

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE CIÊNCIAS DA COMPUTAÇÃO**  
(Bacharelado)

**PROTÓTIPO DE UMA FERRAMENTA DE SOFTWARE  
BASEADA NA METODOLOGIA PERT/CPM PARA O  
PLANEJAMENTO DE PROJETO DE DESENVOLVIMENTO  
DE SISTEMAS**

TRABALHO DE CONCLUSÃO DE CURSO SUBMETIDO À UNIVERSIDADE  
REGIONAL DE BLUMENAU PARA A OBTENÇÃO DOS CRÉDITOS NA  
DISCIPLINA COM NOME EQUIVALENTE NO CURSO DE CIÊNCIAS DA  
COMPUTAÇÃO — BACHARELADO

**ANÍSIO KIENEN**

BLUMENAU, JUNHO/2000

2000/1-06

# **PROTÓTIPO DE UMA FERRAMENTA DE SOFTWARE BASEADA NA METODOLOGIA PERT/CPM PARA O PLANEJAMENTO DE PROJETO DE DESENVOLVIMENTO DE SISTEMAS**

**ANÍSIO KIENEN**

ESTE TRABALHO DE CONCLUSÃO DE CURSO, FOI JULGADO ADEQUADO  
PARA OBTENÇÃO DOS CRÉDITOS NA DISCIPLINA DE TRABALHO DE  
CONCLUSÃO DE CURSO OBRIGATÓRIA PARA OBTENÇÃO DO TÍTULO DE:

**BACHAREL EM CIÊNCIAS DA COMPUTAÇÃO**

---

Prof. Antônio Carlos Tavares — Orientador na FURB

---

Prof. José Roque Voltolini da Silva — Coordenador do TCC

**BANCA EXAMINADORA**

---

Prof. Antônio Carlos Tavares

---

Prof. Oscar Dalfovo

---

Prof. Dalton Solano dos Reis

# DEDICATÓRIA

Dedico este trabalho,

a meus filhos Lucas e Mateus, e principalmente a minha esposa Luciane que com carinho compreenderam as dificuldades deste período para conclusão do curso.

## **AGRADECIMENTOS**

Agradeço a todo o corpo docente do Curso de Ciências de Computação da Furb, e em especial ao professor Antônio Carlos Tavares, que como meu orientador, muito me auxiliou para confecção deste trabalho de conclusão de curso.

# SUMÁRIO

DEDICATÓRIA.....	iii
AGRADECIMENTOS .....	iv
SUMÁRIO.....	v
LISTA DE FIGURAS .....	viii
RESUMO .....	x
ABSTRACT .....	xi
1 INTRODUÇÃO.....	1
1.1 MOTIVAÇÃO.....	2
1.2 OBJETIVOS DO TRABALHO .....	2
1.3 ESTRUTURA DO TRABALHO.....	2
2 FUNDAMENTAÇÃO TEÓRICA.....	3
2.1 O MODELO PERT .....	3
2.2 O MODELO CPM.....	3
2.3 O MODELO PERT/CPM.....	3
2.4 CARACTERÍSTICAS DO MODELO PERT/CPM .....	4
2.5 VANTAGENS E DESVANTAGENS DO MODELO PERT/CPM.....	5
2.6 CONCEITOS BÁSICOS.....	6
2.7 ESTIMATIVA DE ATRIBUTOS .....	8
2.8 CONCEITUAÇÃO DAS DATAS .....	8
2.9 FOLGA DE ATIVIDADE .....	9
2.10 FOLGA TOTAL.....	9
2.11 FOLGA LIVRE.....	9
2.12 FOLGA DEPENDENTE.....	10

2.13 FOLGA INDEPENDENTE .....	10
2.14 IMPLANTAÇÃO E CONTROLE .....	10
2.15 CAMINHO CRÍTICO .....	12
2.15.1 Conceituação de Caminho Crítico.....	12
2.15.2 Evento Crítico.....	13
2.15.3 Atividade Crítica .....	13
2.15.4 Determinação de Caminho Crítico .....	13
2.15.5 Conclusões sobre Caminho Crítico .....	14
2.16 ANÁLISE E PROJETO ESTRUTURADO DE SISTEMAS .....	14
2.16.1 Introdução.....	14
2.16.2 Benefícios e Problemas .....	15
2.16.3 Benefícios do Uso da Análise Estruturada de Sistemas .....	16
2.16.4 Problemas Potenciais.....	18
2.16.5 Objetivos.....	19
2.16.6 Diagrama de Fluxos de Dados (DFD).....	20
2.16.7 Dicionário de Dados .....	21
2.17 Ciclo de Vida do Sistema .....	21
2.17.1 Definição do Problema .....	22
2.17.2 O Estudo de Viabilidade.....	23
2.17.3 Análise .....	24
2.17.4 Projeto de Sistema ou Projeto de Alto Nível.....	24
2.17.5 Projeto Detalhado .....	26
2.17.6 Implementação.....	27
2.17.7 Manutenção .....	27
2.18 TRABALHOS CORRELATOS .....	28

2.18.1 Descrição dos Trabalhos.....	28
2.18.2 Avaliação Comparativa .....	28
3 TÉCNICAS E FERRAMENTAS UTILIZADAS PARA O DESENVOLVIMENTO DO PROTÓTIPO .....	31
3.1.1 Ambiente Visual Basic 6.0.....	31
3.1.2 Flow Charting 4.0 .....	33
3.1.3 Erwin/ERX 3.0.....	34
3.1.4 Microsoft Access 97.....	35
4 IMPLEMENTAÇÃO DO PROTÓTIPO .....	36
4.1 ESPECIFICAÇÃO DO SISTEMA .....	36
4.1.1 Operacionalidade da Implementação .....	37
4.1.2 Diagrama do Fluxo de Dados.....	38
4.1.3 Fluxograma de Lógica de Programa .....	40
4.1.4 Modelo Entidade Relacionamento .....	41
4.1.5 Atributos do Dicionário de Dados.....	42
4.1.6 lógica do cálculo do caminho crítico .....	44
4.2 APRESENTAÇÃO DAS TELAS E OPERACIONALIDADE.....	46
5 RESULTADOS .....	57
6 CONCLUSÃO .....	60
6.1 CONSIDERAÇÕES FINAIS .....	60
REFERÊNCIAS BIBLIOGRÁFICAS .....	62

## LISTA DE FIGURAS

Figura 1: Visual Basic 6.0 executando sob o Windows 95.....	32
Figura 2: Flow Charting 4.0 executando sob o Windows 95.....	33
Figura 3: Erwin/ERX 3.0 executando sob o Windows 95.....	34
Figura 4: Microsoft Access 97 executando sob o Windows 95.....	35
Figura 5: Fluxograma da Lógica de Programa.....	38
Figura 6: Diagrama de Contexto do Protótipo.....	39
Figura 7: Diagrama de Fluxo de Dados do Protótipo.....	40
Figura 8: Modelo Entidade Relacionamento do Protótipo.....	41
Figura 9: Array EventosAtividades.....	44
Figura 10: Array de Pilha.....	44
Figura 11: Menu Principal do Protótipo.....	46
Figura 12: Manutenção do Cadastro de Projetos do Protótipo.....	47
Figura 13: Manutenção das Atividades de Análise do Protótipo.....	48
Figura 14: Manutenção dos Recursos do Protótipo.....	49
Figura 15: Gráfico PERT/CPM do Protótipo.....	50
Figura 16: Relacionamento Recursos a Atividades do Protótipo.....	51
Figura 17: Opções de Relatórios do Protótipo.....	52
Figura 18: Relatório do Cadastro de Projetos do Protótipo.....	53
Figura 19: Relatório do Cadastro de Atividades de Análise do Protótipo.....	54



Figura 20: Relatório do Cadastro de Recursos do Protótipo.....	54
Figura 21: Relatório de Recursos por Atividade do Protótipo.....	55
Figura 22: Relatório das Durações das Atividades do Protótipo.....	56
Figura 23: Relatório do Custo Total por Projeto do Protótipo.....	56
Figura 24: Gráfico da rede para Sistema de Transportadoras.....	58
Figura 25: Gráfico da rede para Sistema de Transportadoras calculado.....	59

## **RESUMO**

Este trabalho visa demonstrar a utilização da técnica da rede PERT/CPM voltada para o planejamento e desenvolvimento de sistemas aplicativos que sigam a metodologia de Análise Estruturada de Sistemas, disponibilizando uma ferramenta onde o desenvolvedor de software entre com as informações relevantes ao novo projeto de software, tais como, processos a serem desenvolvidos com seus respectivos tempos de execução, custo de cada processo, recursos envolvidos. A partir destas informações, o sistema calcula o caminho crítico e o custo do sistema. A utilização da metodologia de análise estruturada de sistemas se deve ao fato da ferramenta auxiliar aos desenvolvedores de softwares que utilizam esta metodologia.

## **ABSTRACT**

This work seeks demonstrate the use of the net technique PERT/CPM for to do the planning and development of systems applications that follow the methodology of Structured Analysis of Systems, availabling a tool where the software developer enters with the important information to the new software project, such as, processes be developed with its respective times of execution, cost of each process, involved resources. With these information, the tool calculates the critical path and the cost of the system. The use of the methodology of structured analysis of systems is due to the fact of the tool auxiliary the developers of softwares that use this methodology.

# 1 INTRODUÇÃO

O desenvolvimento de sistemas informatizados consiste de várias atividades distribuídas ao longo de várias etapas, as quais apresentam interdependência entre si, sendo que uma falha em qualquer das etapas pode comprometer a eficiência e a eficácia de um sistema de informação. Com o desenvolvimento de softwares aplicativos cada vez mais complexos e sempre com maior rapidez, surge a necessidade de planejamento e controle desse desenvolvimento. Desta forma, torna-se imprescindível que os desenvolvedores consigam fazer bons sistemas no menor tempo possível e com um custo reduzido. Com isto, ferramentas de planejamento e controle para o desenvolvimento de softwares aplicativos são também cada vez mais imprescindíveis e indispensáveis, possibilitando aos desenvolvedores respostas rápidas sobre seus próprios sistemas, auxiliando no controle, planejamento e redução de custos do desenvolvimento.

A utilização do método PERT/CPM, apesar dos sucessos alcançados em grandes projetos, ainda encontra uma certa resistência. Resistência esta, originada pela idéia errônea, mas infelizmente bastante difundida, de que estas técnicas não estão no nível de empresas médias ou pequenas, nem de projetos de importância modesta [GEN1966].

Grande parte do sucesso do método se deve, sem dúvida, ao fato de, ao contrário de outros métodos de pesquisa e planejamento operacional, não exigir uma formação matemática muito profunda.

Seus princípios, quase todos bastante intuitivos, são de fácil assimilação e seu valor para o planejamento de projetos, sejam eles grandes ou pequenos, é inestimável.

Neste trabalho será apresentada uma visão geral sobre a metodologia PERT/CPM, suas características e sua utilização para planejamento de projetos. Serão apresentadas também as etapas de elaboração de projetos para construção de software seguindo a metodologia de Análise Estruturada de Sistemas.

## **1.1 MOTIVAÇÃO**

A grande motivação para a elaboração deste trabalho foi e é a dificuldade que tem os desenvolvedores de softwares em estimar o tempo de duração para confecção de qualquer software aplicativo, também como estimar o custo deste desenvolvimento.

## **1.2 OBJETIVOS DO TRABALHO**

O objetivo principal do trabalho é desenvolver uma ferramenta que faça o cálculo de caminho crítico e cálculo de custo, para auxiliar o planejamento no desenvolvimento de sistemas de software.

## **1.3 ESTRUTURA DO TRABALHO**

O texto está organizado em 5 (cinco) capítulos, distribuídos conforme segue:

Primeiramente é apresentada a introdução, a motivação, os objetivos e a organização do trabalho.

O segundo capítulo apresenta a fundamentação teórica.

O terceiro capítulo apresenta as técnicas e ferramentas utilizadas para o desenvolvimento do protótipo.

O quarto capítulo apresenta a implementação do protótipo de uma ferramenta de software baseada na metodologia PERT/CPM para o planejamento de projeto de desenvolvimento de sistemas.

O quinto capítulo apresenta os resultados obtidos, aplicando a ferramenta no desenvolvimento de sistema fictício.

O sexto capítulo apresenta a conclusão e considerações gerais.

## **2 FUNDAMENTAÇÃO TEÓRICA**

Segundo [LOE1999], as etapas de um projeto são o planejamento, a programação e o controle. Durante o planejamento responde-se à pergunta “o que fazer?”; assim são detalhadas as atividades e estabelecida a relação de interdependência entre as atividades. Durante a programação responde-se à pergunta “quando fazer?”; assim é estabelecido um cronograma para execução dessas atividades determinando data para início e término das atividades. Durante o controle responde-se à pergunta “como fazer?”; deve ocorrer durante a execução do projeto e estabelecer medidas para que a programação estabelecida seja cumprida sem atrasos.

### **2.1 O MODELO PERT**

O sistema Program Evaluation and Review Technique (PERT), que segundo [GEN1966] foi o “termo inicialmente empregado para caracterizar o tempo probabilístico como atributo de cálculo”. Esta técnica foi desenvolvida pela empresa de consultoria Booz-Allen and Hamilton, para a marinha dos Estados Unidos da América no programa Polaris, onde cerca de 10.000 empresas (entre contratantes e sub-contratantes) tinham que ser coordenadas e necessitavam comunicar-se numa mesma linguagem. O método reduziu a duração do projeto de 5 para 3 anos.

### **2.2 O MODELO CPM**

O sistema Critical Path Method (CPM), que segundo o mesmo autor, foi o “termo inicialmente empregado para caracterizar o tempo determinístico como atributo de cálculo”. Desenvolvido pela Dupont e Univac, mais ou menos na mesma época do PERT (1957/1958). A diferença entre os dois é irrelevante, de pequenos detalhes. No primeiro, PERT, predominam os chamados esquemas probabilísticos, e no segundo, CPM, os chamados esquemas determinísticos, não havendo maiores vantagens práticas em considerá-los como dois sistemas diferentes. Hoje em dia, tais sistemas acham-se integrados sob a denominação PERT/CPM.

### **2.3 O MODELO PERT/CPM**

Fundamentalmente PERT/CPM são técnicas úteis de gestão de projetos nas funções básicas: planejamento, programação e controle.

A fase de planejamento de qualquer projeto envolve uma lista de operações ou tarefas que devem ser executadas para o projeto considerar-se realizado. As necessidades de material, equipamento e mão-de-obra, são também determinadas nesta fase, sendo também feitas as estimativas dos custos e duração das várias tarefas.

Programação, por outro lado, corresponde à colocação das tarefas do projeto na ordem temporal em que devem ser executadas. Calculam-se as necessidades em material e mão-de-obra em cada fase de produção juntamente com o tempo esperado de execução de cada tarefa.

Controle, geralmente considerado como uma “função subjacente de gestão” começa com a revisão da diferença no programa e o andamento real durante a execução do projeto. A análise e a correção dessa diferença correspondem ao aspecto básico do controle [BEL1974].

O método PERT/CPM consiste em figurar uma operação (obra, projeto, desenvolvimento de sistema, etc) numa rede onde estão representadas as atividades que são necessárias à sua execução. Estas são representadas por setas que indicam as relações de correspondência entre os eventos ou acontecimentos importantes entre a operação, de modo que o conjunto mostre a seqüência entre todas as atividades do empreendimento que devem ser executadas. Junto de cada seta anota-se o nome e a duração e o custo estimado da atividade que ela representa [CUK1978].

## **2.4 CARACTERÍSTICAS DO MODELO PERT/CPM**

Segundo [CUK1978] , o modelo PERT/CPM, técnica de redes, é um conjunto de processos e técnicas para planejamento, programação e controle de um empreendimento ou operação, ou projeto, tendo como característica fundamental a indicação, dentre as várias seqüências operacionais, daquela que possui duração máxima, além de permitir a indicação de graus de prioridade relativos, demonstrando distribuição de recursos e interdependência entre as várias ações necessárias ao desenvolvimento do projeto.

As principais características do Modelo PERT/CPM são:

- a) abordagem de um projeto de um ponto de vista sistêmico, já que:
  - fornece uma visão de totalidade do projeto (graças à necessidade de se estabelecer a “interdependência entre as várias ações necessárias ao desenvolvimento do projeto”);

- ressalta as entradas (diretrizes e recursos), o processo (o desenvolvimento das ações de acordo com as relações de correspondência entre elas) e as saídas (resultado final desejado);
  - conduz à montagem de um esquema de “feed-back”, através do estabelecimento de um sistema de comunicações, que abasteça e permita à administração decidir em função dos dados e informações que convergem de diversos setores, sobre o andamento do projeto.
- b) dá ênfase aos objetivos;
  - c) visa a otimização da chamada “regra dos 5 P’s” (política, performance, prazo, preço e perigo);
  - d) é uma disciplina inter-disciplinar e de comunicação (bastando analisar as próprias origens da técnica).

## **2.5 VANTAGENS E DESVANTAGENS DO MODELO PERT/CPM**

As principais vantagens do método PERT/CPM segundo [HIR1978] são:

- a) é um instrumento disciplinado para planejar, mostrando a seqüência e a interdependência das atividades;
- b) é um método detalhado, permitindo mais segurança no acompanhamento e nas decisões;
- c) as alterações do plano inicial do empreendimento são feitas com maior facilidade e o emprego de sub-redes, ou redes parciais, permitem organizar melhor o trabalho;
- d) obriga o planejamento total do empreendimento, facilitando a tomada de decisões;
- e) os relatórios de controle, embora sucintos, são objetivos e detalhados, não permitindo informações dúbias e sem sentido;
- f) indica as áreas críticas e pontos de estrangulamento com bastante antecedência, inclusive antes do início da execução do projeto.

As principais desvantagens ou dificuldades, segundo o mesmo autor, do método PERT/CPM são:



- a) resistência do pessoal, em qualquer nível da organização, para aplicação de qualquer método de controle;
- b) custo fixo adicional, na administração do projeto, muitas vezes, incompatível com o valor do empreendimento;
- c) necessidade de maior coordenação entre a equipe planejadora, as turmas de execução e a administração da empresa, para que os dados estatísticos e contábeis fiquem disponíveis, em tempo útil, aos planejadores.

## 2.6 CONCEITOS BÁSICOS

O quadro que se segue, segundo [BEL1974], sintetiza o que inicialmente necessitamos para desenvolver o assunto.

NOME	CONCEITO	REPRESENTAÇÃO GRÁFICA	EXEMPLO
Projeto	Conjunto de ações e processos, envolvendo recursos humanos, materiais, financeiros, etc., organizados para a realização de um objetivo, concretamente definido, a partir de uma situação inicial conhecida ou convencionada, chamada, Operação Empreendimento.	Rede, diagrama ou grafo.	Lançamento de uma rede de abastecimento de águas
Atividade	É a identificação de uma etapa de um projeto que consome tempo e recursos, estabelecida em nível compatível às necessidades e possibilidades de sua mensuração	Em flechas ou setas	Serviço de escavação
Evento	Início ou término de uma ou mais atividades	Nó	Início (ou término) do serviço de escavação
Atributo	É toda característica quantitativa específica de uma atividade, indicando qualquer dos recursos necessários a sua efetivação (exemplo: tempo, material, mão-de-obra, equipamento, etc)	Em flechas ou setas	3 semanas para executar a escavação

Os iniciantes, com esta técnica geralmente costumam perguntar “Qual o nível de detalhe que se deve ir na decomposição de um projeto?”, “Quantas atividades justificam o uso de PERT/CPM?”

À segunda responde-se: não é o número de atividades que justifica o uso do modelo, mas sim, a complexidade do problema ou projeto que se está querendo estudar.

À primeira pergunta costuma-se responder que o nível de decomposição deve ser compatível com as nossas necessidades e/ou possibilidades (tal como foi conceituada a atividade) em relação ao projeto. Qualquer dos dois extremos não é aconselhado ou seja: um nível muito geral de decomposição pode não revelar certos detalhes importantes, bem como, um nível muito exagerado de detalhes pode exigir muito tempo de estudo sem que se tenha meios de medir ou mesmo avaliar a execução destes mesmos detalhes.

