ao Locatário



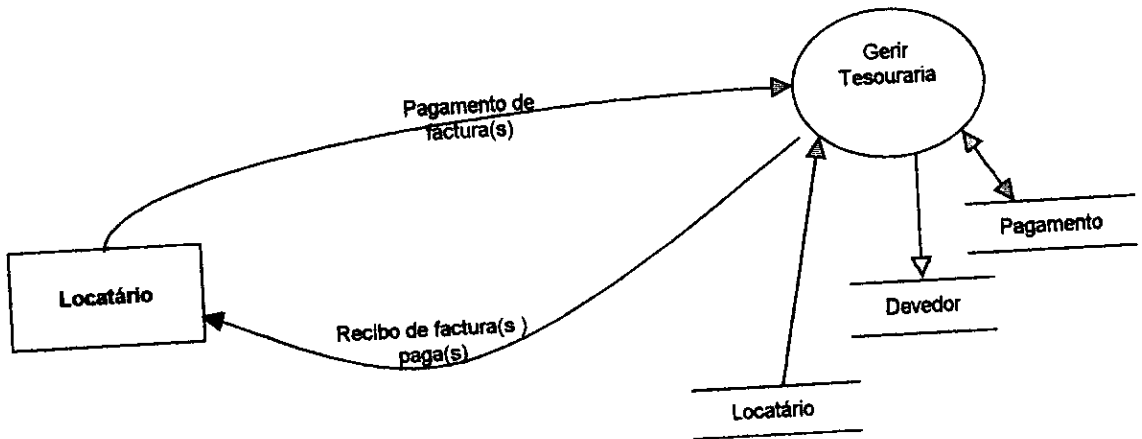
Este evento ocorre mensalmente, pois no dia 01, todas facturas devem começar a ser remetidas aos seus legítimos destinatários para efeitos de pagamento. Então, encarrega-se a um grupo de funcionários para procederem as respectivas distribuições.

Evento 18: Locatário indaga sobre a situação de uma factura



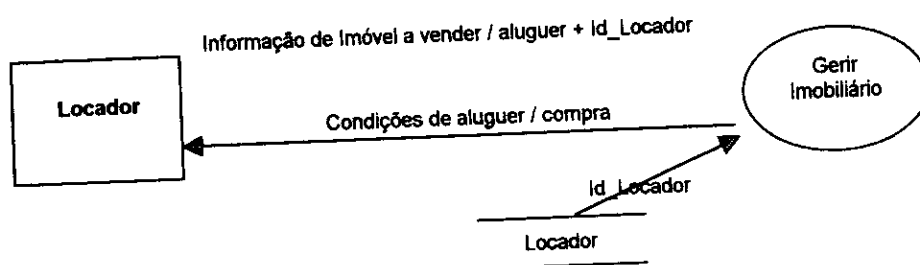
Este evento é activado quando o cliente recebe uma factura diferente do valor previsto no contrato, ou mesmo se não recebe a factura no prazo previsto, sendo obrigado a reclamar a data que possa vir a receber.

Evento 19: Locatário envia pagamento de factura



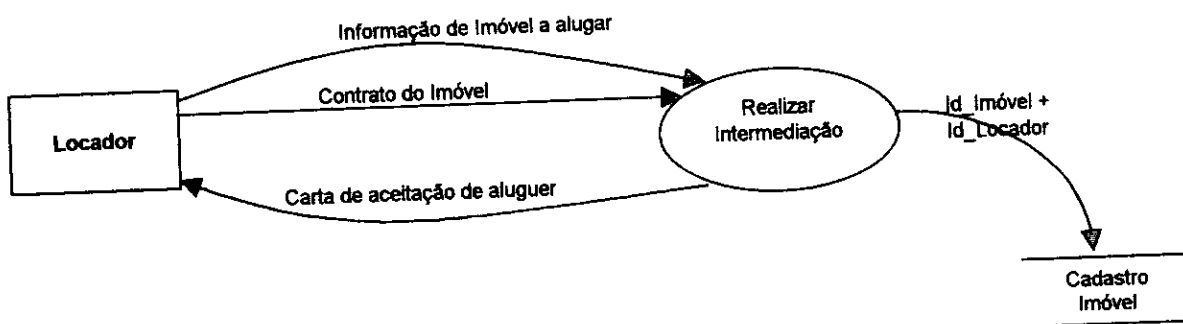
Activado mensalmente quando o Locatário recebe a factura, este evento vela pela situação do pagamento pela parte do cliente.

Evento 21: Locador informa à intermediação do aluguer/venda de Imóvel



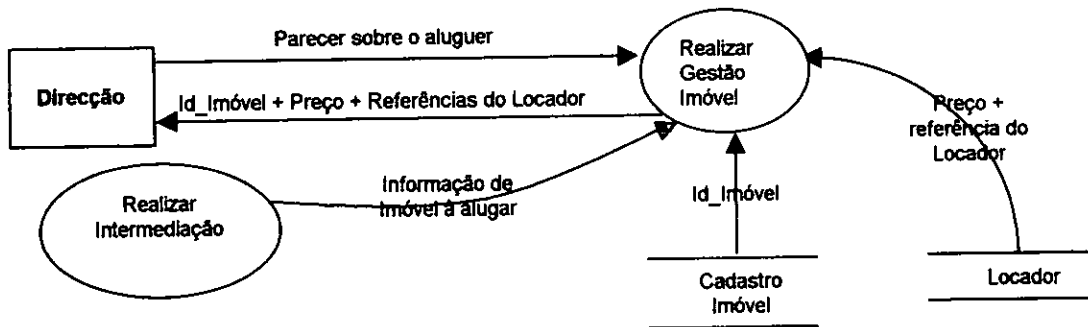
A Imovisa para além de arrendar casas aos seus clientes, também aluga ou compra casas de Locadores que estejam interessados, e quando as tais casas oferecem excelentes condições. Nesse caso, o Locador que pretende ver a sua casa alugada ou vendida, pode contactar a Imovisa no intuito de ter a pretensão satisfeita. Daí, ele informa à Imovisa oferecendo todas as condições para o aluguer/venda. Depois da avaliação feita à casa, a Direcção da Imovisa pode ordenar a aquisição desta casa, conforme algo que seja epigrafado no contrato.

Evento 22: O Locador aceita o aluguer do seu Imóvel



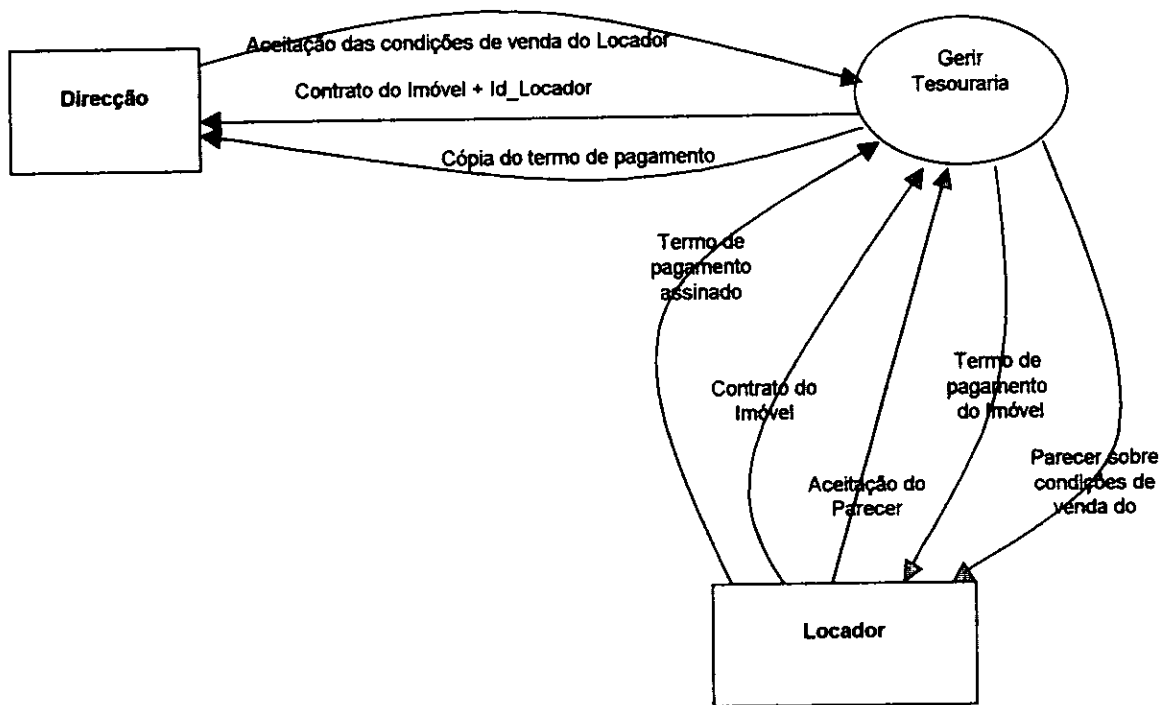
Este evento é activado quando um determinado locador pretenda que a sua casa seja entregue à Imovisa para efeitos de arrendamento, entre em acordo de comissões mensais por entregar, para que se proceda a concessão a um Locatário.

Evento 23: Direcção pede informação de Imóvel a alugar /comprar à Gestão



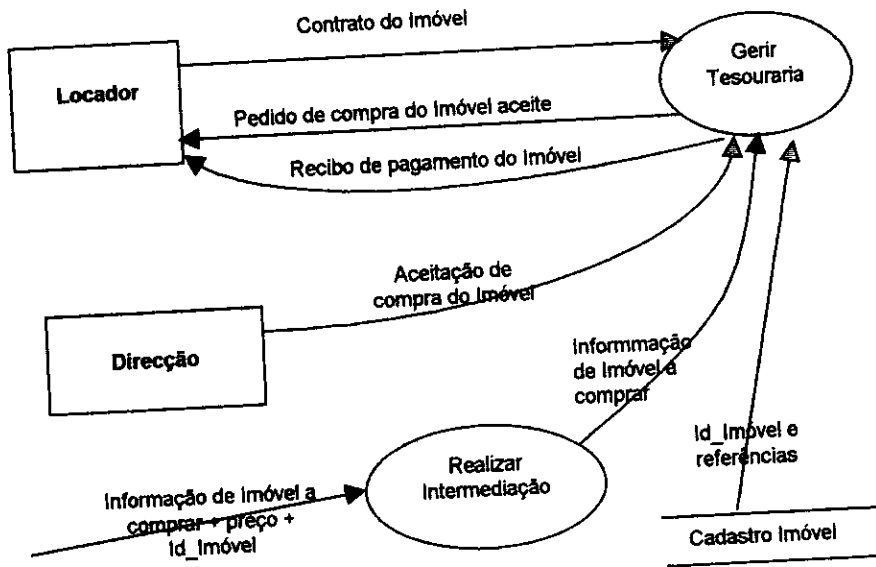
Para que ocorra este evento, espera-se que um determinado Locador apareça na Imovisa com intuito de arrendar a sua casa em troca de alguma comissão mensal.

