

ALDINEI SAMPAIO
FABIO DE SÁ MAINARDES
MAURO ANDERSON FELÍCIO GUIMARÃES



MILLENIUM: GRADE HORÁRIA

Trabalho de Conclusão de Curso
apresentado à Disciplina de Projetos como
requisito parcial à conclusão do Curso de
Tecnologia em Informática, Setor Escola
Técnica, Universidade Federal do Paraná.

Orientador: Professor Dieval Guizelini

Co-orientador: Prof. Dr. Roberto Tadeu
Raittz

MG
005.1
5192

CURITIBA

2007

TERMO DE APROVAÇÃO

ALDINEI SAMPAIO
FÁBIO DE SÁ MAINARDES
MAURO ANDERSON FELÍCIO GUIMARÃES

MILLENIUM: GRADE HORÁRIA

Trabalho de Conclusão de Curso aprovado como requisito parcial para obtenção do grau de Tecnólogo em Informática no Curso de Tecnologia em Informática, Setor Escola Técnica, Universidade Federal do Paraná.

Orientador: Prof. Dieval Guizelini
Setor Escola Técnica, UFPR – Presidente da Banca

Prof^ª. Dr^ª. Jeroniza Nunes Marchaukoski
Setor Escola Técnica, UFPR – Membro da Banca

Prof. MsC. Mário de Paula Soares
Setor Escola Técnica, UFPR – Membro da Banca

Curitiba, 20 de dezembro de 2007.

RESUMO

Aplicativo desenvolvido a fim de demonstrar o desenvolvimento de um software para o curso de Tecnologia em Informática do setor da Escola Técnica da Universidade Federal do Paraná (ET-UFPR). Com o uso de metodologias e ferramentas de desenvolvimento de projetos de software, foi desenvolvido um aplicativo de Grade Horária. O sistema tem a finalidade de gerar qualquer tipo de horário para instituições de ensino de acordo com suas particularidades. Também conhecida como restrições ou preferências de professores, turmas e/ou disciplinas as restrições especificam o comportamento do programa tornando a solução mais lenta ou rápida. Nesse sistema gerar significa: localizar, pesquisar e registrar qualquer grade que seja considerada válida pela instituição. Trata-se de um Sistema *desktop* que utiliza recursos de IA (Inteligência Artificial), principalmente os conceitos relacionados a Algoritmo Genético para a resolução dos problemas.

Palavras-chave: Metodologia. Gerar. Grade Horária. Colisão. População. Seleção. Elitismo. Cruzamento. Mutação. Evolução. Indivíduo. Algoritmo Genético.

ABSTRACT

Application research is to present the development of a software for the course on Informatics Technology, of the Technical School of Parana Federal University (ET-UFPR). Using methodologies and tools for software projecting development, we developed a timetable generating program. This system is designed to generate any kind of timetable for educational organizations, even considering their peculiarities. Also known as teacher's preferences for classes or subjects, the restrictions specify the program's behavior, making its solution slower or faster. For this system, generating means: finding, searching and registering any timetable which the institution could consider appropriate. In fact, it is a desktop system that uses AI (Artificial Intelligence) features for solving problems, especially the concepts related to Genetic Algorithm.

Key-words: Methodology .Generate. Grade week. Collision. Population. Selection. Elitismo. Crossing. Change. Evolution. Individual. Genetic Algorithm.