O que se aconselha é tentar decompor dentro do nível de detalhe que se julgue compatível, sem exageros; parte-se então para o estudo da seqüenciação o que exigirá o conhecimento dos recursos disponíveis e conseqüentemente traçado da rede. Por acasão da elaboração desta, certos detalhes não revelados anteriormente, nela serão evidenciados e vice-versa, ou seja, certas atividades antes consideradas, quando do traçado da rede poderão se mostrar desnecessárias.

Finalmente, na estimativa de duração de cada atividade pode-se ainda aumentar ou diminuir o nível de decomposição, já que certas atividades, quando da estimativa das durações respectivas poderão ter valores muito pequenos em relação às demais e, portanto, sem condições de serem avaliadas e controladas. Por outro lado, haverão atividades para as quais não tinha-se sentido necessidade e/ou possibilidade de uma maior decomposição, mas que na estimativa das durações respectivas poderão revelar uma necessidade de maior diferenciação para melhor avaliação e controle.

Em conclusão, o nível de detalhamento e a estimativa de duração das atividades é um processo de aproximação sucessiva que cessa antes da execução do projeto, mas com o início do desenvolvimento do projeto, serve de controle e acompanhamento.

## 2.7 ESTIMATIVA DE ATRIBUTOS

Segundo [GEN1966], uma vez traçado o diagrama de flechas, isto é, planejada a realização de empreendimento, passa-se à fase da programação. Para isso, ter-se-á que estimar a duração de execução de cada atividade em função dos recursos e características inerentes ao projeto. As atividades poderão receber, segundo o sistema PERT/CPM tradicionais, uma sistemática distinta de obtenção.

No CPM admite-se que a estimativa de duração seja feita com uma precisão aceitável, correspondendo este valor à duração mais provável e, portanto, com um grau de incerteza desprezível, correspondendo ao chamado esquema determinístico.

No sistema PERT considera-se difícil uma precisão aceitável para a duração de uma atividade, adotando-se, então, mais de uma estimativa de tempo para cada atividade que nos permita deduzir a probabilidade de ocorrências de datas; reduz-se este grau de incerteza com a introdução de mais de uma estimativa de duração para cada atividade, corresponde ao chamado esquema probabilístico [GEN1966].

## 2.8 CONCEITUAÇÃO DAS DATAS

Segundo [GEN1966], uma vez estimados os valores das durações das atividades, passa-se ao cálculo das datas ao longo do projeto.

A data mais cedo do evento fim da rede define a duração do projeto.

A data mais cedo de qualquer evento deve ser interpretada como uma data possível de ocorrência do mesmo, sendo determinada em função de condições internas ao projeto que amarram o seu desenvolvimento até aquele evento.

Mas a data em que o projeto deve acabar nem sempre coincide com aquela que se pode realizar, nem sempre o que é possível é permissível.

Generalizando, em qualquer evento, a data correspondente a esta permissão, ao que deve ser feito, é o que se chama de data mais tarde do evento.

Segundo o autor “Data mais tarde do evento (abreviatura Tt), é a maior data em que o evento pode ocorrer, sem atrasar a conclusão do empreendimento.

A data mais tarde de qualquer evento deve ser interpretada como uma data permissível de ocorrência do mesmo, sendo o reflexo de condições externas ao projeto.

## **2.9 FOLGA DE ATIVIDADE**

Folga de atividade, segundo [CUK1978], é a disponibilidade de tempo que a atividade pode utilizar, além de sua duração prevista, sem prejudicar a duração pré-estabelecida para o projeto

Ainda segundo o autor “ folga do evento é a disponibilidade de tempo medida pela diferença entre as datas mais tarde e mais cedo de um evento”.

### **2.10 FOLGA TOTAL**

Folga Total segundo [CUK1978], é a disponibilidade de tempo que a atividade pode utilizar de forma que, iniciada na data mais cedo do seu evento início, tenha sua conclusão na data mais tarde do seu evento fim.

### **2.11 FOLGA LIVRE**

Folga Livre segundo [CUK1978], é a disponibilidade de tempo, além da duração prevista, que a atividade pode utilizar, supondo-se que comece na data mais cedo de seu evento início e termine na data mais cedo de seu evento fim. A fim de deduzirmos a folga livre, parte-se também do princípio de que a atividade teve o seu início na data mais cedo do evento início, isto é, não consumiu a folga do evento início.

A folga livre será igual ao tempo que restou, após a sua conclusão, até a data mais cedo do seu evento fim.

## 2.12 FOLGA DEPENDENTE

Folga Dependente segundo [CUK1978], é a disponibilidade de tempo que a atividade pode utilizar de forma que, iniciada na data mais tarde do seu evento início, tenha sua conclusão na data mais tarde do seu fim.

## 2.13 FOLGA INDEPENDENTE

Folga Independente segundo [CUK1978], é a disponibilidade de tempo que a atividade pode utilizar de forma que, iniciada na data mais tarde de seu evento início, tenha sua conclusão na data mais cedo do seu evento fim.

## 2.14 IMPLANTAÇÃO E CONTROLE

Segundo [FED1967], após definido o projeto, procura-se detalhá-lo ou subdividi-lo até um nível que se necessite ou que seja possível controlar as suas atividades, para o que pode-se recorrer à utilização da Estrutura Analítica. Levantadas as atividades, elabora-se o quadro de seqüenciação relativa, estima-se os atributos ( o que praticamente é feito em paralelo ao estudo de seqüência, já que este está amarrado aos recursos disponíveis), e constrói-se a rede correspondente.

Esta é a etapa que se caracteriza como de planejamento onde se defini “o que” e “como” fazer, sendo a rede, como já se disse a “imagem síntese deste planejamento”.

A partir daí, começa-se a etapa de programação onde calcula-se as datas, folgas, graus de prioridade e obtêm-se um primeiro cronograma diretamente da rede, que representa a “imagem síntese da programação” (“quando” e “com que” fazer em função dos recursos alocados para o projeto).

Em seguida, tornam-se necessário a realização de análises de estudos compatíveis com projeto tais como: alocação de recursos considerados problemas para o mesmo, dimensionamento dos riscos associados a datas de eventos, aceleração do projeto, etc..

Neste momento está-se em condições de implantar o projeto através da montagem de uma estrutura de execução e de um sistema de controle correspondente.

Nesta fase reside o sucesso ou o fracasso de todo o projeto. O estabelecimento de uma boa coordenação e de um sistema de informações eficiente, que garante rapidez, precisão e frequência no que se refere ao acompanhamento do projeto, permitirá a tomada de decisão e conseqüente ações corretivas garantindo a manutenção que se está executando em conformidade com que foi formulado.

Problemas, imprevistos e falhas poderão ocorrer ao longo da execução, acarretando uma necessidade de reprogramação (o que alterará os cálculos feitos e conseqüentemente o cronograma) ou de replanejamento face à dimensão das ocorrências, (o que alterará a definição do projeto e por sua vez todas as demais fases desde o estabelecimento de um novo detalhamento de atividades, ou seja, de uma nova rede).

Assim sendo, com uma “central de coordenação de controle”, para onde e de onde fluirão todas as informações relativas ao projeto, o centro (ou centros) com execução deverão entrar em constante contato, para as necessárias informações relativas à execução do projeto.

A central recebe as informações, analisa, refaz cálculos, emite relatórios para diversos níveis, enviando ao(s) centro(s) de execução, instruções, chamadas “agendas de execução”, de acordo com as “informações de andamento” recebidas desse(s) mesmo(s) centro(s).

Segundo [NBR1994], existem os seguintes conceitos referentes a controle de projetos:

- a) órgão de coordenação e controle é aquele ao qual convergirão as informações relativas à execução e empreendimento para sua análise e eventual replanejamento:
- agenda de execução é o documento elaborado pelo órgão de coordenação e controle; encaminhado ao local de execução, contendo instruções e especificações relativas ao desenvolvimento do projeto durante um determinado período;
  - boletim de controle é o documento elaborado no local de execução do projeto, e encaminhado ao órgão de coordenação e controle, contendo informações relativas ao desenvolvimento durante um período pré-estabelecido;
  - intervalo de controle é o intervalo de tempo em que é feita a verificação do andamento do projeto;

- unidade de apuração é o conjunto de atividades ou segmentos afins do projeto agregados com finalidade específica de análise, avaliação e replanejamento e programação.

Complementando, não existe uma padronização quanto a que informações enviar ou solicitar, nem do intervalo de frequência a adotar para estas transações; isto é função do ambiente em que o sistema vai se desenvolver, dos recursos disponíveis, dos valores das durações, enfim, de uma gama de características inerentes ao trabalho e, portanto, impossíveis de se estabelecerem padrões de referência [NBR1994].

O estabelecimento de tal sistemática fica à critério do responsável pelo projeto, de acordo com a sua sensibilidade e conhecimento das condições de execução do projeto.

## **2.15 CAMINHO CRÍTICO**

### **2.15.1 CONCEITUAÇÃO DE CAMINHO CRÍTICO**

Segundo [BEL1974], caminho crítico é todo o caminho de maior duração em um projeto, compondo-se, embora não necessariamente, de uma seqüência de atividades críticas. Nesta norma define-se ainda caminho semi-crítico aquele cuja duração mais se aproxima daquela do caminho crítico.

Assim sendo, partindo-se do evento início do projeto até o seu evento fim, o caminho crítico define o caminho de maior duração ou seja, qualquer outro trajeto que ligue o evento início ao evento fim tem duração menor do que a do caminho crítico.

A data mais cedo de evento fim, define a duração do projeto e a duração do caminho crítico.

Caso não existam folgas nos eventos do caminho crítico, qualquer atraso em uma de suas atividades pode acarretar um atraso do mesmo valor, em tempo, igual ao do atraso da atividade. Uma diminuição na duração do projeto será possível com a redução correspondente de umas das atividades do caminho crítico. Desta forma, justifica-se a determinação deste caminho na rede, onde se exerce mais controle e concentração de recursos para evitar atrasos ou até mesmo para acelerar um projeto [BEL1974].

## 2.15.2 EVENTO CRÍTICO

Segundo [BEL1974], evento crítico é aquele que apresenta a menor folga de evento. É o evento que tem a menor disponibilidade de tempo que a atividade pode utilizar sem comprometer a sua duração prevista.

## 2.15.3 ATIVIDADE CRÍTICA

Atividade crítica, ainda segundo [BEL1974], é a atividade compreendida entre eventos críticos e correspondendo à maior duração, entre os mesmos, limitada pelos valores de datas mais cedo de início e fim. É a atividade de menor folga em um projeto.

## 2.15.4 DETERMINAÇÃO DE CAMINHO CRÍTICO

Segundo [BEL1974], a determinação de caminho crítico se consegue com as seguintes etapas:

- a) identificação dos eventos críticos, conforme anteriormente conceituado, são todos aqueles que tiveram por folga, a menor folga de evento da rede. Esta menor folga de evento, numa situação inicial, aparece no evento final do projeto, já que é ali que se tem calculada a duração possível do projeto e imposta a duração máxima permissível ao mesmo. Assim sendo, todos os eventos que tiverem uma folga igual a deste último, serão eventos críticos.
- b) identificação das atividades críticas. Para uma atividade ser crítica, ela deverá, pela definição apresentada, satisfazer a duas condições:
  - condição necessária, estar entre eventos críticos;
  - condição suficiente, ter a maior duração entre estes eventos, o que vale dizer, a diferença entre as duas datas deve ser igual à duração desta atividade.
- c) o caminho crítico, fica evidenciado com a determinação das atividades críticas. No conceito de caminho crítico diz-se que este caminho “compõe-se, embora, não necessariamente, de uma seqüência de atividades críticas”, o que pode se explicar notando que, se ao longo do projeto certos eventos (não críticos) tiverem sua data de ocorrência alterada (atrasada), não por atividades internas ao projeto mas devido a causas externas, como por exemplo ocorrências de chuvas, atrasos na chegada de



técnicos necessários ao início de certas atividades etc., teremos, evidentemente, em função da dimensão deste atraso, uma alteração do caminho crítico. Uma outra forma de apresentar o caminho crítico é dizer que ao longo do mesmo a folga de evento é constante e igual a menor folga de evento da rede (que só será nula no caso limite).

### **2.15.5 CONCLUSÕES SOBRE CAMINHO CRÍTICO**

As conclusões que podem ser tiradas sobre o cálculo de caminho crítico segundo [BEL1974] são:

- a) um projeto pode ter mais de um caminho crítico, podendo toda a rede, no caso limite, ser crítica;
- b) as folgas dos eventos das atividades fora do caminho crítico são sempre maiores do que as folgas dos eventos críticos;
- c) a folga dos eventos do caminho crítico é constante e igual à menor folga de evento da rede;
- d) qualquer atraso em uma atividade do caminho crítico (caso não hajam folgas nos eventos críticos) acarretará um atraso no projeto;
- e) a data mais cedo do evento fim é a soma das durações das atividades do caminho crítico.

## **2.16 ANÁLISE E PROJETO ESTRUTURADO DE SISTEMAS**

### **2.16.1 INTRODUÇÃO**

A Análise Estruturada é uma metodologia que tem a finalidade de auxiliar na análise e definição de sistemas. O conceito fundamental é a construção de um modelo do sistema utilizando técnicas gráficas. A metodologia envolve a construção top-down (de cima para baixo) do sistema por refinamentos sucessivos [GAN1983].

O desenvolvimento top-down é um método empolgante. Com planejamento e execução adequados, ele pode evitar muitos problemas que tem atormentado os desenvolvedores durante anos no desenvolvimento de sistemas. No estudo feito por Walston e

Felix sobre produtividade na IBM, projetos que não utilizaram o desenvolvimento top-down obtiveram uma produtividade média de 196 linhas de código por homem-mês; projetos que o usaram, obtiveram, em média 321 linhas por homem-mês, ou seja, um melhoramento de 60%. Talvez o aspecto mais importante, pela visão dos analistas, é o aumento da participação dos usuários à medida que o sistema é desenvolvido, tanto em termos dos comentários e ajuda valiosa que eles podem dar, enquanto o sistema evolui, como também em termos da própria confiança deles de que um sistema certo está sendo construído e mostrando um progresso tangível [GAN1983].

Segundo [DAV1987], há uma certa dose de arte envolvida na análise de sistemas. Embora a abordagem puramente criativa de projetar à medida em que se progride possa funcionar para projetos pequenos ou relativamente simples, ela pode ser um desastre para um sistema grande e complexo.

## **2.16.2 BENEFÍCIOS E PROBLEMAS**

O uso de codificação estruturada ou desenvolvimento top-down torna possível quantificar alguns benefícios resultantes: melhor produtividade em linhas de codificação por dia, uso mais apropriado do tempo de teste e assim por diante [GAN1983].

Com projeto estruturado, os benefícios também são reais porém mais difíceis de quantificar. Pode-se pedir a um grupo de programadores de manutenção para avaliar a alterabilidade de um sistema que utilize projeto estruturado comparando-o a um que não use projeto estruturado; na teoria poder-se-ia medir o custo de manutenção de um grupo desses sistemas e compará-lo com o de um grupo de sistemas não-estruturados. Um estudo não publicado sugere que a modificação de um sistema que utilize projeto estruturado chega a ser sete vezes mais fácil e mais barato do que a projetos tradicionais. Outros estudos tendem a confirmar este resultado dramático, mas são anedóticos; Bill Inmon comentou a respeito de um sistema com projeto estruturado: “A maior modificação precisou de quatro dias, e o que veio logo a seguir menos de um dia” [GAN1983].

Os benefícios trazidos pela Análise Estruturada de Sistemas são ainda mais difíceis de quantificar. Realmente, sob certos aspectos, se o trabalho de análise fosse realizado de forma perfeita, o único resultado seria ausência de problemas [GAN1983] !

### 2.16.3 BENEFÍCIOS DO USO DA ANÁLISE ESTRUTURADA DE SISTEMAS

Segundo [GAN1983], os principais benefícios da utilização da análise estruturada de sistemas são:

- a) os usuários obtêm uma idéia mais clara do sistema proposto pelos diagramas de fluxos de dados lógicos do que a obtida através de narrativas de fluxogramas de sistemas físicos. Porque eles o entendem, tomam uma atitude mais positiva quanto ao projeto. A probabilidade de construir um sistema que, embora excelente, não atenda as necessidades do usuário é bem reduzida;
- b) a apresentação do sistema em termos de fluxo de dados lógicos mostra mal-entendidos e pontos controversos muito mais cedo do que normalmente é o caso. De modo geral é que com especificação sob forma narrativa, todo mundo desenha em sua mente o seu próprio diagrama de fluxo de dados. A partir do momento em que esses fluxos de dados mentais são colocados no papel e tornados públicos, muitas diferenças entre as idéias particulares das pessoas sobre o sistema tornam-se óbvias. Por exemplo, uma narrativa por escrito poderia especificar que um arquivo Histórico-de-Pedidos será criado com detalhes de todos os pedidos processados. A forma desta declaração poderia ser perfeitamente aceitável para os usuários. Entretanto, quando eles percorrerem o diagrama de fluxo de dados, tornar-se-á bastante claro, a partir do trecho do diagrama onde o fluxo de dados histórico-de-pedidos se origina, o que será realmente capturado no Histórico-de-Pedidos. Serão somente aqueles pedidos que foram realmente despachados? Ou serão todos os pedidos, tenham ou não sido remetidos, incluindo os rejeitados por falta de crédito? Esta clara exposição de incertezas através do diagrama de fluxo de dados significa que mais discussão se realiza em torno do fluxo de dados do que em torno da típica narrativa de especificação. É, entretanto, uma discussão muito produtiva. Modificações feitas em um pedaço de papel são muito mais baratas do que as que são feitas no código;
- c) as interfaces entre o novo sistema e os sistemas automatizados e/ou de rotinas manuais, já existentes, são mostradas de modo bem claro pelo diagrama de fluxo de dados, e a necessidade de documentar os detalhes dos fluxos de dados no

dicionário de dados força uma definição clara dessas interfaces nos estágios iniciais. Algumas organizações especificam que um diagrama de fluxo de dados deve ser desenhado, não somente para o sistema em estudo, mas também para cada um dos outros sistemas, automatizados ou os de rotina manuais, com a qual se comunica. Esse exercício, embora envolva uma quantidade considerável de trabalho, mostra duplicação de função e aponta para oportunidades de se incluir funções de rotina no novo sistema;

- d) o uso do modelo lógico suprime uma certa quantidade de duplicação de esforço que ocorre nos projetos tradicionais. No passado, o comum era o representante do usuário e o analista de sistemas trabalharem juntos na produção de uma narrativa da especificação dos sistema. Quando esta narrativa fosse aceita, o grupo de projeto/programação a reanalisaria efetivamente, realizando muito do trabalho de definição dos dados e da lógica novamente. Note-se que as ferramentas de análise estruturada de sistemas são igualmente valiosas para usuários e técnicos. Do momento em que os usuários concordam com o fluxo de dados, com a análise de acesso imediato e com a lógica da política, esses documentos podem ser usados diretamente como entradas para o projeto físico. Esta vantagem é particularmente visível para o projetista de banco de dados, que antes precisava pesquisar a narrativa de especificação de forma a extrair os itens de dados e requisitos de acesso;
- e) o uso do dicionário de dados para guardar os itens de glossário do projeto economiza tempo ao resolver rapidamente os casos em que as pessoas chamam as mesmas coisas por diferentes nomes ou onde um mesmo termo representa coisas diferentes dependendo do contexto. Esse emprego de palavras passa despercebido pelas pessoas da comunidade de usuários, pois fazem parte da vida diária dessas pessoas, mas podem causar bastante surpresa aos analistas.