Evento 24: Tesouraria propõem a compra do Imóvel ao Locador



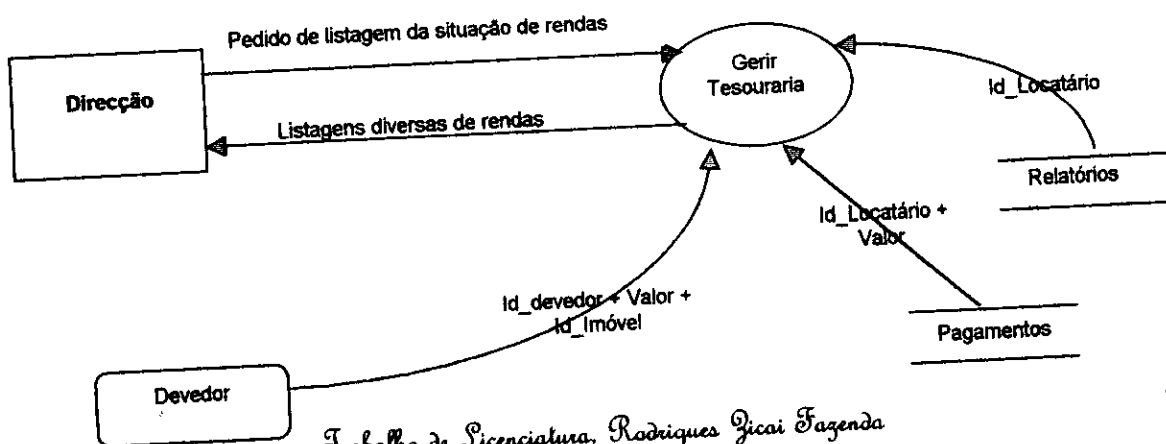
Se a Direcção aprova a compra duma casa já proposta pela Tesouraria e esta tenha já recebido o contrato da casa, a Tesouraria prepara o pagamento da casa num determinado período.

Evento 27: Tesouraria efectua o pagamento da compra de Imóvel



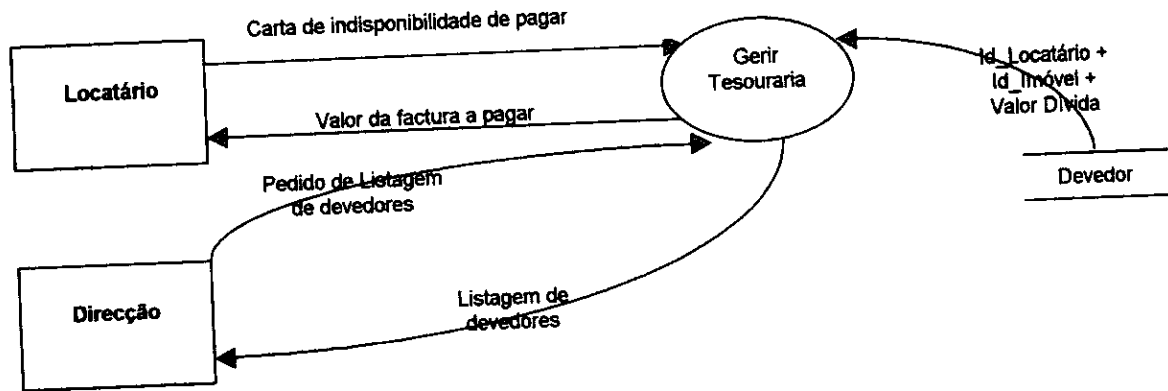
Se a Direcção concorda com as características que a casa apresenta e após um mandato e parecer da inspecção, envia uma informação á Tesouraria admitindo que se faça o pagamento da casa (caso seja para compra) ou admitindo que se faça um contrato com o Locador sobre a verba mensal (caso seja o aluguer)

Evento 29: Direcção pretende saber a situação de pagamento de rendas



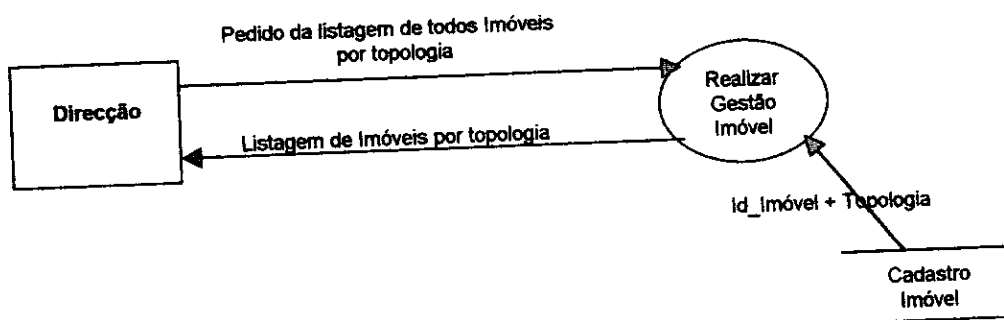
A qualquer momento a Direcção pode querer saber a situação relativa aos últimos pagamentos, para evitar casos de indisciplina pela parte dos Locatários, pois sempre interessa saber se o seu comportamento para com as facturas é normal.

Evento 30: Direcção exige Listagem de devedores de cada mês



Após ter passado o dia 10 de cada mês, considera-se o cliente (Locatário) que não tiver feito o pagamento da renda de devedor, pelo que a direcção precisa de ter a listagem para efeitos de tomada de medidas.

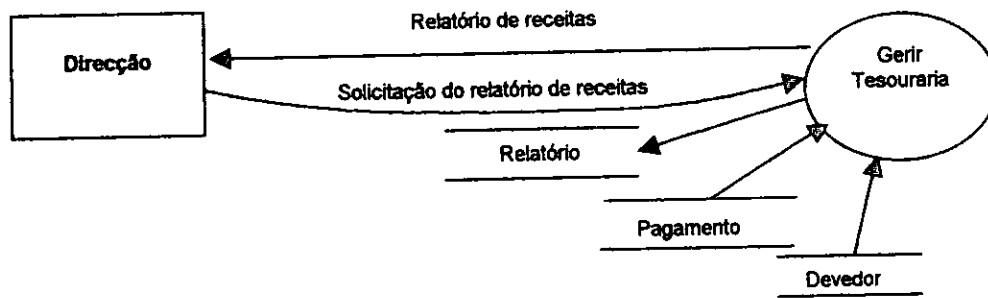
Evento 31: Direcção solicita listagem de Imóveis pela Topologia



Para efeitos de controle dos Imóveis de um determinado tipo, a direcção pode pedir que esta listagem seja efectuada. O exemplo mais nítido é o caso em que

se queira saber, quantas e quais casas do tipo dois existem como propriedades da Imovisa.

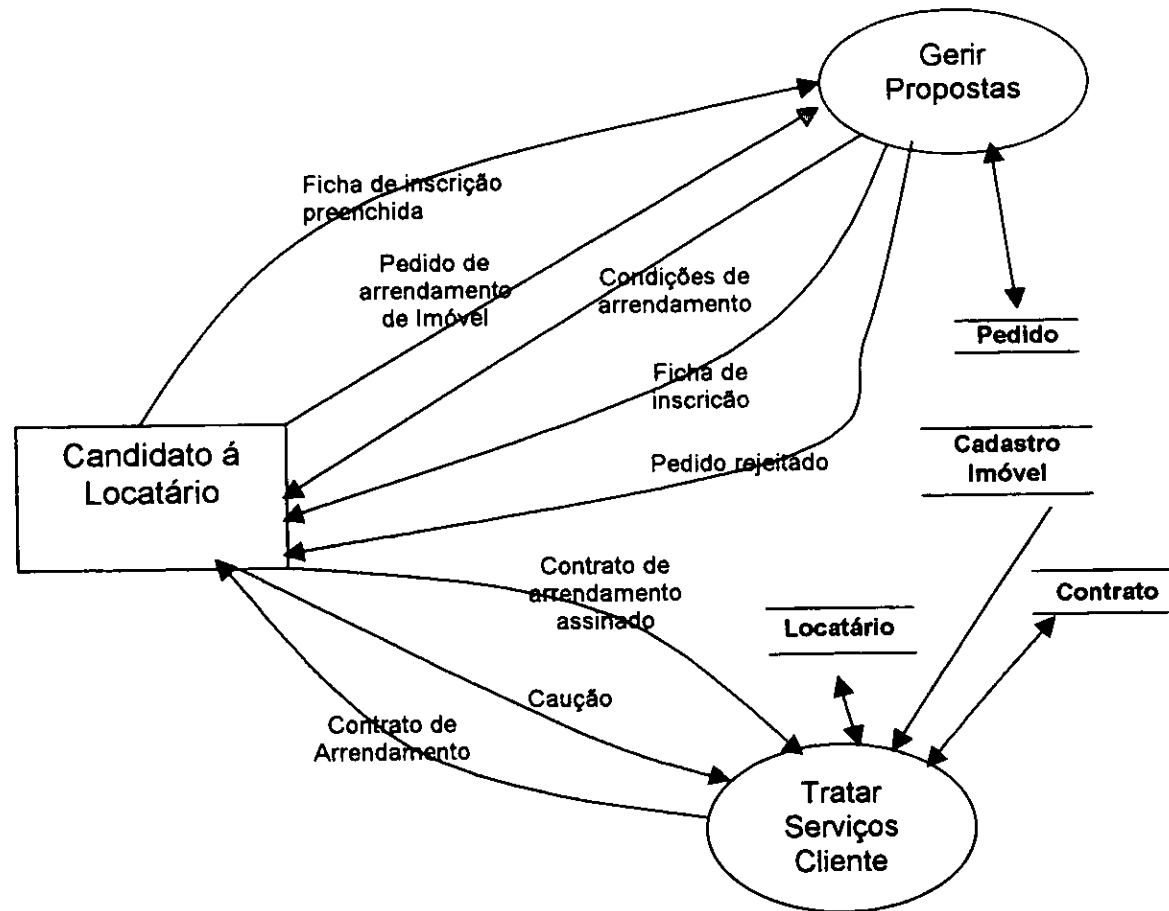
Evento 32: Direcção precisa do relatório "mensalmente" de receitas



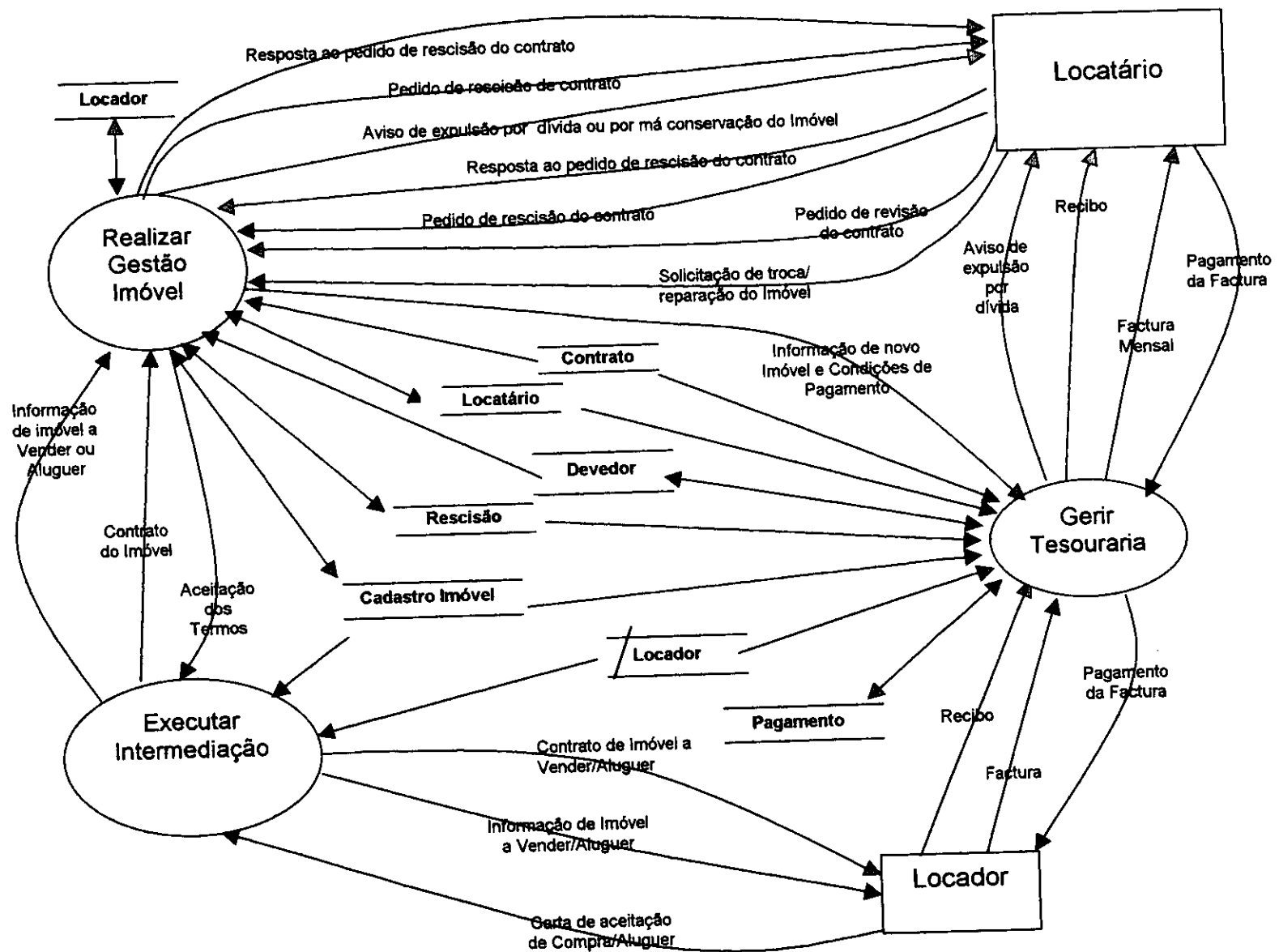
Se a tesouraria fizer o fecho dos valores mensais comunica a Direcção para efeitos de balanço mensal.