LISTA DE FIGURAS

FIGURA 1 – MODELO DO PROBLEMA DE GRADE HORÁRIA MANUAL	14
FIGURA 2 – MODELO DO PROBLEMA DE GRADE HORÁRIA MANUAL	15
FIGURA 3 – DIAGRAMA DE EXECUÇÃO DE UM ALGORITMO GENÉTICO	26
FIGURA 4 – EXEMPLO DE CRUZAMENTO DE UM PONTO	30
FIGURA 5 – EXEMPLO DE MUTAÇÃO	31
FIGURA 6 – FLUXO DE DADOS ENTRE A TELA DE EXECUÇÃO E O GENÉTICO	32
FIGURA 7 – PSEUDO-CÓDIGO REPRESENTANDO O LOOP PRINCIPAL DE EXECUÇÃO DO GENÉTICO	33
FIGURA 8 – ESTRUTURA DO CROMOSSOMO	34
FIGURA 9 – ESTRUTURA DO MOLDE DO CROMOSSOMO	35
FIGURA 10 – ESTRUTURA DE UMA "TABELAMODELO"	36
FIGURA 11 – LINHA DO CROMOSSOMO COM LINHAS RELACIONADAS	37
FIGURA 12 – O ALELO HERDA AS RESTRIÇÕES DE TODOS OS ELEMENTOS QUE O COMPÕEM	38
FIGURA 13 – FÓRMULA USADA NO SORTEIO DE CROMOSSOMOS	41
FIGURA 14 – PONTO DE CORTE DOS CROMOSSOMOS 1 E 2 E ALELOS COPIADOS PARA O NOVO CROMOSSOMO	44
FIGURA 15 – NOVO CROMOSSOMO CRIADO POR MUTAÇÃO	46
FIGURA 16 – CASO DE USO DO SISTEMA MILLENIUM	55
FIGURA 17 – TELA DE SELEÇÃO DE INSTITUIÇÃO	58
FIGURA 18 – TELA DE CADASTRO DE INSTITUIÇÃO	59
FIGURA 19 – TELA PRINCIPAL	61
FIGURA 20 – TELA DE ALTERAÇÃO DE SENHA	61
FIGURA 21 – TELA DE MANUTENÇÃO DE USUÁRIO	65
FIGURA 22 – TELA DE MANUTENÇÃO DE CURSOS	67
FIGURA 23 – TELA DE MANUTENÇÃO DOS TURNOS	68
FIGURA 24 – TELA DE MANUTENÇÃO DE PROFESSOR	70
FIGURA 25 – TELA DE MANUTENÇÃO DE DISCIPLINA	72
FIGURA 26 – TELA DE MANUTENÇÃO DE TURMAS	73
FIGURA 27 – TELA DE EDIÇÃO DE AULAS	75
FIGURA 28 – TELA DE MANUTENÇÃO DE GRADE HORÁRIA	76
FIGURA 29 – TELA DE EDIÇÃO DOS DIAS	78
FIGURA 30 – TELA DE EDIÇÃO DE HORÁRIOS	79
FIGURA 31 – TELA DE GERÇÃO DE GRADE HORÁRIA	80
FIGURA 32 – TELA DE EXECUÇÃO DO GENÉTICO	81
FIGURA 33 – TELA RESULTADO	82
FIGURA 34 – TELA SALVAR RESULTADO	82
FIGURA 35 – TELA VISUALIZAR RESULTADO	84
FIGURA 36 – DIAGRAMA DE PACOTES	88
FIGURA 37 – RELACIONAMENTO DE GENERALIZAÇÃO	89
FIGURA 38 – RELACIONAMENTO DE GENERALIZAÇÃO	90
FIGURA 39 – RELACIONAMENTO DE DEPENDÊNCIA	91
FIGURA 40 – RELACIONAMENTO DE DEPENDÊNCIA	92
FIGURA 41 – RELACIONAMENTO DE DEPENDÊNCIA	93
FIGURA 42 – RELACIONAMENTO DE ASSOCIAÇÃO	94
FIGURA 43 – RELACIONAMENTO ASSOCIAÇÃO	94
FIGURA 44 – RELACIONAMENTO DE DEPENDÊNCIA	95
FIGURA 45 – RELACIONAMENTO DE DEPENDÊNCIA/ASSOCIAÇÃO	96
FIGURA 46 – RELACIONAMENTO DE DEPENDÊNCIA	97
FIGURA 47 – RELACIONAMENTO DE DEPENDÊNCIA	97
FIGURA 48 – DIAGRAMA DE SEQUENCIA EFETUAR LOGIN	98
FIGURA 49 – DIAGRAMA DE SEQUENCIA SELECIONAR INSTITUIÇÃO	99
FIGURA 50 – DIAGRAMA DE SEQUENCIA MANTER DADOS	100
FIGURA 51 – DIAGRAMA DE SEQUENCIA MANTER INSTITUIÇÃO	101