De forma resumida, os benefícios reduzem-se a apenas dois:

- a) mostrar claramente o que se construirá para que todos possam ter certeza de que se está construindo o sistema certo;

- b) desenvolver as alternativas e os detalhes com o mínimo possível de perda de tempo.

Estas duas afirmações parecem quase triviais, porém gasta-se bastante tempo e dinheiro porque não tem-se sido capaz de fazê-las.

#### **2.16.4 PROBLEMAS POTENCIAIS**

É claro, segundo [GAN1983], que os benefícios trazidos pelas novas ferramentas analíticas não são gratuitos; existem alguns custos e problemas em potencial associados à sua introdução. Em parte são problemas associados a qualquer mudança; em parte são resultado da maior formalidade e da disciplina das ferramentas lógicas:

- a) orientação dos usuários e treinamento dos analistas são necessários. Como a introdução da Análise Estruturada de Sistemas é percebida como “mudança de regras”, todos devem ser bem esclarecidos quanto às novas regras e à maneira como elas melhoram a forma de trabalho;
- b) o esforço, a formalidade e o grau de detalhe necessários, especialmente na construção do dicionário de dados, muitas vezes sofrem resistência. Esta é uma questão de investir esforço durante a análise, para que o projeto seja mais suave posteriormente, de fazer as coisas certas da primeira vez, para não ser necessário fazê-las novamente. Em parte a resistência advém dos usuários, porque projetos anteriores não necessitaram de uma definição tão clara dos termos e significados; a equipe de projeto é que sofria para que os usuários pudessem prosseguir com sua velha e confusa metodologia. A resistência pode surgir por causa das tentativas de definir, muito cedo, detalhes excessivos; uma decisão equilibrada precisa ser tomada quanto ao nível de documentação detalhada, especialmente das funções do sistema atual que não serão excluídas no novo sistema. Mas o fato necessita ser encarado; para que a análise seja boa é necessário esforço tanto por parte dos usuários como dos analistas. A compensação é que é um esforço mais produtivo;
- c) tem havido uma certa preocupação por parte dos programadores de que ao obterem especificações detalhadas da lógica no português estruturado, acabarão “retirando todo o prazer da programação, tornando-os meros codificadores”. Esses receios se dissipam quando o projetista e os programadores percebem que a Análise

Estruturada de Sistemas lhes dá um trabalho muito mais extenso, partindo do modelo lógico. Nessa discussão sobre vantagens/desvantagens de projeto, deve ficar claro quanto trabalho resta depois que o modelo lógico está terminado. Acha-se que o problema surge porque, até que os programadores tenham alguma experiência em lidar com modelos lógicos, eles não valorizam a diferença entre a lógica externa de validação de crédito, expressa, por exemplo, no português estruturado, e a lógica interna do módulo físico. A lógica externa é dada à eles pelo analista; o projeto da lógica interna é toda deles;

- d) por último, em algumas organizações, surge uma pergunta após a primeira experiência positiva com Análise Estruturada de Sistema. “Que pena que não se tinha essas ferramentas para o projeto XYZ; está-se trabalhando nele há seis meses e ainda não se terminou a análise. Será que se pode utilizar as ferramentas de Análise Estruturada de Sistemas no projeto XYZ agora, depois de tê-lo iniciado há algum tempo?” A lição da experiência parece indicar que vale a pena usar-se as técnicas estruturadas para a análise, projeto e desenvolvimento, começando em qualquer ponto de um projeto. Mesmo que os usuários do projeto XYZ tenham a impressão de que irão desperdiçar os últimos seis meses de trabalho e começar de novo (o que não é verdade), eles se convencerão rapidamente quando virem os melhoramentos no andamento das coisas. Como o analista de uma companhia de seguros disse, “Quando você tiver visto o que esses novos métodos podem fazer, detestará ter de continuar a usar os métodos antigos”.

### **2.16.5 OBJETIVOS**

A Análise Estruturada de Sistemas, segundo [GAN1983], tem por objetivos:

- a) dotar o desenvolvimento do software de uma linguagem comum, facilitando o relacionamento entre analistas e usuários;
- b) propiciar o uso de ferramentas que possibilitem de forma gráfica, uma visão dos fluxos de informações e as transformações existentes entre esses fluxos.

### 2.16.6 DIAGRAMA DE FLUXOS DE DADOS (DFD)

Diagrama de Fluxo de Dados segundo [GAN1983], é uma representação em rede dos processos (funções) do sistema e dos dados que ligam esses processos. Ele mostra o que o sistema faz e não como é feito. É a ferramenta de demonstração central da análise estruturada. O Diagrama de Fluxo de Dados é um retrato de um sistema (ou parte de um). Ele mostra Particionamento, Interfaces e Movimento dos Dados.

Um Diagrama de Fluxo de Dados é uma ferramenta para análise top-down. Podemos utilizar diagramas de fluxos de dados para fornecer tanto visões globais (Diagrama de Contexto), com visões detalhadas de um sistema (Diagrama de Fluxo de Dados nível 0). O que é colocado dentro de uma bolha em um diagrama de fluxo de dados, pode ser mostrado em detalhes em outra bolha no digrama de fluxo de dados nível mais inferior. Este processo de definir um sistema em uma maneira top-down é denominado de nivelamento de um diagrama de fluxo de dados [GAN1983].

Um Diagrama de Fluxo de Dados, segundo o mesmo autor, é composto de quatro elementos que são:

- a) Fluxo de Dados;
- b) Processo;
- c) Depósito de Dados;
- d) Terminal (Entidade Externa).

Fluxo de Dados é um conjunto de setas através do qual fluem pacotes de composições conhecidas (informações em movimento). Utiliza-se para sua representação uma flecha, indicando a direção do fluxo.

Processo é a transformação de fluxo de dados de informações em fluxos de dados de resultados. Utiliza-se normalmente um círculo ou bolha para sua representação, contendo um nome para sua identificação.

Depósito de Dados é um repositório de informações (informações estáticas). A simbologia utilizada para sua representação normalmente é um retângulo aberto ou duas linha paralelas na horizontal.

Terminal ou Entidade Externa é a origem ou destino de informações do sistema. Pode ser um outro sistema, pessoa ou órgão, situação fora do contexto de estudo o qual dá origem a uma rede ou recebe fluxo de dados que são parte do estudo. É representado por um retângulo ou quadrado.

### **2.16.7 DICIONÁRIO DE DADOS**

Segundo [GAN1983], o dicionário de dados é um repositório de dados sobre os dados do sistema. De maneira figurativa, podemos dizer que trata-se de um banco de dados sobre os dados do sistema. Ele deve conter definições dos elementos que tornam do diagrama de fluxo de dados precisos, os quais são:

- a) Processo;
- b) Fluxo de Dados;
- c) Depósito de Dados;
- d) Entidade Externa;
- e) Elementos de Dados;
- f) Chaves primárias e estrangeiras das entidades.

### **2.17 CICLO DE VIDA DO SISTEMA**

Segundo a Norma NBR 1994 [NBR1994], no capítulo 1.5 cita que a mesma não prescreve um modelo específico de ciclo de vida ou método de desenvolvimento de software. As partes envolvidas com esta Norma são responsáveis pela seleção de um modelo de ciclo de vida para o projeto de software, e pelo mapeamento dos processos, atividades e tarefas desta Norma dentro deste modelo. As partes envolvidas são também responsáveis pela seleção e aplicação dos métodos de desenvolvimento de software e pela execução das atividades e tarefas adequadas ao projeto de software.

A análise e projeto estruturado de sistemas está vinculado ao ciclo de vida do sistema. Na medida em que o sistema progride de uma idéia até sua implementação, terá de passar por cada uma dessas etapas. Quando for utilizada uma abordagem estruturada, o analista de sistemas terá de progredir de uma fase para outra de modo muito cuidadoso e metódico,



obedecendo a alguns critérios bem definidos de saída, para cada uma das etapas. O ciclo de vida de um sistema, segundo [DAV1987], pode seguir as seguintes etapas:

- a) definição do problema;
- b) estudo de viabilidade;
- c) análise;
- d) projeto do sistema;
- e) projeto detalhado;
- f) implementação;
- g) manutenção.

### **2.17.1 DEFINIÇÃO DO PROBLEMA**

Qual é o problema? Esta é a pergunta-chave que terá que ser respondida durante a definição do problema; não faz muito sentido tentar resolver um problema que não se sabe qual é. Embora a necessidade de definição do problema possa parecer óbvia, este talvez seja o passo mais freqüente omitido em todo o processo de análise e projeto de sistemas.

Qual é a fonte da definição do problema? Obviamente, alguém terá de reconhecer que ele existe. Muitas vezes o usuário vai se defrontar com dificuldades, solicitando ajuda. Talvez a administração identifique uma área de desempenho fraco dentro da função do usuário; freqüentemente o analista de sistemas apontará o problema. As discussões iniciais a ele relativas muitas vezes são bastante informais. Contudo, eventualmente essas discussões atingem o ponto em que o usuário, a administração, e o analista de sistemas concordam: “Sim, realmente há um problema”.

Se o problema for considerado significativo, a administração e o usuário poderão querer que o analista o estude. Uma vez designado o analista, o processo sofre uma mudança sutil. A designação do analista pressupõe o comprometimento de recursos; as discussões informais repentinamente tornam-se um projeto definido.

A primeira responsabilidade do analista de sistemas é a de preparar uma declaração escrita dos objetivos e a delimitação do problema. Baseado em entrevistas com a

administração e o usuário, o analista redige uma sucinta descrição de sua compreensão do problema, e o revisa com os dois grupos, em uma reunião conjunta usuário/administração. As pessoas respondem a perguntas escritas, solicitando esclarecimentos; corrigem erros ou incompreensões óbvias. É por isso que é tão importante uma colocação clara dos objetivos.

A definição do problema deve ser breve, durando um único dia, ou até menos. A intenção é definir os objetivos e a delimitação do sistema proposto. Podem ocorrer falhas na comunicação, sendo essencial que o usuário, a administração e o analista de sistemas concordem sobre um sentido genérico bem no início do projeto. Uma definição do problema, mal-entendida, praticamente garante que o sistema não vai resolvê-lo.

### **2.17.2 O ESTUDO DE VIABILIDADE**

Estudo de viabilidade, basicamente, é uma versão resumida e de alto nível do processo como um todo, direcionada a responder a várias perguntas. Qual é o problema? Há uma solução viável para o problema? Será que vale a pena resolver o problema? O estudo de viabilidade deve ser relativamente curto; a tarefa não é resolver o problema, mas obter um sentido de sua delimitação. O usuário será solicitado a reagir ao estudo de viabilidade; de que outra maneira poder-se-ia ter certeza de se estar atacando o problema certo? A administração está vitalmente vinculada aos resultados, pois eles serão solicitados a comprometer recursos e pessoal com base no estudo de viabilidade.

A declaração de interesses e objetivos, preparada durante a definição do problema, em geral é um tanto vaga. Essencialmente, o analista promete investigar (não resolver) um problema amplamente definido. Durante o estudo de viabilidade a definição do problema é enfocada de forma mais nítida. São fixados objetivos específicos para o sistema, e os aspectos do problema que serão excluídos do sistema são claramente identificados. Consequentemente, o analista deve ser capaz de calcular os custos e benefícios do sistema com maior precisão. Uma análise de custo/benefício do sistema proposto é uma parte importante do estudo.

O estudo de viabilidade termina com uma apresentação formal ao usuário e à administração. Esta apresentação demarca um ponto de decisão crítico na vida do projeto. Muitos projetos morrerão exatamente aqui; apenas aqueles que prometem um rendimento significativo em relação ao investimento devem ser continuados. Pressupondo que a

aprovação da administração será concedida, o estudo de viabilidade apresenta um modelo excelente de compreensão, por parte do analista de sistemas, do problema (comparável aos esboços do engenheiro), e fornece um sentido claro de direção para o desenvolvimento posterior do sistema.

### **2.17.3 ANÁLISE**

A análise é um processo lógico. O objetivo desta fase não é efetivamente resolver o problema, mas determinar exatamente o que precisa ser feito para o resolver. O usuário sabe o que tem de ser feito, mas não como fazê-lo. Durante a análise, a analista de sistemas trabalha com o usuário para desenvolver um modelo lógico do sistema.

Muitos analista de sistema possuem experiência técnica. A tentação de muitas pessoas com treinamento técnico é seguir rápido demais para o projeto de programa, de concretizá-lo prematuramente. É preciso resistir a essa tentação. O objetivo, não pode-se esquecer, é resolver o problema do usuário. O usuário conhece o problema, e é a fonte principal de informações, neste estágio, para o analista. Se este começar a falar sobre detalhes de programação, o usuário poderá ficar perdido, e incapaz de contribuir, fazendo que o analista desenvolva um sistema que não resolve o problema.

Como o analista pode evitar de passar prematuramente para a fase física? A metodologia estruturada pode ajudar. Com a abordagem estruturada é preciso atender critérios específicos de saída para cada etapa do processo. O objetivo básico do estágio de análise é desenvolver um modelo lógico de sistema, utilizando ferramentas como diagrama de fluxo de dados, um dicionário elementar de dados, e descrições gerais dos algoritmos relevantes. Este modelo lógico está sujeito a revisão tanto pelo usuário como pela administração que terão de concordar que o modelo, de fato, reflete o que deve ser feito para resolver o problema.

### **2.17.4 PROJETO DE SISTEMA OU PROJETO DE ALTO NÍVEL**

Uma vez terminado o estágio de análise, o analista de sistemas sabe o que terá de ser feito. O passo seguinte é determinar, em forma de amplo contorno, como o problema poderá ser resolvido. Durante o projeto de sistema, está-se começando a se deslocar do lógico para o físico.

Diversas soluções alternativas podem ser consideradas. Por exemplo, um determinado sistema poderá ser implementado através do computador ou por meios manuais. Se for utilizado o computador, o sistema poderá ser tanto de processamento em lotes como interativo. Pode-se utilizar os arquivos de dados tradicionais; outra opção é a base de dados. É importante que o analista evite simplesmente escolher uma implementação do sistema e passar para o projeto de programa. Mais uma vez, a utilização de uma metodologia estruturada pode ajudar.

A questão a ser respondida durante o projeto de sistema é: “Em geral, como o problema deve ser resolvido?” A resposta a esta pergunta é crítica tanto para o usuário como para o programador; desta forma, os critérios de saída para esta fase terão de ser direcionados para os dois grupos. A administração está particularmente preocupada com a orientação futura do sistema. Até esse ponto, o projeto envolveu o tempo de alguns analistas de sistemas, e o custo tem sido limitado. O projeto detalhado bem poderá envolver, além desses analistas, mais alguns outros, de forma que o estágio de projeto detalhado levará mais tempo; assim, os custos começam a se acelerar. Ao deslocar-se para a implementação, programadores, operadores, redatores técnicos, e tempo de computador terão que ser comprometidos, e o custo acumulado aumenta assustadoramente. Além de tudo isso, após a implementação há um comprometimento com a manutenção, a longo prazo e potencialmente cara. Antes de a administração poder apoiar o sistema, precisa ter uma idéia de quanto isso irá custar. É imprescindível ter alguma idéia da provável proporção custo/benefício.

O custo de um sistema acelera-se ao se deslocar através do processo de desenvolvimento; este é outro argumento para utilizar uma abordagem estruturada e cuidadosa. A abordagem metódica da análise e projeto de sistemas estruturado aumenta a probabilidade de que os erros significativos serão encontrados cedo.

Como critérios de saída do estágio de projeto de sistema, muitas vezes se solicita ao analista que delineie diversas soluções alternativas para o problema, incluindo no mínimo:

- a) uma solução de baixo custo que faça apenas o trabalho desejado;
- b) uma solução de custo intermediário que faça bem o trabalho, e seja conveniente para o usuário. Este sistema poderá incluir diversos recursos que o usuário não

solicitou especificamente, mas que o analista, baseado em sua experiência e seus conhecimentos, sabe que virão a ser de valia;

- c) um sistema sofisticado “Mercedes” de alto custo, com tudo que o usuário possa a vir a desejar.

Como apoio a cada alternativa, deverá haver um diagrama de fluxo do sistema ou outra descrição dos sistema, e uma estimativa dos custos e benefícios.

A fase de projeto de sistema termina com uma opção. Talvez a administração ou o usuário decidam que os benefícios a serem obtidos deste sistema não valham seu custo, ou que o setor de programação não é capaz de programar o sistema. Desta forma, o projeto pára. Se valer a pena apoiar o sistema, uma das alternativas será escolhida. A descrição desta alternativa feita pelo analista de sistemas será utilizada como um modelo de alto nível para desenvolver o sistema físico.

### **2.17.5 PROJETO DETALHADO**

Para esta fase a pergunta básica que terá que ser respondida é: “Especificamente, como o sistema deve ser implementado?” durante o projeto detalhado, o analista de sistemas terá de desenvolver (de preferência com o auxílio do departamento de programação) um conjunto de especificações para cada programa. Essas especificações devem conter detalhes suficientes para permitir a codificação. Se o sistema pedir hardware novo, a especificação para o hardware terá que ser redigida numa forma aceitável para o departamento de compras. Geralmente, durante o projeto detalhado a analista de sistemas terá de definir cada componente do sistema ao nível de detalhamento que se fizer necessário para a etapa de implementação.

Como foi utilizada uma abordagem estruturada à análise e projeto de sistemas, um dos objetivos é desenvolver especificações de programa que facilitem a escrita de código estruturado. Neste trabalho, prepara-se um quadro hierárquico e um conjunto de quadros de entrada/processo/saída para cada programa do sistema. Serão utilizados algoritmos utilizando-se pseudocódigos ou português estruturado.

As especificações desenvolvidas durante o projeto detalhado são análogas às plantas do engenheiro. Dadas essas especificações detalhadas, é possível gerar estimativas de custo e programas de implementação altamente precisos, dois fatores extremamente importantes para a administração. Os programadores se utilizarão das especificações para orientar suas ações. Finalmente, o usuário deve ser capaz de ver uma imagem do sistema nas especificações.

### **2.17.6 IMPLEMENTAÇÃO**

Durante o estágio de implementação, o sistema é fisicamente criado. Os programas necessários são codificados, corrigidos e documentados. O novo hardware é escolhido, pedido e instalado. Em muitas organizações, o analista de sistemas também é o programador; neste trabalho, contudo, pressupõe-se que a análise e a programação sejam feitas por equipes diferentes, e sem considerar-se os detalhes da implementação do sistema.

Entretanto, há certas funções que o analista terá de realizar à medida que os programas estiverem sendo compostos e o hardware estiver sendo pedido e instalado. Procedimentos de segurança e de auditoria provavelmente serão necessários. O plano de teste terá de ser estabelecido. É fácil esquecer os procedimentos e o plano de teste, mas sem esses elementos não há sistema. O estágio de implementação normalmente se encerra com um teste formal do sistema envolvendo todos os componentes e procedimentos.

### **2.17.7 MANUTENÇÃO**

Após a implementação, o sistema entra em estágio de manutenção. O objetivo da manutenção é manter o sistema funcionando a um nível aceitável. Ocasionalmente, defeitos no programa passarão pelo teste de sistema sem terem sido aferidos; a correção de tais erros é uma função de manutenção. Com maior frequência, os parâmetros e os algoritmos utilizados para desenvolver os programas originais modificar-se-ão, significando que os programas terão de ser atualizados. A manutenção do hardware é uma exigência óbvia. Até os procedimentos modificam-se.

O que o analista de sistemas faz durante o estágio de manutenção? Normalmente, muito pouca coisa. Na realidade, é provável que o analista que projeta um sistema esteja trabalhando em outro ao começar o estágio de manutenção. Contudo, o analista exerce influência considerável sobre a manutenção. Talvez a mudança seja a única constante no

campo da computação; o sistema bem projetado terá de prever e dar margem a mudança. Projetar no sentido da mudança é responsabilidade do analista, e a abordagem estruturada à análise e ao projeto de sistema procura apoiar este objetivo.