Anexo III
PROCESSOS DE BAIXO NÍVEL

Expansão do Processo: Realizar Controlo Comercial



Expansão do Processo: Gerir Imobiliário



DICIONÁRIO DE DADOS DE ALGUNS FLUXOS

Nome do Fluxo: Pedido de arrendamento de imóvel

Composição: @Número da ficha de inscrição + Nome + data de nascimento + Estado civil + profissão + entidade patronal + Nacionalidade + tempo de permanência + Número de BI + Residência actual + Contacto

Descrição: Do terminador "Candidato á Locatário" para o processo "Gerir Propostas", este fluxo ocorre quando um candidato pretende um imóvel da Imovisa de aluguer para um determinado assunto.

Nome do Fluxo: Contrato de Arrendamento

Composição: @Número do Contrato + Data de início do contrato + Data fim do contrato + Valor de caução *este valor deve ser igual ao da renda mensal* + Renda mensal

Descrição: Do processo "Tratar Serviços Cliente" para o terminador "Candidato á Locatário". Este fluxo ocorre após a manifesta vontade de pretender um imóvel de aluguer, de ter-se recebido e analisadas todas condições de arrendamento.

Nome do Fluxo: Recibo

Composição: @Número do recibo + Número de contrato + localização + tipo de imóvel + topologia + Nome do Inquilino + Valor Pago + Último mês + Ano

Descrição: Após o pagamento do valor da factura resta a Imovisa, passar o respectivo recibo para o seu cliente como forma de confirmar a recepção do valor cobrado.

Nome do Fluxo: Aviso de expulsão por dívida ou má Conservação do Imóvel

Composição: @Número do aviso + Data + Nome do locatário + Motivos + Localização da casa + Assinatura da Direcção

Descrição: Este fluxo é activado quando o cliente passa o dia 10 do mês sem ter regularizado as suas contas ou quando após uma inspecção realizada ao imóvel, descubra-se que o cliente não conserva a casa á condições mínimas exigidas.

O cliente tomará conhecimento do resultado e medidas

Nome do Fluxo: Informação de Imóvel a vender ou alugar

Composição: Dados do Locador *veja no DD do Locador* + Detalhes do Imóvel + referências sobre o estado da casa + @Código do título de propriedade

Descrição: Quando um Locador pensa em vender/alugar seu imóvel à Imovisa, ele contacta-a no intuito de saber se esta pode estar interessada, mediante as referências do imóvel, como são os casos da zona de localização, o tipo, o contrato, etc.

DICIONÁRIO DE DADOS DE PROCESSOS

Nome do Processo: Gerir Propostas

Descrição: Este processo se encarrega de realizar a recepção de pedidos de arrendamento, encaminhá-los para análise à nível do Departamento Comercial ou mesmo à Direcção Geral, arquivá-los e posterior entrega de resposta do pedido.

Nome do Processo: Tratar Serviços Cliente

Descrição: Após a declaração de existência dum Candidato á Locatário, recepção do contrato de arrendamento assinado e do recibo de pagamento do valor de caução, transforma o candidato em Locatário, guardando respectivo contrato no respectivo processo. Em suma: Gere e trata de todas situações contratuais.

Nome do Processo: Realizar Gestão Imóvel

Descrição: Este processo trata de casos relacionados com contenciosos entre a Imovisa e o Locatário, atendimento de processos do Locador que lidam com alguma intermediação de compra ou aluguer de Imóvel e também faz a gestão de todo o Parque Imóvel da Imovisa.

Nome do Processo: Executar Intermediação

Descrição: Recebe e gere toda informação de propostas de venda ou aluguer de Imóveis dos Locadores, consulta constantemente o cadastro de Imóveis no intuito de dar qualquer informação sobre um Imóvel que se queira alugar/comprar a outras estruturas, sempre que assim o desejar.

Nome do Processo: Gerir Tesouraria

Descrição: Este processo recebe os valores de caução e de renda, emite facturas e recibos. a qualquer momento pode oferecer, dados sobre a situação de pagamentos.

Nome do Arquivo: Pagamento

Descrição: Guarda informação do cliente em relação aos pagamentos. Ele exceptua registar dentro de si todo o Locatário em dívida.

Composição: @Codpag + Mesreferencia + Valorrenda + Num_contrato + Nome + Data_Pagamento + Ano + Meticais

Nome do Arquivo: Cadastro Imóvel

Descrição: Armazena todos dados do Imóvel. Esses dados podem ser dum imóvel do terceiro (Locador) ou da própria Imovisa.

Composição: @I_Id + I_Localização + I_Nome_prop + I_Bloco + I_Area + Loc_Id + Edificio_Id + Nome_Edificio + Num_Entrada + C_Piscina + C_Garagem + Flat_Id + Num_Flat + Num_Andar + Tipo_Casa + N_Contrato + Parque

Nome do Arquivo: Contrato

Descrição: Armazena os dados de todos Locatários. Isso facilita para que quem não é inquilino agora, mas que outrora fora, mantenha o seu registo curricular dentro da empresa.

Composição: @C_Id + C_Num_Contrato + C_data_Inicio + C_Data_Fim + C_Valor Caução + C_Valor_Renda + C_Imóvel + AF_Id + AF_Nome + AF_Idade + AF_Parent + AF_Prof + AF_Pid

Nota: AF – Agregado Familiar, C - Contrato

DICIONÁRIO DE DADOS DE ARQUIVOS

Nome do Arquivo: Pagamento

Descrição: Guarda informação do cliente em relação aos pagamentos. Ele exceptua registar dentro de si todo o Locatário em dívida.

Composição: @Codpag + Mesreferencia + Valorrenda + Num_contrato + Nome + Data_Pagamento + Ano + Meticais

Nome do Arquivo: Cadastro Imóvel

Descrição: Armazena todos dados do Imóvel. Esses dados podem ser dum imóvel do terceiro (Locador) ou da própria Imovisa.

Composição: @I_Id + I_Localização + I_Nome_prop + I_Bloco + I_Area + Loc_Id + Edificio_Id + Nome_Edificio + Num_Entrada + C_Piscina + C_Garagem + Flat_Id + Num_Flat + Num_Andar + Tipo_Casa + N_Contrato + Parque

Nome do Arquivo: Contrato

Descrição: Armazena os dados de todos Locatários. Isso facilita para que quem não é inquilino agora, mas que outrora fora, mantenha o seu registo curricular dentro da empresa.

Composição: @C_Id + C_Num_Contrato + C_data_Inicio + C_Data_Fim + C_Valor Caução + C_Valor_Renda + C_Imóvel + AF_Id + AF_Nome + AF_Idade + AF_Parent + AF_Prof + AF_Pid

Nota: AF – Agregado Familiar, C - Contrato

Nome do Arquivo: Pedido

Descrição: Ele armazena dentro de si, todos pedidos dos clientes, quer sejam individuais ou colectivos.

Composição: @Pc_Fich_Insc + Pc_Denominação + P_Nome + P_Data_Nasc + P_Estado_Civil + P_Prof + P_Ent_pat + P_nat + P_Nacio + P_Tem_Per + P_BI + P_Dire + P_Emit + P_Arquivo + P_Resid_Act + P_Telef_serv + P_Tel_Resid + Nacio_Id + Pc_Tipo + PC_sede + P_Data_Const + P_Reg_Comecial + P_Ninc + P_Bairro_Fis + P_Act_Princ + P_Telef + P_N_Insc

Nota: P – Pedido Individual, Pc – Pedido colectivo

Nome do Arquivo: Locatário/Representante

Descrição: Guarda-se aqui todo o cadastro do candidato à Locatário, Locatário ou Representante da empresa para o caso de terceiros.

Composição: @R_Id + R_Nome + R_Func + R_Data_Nasc + R_Estado_Civil + R_Nat + R_Nacio + R_BI + R_Dire + R_Emit + R_Arquivo + R_N_Siab + R_Cart_Not + R_Data_Absin + R_Resid + R_Telef + Pc_Id

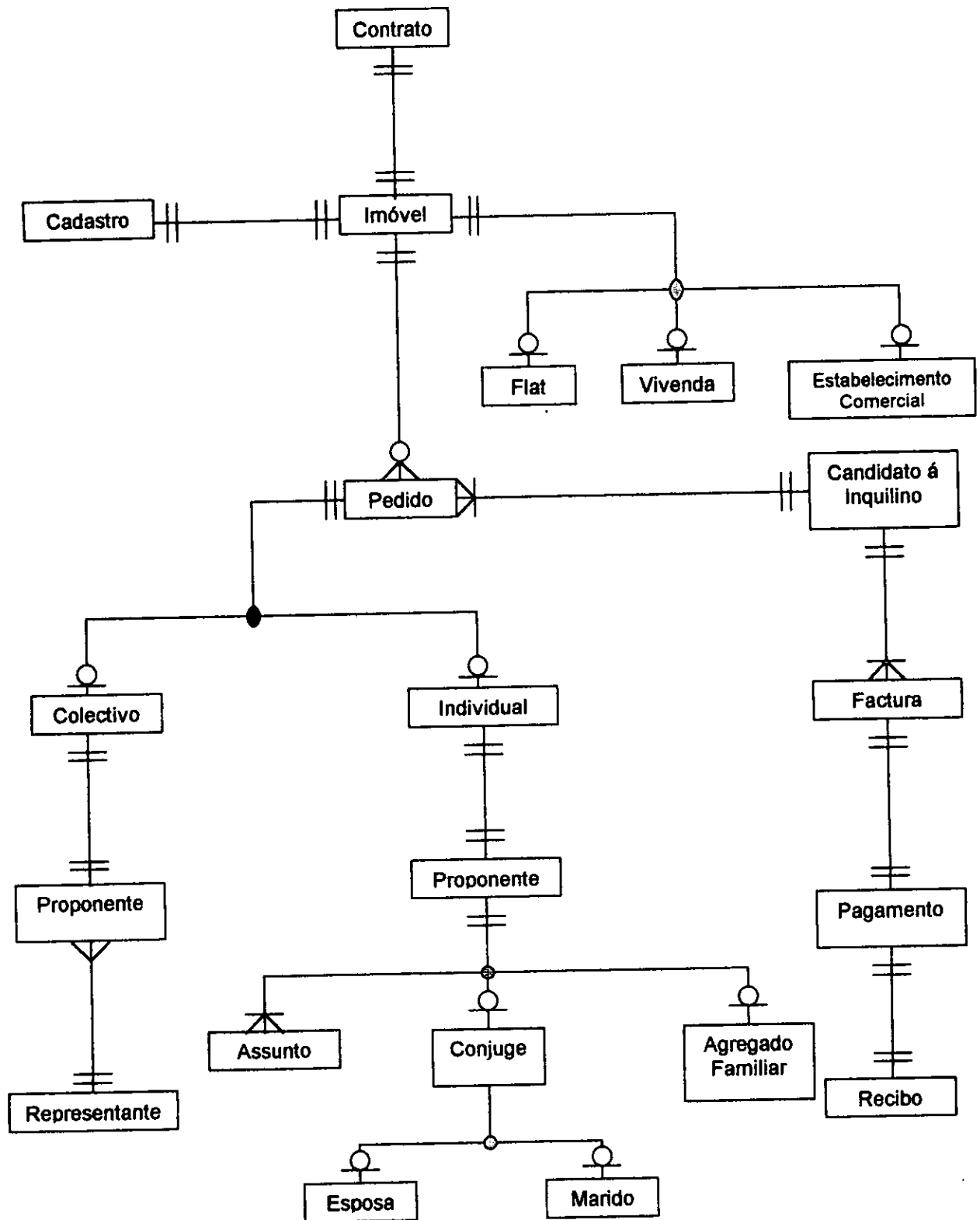
Nota:

Nome do Arquivo: Devedor

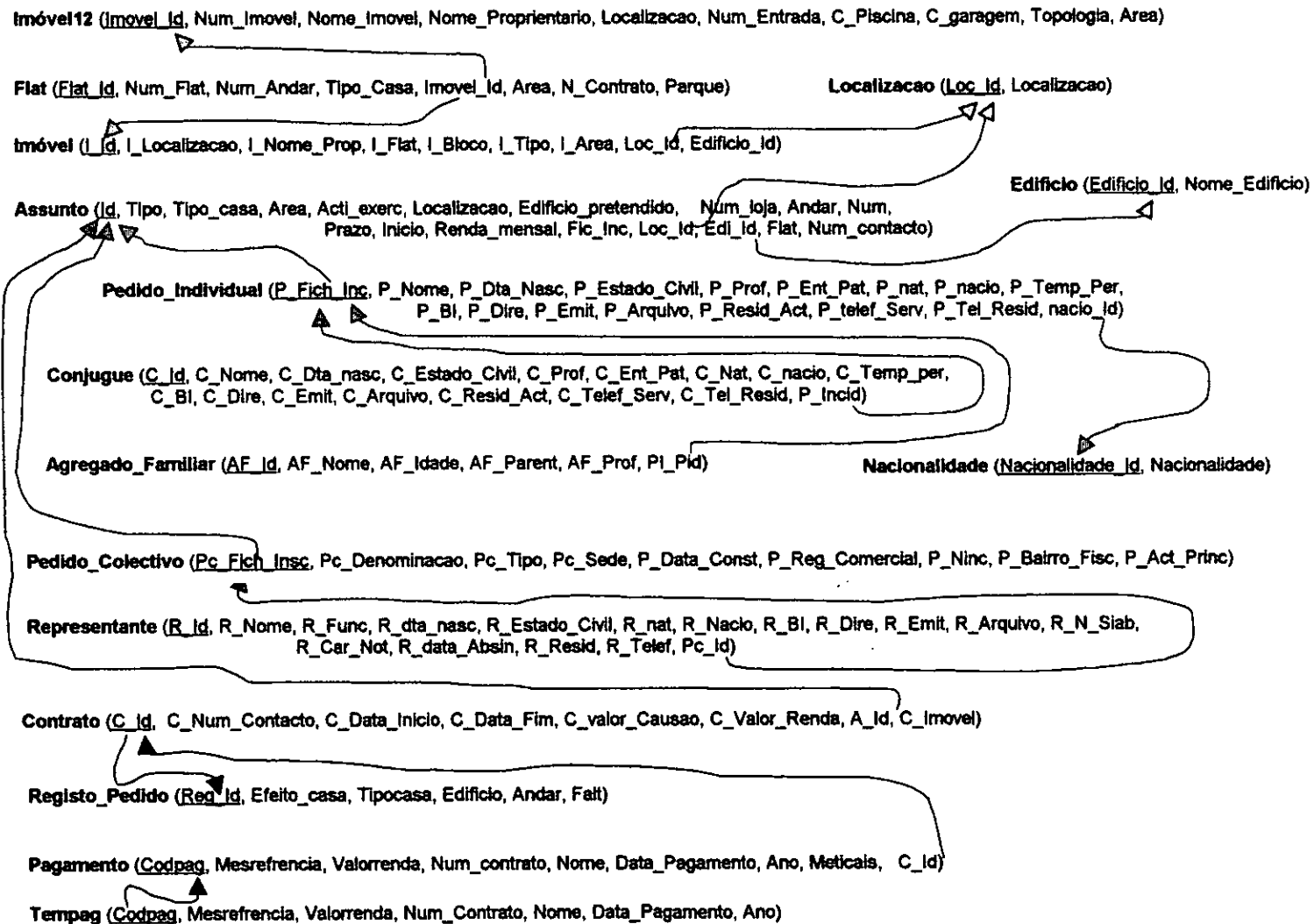
Descrição: Guarda toda informação do cliente que não tenha feito o pagamento de renda até dia 10 de cada mês. Acumula os valores em dívida.

Composição: @Cod_Locatário + Cod_Factura + mês_dívida + Valor_dívida

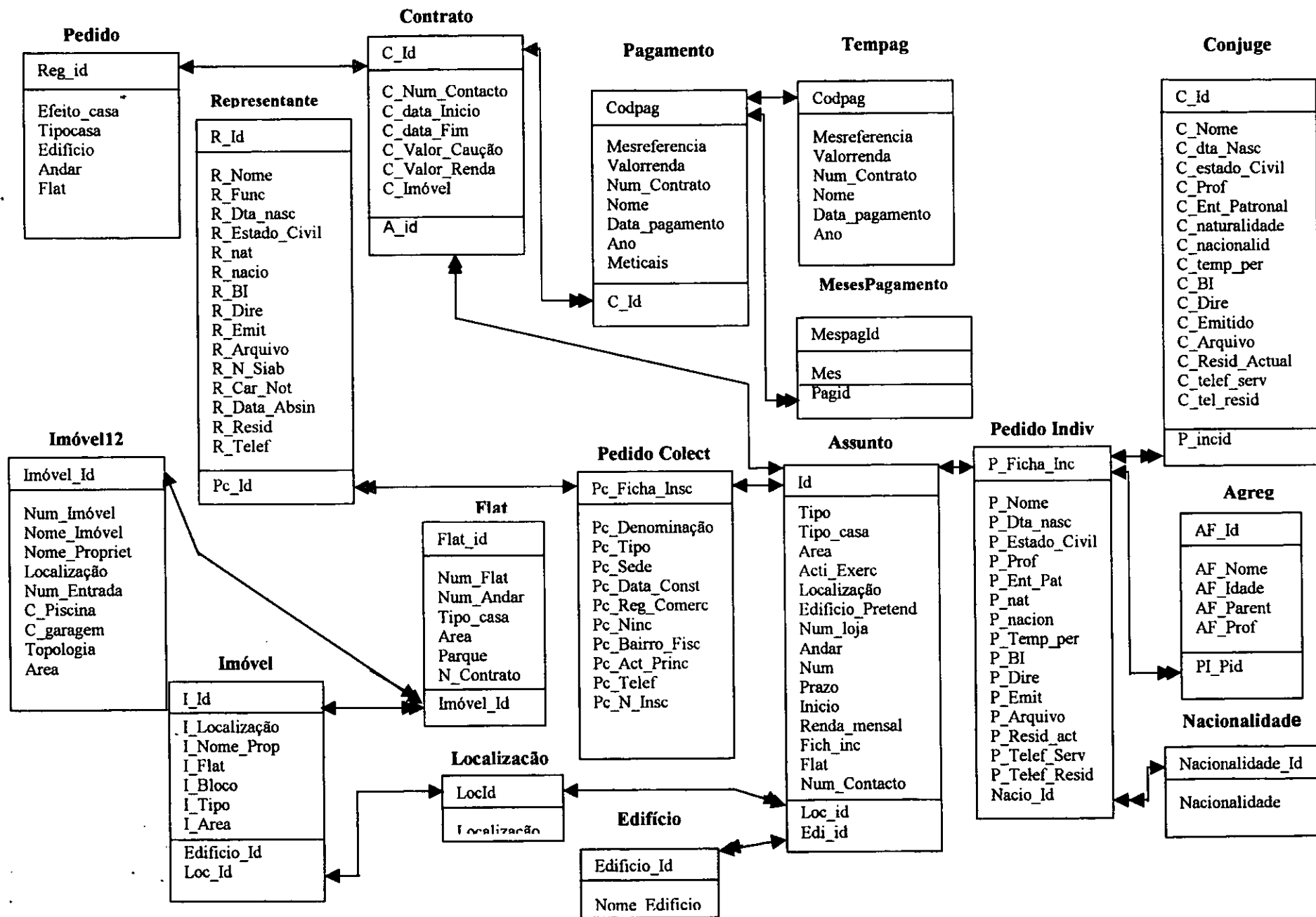
Anexo V: O DIAGRAMA ENTIDADE RELACIONAMENTO



Anexo VI: ESQUEMA RELACIONAL



Anexo VII: ESQUEMA DE TABELAS



Anexo VIII: CRIAÇÃO DE TABELAS USANDO O SQL

1- Criação da tabela Localização

Create Table Localização

(LocId Counter Primary Key,
Localização Char)

2- Criação da tabela Contrato

Create Table Contrato

(C_Id Counter Primary Key ,
C_Nun_Contacto Integer,
C_data_Inicio Date,
C_data_Fim Date,
C_Valor_causao Float,
C_Valor_Renda Float,
A_id Integer,
C_imovel Char);

3- Criação da tabela Assunto1

Create table Assunto1

(Id Counter,
Tipo char Not Null Check (Tipo In ('Apartamento','P/Habitação','P/Escritório',
'Espaço para Comércio')),
Tipo_Casa Char,
Área Float,
Act_Exerc Char,
Localização = (Select Distinct Localização.Localização From Localização),
Edifício_Pretendido = (Select Distinct Imovel12.Nome_Imóvel From Imovel12),
Num_loja Integer,
Andar Integer,
Num Integer,
Prazo Integer,
Inicio Date,
Renda_mensal Float,

```

Fic_inc      Char,
loc_id,      Number
Edi_id       Number,
Flat = (Select Distinct Flat.Num, Flat.Tipo_Casa, Flat.Imovel_Id,
        Imovel12.Num_Imovel, Imovel12.Nome_Imovel From Flat Inner Join
        Imovel12 on Flat.Imovel_Id = Imovel12.Imovel_Id Where
        (Imovel12.Nome_Imovel) = [Forms]![Assunto1]![Edificio]),
Num_contacto Integer);

```

4- Criação da tabela Agregado_Familiar

Create table Agregado_Familiar

```

(AF_Id Counter Primary Key,
AF_Nome Char,
AF_Idade Numeric(2),
AF_Parent Char,
AF_Prof Char,
PI_Id Numeric);