FIGURA 52 – DIAGRAMA DE SEQUENCIA DEFINIR SENHA	102
FIGURA 53 – DIAGRAMA DE SEQUENCIA MANTER USUÁRIO	102
FIGURA 54 – DIAGRAMA DE SEQUENCIA MANTER CURSO	103
FIGURA 55 – DIAGRAMA DE SEQUENCIA MANTER TURNO	103
FIGURA 56 – DIAGRAMA DE SEQUENCIA MANTER PROFESSOR	104
FIGURA 57 – DIAGRAMA DE SEQUENCIA MANTER DISCIPLINAS	104
FIGURA 58 – DIAGRAMA DE SEQUENCIA MANTER TURMA	105
FIGURA 59 – DIAGRAMA DE SEQUENCIA MANTER GRADE	105
FIGURA 60 – DIAGRAMA DE SEQUENCIA CRIAR GENETICO	106
FIGURA 61 – DIAGRAMA DE SEQUENCIA CRIAR GENÉTICO	106
FIGURA 62 – VISUALIZAR GRADE	107
FIGURA 63 – DIAGRAMA DE SEQUENCIA DESISTE DE <i>LOGIN</i> (A1).....	108
FIGURA 64 – DIAGRAMA DE SEQUENCIA CANCELAR SELEÇÃO DE INSTITUIÇÃO (A2).....	109
FIGURA 65 – DIAGRAMA DE SEQUENCIA CANCELAR ALTERAR SENHA (A3)	109
FIGURA 66 – DIAGRAMA DE SEQUENCIA EDITAR (A4)	110
FIGURA 67 – DIAGRAMA DE SEQUENCIA CANCELAR (A5).....	110
FIGURA 68 – DIAGRAMA DE SEQUENCIA EXCLUIR (A6).....	111
FIGURA 69 – DIAGRAMA DE SEQUENCIA EDITAR RESTRICOES (A7).....	111
FIGURA 70 – DIAGRAMA DE SEQUENCIA EDITAR TUPLAS (A8)	112
FIGURA 71 – DIAGRAMA DE SEQUENCIA EDITAR DIAS	112
FIGURA 72 – DIAGRAMA DE SEQUENCIA EDITAR HORARIOS (A10).....	113
FIGURA 73 – DIAGRAMA DE SEQUENCIA INTERROMPER GENÉTICO (A11).....	114
FIGURA 74 – DIAGRAMA DE ATIVIDADES DO MILLENIUM.....	115
FIGURA 75 – DIAGRAMA DE ATIVIDADES SALVAR/ALTERAR	116
FIGURA 76 – DIAGRAMA DE ESTADO DO MILLENIUM	116
FIGURA 77 – MODELO ENTIDADE RELACIONAMENTO	117
FIGURA 78 – COMPOSIÇÃO BÁSICA DO PROJETO GRADE – WBS	145
FIGURA 79 – COMPOSIÇÃO DO PLANEJAMENTO – WBS	145
FIGURA 80 – COMPOSIÇÃO DA ELABORAÇÃO – WBS.....	146
FIGURA 81 – COMPOSIÇÃO DA CONSTRUÇÃO – WBS	146
FIGURA 82 – DIAGRAMA DE GANTT	147

LISTA DE GRÁFICOS

GRÁFICO 1 – PROPORÇÃO GERALMENTE USADA ENTRE OS CROMOSSOMOS CRIADOS DE UMA NOVA POPULAÇÃO	40
GRÁFICO 2 – COMPORTAMENTO DA FÓRMULA DE SORTEIO DO CROMOSSOMO	42