## **2.18 TRABALHOS CORRELATOS**

### **2.18.1 DESCRIÇÃO DOS TRABALHOS**

Cabe aqui mencionar a série de modelos de caminhos críticos ou variantes com nomes patenteados e específicos, oriundos de tentativas de aprimorar ou adaptar o PERT/CPM a atividades específicas ou usos especializados. A diferença entre eles em relação ao PERT/CPM é pequena [HIR1978].

Alguns exemplos:

- a) Cost Planing and Appraisal (CPA);
- b) Critical Path Scheduling (CPS);
- c) Critical Path Technique (CPT);
- d) Project Control System (PCS) (programa da IBM) ;
- e) Project Management System (PMS) (programa da IBM);
- f) Project Analysis and Control System (PROJACS) (programa da IBM);
- g) MS PROJECT (programa da Microsoft );
- h) MS PROJECT PLUS (programa da Microsoft);
- i) MS SUPER PROJECT (programa da Microsoft).

### **2.18.2 AVALIAÇÃO COMPARATIVA**

A avaliação mais aprofundada sobre os softwares supra citados foi feita sobre o MS Project da Microsoft, por não ter-se conseguido nenhum dos outros softwares.

O MS Project, da Microsoft, é um dos mais populares entre os programas atualmente existentes. Sua versão inicial rodava no ambiente DOS e foi lançada em 1985. A principal característica deste programa sempre foi a facilidade de uso. Assim, apesar de nem sempre ter

oferecido o “estado-da-arte” no assunto gerência de projetos, sua simplicidade de uso atraía simpatizantes, mesmo em empresas que também utilizavam outros softwares mais poderosos com a mesma finalidade [PRA1998].

Segundo [PRA1998], dentre os inúmeros recursos que o programa oferece, no que se refere a aspectos de gerência de projetos, destacam-se:

a) tempo

- baseia-se no modelo Diagrama de Precedências. Portanto, as atividades do projeto são criadas na forma de blocos, não na forma de setas;
- utiliza o Gráfico de Gant ou de barras como ferramenta básica a ser utilizada durante a entrada de dados;
- aceita relações de precedência tipo Fim-Início, Início-Início, Fim-Fim e Início-Fim;
- permite tarefas concorrentes: inclusão de tarefas que ocorrem de forma repetitiva. Por exemplo: em um projeto pode-se ter reuniões todas as segundas-feiras;
- permite estabelecer níveis hierárquicos para as tarefas, isto é, admite existência de tarefas de resumo;
- permite uso de subprojetos;
- permite uso de “datas programadas” para as atividades;
- permite uso do modelo probabilístico.

b) recursos, os quais são ligados diretamente às atividades;

c) custos, os quais são ligados diretamente às atividades na forma de custos fixos ou de custos dos recursos.

O MS Project, por ser desenvolvido no ambiente Windows, possui todas as vantagens que este ambiente oferece (tais como copiar, colar, duplicar, etc). Visualmente, ele se apresenta como um conjunto de planilhas e de gráficos.



Outro trabalho analisado, foi o trabalho de monografia de Marcelo Loffi “Protótipo de Sistemas de Controle da Utilização de Recursos para Produção nas Indústrias de Manufatura”, de 1999/1, orientado pelo professor Oscar Dalfovo, porque no referido trabalho utiliza-se a técnica PERT/CPM. A utilização da técnica PERT/CPM nesta monografia serve de auxílio para um sistema especialista de controle da programação da produção para as indústrias de manufatura.

A diferença dos trabalhos supra citados a este, é que este fará o desenho gráfico de uma rede PERT e o cálculo do caminho crítico, conforme os eventos, atividades, recursos, tempo das atividades, custo dos recursos que o desenvolvedor de software achar necessários para o desenvolvimento de seu sistema.

### **3 TÉCNICAS E FERRAMENTAS UTILIZADAS PARA O DESENVOLVIMENTO DO PROTÓTIPO**

Para desenvolver este trabalho foram utilizadas as seguintes tecnologias:

- a) ambiente Visual Basic 6.0 como linguagem para o desenvolvimento;
- b) Flow4 para definição do fluxo de dados;
- c) ERwin/ERX 3.0 para confeccionar o modelo Entidade Relacionamento;
- d) Microsoft Access 97 como banco de dados;
- e) metodologia de análise estruturada de sistemas,
- f) técnica PERT/CPM.

#### **3.1.1 AMBIENTE VISUAL BASIC 6.0**

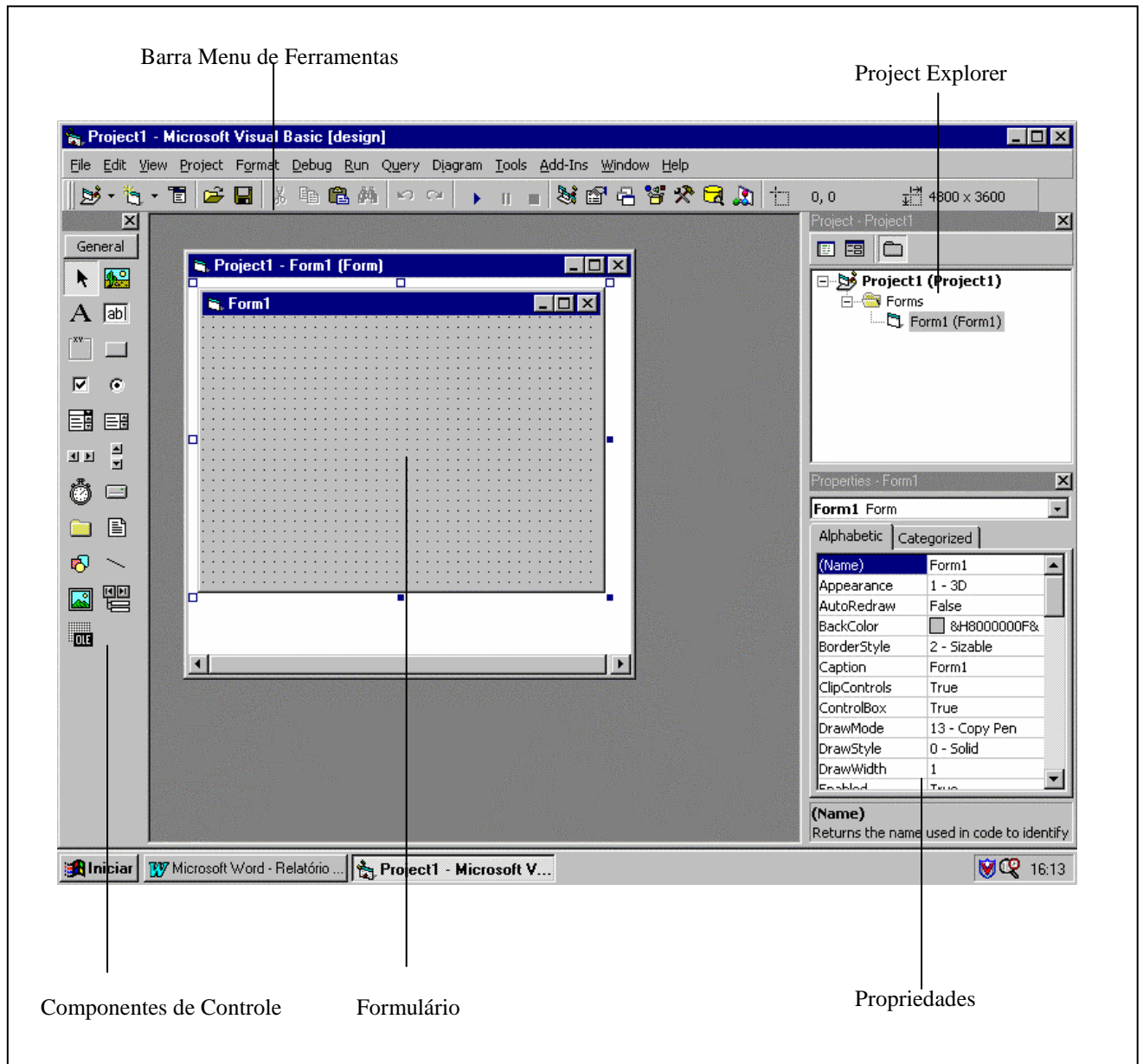
O Visual Basic é um padrão de linguagem de programação baseada em objetos. Ele é considerado uma ferramenta RAD (rapid application development) que permite desenvolver aplicativos visuais rapidamente com o uso de componentes gráficos pré-definidos. Ele trabalha com eventos que dão início à alguma rotina de trabalho, ou seja, o programa fica parado até que um evento ocorra.

O ambiente para desenvolvimento do Visual Basic possui vários elementos, os principais são:

- a) barra de menu e ferramentas;
- b) janelas do Visual Basic (formulário);
- c) toolbox (controle), que são os objetos que podem ser inseridos num formulário;
- d) janela de propriedades, onde define-se as características de cada objeto;
- e) Project Explorer, que mostra todos os componentes do projeto.

A figura 1 apresenta o Visual Basic sendo executado sob o Windows, apontando cada um dos elementos principais acima citados.

Figura 1: Visual Basic 6.0 executando sob o Windows 95.



Fonte: [MIC1999]

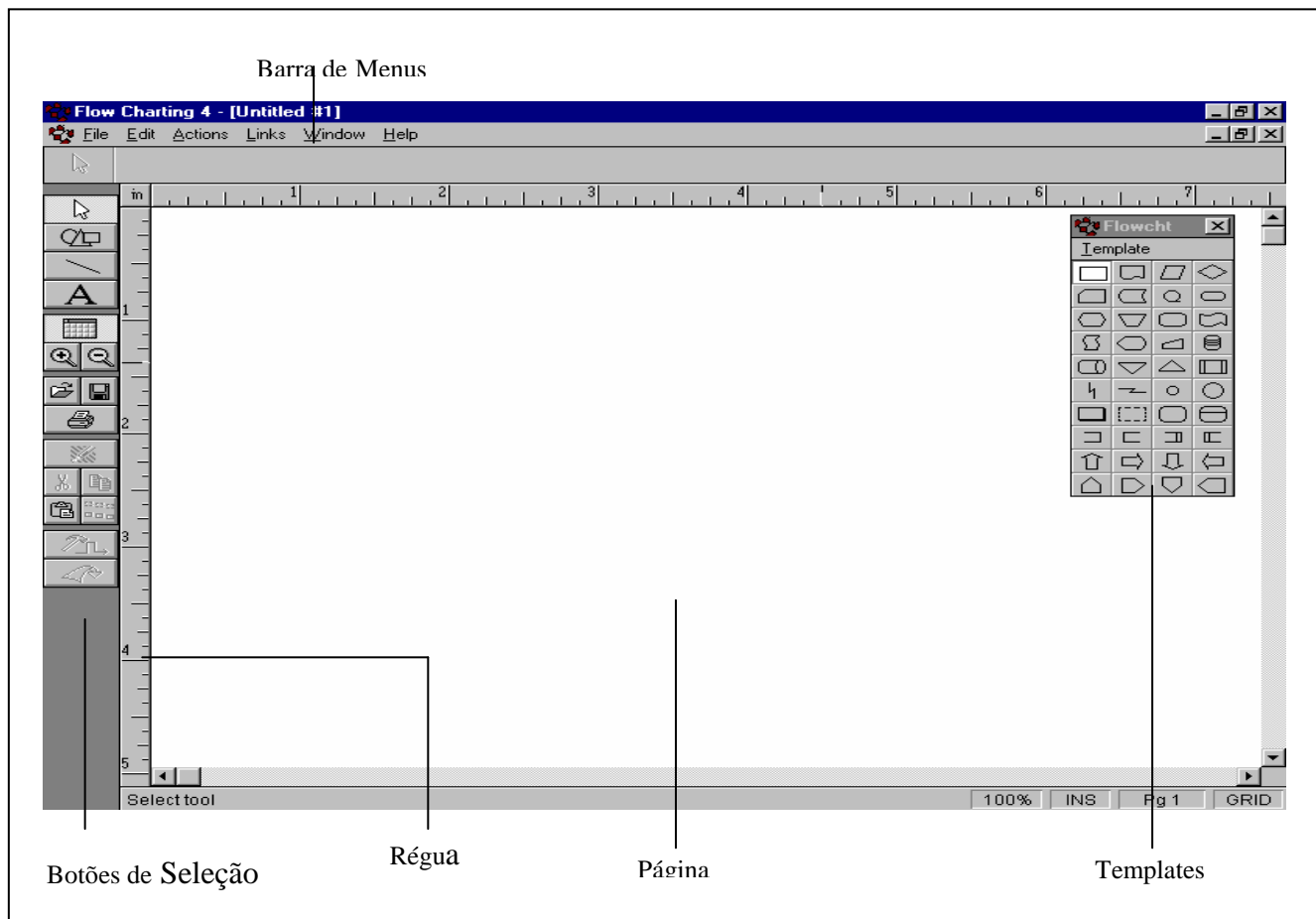
### 3.1.2 FLOW CHARTING 4.0

A ferramenta Flow Charting 4.0, da Patton & Patton Software Corporation, é utilizada para o fluxo de dados. Os principais elementos da figura 2 são:

- barra de menus – menu semelhante ao do Windows;
- botões de seleção – usados para selecionar o que será feito, por exemplo, pressionando no botão “A” poderá ser digitado um texto, no botão com uma setinha, uma seta de ligação entre duas figuras;
- botões de figuras – a figura que estiver no botão, se o mesmo for clicado, aparecerá na área da página;
- página - é a área onde será desenhada a figura;
- régua – auxílio para alinhamento das figuras.

A figura 2 apresenta o Flow Charting sendo executado sob o Windows, apontando cada um dos elementos principais acima citados.

Figura 2: Flow Charting sendo executado sob o Windows



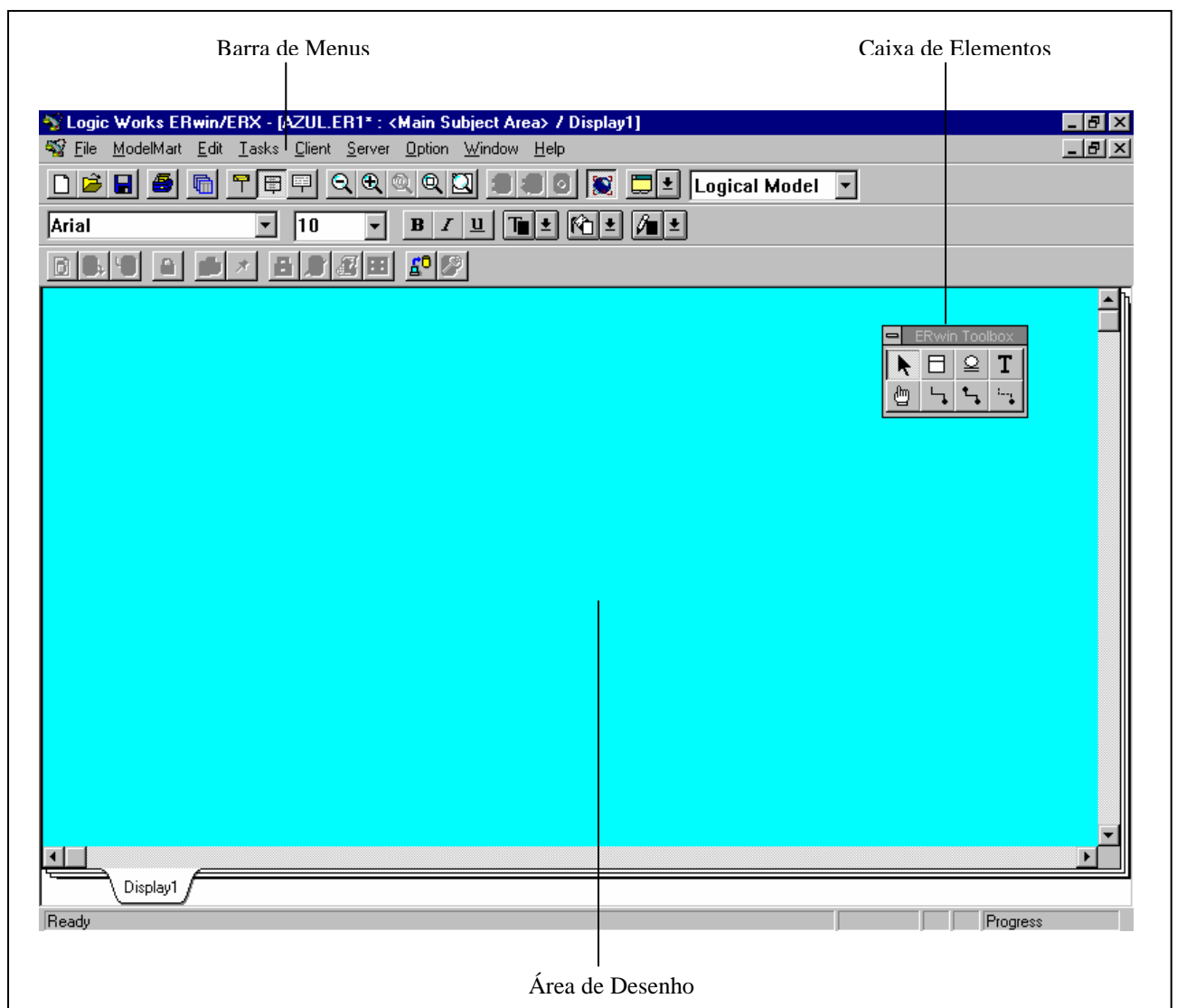
### 3.1.3 ERWIN/ERX 3.0

A ferramenta Erwin/ERX 3.0, da Logic Works, é utilizada para a elaboração do Modelo Entidade Relacionamento. Os principais elementos da figura 3 são:

- a) barra de menus – menu semelhante ao do Windows;
- b) botões de seleção – usados para selecionar o que será feito, por exemplo, pressionando no botão “T” poderá ser digitado um texto, nos botões de setinha são feitas as ligações de cardinalidade entre as tabelas;
- c) página - é a área onde será desenhada a figura;

A figura 3 apresenta o Erwin/ERX 3.0 sendo executado sob o Windows, apontando cada um dos elementos principais acima citados.