```

5- Criação da tabela Conjuge

Create table Conjugue

```

(C_ID          Counter Primary Key,
C_Nome         Char Not Null,
C_Dta_Nasc     Date,
C_Estado_Civil Char Not Null,
C_Prof         Char,
C_Ent_Pat     Char,
C_Nat         Char,
C_Nacio       Char,
C_Temp_Per    Char,
C_BI          Numeric Unique,
C_Dire        Numeric Unique,
C_emit       Date,
C_Arquivo     Char,
C_Resid_Act   Char,
C_Telef_Serv  Numeric
C_tel_resid   Numeric
P_INCID       Char);

```


6- Criação da tabela Edificio

Create table Edificio

(Edificio_id Counter Primary Key,

Nome_edificio Char Not Null);

7- Criação da tabela Edificio

Create table Edificio

(Flat_id Counter Primary Key,

Num_flat Integer,

Num_andar Integer,

Tipo_casa Integer,

Imovel_Id Number,

Area Float,

N_contrato Number,

Parque Char);

8- Criação da tabela Imovel

Create table Imovel

(I_Id Counter Primary Key,

I_localizacao Char,

I_nome_prop Char,

I_Flat Integer,

I_Bloco Char,

I_tipo Integer,

I_Area Float,

loc_id Counter References Localização(Locid),

edificio_id Integer References edificio(edificio_id));

9- Criação da tabela Imovel12

Create table Imovel12

(Imovel_id Counter Primary Key,

Num_imovel Integer,

Nome_imovel Char,

Nome_proprietario Char,

Localizacao Char,
Num_entrada Integer,
C_piscina Char,
C_Garagem Char,
Topologia Char,
Area Numeric (3,2);

10- Criação da tabela Nacionalidade

Create table Nacionalidade
(Nacionalidade_id Counter Primary Key,
Nacionalidade Char);

11- Criação da tabela Pagamento

Create table Pagamento
(codpag Counter Primary Key,
Mesreferencia Char,
Valorrenda Float references Contrato(C_Valor_renda),
Num_contrato Char references Flat(N_Contrato),
Nome Char,
Data_pagamento Date,
Ano Integer,
Meticais float);

12- Criação da tabela Meses que vai pagar

Create table Mesespagamento
(mespagid Counter Primary Key,
Mes Char,
Pagid Integer,
Foreign Key(Pagid) references Pagamento(Codpag));

13- Criação da tabela Tempo de Pagamento

Create table TemPag
(Codpag Counter Primary Key,
Mesreferencia Char,
Valorrenda Float references Contrato(C_Valor_renda),
Num_contrato Integer,
Nome Char,

Data_pagamento Date,
Ano Integer);

14- Criação da tabela Temporária de Nomes

Create table TempP
(Temp_id Counter Primary Key,
Nome Char);

15- Criação da tabela Senha

Create table Senha
(Id Counter Primary Key,
Nome Char,
Senha Char);

16- Criação da tabela Recibo

Create table Recibo
(Recibo_id Counter Primary Key,
Recibo Integer);

17- Criação do Registo de Pedidos

Create table Registo_Pedido
(Reg_id Counter Primary Key,
Efeito_casa Char,
Tipocasa Char,
Edificio Char,
Andar Char,
Flat Char,);

18- Criação do Pedido Colectivo

Create table Pedido_Colectivo
(Pc_Ficha_Insc Char Primary Key,
PC_Denominacao Char,
PC_Tipo Char,
PC_Sede Char,
P_Data_const Date,
P_Reg_commercial Char,

P_Ninc Integer,
P_Bairro_fisc Char,
P_act_Princ Char,
P_Telef Numeric(6),
P_N_insc Integer);

19- Criação da Tabela do Pedido Individual

Create table Pedido_Individual

(P_Ficha_inc Char Primary Key,
P_Nome Char,
P_Dta_Nasc Date,
P_Estado_Civil Char,
P_Prof char,
P_Ent_Pat Char,
P_Nat Char,
P_Nacio Char,
P_Temp_Per Integer,
P_BI Char,
P_Dire Char,
P_emit Date,
P_Arquivo Char,
P_Resid_Act Char,
P_Telef_Serv Numeric(6),
P_tel_resid Numeric(6),
Nacio_Id Integer);

20- Criação da Tabela Representante

Create table Representante

(R_Id Counter Primary Key,
R_Nome Char,
R_Func Char,
R_Dta_Nasc Date,
R_Estado_Civil Char,
R_Nat Char,
R_Nacio Char,
R_BI Char,
R_Dire Char,

R_emit Char,
R_Arquivo Char,
R_N_Siab char,
R_Car_Not char,
R_Data_Absin date,
R_Resid Char,
R_Telef Numeric(7),
PC_ID Integer);

21- criação da Tabela Flat

Create table Flat

(Flat_id Counter Primary Key,

Num_flat Integer,

Num_andar Integer,

Tipo_casa Integer,

Imovel_Id Counter references Imovel12(Imovel_Id),

Area Float,

N_Contrato Char,

Parque Character);

Anexo IX: CRIAÇÃO DE ALGUMAS CONSULTAS USANDO O SQL

1) Seleccionar a topologia de Imóveis ocupados com as respectivas localizações e o total dos imóveis

```
Select Topologia, Localização, Num_entrada, Imovel_Id, Num as Count(Topologia)
From Imovel12
Where Topologia = (Select Tipo_Casa
                   From Registo_Pedido)
```

2) Listar todos apartamentos pela localização e topologia

```
Select Localização, Topologia, Num_Entrada
From Imovel12
Where Imovel12.localização = Localização.localização
Group by Imovel12.localização Desc
```

3) Listar todas flats localizadas em um determinado andar

```
Select Flat_Id, Num_Flat, Num_Andar, Localização
From Flat F, Imovel I, Localização L, Assunto A
Where Localização = (Select I.I_localização
                    From I, A
                    Where I.I_Id = A.Loc_Id or I.Edificio_Id = A.Edi_Id)
Group by Localização
```

4) Listar todos Locadores e a Localização do Imóvel

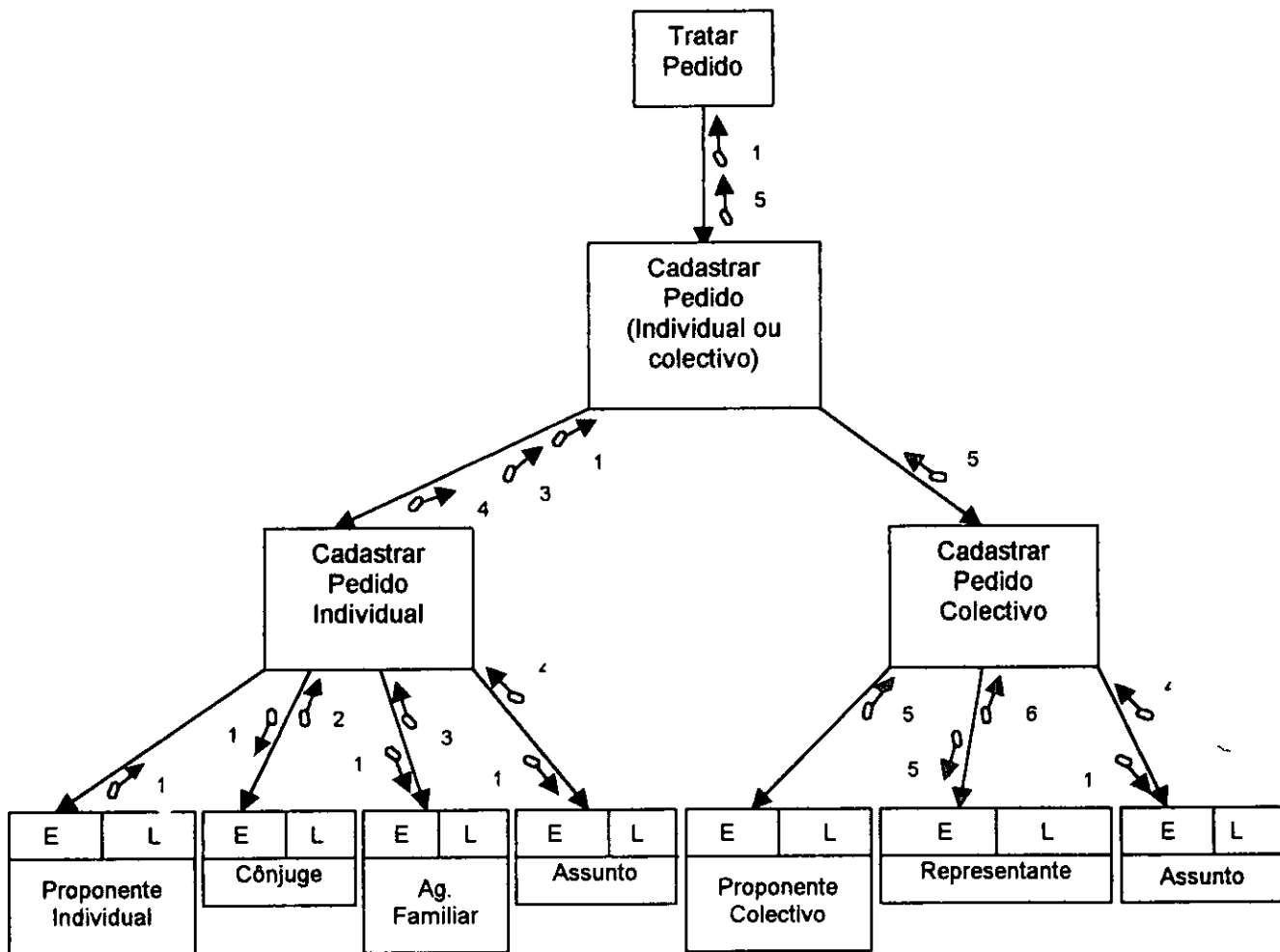
```
Select I_Nome_Prop, Localização
From Imovel I, Localização L, Imovel12 I2, Flat F
Where Localização = (Select I.I_Localização
                    From I, L
                    Where I.I_Localização = L.Localização) and
Nome = (Select I.I_Nome_Prop
        From I, I2
        Where I.I_Nome_Prop = I2.Nome_Proprietario)
```

5) Listar todos Locatários com base no nome, Localização e o tipo de Imóvel onde se encontram a residir