LISTA DE TABELAS

TABELA 1 - ESCOLA ESTADUAL NO PARANÁ POR MODALIDADE.....	16
TABELA 2 - RELAÇÃO DE RISCO DO TRABALHO CONTINUA.....	18
TABELA 3 - USUARIO.....	118
TABELA 4 - CONFIGURACAO.....	119
TABELA 5 - INSTITUICAO.....	119
TABELA 6 - CURSO.....	119
TABELA 7 - TURNO.....	120
TABELA 8 - GRADE.....	120
TABELA 9 - DIA.....	121
TABELA 10 - INTERVALO.....	121
TABELA 11 - DISCIPLINA.....	122
TABELA 12 - PROFESSOR.....	122
TABELA 13 - TURMA CONTINUA.....	122
TABELA 14 - TURMA CONCLUSÃO.....	123
TABELA 15 - TUPLA.....	123
TABELA 16 - RESTRICAO.....	124
TABELA 17 - RESULTADO.....	124
TABELA 18 - RESULTADODIMENSAO.....	125
TABELA 19 - RESULTADOFATO CONTINUA.....	125
TABELA 20 - MODELO.....	126
TABELA 21 - RELAÇÃO DE MICRO-COMPUTADORES UTILIZADOS.....	140
TABELA 22 - CALENDÁRIO DE ATIVIDADES DO PROJETO.....	144
TABELA 23 - WT CRIAÇÃO DE TABELAS.....	148
TABELA 24 - WT PERMISSÕES DO USUÁRIO.....	149
TABELA 25 - WT TROCA DE SENHA DE USUÁRIO.....	150
TABELA 26 - WT MANUTENÇÃO DE USUÁRIO.....	151
TABELA 27 - WT MANUTENÇÃO DE INSTITUIÇÃO.....	152
TABELA 28 - WT SELEÇÃO DE INSTITUIÇÃO.....	153
TABELA 29 - WT MANUTENÇÃO DO CURSO.....	154
TABELA 30 - WT MANUTENÇÃO DO TURNO.....	155
TABELA 31 - WT MANUTENÇÃO DE GRADE HORÁRIA.....	156
TABELA 32 - WT MANUTENÇÃO DO PROFESSOR.....	157
TABELA 33 - WT MANUTENÇÃO DA DISCIPLINA.....	158
TABELA 34 - WT MANUTENÇÃO DA TURMA.....	159
TABELA 35 - WT MANUTENÇÃO DA TUPLA.....	160
TABELA 36 - WT MANUTENÇÃO DE RESTRIÇÃO.....	161
TABELA 37 - WT MANUTENÇÃO DO DIA.....	162
TABELA 38 - WT MANUTENÇÃO DOS HORÁRIOS.....	163
TABELA 39 - WT GERAR GRADE HORÁRIA.....	164
TABELA 40 - WT USO DE RECURSOS.....	165
TABELA 41 - TESTES REALIZADOS COM O MILLENIUM.....	167
TABELA 42 - TESTES REALIZADOS COM O ASC TIMETABLES.....	167
TABELA 43 - TESTES REALIZADOS COM O PROGRAMA A.....	168
TABELA 44 - TESTES REALIZADOS COM O PROGRAMA A.....	169
TABELA 45 - TESTES REALIZADOS COM O ASC TIMETABLES.....	169
TABELA 46 - TESTES REALIZADOS COM O PROGRAMA A.....	170