Figura 3: Erwin/ERX 3.0 executando sob o Windows



Fonte:[LOG1997]

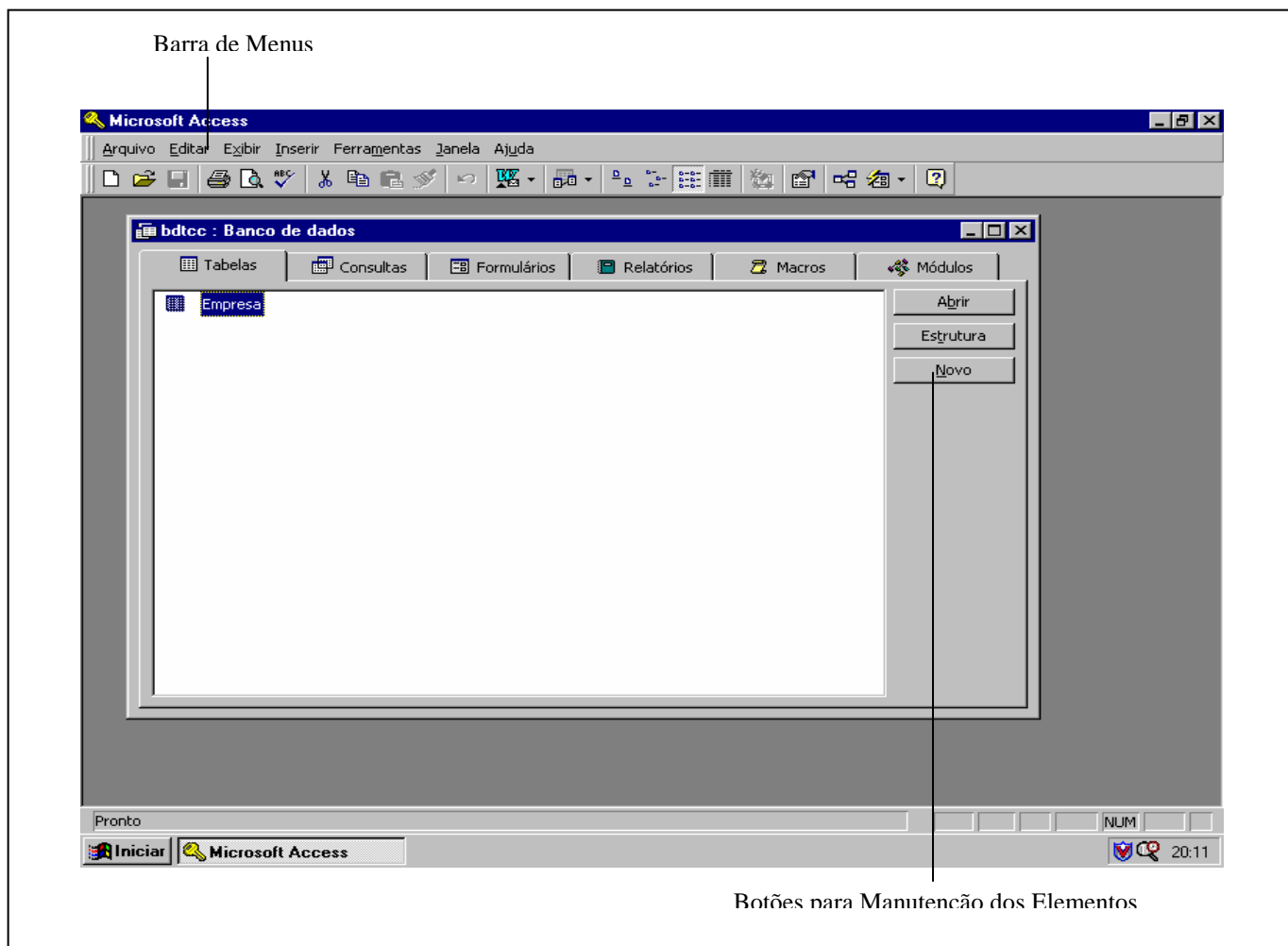
### 3.1.4 MICROSOFT ACCESS 97

A ferramenta Access 97, da Microsoft, é utilizada para armazenar dados, pode-se considerá-la um banco de dados. Os principais elementos da figura 4 são:

- a) barra de menus – menu padrão ao do Windows;
- b) abas de trabalho – utilizadas para escolha de componentes;
- c) botões de criação, alteração de elementos.

A figura 4 apresenta o Access sendo executado sob o Windows, apontando cada um dos elementos principais acima citados.

Figura 4: Access executando sob o Windows



Fonte:[VIE1998]

## 4 IMPLEMENTAÇÃO DO PROTÓTIPO

O protótipo tem por finalidade o cálculo de caminho crítico e cálculo de custo, no planejamento de desenvolvimento de softwares que sigam a análise estruturada de sistemas, utilizando para isto a metodologia PERT/CPM. Para isto deverão ser conhecidas as necessidades de recursos, as etapas de desenvolvimento. Para cada atividade do projeto de desenvolvimento de software deverão ser ligados os recursos necessários e o tempo estimado para execução da atividade.

O protótipo faz o cálculo do caminho crítico, sem utilizar, da técnica PERT/CPM, as folgas dos eventos, porém, sustentado na afirmação do autor ([BEL1974]) “a data mais cedo do evento final é a soma das durações das atividades do caminho crítico”, de uma outra forma, pode-se dizer: caminho crítico é a soma das durações das atividades do projeto, isto é, é o tempo mínimo para execução do projeto, levando em consideração apenas a duração das atividades.

Como o protótipo não fará, por exemplo, a aceleração do projeto, entre outras possibilidades, pois isto não está no escopo do trabalho, ele não utiliza as folgas dos eventos para o cálculo do caminho crítico, e sim, somente a duração das atividades, sem que com isto, esteja comprometida sua veracidade e eficácia.

### 4.1 ESPECIFICAÇÃO DO SISTEMA

De acordo com [NBR1994], o processo de desenvolvimento de softwares contém as atividades e tarefas do desenvolvedor. O processo contém as atividades para análise de requisitos, projeto, codificação, integração, testes, e instalação e aceitação relacionadas aos produtos de software. Pode conter atividades relacionadas ao sistema, se estipulado no contrato. O desenvolvedor executa ou apoia as atividades neste processo, de acordo com o contrato.

Para desenvolver este protótipo utilizou-se a linguagem de programação Visual Basic, que é uma linguagem que possibilita a geração de uma interface amigável para o usuário, e utilizou-se também a metodologia de análise estruturada de sistemas, pela sua penetração de mercado, isto é, pelo número de profissionais que conhecem e utilizam esta metodologia para desenvolver seus softwares. Relembra-se aqui, que o que foi considerado mais importante,

foi a dificuldade dos desenvolvedores de software tem em planejar o desenvolvimento de seus softwares.

No protótipo, os documentos referentes ao desenvolvimento são os seguintes: para o projeto lógico, o diagrama de contexto e o diagrama do fluxo de dados; para o modelo de dados, o diagrama do modelo entidade e relacionamento; para o sistema físico, o fluxograma de lógica de programa e o dicionário de dados; para relatórios, o cadastro de projetos, o cadastro de atividades de análise, o cadastro de recursos, o relacionamento de recursos por atividade, as durações das atividades e o custo total por projeto.

#### **4.1.1 OPERACIONALIDADE DA IMPLEMENTAÇÃO**

O protótipo está dividido em quatro etapas importantes: a primeira é o cadastramento de um projeto, que é como ele será conhecido durante o seu planejamento; a segunda é o cadastramento de todas as atividades para a confecção do projeto, lembra-se aqui, que as atividades não são diretamente ligadas à um projeto específico, podendo assim ser utilizadas ou não, para vários projetos, dependendo da necessidade do mesmo; a terceira é o cadastramento dos recursos, que tem o mesmo comportamento das atividades; o quarto, e provavelmente o mais importante, é a confecção do diagrama PERT/CPM, onde serão feitas as entradas dos eventos e atividades relevantes ao projeto em questão.

É importante lembrar que para utilizar as atividades e recursos em um determinado projeto, as mesmas deverão estar cadastradas, fazendo-se neste momento o relacionamento das atividades e recursos à cada etapa existente no projeto.

Ressalta-se a partir deste momento as etapas que deverão ser seguidas para o bom funcionamento e a boa operacionalidade do protótipo:

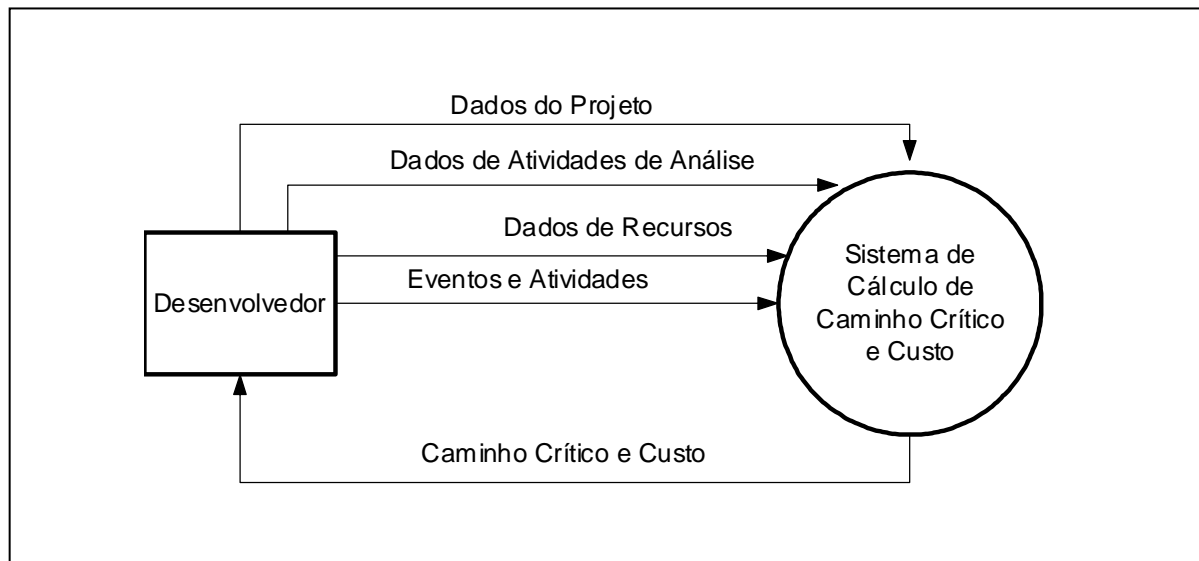
- a) manter atualizado os cadastros de projetos, atividades de análise e recursos, pois os mesmos são indispensáveis para a confecção do diagrama de PERT;
- b) gerar o diagrama de PERT, não esquecendo:
  - todos os eventos criados deverão ter no mínimo uma atividade relacionada;
  - toda atividade deverá ter, obrigatoriamente, uma atividade de análise, uma duração em dias, no mínimo um recurso alocado a executar a mesma.



### 4.1.2 DIAGRAMA DO FLUXO DE DADOS

O diagrama de contexto é apresentado abaixo pela figura 6, é uma apresentação lógica do protótipo, onde são evidenciados os fluxos dos dados de entradas e saídas.

Figura 6: Diagrama de Contexto do Protótipo



No processo Manter Cadastro de Projetos será possível incluir, alterar, excluir e consultar todos os projetos existentes. Sendo que para gerar um diagrama PERT o projeto deverá estar cadastrado.

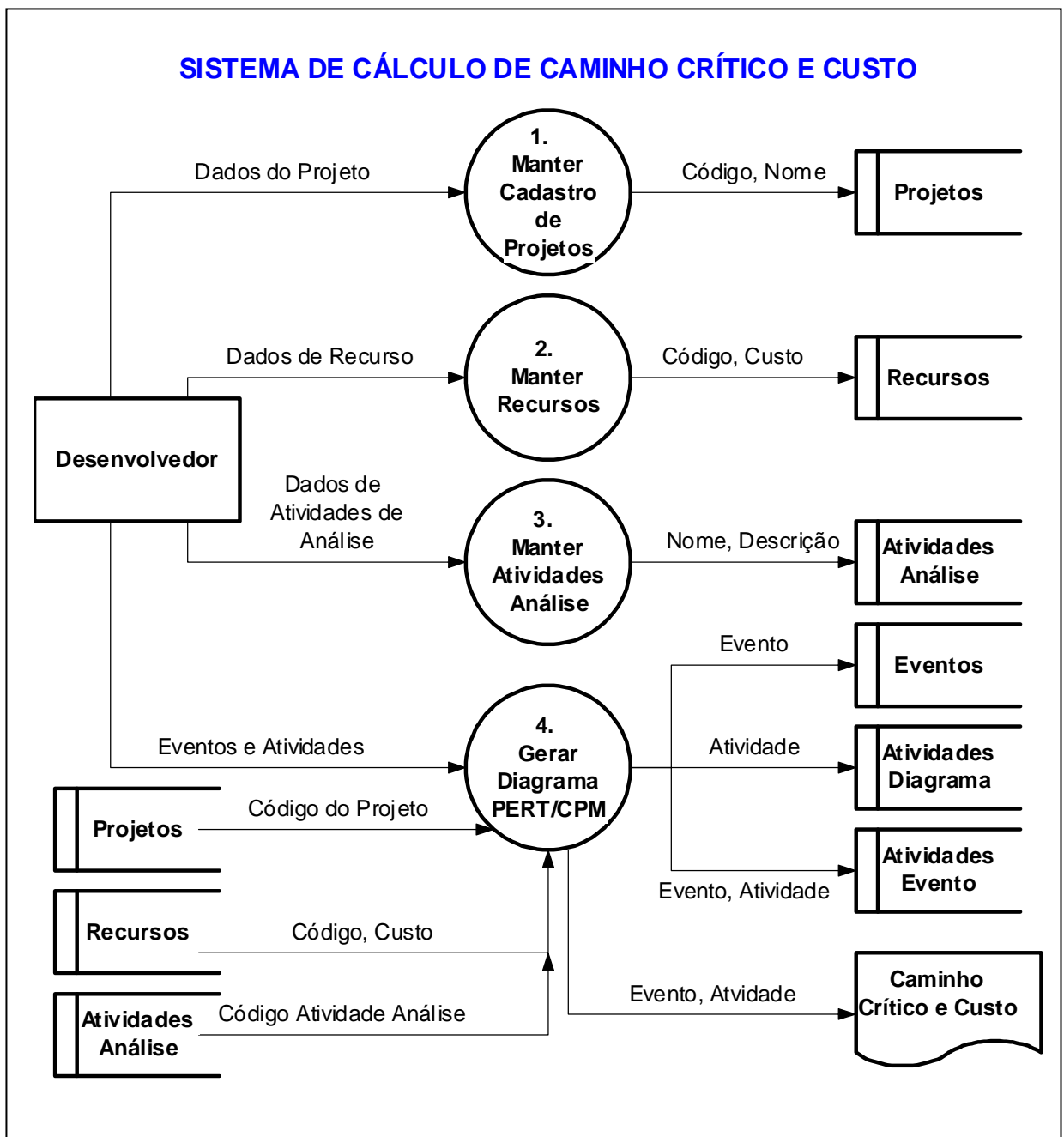
No processo Manter Recursos será possível incluir, alterar, excluir e consultar todos os recursos cadastrados, sendo que eles não estão diretamente relacionados a um projeto, pois um mesmo recurso poderá ser utilizado em vários projetos, sendo este relacionamento feito no momento da criação do gráfico PERT, é neste processo ainda, que se informa qual será o custo hora para cada recurso.

No processo Manter Atividades Análise será possível incluir, alterar, excluir e consultar todas as atividades análise cadastradas, e como os recursos, as atividades também não estão diretamente relacionados a um projeto, sendo este relacionamento feito no momento da criação do gráfico PERT.

No processo Gerar Diagrama PERT será construído o diagrama PERT do projeto a ser construído. O diagrama consiste em criar eventos e ligar estes eventos através de linhas, que indicam a atividade que será executada entre dois eventos, e é no momento da geração das linhas é que se informa qual é a atividade análise que se executará e quais serão os recursos necessários para efetuar a tarefa.

O diagrama de fluxo de dados nível 0 é apresentado abaixo pela figura 7, é uma apresentação lógica do protótipo, onde são evidenciados os processos, o fluxo dos dados, as entradas e saídas.

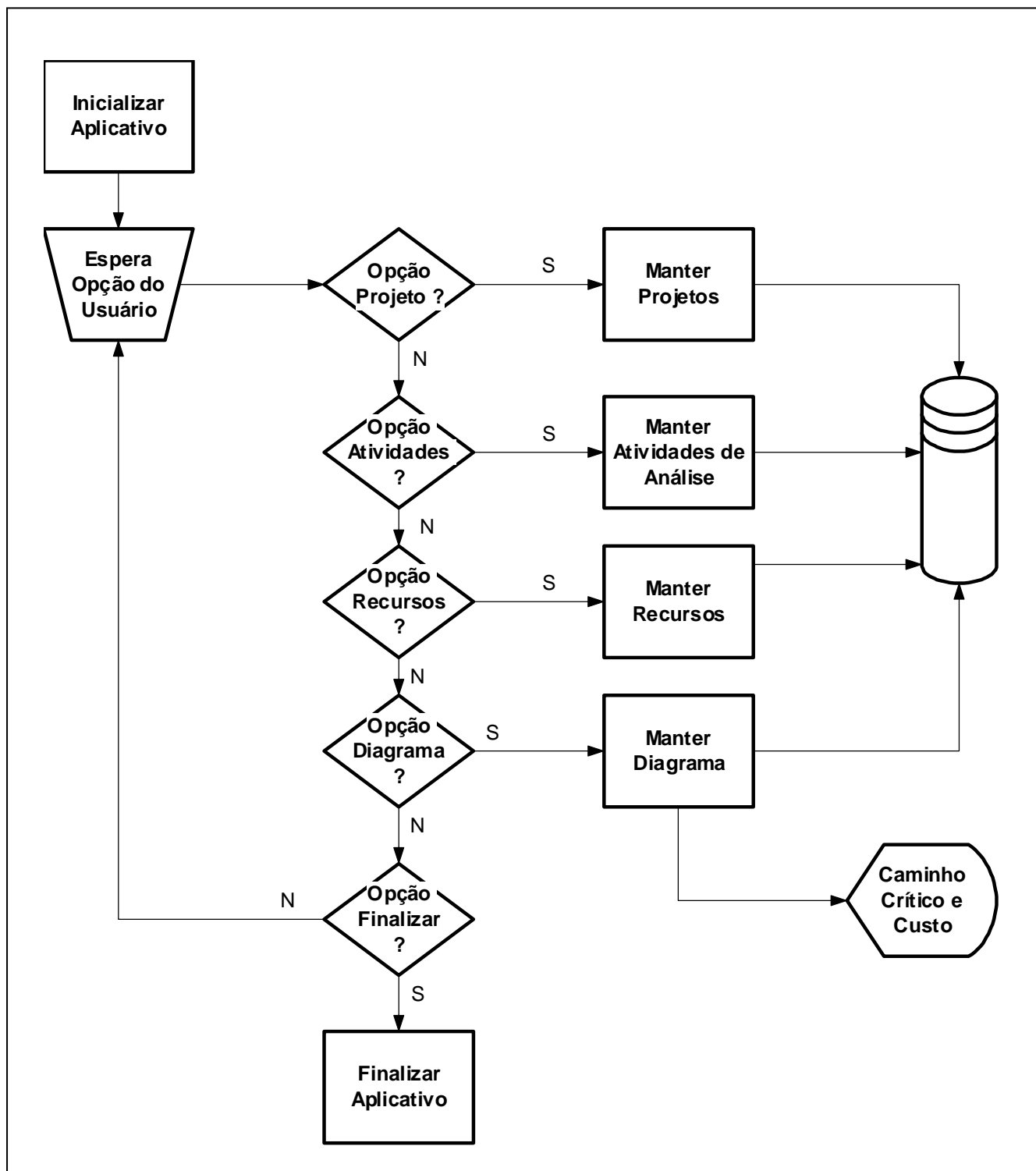
Figura 7: Diagrama do Fluxo de Dados do Protótipo



### 4.1.3 FLUXOGRAMA DE LÓGICA DE PROGRAMA

O fluxograma de lógica de programa é apresentado abaixo pela figura 5, é uma apresentação física do protótipo.