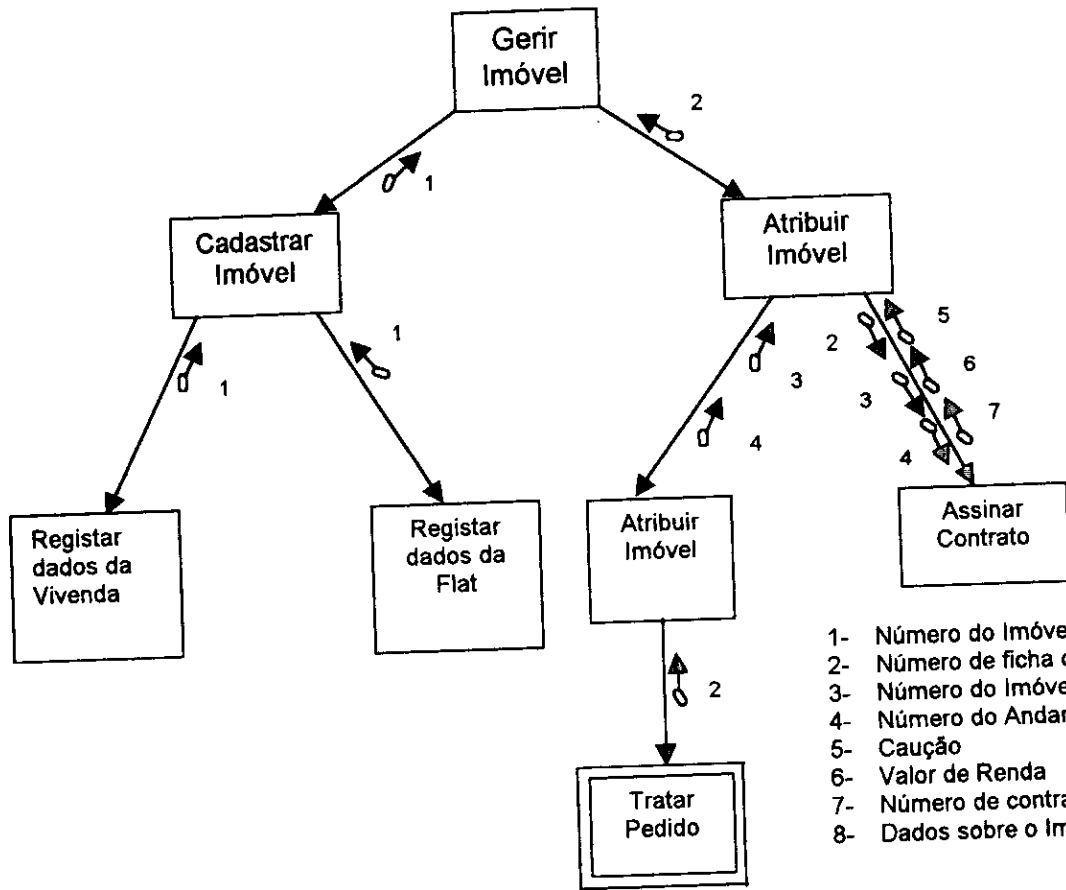
```
Select Nome, Localização, Tipo
From Assunto A, Localização L, Imovel I, Representante R, Tempag T, Pagamento P
Where Nome = (Select R_Nome
              From R, T, P
              Where R.R_Nome = T.Nome and T.Nome = P.Nome) and
Localização = (Select L.Localização
                From A, L, I
                Where I.Localização = (Select I.I_Localização
                                        From A, I
                                        Where I.I_Localização = A.Localização)) and
Tipo = (Select A.Tipo_casa
        From A)
```

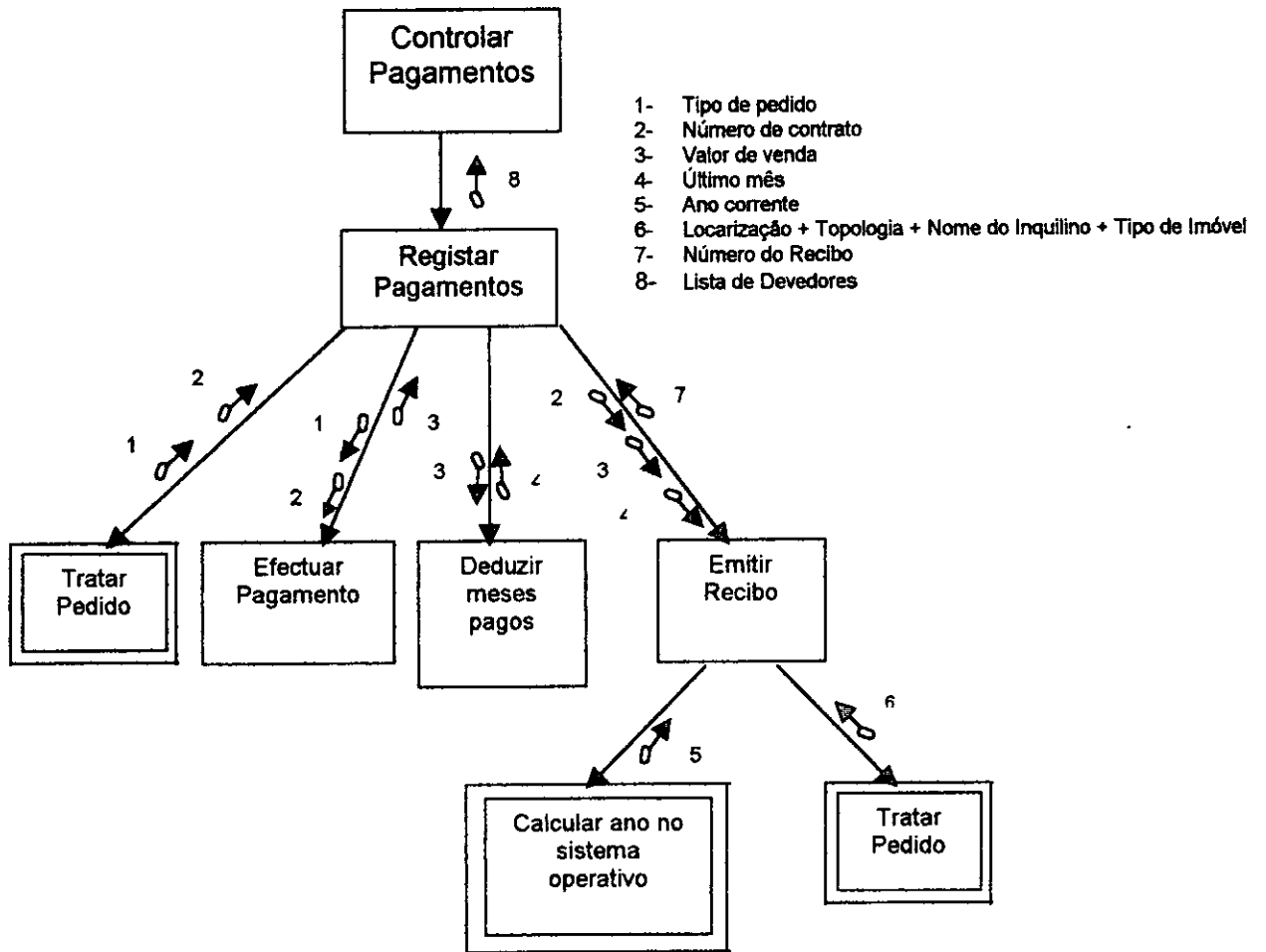
- 1) Listar todos Devedores pelos meses de dívida e o Nome
Select mesrefrencia as Mes, Nome
From Pagamento
Where Data_Pagamento < Now

Anexo X: DIAGRAMA ESTRUTURAL



- 1- Número da Ficha de Inscrição
- 2- Dados do Cônjuge
- 3- Dados do Ag. Familiar
- 4- Dados sobre o assunto a tratar
- 5- Número da ficha de inscrição do colectivo
- 6- Dados do Representante





Como forma de facilitar o entendimento durante a leitura deste trabalho, a maior parte dos termos técnicos usados neste trabalho tem a sua fundamentação e significado (sinónimo), no glossário de termos a seguir:

Anexo XI: GLOSSÁRIO DE TERMOS USADOS

Acesso - habilidade ou meio necessário para consultar ou entrar *dados*, comunicar-se com algum *software* ou usar recursos de um *sistema* informatizado.

Acesso remoto - *capacidade* de *estações de trabalho* em locais distantes terem *acesso* ao *computador* central via canais de *comunicação*.

Acordo de nível de serviço - acordo feito entre *usuários* e *CPD* estabelecendo padrões de atendimento aos *usuários* no que diz respeito a *disponibilidade*, *tempo de resposta* e *performance* dos *sistemas* de informática.

Aderência - execução em conformidade com as regras e normas pré-estabelecidas.

Administração de dados - actividade que engloba projecto, desenvolvimento, operação, segurança, manutenção e utilização de uma *base de dados*. O administrador de *dados* normalmente é o responsável pela definição de regras para *acesso* e armazenamento de *dados*, *integridade*, *segurança*, *performance* e *recuperação* da *base de dados*.

Administrador de sistemas - pessoa responsável por administrar o uso de um *sistema* multi-usuário ou de *comunicações*.

Análise de sistemas - exame de uma actividade, *procedimento*, método, técnica ou negócio para determinar o que deve ser feito e quais as operações necessárias para melhor atender aos objetivos da organização no que diz respeito a informática.

Aplicativo ou aplicação - *programa de computador* que efectua uma ou mais tarefas específicas para o *usuário*. Dependendo do tipo de tarefa para o qual foi projectado, o *aplicativo* pode manipular texto, números, gráficos ou uma combinação de todos estes elementos.

Aquisição - acto de adquirir ou obter, por contrato ou por outro instrumento que o substitua, suprimentos ou serviços para uso da organização através de compra, aluguer ou leasing.

Arquitectura - descrição de todas as actividades funcionais a serem executadas, os elementos de *sistema* necessários e a designação dos níveis de *performance* destes elementos. Uma *arquitectura* também inclui informação a respeito de *tecnologias*, *interfaces* etc...

Arquitetura cliente-servidor - arranjo utilizado em *redes* locais que trata tanto o servidor como as *workstations* individuais como dispositivos programáveis inteligentes, explorando toda a *capacidade* computacional de cada um.

Arquitetura de rede - Estrutura que suporta uma *rede* de *computadores*, incluindo *hardware*, camadas (*layers*) funcionais de *rede*, *interfaces* e *protocolos* utilizados para estabelecer a *comunicação* e garantir a transferência de informações.

Arquivo - uma colecção organizada de informações preparada para ser usada com alguma finalidade e identificada por um nome.

Arquivo Master - principal *arquivo* de referência usado em um *sistema* de computação. Provê informações a serem utilizadas pelo *programa* e pode ser actualizado e mantido para refletir os resultados do *processamento*.

Automação de escritórios - uso de dispositivos eletrônicos e de comunicação, tais como *computadores*, *modems* e fac-símiles (*faxes*), assim como *softwares* associados, para executar, de forma mecânica ou automatizada, funções típicas do ambiente de escritórios.

Autorização - (1) direito de um *usuário* se comunicar com ou utilizar um *sistema* computacional. (2) direito de *acesso* (completo ou restrito) a recursos de informática ou funções.

Backup - (1) pertinente a um *sistema*, dispositivo, *arquivo* ou recurso que pode ser utilizado quando da ocorrência de um defeito ou perda de *dados*. (2) um *arquivo* auxiliar imagem do *arquivo* fonte, utilizado como base de *recuperação* de *dados*.

Banco (ou base) de dados - colecção de *dados* organizados em diversas categorias. Conjunto de todas as informações armazenadas em um *sistema* de computação, fundamental para a organização.

Banco de dados distribuído - *banco de dados* implementado em uma *rede*, cujos *componentes* estão distribuídos em vários *nós* (estações) da *rede*.

Banco de dados relacional - tipo de *base de dados* que armazena informação em tabelas (linhas e colunas de *dados*) e conduz pesquisas usando os *dados* de colunas específicas de uma tabela para encontrar *dados* adicionais em outra tabela.

Bolha - processo ou nó dum diagrama de fluxo de dados

Capacidade - quantidade de informação que um *computador* ou dispositivo a ele conectado pode processar ou armazenar.

Código - *sistema* de símbolos e regras utilizado na representação de informações.

Código de identificação - identifica *usuários*, ou grupo de *usuários*, que podem acessar determinado *arquivo* ou *aplicativo*.

Comando - uma instrução ou sinal de controle para o *computador*.

Compartilhamento de dados - uso de um único *arquivo* de *dados* por mais de uma pessoa ou *computador*. O *compartilhamento de dados* pode ser feito através da transferência física de *arquivos* de um *computador* ou indivíduo a outro ou pode ser feito eletronicamente, possibilitando que dois ou mais *computadores* se comuniquem, acessando *dados* armazenados em um deles.

Compatibilidade - característica de um equipamento de *processamento de dados* através da qual uma *máquina* pode aceitar e processar *dados* preparados por outra *máquina* sem *conversão* ou modificação de *códigos*.

Componente - um único recurso com características definidas.

Comunicação de dados - *transmissão de dados* ou informações de um local ou *componente* a outro.

Concorrência - pertinente à ocorrência de dois ou mais eventos ou actividades dentro de um mesmo intervalo de tempo.

Configuração - (1) composição estrutural de um *sistema*, seja do ponto de vista de *hardware* (equipamentos que compõem o *sistema*), seja do ponto de vista do *software* (*programas* e seus *parâmetros* que executarão as tarefas atribuídas ao *sistema*). (2) um grupo de *máquinas* interconectadas e programadas para operar como um *sistema*.

Controle de acesso - um recurso que permite bloquear *acesso* a *dados* por pessoas não autorizadas.

Controle de erros - método para detectar a presença de erros. Em alguns *sistemas* mais refinados, além de serem detectados, os erros são também corrigidos por operações na recepção dos *dados* ou por retransmissão dos *dados* a partir da sua fonte.

Controles de aplicativo - métodos e *procedimentos* contidos no *software* do *aplicativo* para garantir autoridade para originar *dados*, exatidão dos *dados* de *entrada*, *integridade* do *processamento* e distribuição dos *dados* de *saída*.

Controles de segurança - qualquer acção, dispositivo, *procedimento*, técnica ou outra medida que implemente, efectivamente, segurança de informações e do ambiente computacional.

Cópia backup - cópia de um *arquivo* ou de um conjunto de *dados*, guardada para futura referência ou consulta, caso o *arquivo* ou conjunto de *dados* seja destruído.

CPD - centro de *processamento de dados*.

Dados - representação formalizada de factos, conceitos ou instruções, adequada para comunicação, interpretação ou *processamento*; qualquer representação, tais como caracteres, a qual pode ser associado um significado.

Dados compartilhados - *dados* que podem ser acessados por um ou mais *usuários* ou *programas*.

Dados de teste - conjunto de *dados* escolhidos especificamente para testar se um *sistema* é adequado ou não.

Dados fonte - *dados* originais nos quais o *aplicativo* se baseia.

Data de expiração - data a partir da qual os *dados* contidos num *arquivo* ou meio magnético podem ser libertados ou têm sua protecção expirada.

DBMS (Database management system) - *software* de *computador* usado para armazenar e disponibilizar *dados*. Normalmente, o *software* é limitado ao uso de dispositivos de acesso directo. Os *pacotes DBMS* geralmente utilizam *estrutura hierárquica, de rede ou relacional*.

DC (Diagrama de Contexto) - Representação gráfica que mostra a interação entre o sistema e as entidades externas

Degradação - condição na qual um *sistema* continua a operar, porém a um nível de serviço reduzido. A causa mais comum é a indisponibilidade de alguns *subsistemas*, recursos, equipamentos ou *componentes*.

Depósito de dados - Veja Arquivo

Desenvolvimento de sistemas - processo de definição, *design*, desenvolvimento, *teste* e *implementação* de um novo *sistema*.

DER - *diagrama de entidades* relacionamento.

Design - (1) projecto ou planeamento de um *sistema*. (2) Principal fase no ciclo de vida de um *sistema*, englobando projecto físico e lógico. (3) projeto de *computador*.

DFD - diagrama de fluxo de *dados* que representa o caminho dos *dados* em um *sistema*, define as principais fases do *processamento* e os vários meios utilizados.

Diagrama de entidades - representação gráfica que mostra *entidades* e as *transações* que ocorrem entre elas. Auxilia o *usuário* a ter uma noção global do *sistema* em um único diagrama.

Diagrama lógico - esquema que mostra as conexões entre os circuitos lógicos de um *computador*, especificando as *saidas* esperadas a partir de um conjunto específico de *entradas*.

Dicionário de dados - um repositório centralizado de informações sobre *dados*, como por exemplo, seu significado, relacionamento com outros *dados*, origem, utilização e formato. Auxilia os administradores de *base de dados*, analistas de *sistemas* e programadores no planeamento efectivo, controle e avaliação do *sistema*, armazenamento e utilização dos *dados*.

Disco - na maioria dos *computadores*, o *disco* é o primeiro meio de armazenamento, permanente ou temporário, de *dados*. Pode ser confeccionado com material plástico flexível (floppy disk) ou com metal inflexível (hard disk) coberto de material magnético que pode ser eletronicamente activado para guardar informações na forma digital (ou binária).

Documentação - conjunto de técnicas necessárias para apresentação, organização e comunicação, de forma ordenada, do conhecimento especializado empregado em um *sistema*.

Documento fonte - *registro* original que deverá ser convertido para *linguagem* computacional.

Efectividade - aspecto relacionado com a produção dos resultados desejados. *Efectividade* responde à pergunta, "Estamos fazendo a coisa certa?"

Eficiência - aspecto relacionado com a produção do resultado desejado sem desperdício. *Eficiência* responde à pergunta, "Estamos fazendo de maneira correcta?"

Elemento de dados - (1) um item específico de informação que aparece em um conjunto de *dados*. (2) representação digital de um facto ou evento. "Matéria-prima" necessária para produzir informação e o menor componente de modelagem de *dados*.

Engenharia de informação - conjunto integrado de técnicas para planeamento, análise, *design* e construção de *sistemas* de informação a partir de uma perspectiva institucional.

Engenharia de software - área de *engenharia de informação* cujos objectivos são: produzir *programas* confiáveis atendendo às especificações; produzir *programas* fáceis de serem modificados e mantidos; e fisicamente implementar *programas* da maneira mais prática e eficiente possível.

Entidade - coisa ou classe de coisas independentes que existem no mundo real.

Entrada de dados - método de introduzir *dados* em um *computador*, para *processamento*, normalmente via *terminais* ou *micros*.

Especificação - descrição detalhada. Com relação a *hardware*, as especificações fornecem informação sobre os *componentes*, *capacidades* e *features* especiais da

máquina. No que diz respeito a *software*, as especificações descrevem o ambiente operacional e as funcionalidades do *programa* ou *software*. Em *processamento* de informações, as especificações descrevem os *registos* de *dados*, *programas* e *procedimentos* envolvidos em uma determinada tarefa.

Estação de trabalho - equipamento através do qual um *usuário* ou operador pode enviar ou receber *dados* (comunicar-se com o *computador* central ou servidor).

Estrutura de dados - relacionamento lógico entre unidades de *dados* e descrição de atributos de um dado (tipo, tamanho, etc).

Ferramentas de software - *programas*, utilitários, bibliotecas e outros recursos que podem ser usados para desenvolver *programas* de maneira mais eficiente.

Fita - tira de material impregnado com substância magnética, usada para *entrada*, *saída* ou armazenamento de *dados* em um *computador*.

Gerência de informação - gerência e controle global da informação, incluindo : identificação das necessidades de informação gerencial; garantia de padronização, controle, segurança e *integridade de dados* armazenados e manipulados; *registos* estatísticos e atividades gerenciais; e confidencialidade dos *registos* de informação.

Gerência de performance - processo de análise da *performance* de um *sistema* de computação com intuito de determinar como os recursos estão sendo utilizados e como esta utilização pode ser melhorada. O *planeamento de capacidade* auxilia na previsão das necessidades futuras de recursos.

Gerência de sistemas - função responsável pelo desenvolvimento, modificação, melhorias e *manutenção de sistemas* de informação em uma organização. Como tal, é uma função gerencial relacionada com o *desenvolvimento de sistemas* para satisfazer às necessidades da organização.

Gerenciador de banco de dados - *software* que possibilita a criação e manutenção de uma *base de dados* e a execução de *programas*, utilizando a base.

Hardware - (1) *componentes* físicos (equipamentos) de um *sistema de processamento de dados*. (2) Equipamento mecânico e eletrônico combinado com *software* (*programas*,

instruções etc...) na *implementação* de um *sistema* de *processamento* de informações eletrônicas.

Hierarquia - relacionamento entre um conjunto, seus subconjuntos e elementos.

Host - *computador* primário ou principal em uma *rede* de múltiplos *computadores*.

Normalmente provê serviços de alto nível, tais como *computação*, *acesso a base de dados*, *programas* especiais ou *linguagens de programação* para outros *computadores* na *rede*.

Identificação - processo através do qual um *usuário* ou *sistema* se identifica a um *sistema* de *computador*. Normalmente faz parte do processo de *logon*.

Implementação - fase do ciclo de vida de um *sistema*, a qual corresponde a colocar em produção o *sistema* desenvolvido.

Integridade de dados - (1) qualidade que tem o dado de resistir a uma tentativa de destruição, acidental ou intencional, alteração ou perda. (2) Em um *sistema* de *banco de dados*, para manter a *integridade de dados* deve-se evitar atualização simultânea onde duas *transações* executadas concorrentemente podem interferir uma na outra, produzindo resultados incorretos.

Interface - ponto em que a *conexão* é feita entre dois elementos de forma a trabalharem juntos. Em *sistema* de *computação*, existem *interfaces* de *hardware*, *interfaces* com o *usuário*, etc.

LAN (Local area network) - (1) *sistema* de *comunicação* projetado para *comunicação de dados* em um mesmo prédio. (2) Agrupamento de *computador* e outros *devices* dispersos em uma área relativamente limitada e conectados por *link* de *comunicação*, o qual possibilita que qualquer *device* interaja com outro nesta mesma *rede*.

Layout do relatório - disposição dos *dados* em um relatório.

Linguagem de alto nível - *linguagem* desenvolvida para facilitar a *programação* em relação às *linguagens de baixo nível*. Enquanto as *linguagens de baixo nível* utilizam mnemônicos de *linguagem* assembly, as *linguagens de alto nível* utilizam símbolos e declarações bem próximos do inglês. Além disso, estas *linguagens* são mais compactas,

já que uma declaração em *linguagem de alto nível* pode ser equivalente a várias instruções em *linguagem de máquina*.

Linguagem de baixo nível - *linguagem* dependente da *máquina* e/ou que oferece poucas instruções de controle e tipos de *dados*. Normalmente cada declaração de um *programa* escrito em *linguagem de baixo nível* corresponde a uma instrução de *máquina*. Ex: *linguagem assembly*.

Linguagem de máquina - *linguagem* binária interna para a qual outras *linguagens de programação* mais avançadas devem ser convertidas antes do *computador* processar um *programa*.

Linguagem de programação - *linguagem* usada pelos programadores para escrever *rotinas* de *computador*. As instruções de um *programa* são séries de declarações agrupadas de acordo com a sintaxe (conjunto de regras, como gramática, instruções básicas e *parâmetros*) da *linguagem*.

Linguagem de quarta geração - termo usado para *linguagens* projectadas para interagir com o programador. Tais *linguagens* representam um passo à frente com relação às *linguagens de programação de alto nível* como C, Pascal e COBOL.

Linha - (1) *linha* de informação em um *programa de computador*. (2) *linha* de *comunicação*.

Linha de código - um único *comando*, declaração ou instrução de *programa de computador*. O tamanho de um *programa* em geral é contabilizado por *linhas de código*.

Link - ligação ou *conexão*. Termo normalmente utilizado em *redes de comunicação*.

Listagem - relatório impresso.

Logon - *procedimento* através do qual um *usuário* inicia uma sessão em um *terminal* ou *microcomputador*.

Macro - programadores usam *macros* para economizar tempo, já que estas podem representar um conjunto de instruções repetitivas. Antes da execução de um *programa*, as *macros* são substituídas pelas reais instruções as quais representam.