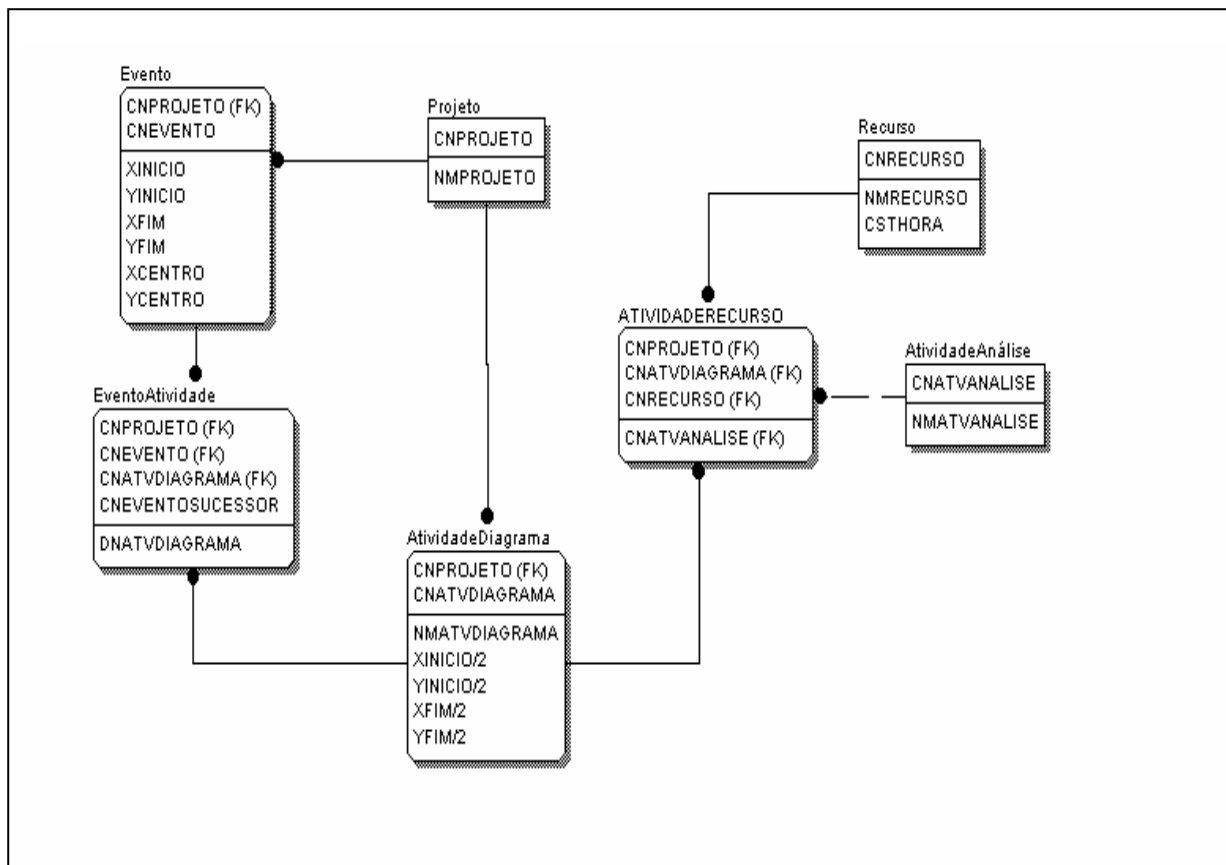
Figura 5: Fluxograma de Lógica de Programa



#### 4.1.4 MODELO ENTIDADE RELACIONAMENTO

O modelo entidade relacionamento é apresentado abaixo pela figura 8, onde são evidenciados os atributos das tabelas do banco de dados e seu relacionamento.

Figura 8: Modelo Entidade Relacionamento do Protótipo



Cada retângulo da figura acima representa uma tabela, e os atributos que aparecem acima da linha que corta o retângulo, formam a chave primária da tabela, que pode vir como chave estrangeira de outra tabela.

Constata-se na figura acima que as tabelas recursos e as atividades análise não tem relacionamento direto com a tabela de projeto, e que isto acontece através da tabela atividade recurso que está ligada a tabela atividade diagrama, a qual conterà a linha mencionada no diagrama de fluxo de dados. Nesta tabela guarda-se também as coordenadas x e y da tela para saber seu ponto início e final. A tabela evento atividade, conterà o evento e seu

evento sucessor e a linha que relaciona os dois eventos. Na tabela evento, tem-se a numeração seqüencial dos eventos e suas coordenadas x e y da tela.

#### 4.1.5 ATRIBUTOS DO DICIONÁRIO DE DADOS

Para demonstrar os dados do dicionário de dados serão apresentados a relação de atributos, que conterà o nome do atributo e sua descrição, a relação de tabelas com suas chaves primárias e a relação de tabelas e suas chaves estrangeiras.

Relação de Atributos:

Nome do Atributo	Descrição do Atributo	Tipo/Tamanho
CNATVANALISE	Código Numérico da Atividade de Análise	Integer
CNATVDIAGRAMA	Código Numérico da Atividade do Diagrama	Integer
CNEVENTO	Código Numérico do Evento	Integer
CNEVENTOSUCCESSOR	Código Numérico do Evento Sucessor	Integer
CNPROJETO	Código Numérico do Projeto	Integer
CNRECURSO	Código Numérico do Recurso	Integer
CSTHORA	Custo Hora do Recurso	Currency
DNATVDIAGRAMA	Número de Dias de Duração da Atividade do Diagrama	Integer
NMATVANALISE	Nome da Atividade de Análise	Text(20)
NMATVDIAGRAMA	Nome da Atividade do Diagrama	Text(20)
NMPROJETO	Nome do Projeto	Text(30)
NMRECURSO	Nome do Recurso	Text(20)
XINICIO	Coordenada Início de X do Evento	Integer
YINICIO	Coordenada Início de Y do Evento	Integer
XCENTRO	Coordenada Centro de X do Evento	Integer
YCENTRO	Coordenada Centro de Y do Evento	Integer

XFIM	Coordenada Fim de X do Evento	Integer
YFIM	Coordenada Fim de Y do Evento	Integer
XINICIO/2	Coordenada Início de X da Atividade do Diagrama	Integer
YINICIO/2	Coordenada Início de Y da Atividade do Diagrama	Integer
XFIM/2	Coordenada Fim de X da Atividade do Diagrama	Integer
YFIM/2	Coordenada Fim de Y da Atividade do Diagrama	Integer

#### Relação de Tabelas e suas Chaves Primárias

Nome da Tabela	Chave Primária
AtividadeAnálise	CNATVANALISE
AtividadeDiagrama	CNPROJETO
	CNATVDIAGRAMA
AtividadeRecurso	CNPROJETO
	CNATVDIAGRAMA
	CNRECURSO
Evento	CNPROJETO
	CNEVENTO
EventoAtividade	CNPROJETO
	CNEVENTO
	CNATVDIAGRAMA
	CNEVENTOSUCCESSOR
Projeto	CNPROJETO
Recurso	CNRECURSO

### Relação de Tabelas e suas Chaves Estrangeiras

FK	Nome da Tabela Pai	Nome da Tabela Filho
está relacionado	AtividadeAnálise	AtividadeRecurso
Contem	AtividadeDiagrama	AtividadeRecurso
Relaciona	AtividadeDiagrama	EventoAtividade
Relaciona	Evento	EventoAtividade
Contem	Projeto	AtividadeDiagrama
Contem	Projeto	Evento
está alocado	Recurso	AtividadeRecurso

### 4.1.6 LÓGICA DO CÁLCULO DO CAMINHO CRÍTICO

Mostra-se abaixo as principais estruturas criadas para se chegar ao caminho crítico do gráfico PERT. Na figura 9 o array chamado `EventosAtividades` conterà todos os eventos do gráfico com identificação do evento sucessor, a atividade e a duração da atividade. Na lógica do português estruturado abaixo, o índice para esta estrutura será a variável “`TotalReg`”. Na figura 10, abaixo, o array tipo Pilha que servirá para armazenar os eventos do gráfico ainda não processados, e que conterà o `Evento`, a `Lista_Eventos`, `Total_Caminho` e `Numero_Eventos`.

Figura 9: Array `EventosAtividades`

NÚMERO EVENTO	EVENTO SUCESSOR	ATIVIDADE	DURAÇÃO

Figura 10: Estrutura de Pilha

EVENTO	LISTA_EVENTOS	TOTAL_CAMINHO	NUMERO_EVENTOS

Abaixo é apresentado em português estruturado como se faz o processamento do gráfico PERT, para identificar quais são os eventos e atividades críticas, para se saber qual é o caminho crítico do gráfico.

```

Início
Evento = 1; Maior_Caminho = 0; Lista_Eventos_Caminho Crítico = ' ';
Lista_Eventos[1] = Evento; {Insere primeiro Evento}
Total_Caminho = 0; Numero_Eventos = 1; Numero_Eventos_Caminho_Critico = 0;
{Cria Pilha}
Empilha (Evento, Lista_Eventos, Total_Caminho, Numero_Eventos);
Enquanto Pilha não estiver vazia faça
Início
  Desempilha (Evento, Lista_Eventos, Total_Caminho, Numero_Eventos)
  Para I = 1 até TotalReg faça
  Início
    Se Evento = Tab_Atividades[I] então      {Insere Sucessor na lista de Eventos}
    Início
      Lista_Eventos[Numero_Eventos +1] = Tab_Atividades[I].Sucessor
      Empilha (Sucessor, Lista_Eventos, Total_Caminho + Duração_Atividade,
              Numero_Eventos + 1);
    Fim se;
  Fim Para;
  Se não há Atividade a partir deste Evento então {é o Evento Final}
  Se Total_Caminho > Maior_Caminho então Início
    Maior_Caminho = Total_Caminho;
    Lista_Eventos_Caminho_Critico = Lista_Eventos;
    Número_Eventos_Caminho_Critico = Numero_Eventos;
  Fim se
  Fim se
Fim enquanto.

```

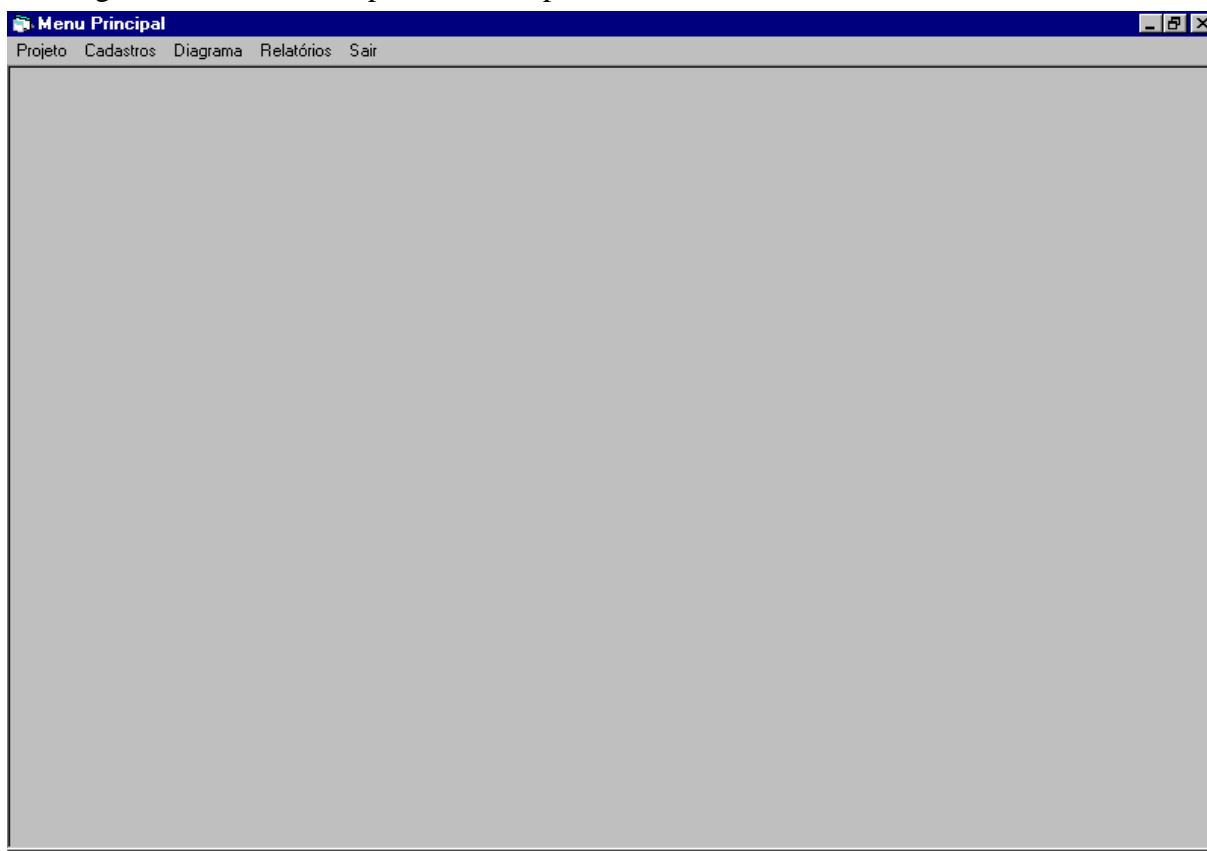
Como resultado final deste processamento tem-se o array “Lista\_Eventos\_Caminho\_Critico” o qual conterà todos os eventos que compõe o caminho crítico, e a variável “Numero\_Eventos\_Caminho\_Critico que conterà o numero de eventos críticos, identificando-se com estes dois elementos o caminho crítico do gráfico.



## 4.2 APRESENTAÇÃO DAS TELAS E OPERACIONALIDADE

Abaixo é apresentada a figura 11, que é o menu principal do protótipo, de onde se escolhe uma tarefa, dentre elas, manter o cadastro de projetos, o cadastro das atividades de análise estruturada de sistemas, o cadastro de recursos, o desenho do diagrama, opções de relatórios e por fim a possibilidade de fechamento da execução do protótipo.

Figura 11: Menu Principal do Protótipo



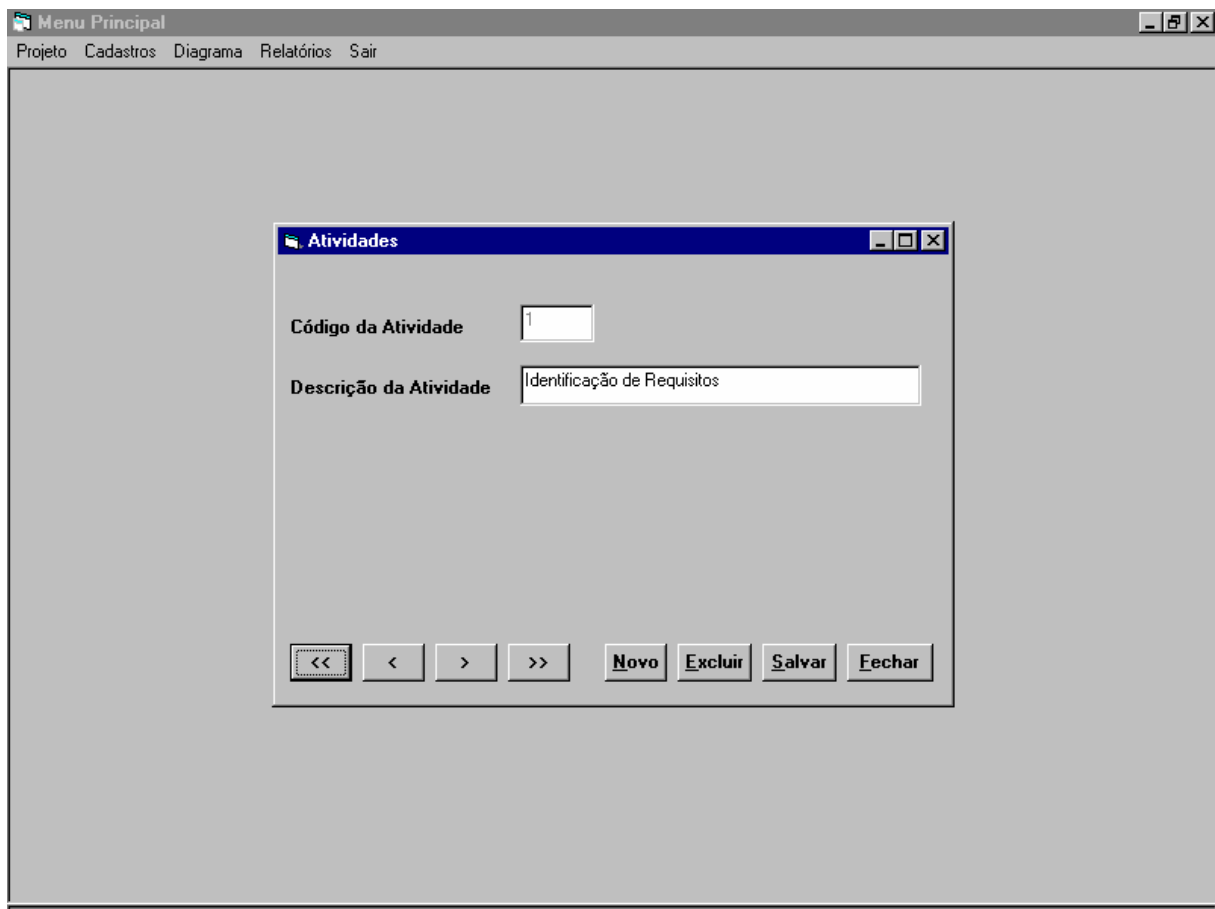
Com a escolha da opção projeto, será mostrada abaixo a figura 12 do protótipo, onde pode-se incluir, alterar, excluir, consultar e navegar por todos dados existentes cadastrados para projetos.

Figura 12: Manutenção do Cadastro de Projetos

The image shows a screenshot of a software application. The main window is titled "Menu Principal" and has a menu bar with the following items: "Projeto", "Cadastros", "Diagrama", "Relatórios", and "Sair". A sub-window titled "Controle de Projeto" is open, displaying two input fields: "Código do Projeto" with the value "2" and "Nome do Projeto" with the value "Projeto de TCC". At the bottom of the sub-window, there are several buttons: "<<", "<", ">", ">>", "Novo", "Excluir", "Salvar", and "Fechar".

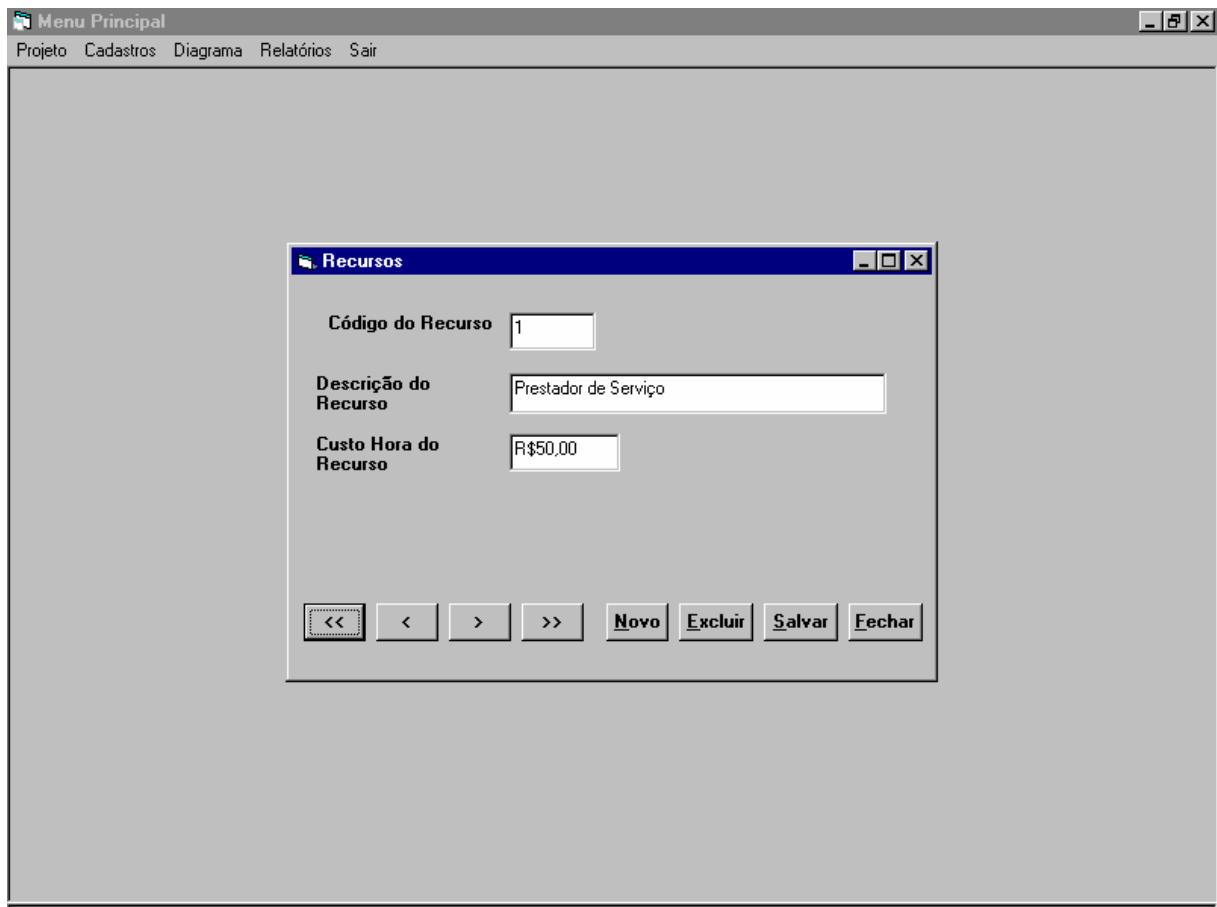
Com a escolha da opção cadastros, optar-se-á pela manutenção do cadastro das atividades da análise estruturada de sistemas, é mostrada abaixo a figura 13 do protótipo, onde pode-se incluir, alterar, excluir, consultar e navegar por todos dados existentes cadastrados para atividades, as quais são as atividades de análise estruturada de sistemas.

Figura 13: Manutenção do Cadastro de Atividades



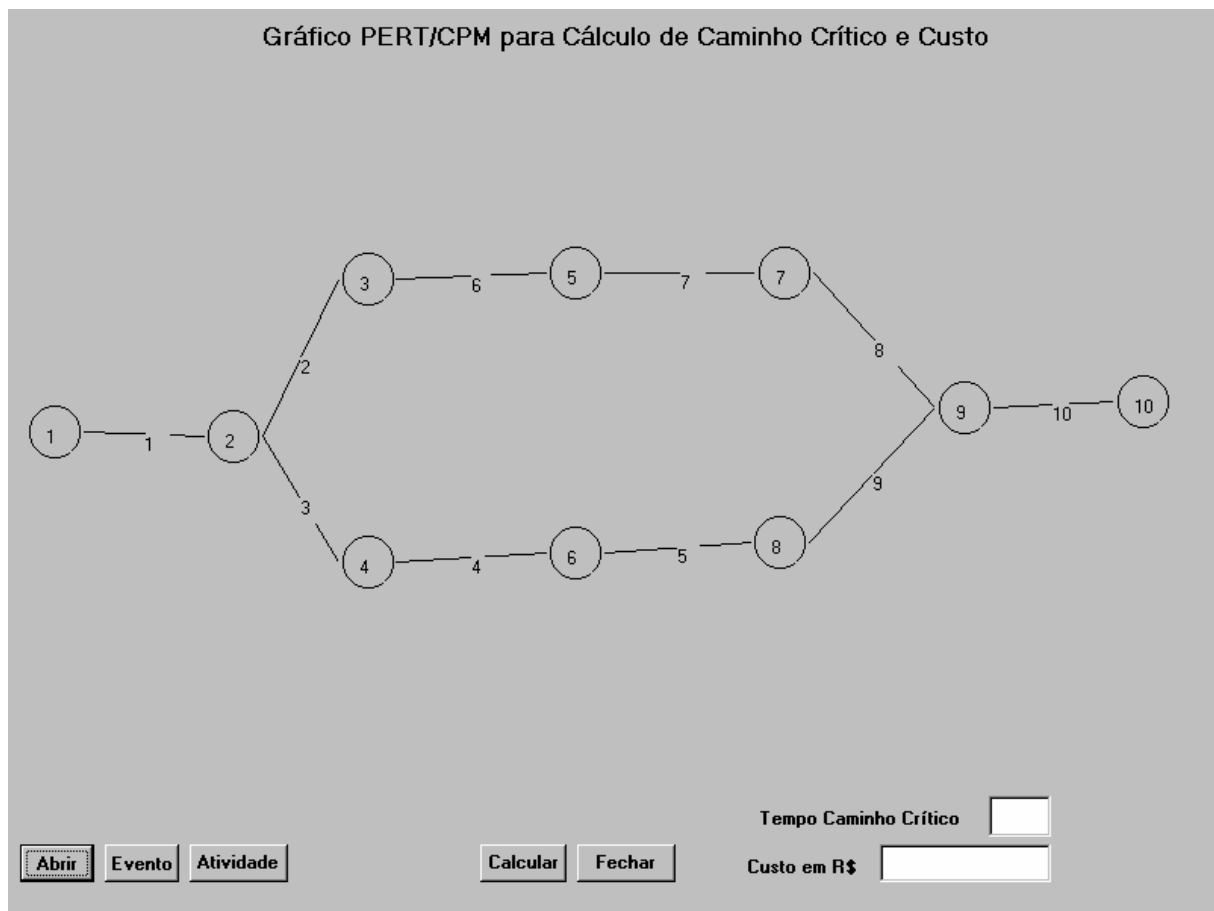
Ainda na opção cadastros, optar-se-á pela manutenção do cadastro de recursos, será mostrada abaixo na figura 14 do protótipo, onde pode-se incluir, alterar, excluir, consultar e navegar por todos dados existentes cadastrados para recursos, ressalta-se que, é neste momento que é definido o custo de cada recurso.

Figura 14: Manutenção dos Recursos do Protótipo



Em seguida, dever-se-á utilizar a opção diagrama do menu principal para construir o diagrama PERT/CPM. A figura 15, mostrada abaixo, já apresenta um exemplo de diagramação de um projeto de software, neste caso, utilizou-se o evento “abrir” da tela do diagrama. Para um projeto novo, deve-se começar construindo os eventos e atividades na tela, basta para isto clicar no comando Evento e depois posicionar o mouse na tela na posição que se desejar, e neste ponto, através das coordenadas x e y será criado o evento pela representação de um círculo. Repete-se isto até o gráfico estar concluído. Para criar uma atividade, é necessário que existem dois eventos na tela que ainda não tenham sido relacionados, clica-se então no comando atividade e depois clica-se, primeiro no evento origem e depois no evento destino, desta forma será criada a atividade que será representada por uma linha, e é neste momento que se atribuirá a esta atividade um nome e os recursos necessários para executá-la, através da figura 16, apresentada após a figura 15.

Figura 15: Gráfico PERT/CPM do Protótipo



Mostra-se abaixo, na figura 16, a tela do protótipo que será exibida no momento de relacionar dois eventos, pois é onde se identifica qual é a atividade que está sendo efetuada e quais são os recursos necessários para sua conclusão.

Figura 16: Relacionamento Recursos a Atividades do Protótipo

Gráfico PERT/CPM para Cálculo de Caminho Crítico e Custo

1 — 1 — 2

**Recursos Alocados para Atividade**

Código do Projeto: 6      Código da Atividade do Diagrama: 6

Código da Atividade de Análise: Criação do Dicionário de Dados

Tempo da Atividade em Dias: 5

**Recursos Disponíveis**

- Prestador de Serviço
- Analista de Sistemas
- Programador de Computador
- Usuário
- Gerente do Projeto

**Recursos da Atividade**

- Administrador do Banco de Dados

OK

Tempo Caminho Crítico:

Custo em R\$:

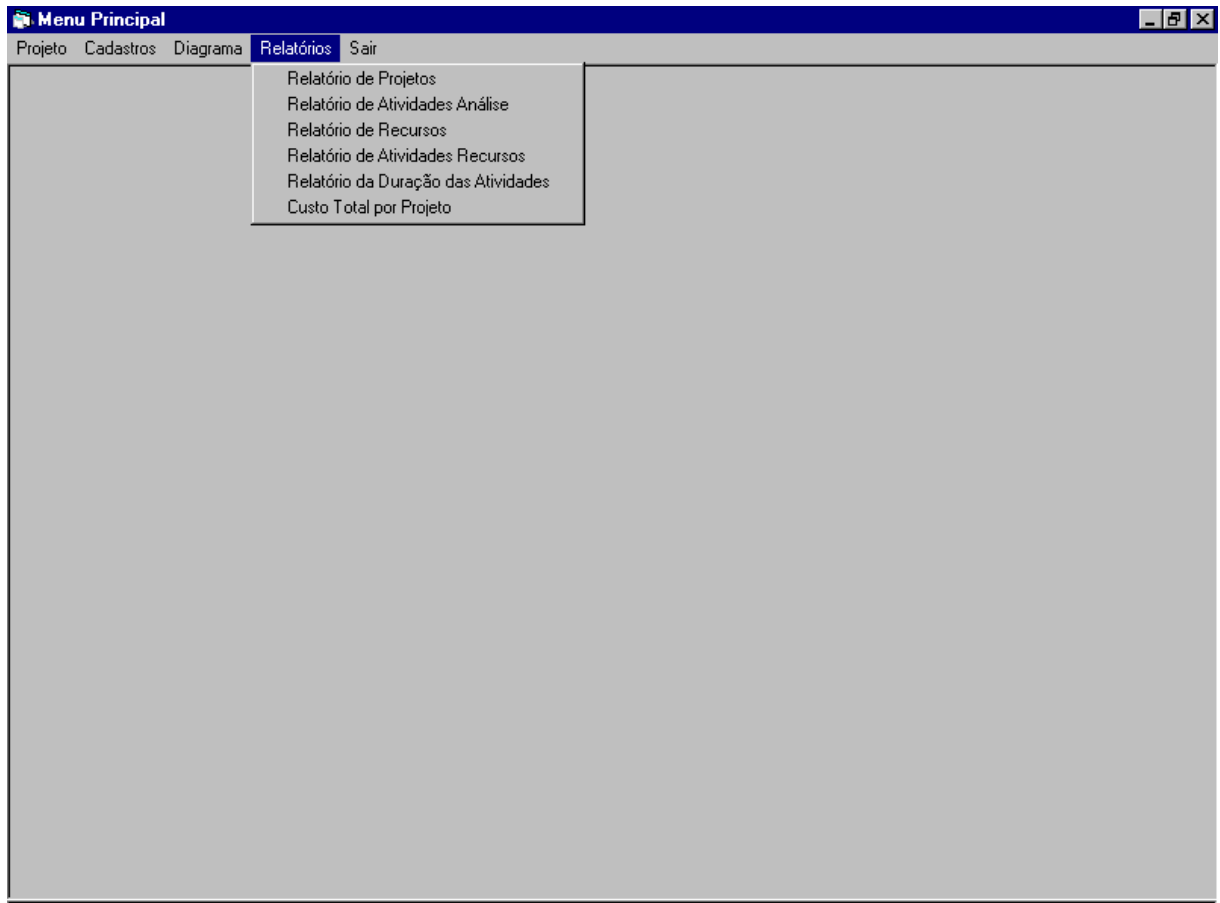
Abrir   Evento   Atividade   Calcular   Fechar

Como possibilidade de relatórios, a figura 17, apresenta todos os relatórios do protótipo, que servirão de acompanhamento dos projetos existentes, os quais são:

- relatório de projetos, imprime o projeto desejado, com Código e Nome do projeto;
- relatório de atividades análise, imprime todas as atividades cadastradas, as quais poderão ser utilizadas em um projeto;
- relatório de recursos, imprime todos os recursos cadastrados, os quais poderão ser utilizados em um projeto;
- relatório de atividades recursos, imprime todas as atividades de um projeto, com seus recursos relacionados;
- relatório da duração das atividades, imprime as atividades do projeto com seu respectivo tempo necessário para execução;

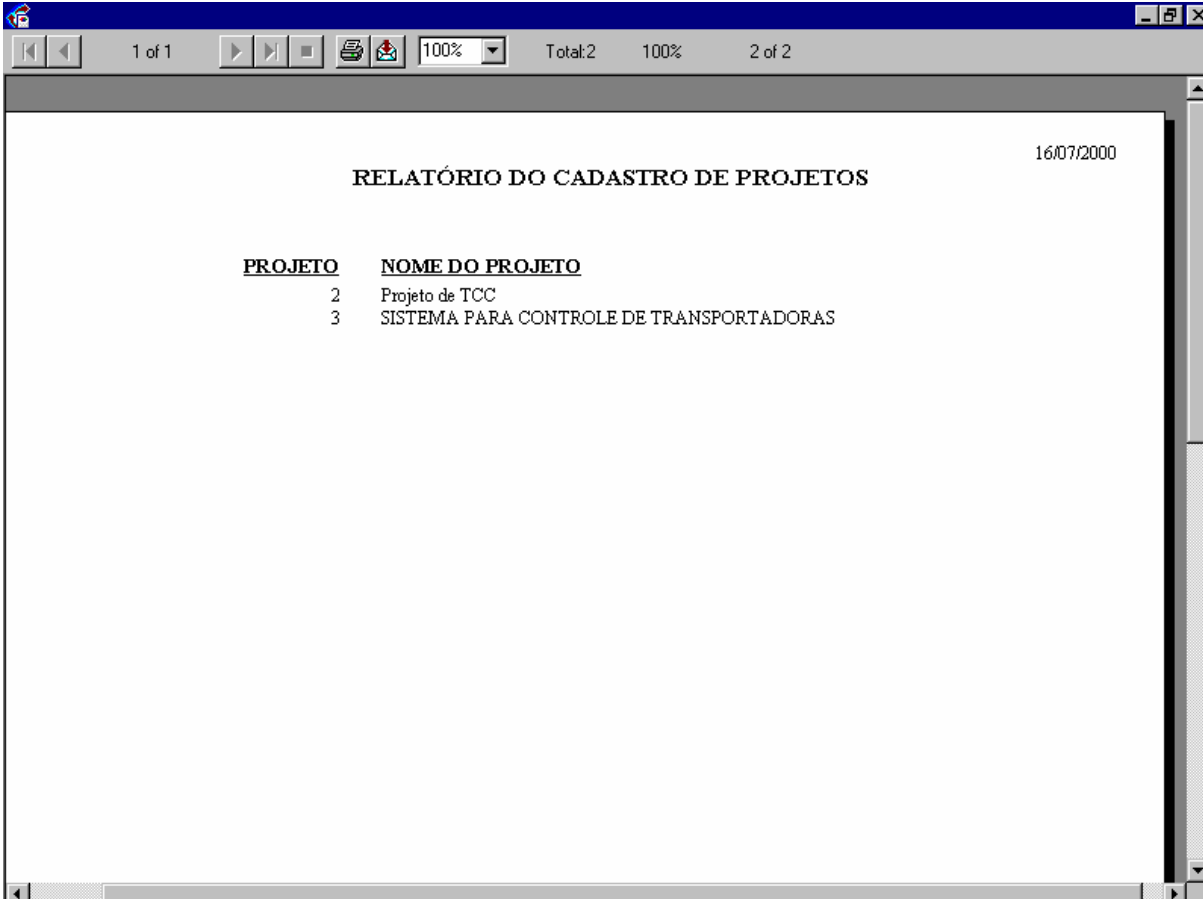
- f) relatório custo total por projeto, imprime o custo total do projeto, que é calculado somando-se a multiplicação do custo hora de cada recurso relacionado a uma atividade com a duração da mesma, faz-se isto para todas as atividades de um projeto.

Figura 17: Opções de Relatórios do Protótipo



Como resultado da execução da primeira opção de relatórios, que é relatório de projetos, teremos na figura 18 mostrada abaixo, como exemplo.

Figura 18: Relatório do Cadastro de Projetos do Protótipo



16/07/2000

**RELATÓRIO DO CADASTRO DE PROJETOS**

<u>PROJETO</u>	<u>NOME DO PROJETO</u>
2	Projeto de TCC
3	SISTEMA PARA CONTROLE DE TRANSPORTADORAS

Ainda como possibilidade de relatórios, a figura 19, apresenta o relatório do cadastro de Atividades de Análise.



Figura 19: Relatório do Cadastro de Atividades de Análise do Protótipo

<u>Código da Atividade</u>	<u>Nome da Atividade</u>
1	Identificação de Requisitos
2	Análise das Necessidades
3	Especificação Técnica
5	Programação
6	Definição da Base de Dados
7	Manutenção
8	Criação do Dicionário de Dados
9	Desenho do Sistema (DFD nível 0 - contexto)
10	Desenho das Funções (DFD nível 1)
11	Codificação da Função Cadastro de Transportadoras
12	Testes da Função Cadastro de Transportadoras
13	Codificação da Função Cadastro de Frota
14	Testes da Função Cadastro de Frota
15	Codificação da Função Cadastro de Motoristas
16	Testes da função Cadastro de Motoristas
17	Codificação da Função Cadastro de Clientes
18	Testes da Função Cadastro de Clientes
19	Codificação da Função Controle de Pedidos
20	Testes da Função Controle de Pedidos
21	Codificação da Função Cadastro de Preço de Frete
22	Testes da Função Cadastro de Preço de Frete
23	Codificação da Função Geração de Nota Fiscal
24	Testes da Função Geração de Nota Fiscal
25	Testes Gerais (Caixa Preta)
26	Implantação
27	Treinamento dos Usuários
28	Operacionalização do Sistema

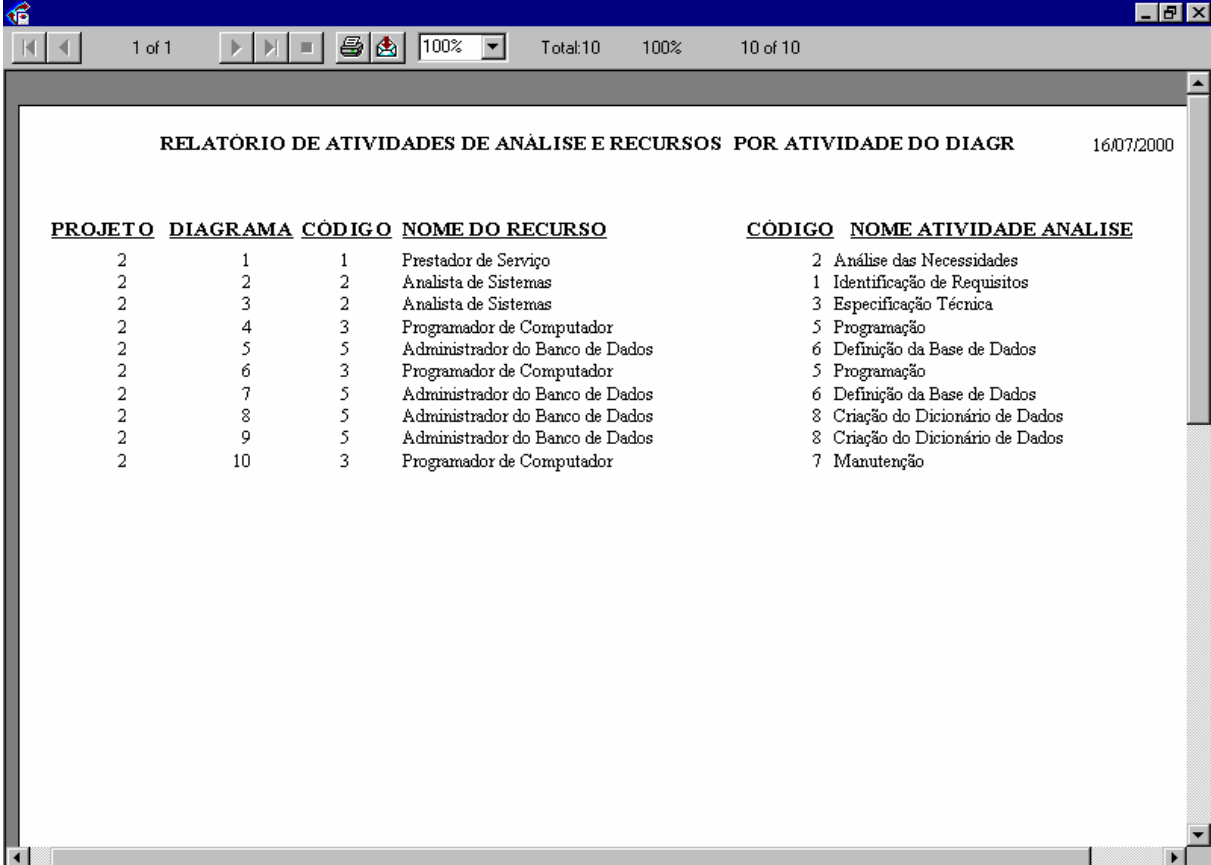
Na figura 20, mostrada abaixo, apresenta-se o relatório do cadastro de Recursos com seu custo hora.