Mainframe - *computador* central de um *sistema* de computação. Os *computadores mainframe* em geral são compartilhados por múltiplos *usuários* a eles conectados via *terminais*.

Manutenção de programas - processo de *suporte*, depuração e actualização de *programas* em resposta ao feedback dos *usuários* da organização ou do mercado em geral.

Manutenção de sistemas - modificação periódica de informação (*arquivos, programas, etc*) para incorporar mudanças ocorridas durante um determinado período ou solicitadas por fontes externas.

Manutenção preventiva - acções necessárias para manter os equipamentos ou software funcionando correctamente. Engloba actividades de alteração do código comando, limpeza, reposição de peças, etc. Estas actividades geralmente são executadas por representantes do designer ou do fabricante de acordo com o contrato de manutenção dos software ou equipamentos.

Máquina - nome genérico de equipamento que pode armazenar e processar informações numéricas e alfabéticas.

Método de acesso - (1) técnica usada para acessar *dados* em um dispositivo de armazenamento físico. Pode ser sequencial, sequencial indexado ou baseado em *estrutura hierárquica, de rede ou relacional*. (2) método de transferência de *dados* entre a memória central do *computador* e os *devices* de *entrada/saída*.

Modem - MODulador-DEModulador . Dispositivo usado em *sistemas* de *comunicação* para converter sinais digitais de *processamento de dados* em sinais analógicos ou de voz para *transmissão* pela *linha* telefônica. Do outro lado da *linha*, outro *modem* converte os sinais analógicos em sinais digitais.

Objectivos de segurança - são três os *objetivos de segurança* : *integridade* de *software* e de *dados*; *confidencialidade* de *dados*; e *disponibilidade* de *dados* e do *sistema*.

Online - em *processamento online*, o *computador* se comunica directamente com a fonte (origem) e o destino dos *dados* via *teleprocessamento* e o *processamento* é imediato na base de *tempo real*.

Password - senha de *acesso* (seqüência de caracteres) que permite a *entrada* em *sistemas, arquivos* ou *programas* reservados.

Performance - desempenho.

Periférico - adjectivo usado para descrever qualquer unidade ou equipamento que fornece algum meio de comunicação externa à unidade de processamento.

Política de segurança - leis, regras e práticas que regulam como uma organização gerencia, protege e distribui informação.

Procedimento - descrição lógica de um conjunto de acções necessárias para cumprir uma tarefa. Um *procedimento* pode ser implementado tanto na forma manual quanto por meio de *computadores*.

Processador - *computador* capaz de receber, manipular *dados* e apresentar os resultados de operações com esses *dados*.

Processamento - manipulação de *dados* em um *sistema* computacional.

Processamento batch - tipo de *processamento* que se aplica a *dados* que podem ser agrupados em lotes com periodicidade definida e tratados de uma só vez em um *sistema*. Aplica-se a *dados* que não exijam *processamento* imediato na base de *tempo real*.

Processamento de dados - execução de uma seqüência sistemática de operações sobre *dados*. Sinônimo de *processamento* de informações.

Processamento de dados distribuído - *processamento de dados* no qual parte ou todo o *processamento*, armazenamento, funções de controle e funções de *entrada/saída* são executados por *computadores* situados em diferentes locais e conectados entre si através de *rede de comunicação*.

Processamento descentralizado - distribuição dos equipamentos e operação computacionais em mais de um local.

Processamento em tempo real - operações executadas em um *computador* ao mesmo tempo do processo ou actividade física real.

Programação - projecto, escrita e *teste de programas de computador*. Dependendo da filosofia da organização, o processo de *programação* pode envolver ainda *análise de sistemas*.

Protótipo – Modelo físico actual

Rede - série de pontos interconectados por canais de *comunicação*. A *rede* telefônica é composta de *linhas* públicas normalmente utilizadas para chamadas telefônicas comuns. Uma *rede* privada constitui-se de uma *configuração* de canais de *comunicação* reservados para uso de um único cliente.

Rede de comunicação - diversos *computadores* conectados entre si e que podem compartilhar *arquivos* de *dados* e dispositivos *periféricos* comuns.

Redundância - parte da informação completa contida em uma mensagem, a qual pode ser eliminada sem perda da informação essencial, tais como caracteres utilizados apenas para verificação. Este termo também é usado para designar um *computador* que contenha *componentes* importantes de seu *sistema* em duplicidade, por motivos de segurança.

Registo - grupo de factos ou campos de informação relacionados entre si, tratados como unidade.

Regras de acesso - de acordo com o perfil do *usuário* são estabelecidos os tipos de *acesso* autorizados (nenhum, consulta, actualização e remoção).

Requerimentos de usuário - funções a serem executadas pelo *sistema* para atender às expectativas dos *usuários*.

Restart - reiniciar, voltando a um ponto específico em uma *rotina*, normalmente quando ocorre um defeito na *máquina*, com a finalidade de reprocessar a parte onde ocorreu o erro.

Rotina - *procedimento* ou método de solucionar um problema, executar uma tarefa, etc.

Saída de dados - informação transferida a partir de um *computador* para o mundo externo.

Segurança da informação - geralmente considerada como gerência, *procedimentos* e controles necessários para garantir confidencialidade, *integridade* e continuidade de operação de um *sistema* de informação.

Segurança de dados - prevenção contra *acesso* ou uso de *dados* por pessoal não autorizado.

Segurança de rede - técnicas e práticas que limitam o *acesso remoto* a *registros* da organização e protegem as informações transmitidas pela *rede* contra *acesso* não autorizado e alterações.

Segurança física - medidas de segurança para proteger os *sistemas* de informação, prédios e equipamentos contra incêndio e outras intempéries naturais, ataques deliberados, invasão e acidentes. Normalmente é feita através de controles administrativos, alarmes, trancas, crachás de identificação, etc.

Segurança física e lógica de um sistema - todas as salvaguardas tecnológicas e administrativas estabelecidas e aplicadas para protecção de *hardware*, *software* e *dados* contra alterações acidentais, propositais ou destruição.

Sistema - conjunto de processos ou elementos interrelacionados que trabalham juntos para atingir um determinado objectivo comum.

Sistema de processamento de dados - conjunto de equipamentos e *programas* de *processamento de dados* capaz de aceitar informações, processá-las de acordo com um plano e produzir os resultados desejados.

Sistema gerenciador de banco de dados (SGBD) - um *programa* projectado para permitir a criação de *arquivos* especialmente organizados, bem como *entrada*, manipulação, remoção e elaboração de um relatório dos *dados* existentes nesses *arquivos*.

Sistema online - *sistema* em que os *dados* são introduzidos no *computador* directamente pelo ponto de origem (o qual pode ser *remoto* ou *local*) e a *saida* é retornada imediatamente a este mesmo ponto.

Sistema operacional - conjunto de *programas* básicos ou de apoio do *computador* que auxilia, supervisiona e controla a execução dos *programas* do *usuário*.

Software - o conjunto de *programas* e instruções que operam o *computador*. São dois os tipos de *software* de *computador* : *software de sistema*, o qual engloba operações básicas necessárias para operar o *hardware* (por exemplo, *sistema operacional*, utilitários de *comunicação*, monitores de *performance*, editores, compiladores etc.) e *software aplicativo*, o qual executa tarefas específicas para auxiliar aos *usuários* em suas actividades.

Software aplicativo - Há dois tipos de *software* : *software aplicativo* e *software de sistema*. Ambos são escritos em *linguagem* de *computador* e podem ser chamados de *programas*. *Software aplicativo* é todo aquele que executa algo para o *usuário*.

Software de sistema - conjunto de *programas* e *dados* que constituem e se relacionam com o *sistema operacional*.

Stand-alone - adjectivo que descreve um *device* ou operação independente que não necessita de *suporte* de outro *device* ou *sistema*.

Suporte - assistência técnica dada aos clientes pelo fabricante ou fornecedor de um produto de *hardware* ou *software*, podendo também ser executada por analistas e programadores da própria organização para apoio aos *usuários*.

Tecnologia - *aplicação* de ciência e engenharia no desenvolvimento de *máquinas* e *procedimentos* com o objectivo de melhorar as condições humanas de alguma forma.

Telecomunicações - recepção e/ou *transmissão* de informação por meio de telefone, telégrafo, rádio ou outros meios similares.

Tempo de recuperação - intervalo de tempo entre uma falha ou defeito no *sistema* e sua perfeita restauração.

Tempo de resposta - tempo que decorre entre a submissão de uma tarefa a um *sistema* de computação e o retorno dos resultados. Em *comunicação de dados*, *tempo de resposta* engloba tempo de *transmissão* até o *computador*, tempo de *processamento* (incluindo *acesso aos registos de arquivos*) e tempo de *transmissão* de retorno ao *terminal*.

Terminador - expressão usada para referir a entidade externa

Terminal inteligente - *terminal* com recursos próprios de memória, *processador* e *software*, capaz de executar certas funções independentes do *host*.

Time sharing - (1) compartilhamento de tempo. (2) uso de um *sistema* por mais de um indivíduo ao mesmo tempo.

Token - objecto ou mensagem de *dados* estruturados que circula continuamente entre *nós* de uma *token ring* (tipo de *rede em anel*) e descreve o estado atual da *rede*. Antes de enviar qualquer mensagem, o *nó* deve adquirir controle do *token*.

Topologia de rede - arranjo esquemático das ligações e *nós* de uma *rede*.

Transação - qualquer conjunto de atividades.

Transmissão de dados - envio de *dados* de uma parte do *sistema* a outra. Ver também *Comunicação de Dados*.

Trilha de auditoria - (1) *rotinas* específicas programadas nos *sistemas* para fornecerem informações de interesse da auditoria. (2) conjunto cronológico de *registros* que proporcionam evidências do funcionamento do *sistema*. Estes *registros* podem ser utilizados para reconstruir, revisar e examinar *transações* desde a *entrada de dados* até a *saída* dos resultados finais. Podem ser usados também para rastrear o uso do *sistema*, detectar e identificar *usuários* não autorizados.

Trivial - (1) O que é de fácil aprendizagem. (2) Conhecido por todos

Usuário - qualquer pessoa que utiliza *computadores* e *aplicativos*.

Validação - (1) processo de avaliação de um *sistema* ou *componente* durante ou ao final do processo de desenvolvimento para determinar se este satisfaz os requerimentos especificados. (2) processo de verificação dos *dados de entrada* de uma *aplicação*.

Versão - um ciclo de evolução de um *sistema*. Novas versões resultam em alterações do *sistema*.

Vida útil de um sistema - período de tempo que começa com a instalação de um *sistema* e termina quando este deixa de ser necessário para a organização.

Anexo XII: MATERIAL USADO

PARTE TEÓRICA

- + Um computador com processador Intel Pentium, 233 Mhz com 64 MB de RAM e um computador com Processador Intel Pentium II, 560 Mhz com 64 MB de RAM
- + Sistema Operativo Windows 98 e Windows NT Workstation, respectivamente
- + Foram usados como processadores de texto, respectivamente Microsoft Word 2000 e como impressoras HP Laserjet 6L e HP Deskjet 850 C, respectivamente

PARTE PRÁTICA

- + Um computador com processador Intel Pentium, 266 Mhz com 64 MB de RAM e um computador com Processador Intel Pentium II, 560 Mhz com 64 MB de RAM
- + Sistema Operativo Windows 98 e Windows NT Workstation, respectivamente
- + Impressora HP Deskjet 850 C
- + Microsoft Access 97 e 2000 e SQL Server para a implementação do sistema, tendo o Access como linguagem hospedeira. Não se pode esquecer o uso do Microsoft Word e Excel para exportação de reports (Saídas do Modelo Implementado) e o Powerpoint para mostragem dos Slides durante a apresentação.