Figura 20: Relatório do Cadastro de Recursos do Protótipo

<u>Código do Recurso</u>	<u>Nome do Recurso</u>	<u>Custo Hora</u>
1	Prestador de Serviço	R\$50,00
2	Analista de Sistemas	R\$40,00
3	Programador de Computador	R\$22,00
4	Usuário	R\$22,00
5	Administrador do Banco de Dados	R\$35,00
6	Gerente do Projeto	R\$50,00

Apresenta-se abaixo, na figura 21, o relatório de atividades recursos, onde são mostrados todos os recursos relacionados a uma atividade.

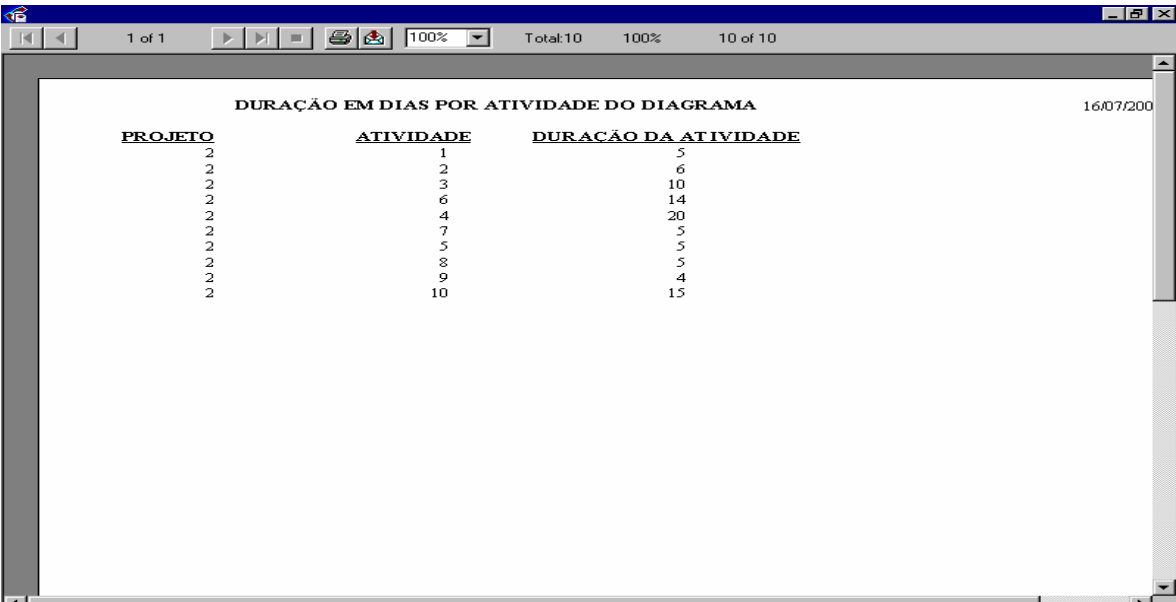
Figura 21: Relatório de Recursos por Atividade do Protótipo



<u>PROJETO</u>	<u>DIAGRAMA</u>	<u>CÓDIGO</u>	<u>NOME DO RECURSO</u>	<u>CÓDIGO</u>	<u>NOME ATIVIDADE ANALISE</u>
2	1	1	Prestador de Serviço	2	Análise das Necessidades
2	2	2	Analista de Sistemas	1	Identificação de Requisitos
2	3	2	Analista de Sistemas	3	Especificação Técnica
2	4	3	Programador de Computador	5	Programação
2	5	5	Administrador do Banco de Dados	6	Definição da Base de Dados
2	6	3	Programador de Computador	5	Programação
2	7	5	Administrador do Banco de Dados	6	Definição da Base de Dados
2	8	5	Administrador do Banco de Dados	8	Criação do Dicionário de Dados
2	9	5	Administrador do Banco de Dados	8	Criação do Dicionário de Dados
2	10	3	Programador de Computador	7	Manutenção

Ainda como possibilidade de relatórios, a figura 22, apresenta o relatório de atividades do diagrama PERT com suas respectivas durações.

Figura 22: Relatório das Durações das Atividades do Protótipo



<u>PROJETO</u>	<u>ATIVIDADE</u>	<u>DURAÇÃO DA ATIVIDADE</u>
2	1	5
2	2	6
2	3	10
2	6	14
2	4	20
2	7	5
2	5	5
2	8	5
2	9	4
2	10	15

O relatório do custo total por projeto, onde pode-se também identificar qual é o custo de cada recurso por atividade existente, é apresentada abaixo pela figura 23.

Figura 23: Relatório de Custo Total por Projeto

CUSTO DO PROJETO POR RECURSO						16/07/2000
<u>PROJETO</u>	<u>ATIVIDADE</u>	<u>RECURSO</u>	<u>NOME DO RECURSO</u>	<u>DURACÃO.\$ HORA</u>	<u>Custo</u>	
2						
2	1	1	Prestador de Serviço	5 R\$50,00	R\$250,00	
2	2	2	Analista de Sistemas	6 R\$40,00	R\$240,00	
2	3	2	Analista de Sistemas	10 R\$40,00	R\$400,00	
2	6	3	Programador de Computador	14 R\$22,00	R\$308,00	
2	4	3	Programador de Computador	20 R\$22,00	R\$440,00	
2	7	5	Administrador do Banco de Dados	5 R\$35,00	R\$175,00	
2	5	5	Administrador do Banco de Dados	5 R\$35,00	R\$175,00	
2	8	5	Administrador do Banco de Dados	5 R\$35,00	R\$175,00	
2	9	5	Administrador do Banco de Dados	4 R\$35,00	R\$140,00	
2	10	3	Programador de Computador	15 R\$22,00	R\$330,00	
2			<b>Sub-Total</b>		<b>2.633,00</b>	
			<b>Total Geral</b>		<b>2.633,00</b>	
<b>Grand Tot:</b>					<b>2.633,00</b>	

## 5 RESULTADOS

Apresenta-se a seguir um exemplo prático da utilização do protótipo. Utilizar-se-á o desenvolvimento de um sistema fictício de controle de transportes para transportadoras.

O sistema terá as seguintes tabelas de base de dados: a tabela de cadastro de transportadoras, o cadastro de clientes, o cadastro de preço de frete, o cadastro da frota, o cadastro de motoristas, o cadastro de pedidos e o cadastro de notas fiscais.

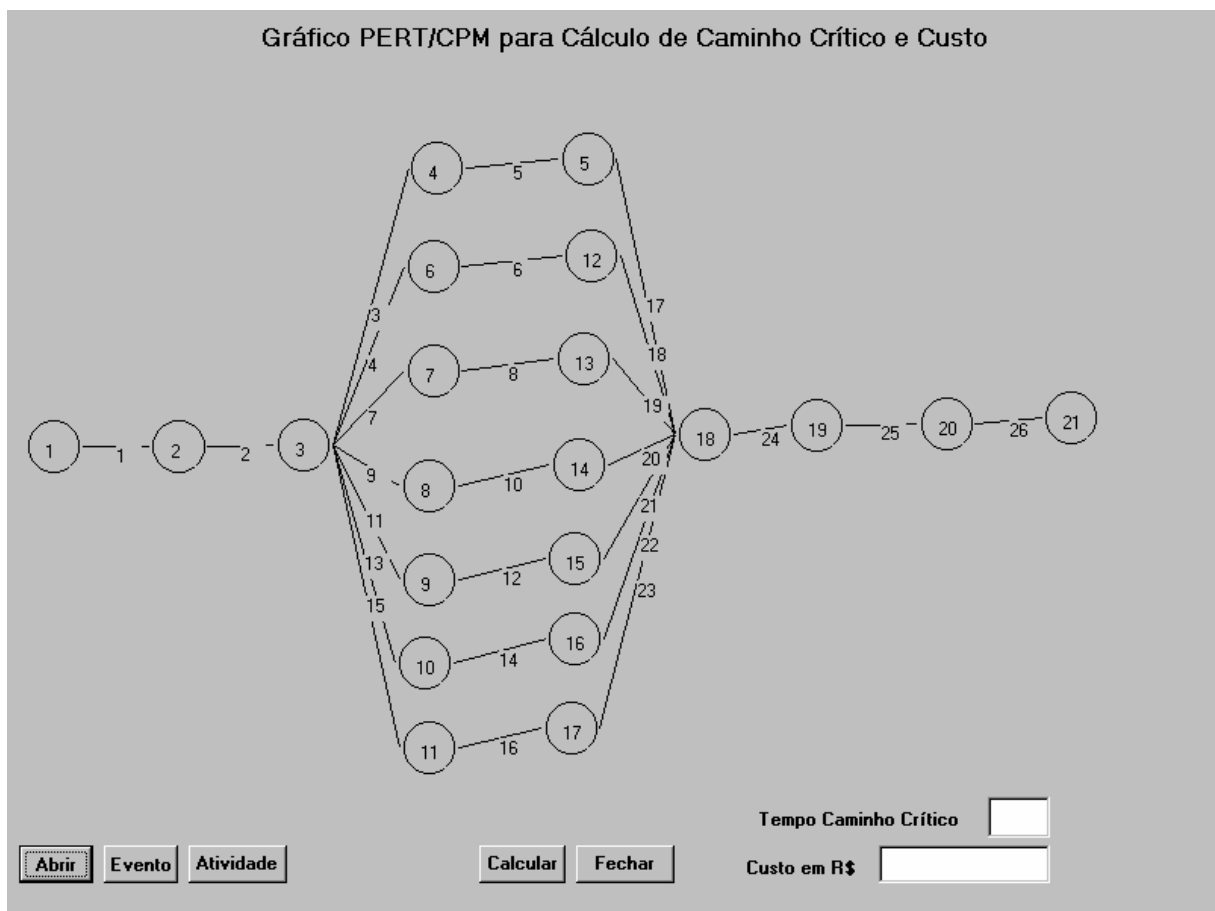
Considera-se para o sistema as atividades da tabela abaixo:

Atividade	Duração	Recurso(s)
1) desenho do sistema (DFD nível 0 – contexto)	10	analista
2) desenho das funções (DFD nível 1)	30	analista
3) codificação da função cadastro de transportadoras	8	programador
4) testes da função cadastro de transportadoras	2	programador
5) codificação da função cadastro da frota	5	programador
6) teste da função cadastro da frota	1	programador
7) codificação da função cadastro de motoristas	5	programador
8) teste da função cadastro de motoristas	1	programador
9) codificação da função cadastro de clientes	7	programador
10) teste da função cadastro de clientes	2	programador
11) codificação da função controle de pedidos	6	programador
12) teste da função controle de pedidos	3	programador
13) codificação da função cadastro de preço de frete	4	programador
14) teste da função cadastro de preço de frete	1	programador
15) codificação da função geração de nota fiscal	20	programador
16) teste da função geração de nota fiscal	5	programador
17) testes gerais (caixa preta);	5	analista/programador

18) implantação	7	analista/programador
19) treinamento dos usuários	5	analista/usuário
21) operacionalização do sistema	3	operador

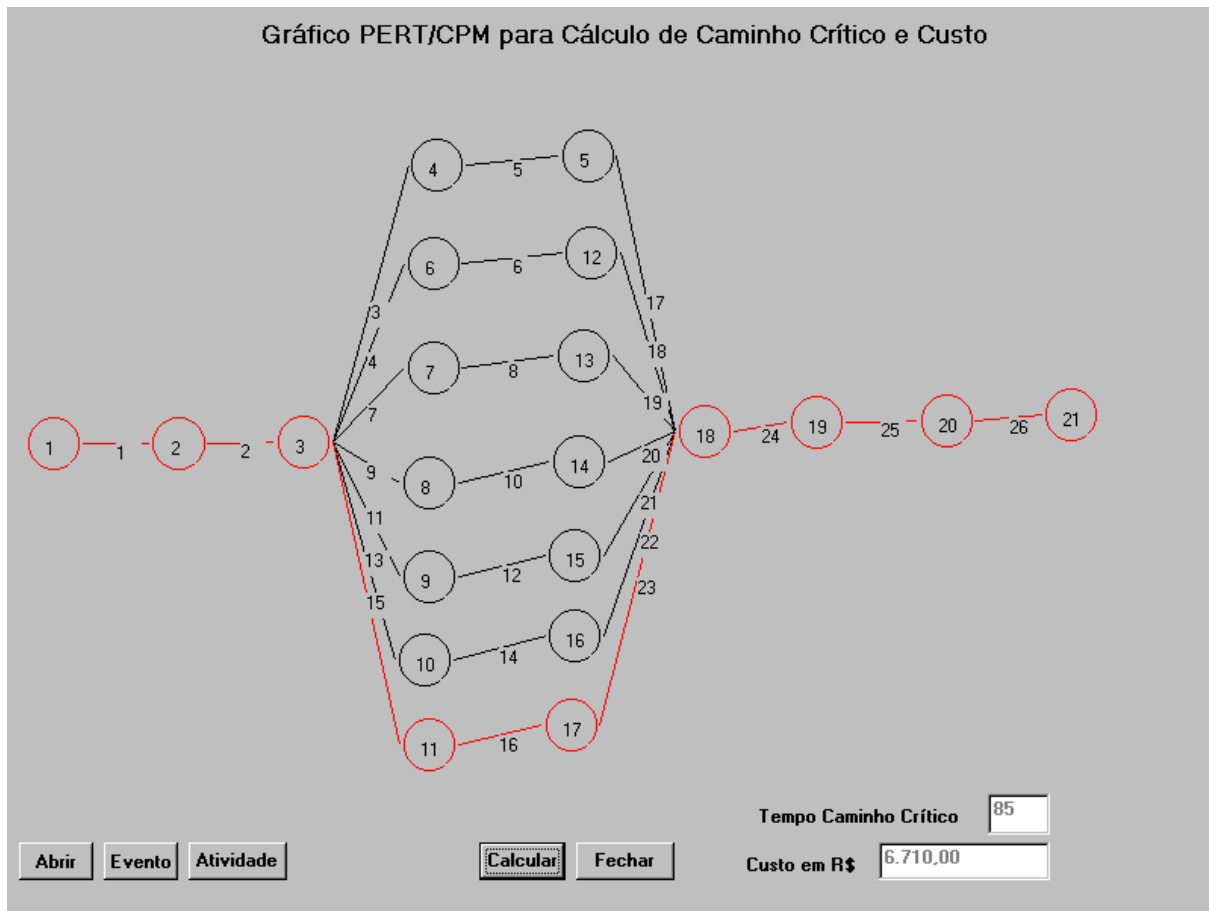
Após as atividades identificadas gera-se pelo protótipo, a rede PERT/CPM na função diagrama, de onde obtêm-se como resultado a figura 24 mostrada abaixo.

Figura 24: Gráfico da Rede para Sistema de Transportadoras



Com a rede PERT gerada, pode-se calcular o caminho crítico e o custo da mesma, para isto deve-se utilizar a função calcular. Executando-se esta função sobre a rede PERT do gráfico acima, obtêm-se como resultado a figura 25 mostrada abaixo, onde aparecerá em cor vermelha os eventos e atividades do caminho crítico e aparecerá também o total de dias do caminho crítico e o custo total do projeto no canto inferior direito.

Figura 25:Gráfico da rede para Sistema de Transportadoras calculado



## 6 CONCLUSÃO

Até há bem pouco tempo a palavra “Planejamento” não era considerada algo de relevante importância. Hoje esta função passou a influenciar diretamente os processos organizacionais e os planejadores começaram a trabalhar em contato constante com o pessoal de execução, reconhecendo o planejamento como parte integrante do processo global.

Embora o problema da integração entre planejamento e controle como variáveis interdependentes já tivesse sido despertado, ainda assim essa conjugação era feita assistematicamente, sem um mecanismo disciplinador.

O PERT/CPM é uma boa técnica, tornando-se um dos progressos do velho ideal dos administradores: o de um sistema integrado de administração, que completasse o ciclo das atividades administrativas numa seqüência unificada.

O protótipo alcançou o objetivo principal, que era o de permitir construir um gráfico PERT, com eventos e atividades, permitindo seqüenciá-las em ordem cronológica de execução, dando para isto total liberdade ao planejador, e ao final da construção do gráfico, calcular o caminho crítico e o custo do projeto.

Portanto, o protótipo construído neste trabalho aumenta-se ainda mais a ênfase em planejamento para construção de softwares, e ele vem de encontro a este objetivo, permitindo o planejamento e o controle na construção de sistemas de informação, utilizando a técnica PERT/CPM e a metodologia de análise estruturada de sistemas para alcançar este objetivo.

### 6.1 CONSIDERAÇÕES FINAIS

Uma das etapas com a qual dever-se-ia gastar, se não a maior, mas uma boa quantidade de tempo, é com o planejamento e conseqüentemente o controle no desenvolvimento de softwares aplicativos, e é exatamente este ponto que o protótipo vem atender, planejar para fazer softwares cada vez mais eficientes e eficazes, reduzindo, em contrapartida, cada vez mais o retrabalho por causa de um planejamento mal efetuado.

Vê-se o protótipo, apenas como um ponto de partida, muitas tarefas a mais poderiam ser incorporadas ao mesmo, tais como:

- a) abrir a possibilidade de se trabalhar com datas cedo e tarde da técnica PERT/CPM;
- b) implementar a aceleração do projeto;
- c) implementar o cálculo do tempo das atividades partindo de custo desejado informado;
- d) inúmeras outras possibilidades de relatórios.



## REFERÊNCIAS BIBLIOGRÁFICAS

- [BEL1974] BELCHIOR, Procópio G. O.. **Métodos de caminho crítico (PERT/CPM) na administração de projetos.** Rio de Janeiro : Ed. Americana, 1974.
- [CUK1978] CUKIERMAN, Sigmundo Salomão. **O modelo PERT/CPM aplicado a projetos :** Rio de Janeiro : Rio, 1978.
- [DAV1987] DAVIS, William S. **Análise e projeto de sistemas; uma abordagem estruturada.** Rio de Janeiro : Livros Técnicos e Científicos Editora S.A., 1987.
- [FED1967] FEDERAL ELECTRIC CORPORATION. **Uma introdução programada ao PERT técnica de avaliação e revisão de programas.** São Paulo : Livraria Pioneira Editora, 1967.
- [GAN1983] GANE, Chris; TRISH, Sarson. **Análise estruturada de sistemas.** Rio de Janeiro : Livros Técnicos e Científicos Editora S.A., 1983.
- [GEN1966] GENARI, Breno. **Introdução ao PERT básico.** Rio de Janeiro : Fundação Getúlio Vargas, 1966.
- [HIR1978] HIRSCHFELD, Henrique. **Planejamento com PERT-CPM e análise do desempenho.** Rio de Janeiro : Atlas, 1978
- [KEL1990] KELLER, Robert. **Análise estruturada na prática.** São Paulo : McGraw-Hill Ltda, 1990.
- [LOE1999] LOESCH, Cláudio; HEIN, Nelson. **Pesquisa operacional: fundamentos e modelos..** Blumenau : Ed. da Furb, 1999.
- [LOG1997] LOGIC WORKS. **Erwin/ERX 3.0** São Paulo : Makron Books, 1997.

- [MIC1999] MICROSOFT CORPORATION. **Microsoft Visual Basic 6.0. : guia do programador.** São Paulo : Makron Books, 1999.
- [NBR1994] ISO 9001/1994. **Sistemas da qualidade – modelo para garantia da qualidade em projeto, desenvolvimento, produção, instalação e serviços associados.** 1994.
- [PRA1998] PRADO, Darci Santos do. **Usando o ms project 98 em gerência de projetos.** Belo Horizonte : Editora de Desenvolvimento Gerencial, 1998.
- [STA1967] STANGER, Luiz B.. **PERT-CPM técnica de planejamento e controle.** Rio de Janeiro : Ao Livro Técnico, 1967.
- [VIE1998] VIASCAS, Jonh. **Microsoft Access 97 : guia autorizada Microsoft.** São Paulo : Makron Books, 1998.
- [YOU1990] JOURDON, EDWARD. **Análise estruturada moderna.** Rio de Janeiro : Campus Ltda, 1990.