SUMÁRIO

1. TEMA	13
2. PROBLEMA	14
3. DELIMITAÇÃO	16
3.1. PÚBLICO-ALVO DO TRABALHO	16
3.2. IMPORTÂNCIA DO TRABALHO	17
3.3. LIMITAÇÃO DO TRABALHO	18
3.4. RISCO DO TRABALHO	18
4. JUSTIFICATIVA	20
5. OBJETIVOS	21
5.1. OBJETIVO GERAL	21
5.2. OBJETIVOS ESPECÍFICOS	21
6. REVISÃO DE LITERATURA	22
6.1. GERAÇÃO DE HORÁRIO.....	22
6.2. INTRODUÇÃO A ALGORITMO GENÉTICO.....	23
6.2.1 Definição de algoritmo genético.....	24
6.2.2 Características do algoritmo genético.....	25
6.2.3 Componentes do algoritmo genético	25
6.2.4 Funcionamento do algoritmo genético	26
6.2.4.1 Codificação do indivíduo	27
6.2.4.2 Sorteio e aleatoriedade	27
6.2.4.3 Base de dados	27
6.2.4.4 Criação da população inicial	28
6.2.5 Operadores genéticos.....	28
6.2.5.1 Seleção	29
6.2.5.2 Elitismo.....	29
6.2.5.3 Cruzamento.....	29
6.2.5.4 Mutação	30
6.2.5.5 Evolução	31
7. METODOLOGIA	32
7.1. O MODELO GENÉTICO	32
7.1.1. Funcionamento Geral	32
7.1.2. A classe genético.....	33
7.1.3. A classe população.....	33
7.1.4. A classe cromossomo.....	33
7.1.5. O molde	35
7.1.6. O cálculo da "fitness"	37
7.1.7. O ciclo de vida do algoritmo genético	39
7.1.8. O funcionamento da "iteração".....	40
7.1.8.1 Sorteio.....	41
7.1.8.2 Cruzamento.....	42
7.1.8.3 Mutação	45
7.2. ANÁLISE DE REQUISITOS.....	46
7.3. DESCRIÇÃO DAS REGRAS DO NEGÓCIO	47
7.3.1 Efetuar login.....	47

7.3.2	Selecionar instituição	48
7.3.3	Manter instituição	48
7.3.4	Definir senha	49
7.3.5	Manter usuário	49
7.3.6	Manter curso	50
7.3.7	Manter turno	50
7.3.8	Manter professor	50
7.3.9	Manter disciplina	51
7.3.10	Manter turmas	51
7.3.11	Manter grade	52
7.3.12	Edição de restrições	52
7.3.13	Editar dias	53
7.3.14	Editar horários	53
7.3.15	Editar tuplas	54
7.3.16	Gerar grade	54
7.4.	DIAGRAMA DE CASO DE USO DO SISTEMA	55
7.5.	DESCRIÇÃO DOS CASOS DE USO DO SISTEMA MILLENIUM	56
7.5.1	Caso de uso 1: efetuar login	56
7.5.2	Caso de uso 2: selecionar instituição	57
7.5.3	Caso de uso 3: manter instituição	59
7.5.4	Caso de uso 4: Definir Senha	60
7.5.5	Caso de uso 5: manter dados	63
7.5.6	Caso de uso 6: Manter Usuario	64
7.5.7	Caso de uso 7: Manter Curso	66
7.5.8	Caso de uso 8: Manter Turno	68
7.5.9	Caso de uso 9: Manter Professor	69
7.5.10	Caso de uso 10: manter disciplina	71
7.5.11	Caso de uso 11: manter turmas	73
7.5.12	Caso de uso 12: manter grade	76
7.5.13	Caso de uso 13: gerar grade	80
7.5.14	Caso de uso 14: visualizar grade	84
7.6.	CENÁRIO	85
7.7.	DIAGRAMAS DO PROJETO	86
7.7.1	Diagrama de pacotes do millenium	88
7.7.2	Relacionamento de Generalização	89
7.7.3	Diagrama de classes do pacote	91
7.7.3.1	Diagrama de classes do pacote dados	91
7.7.3.2	Diagrama de classes do pacote genético	93
7.7.3.3	Diagrama de classes do pacote modelo	94
7.7.3.4	Diagrama de classes do pacote telas	95
7.7.3.5	Diagrama de classes do pacote resultados	96
7.7.3.6	Diagrama de classes do pacote util	97
7.7.3.7	Diagrama de classes do pacote principal	97
7.7.4	Diagrama de seqüência	98
7.7.4.1	Diagrama de seqüência efetuar login	98
7.7.4.2	Diagrama de seqüência selecionar instituição	99
7.7.4.3	Diagrama de seqüência manter dados	100
7.7.4.4	Diagrama de seqüência manter instituição	101
7.7.4.5	Diagrama de seqüência definir senha	102
7.7.4.6	Diagrama de seqüência manter usuário	102

7.7.4.7	Diagrama de seqüência manter curso.....	103
7.7.4.8	Diagrama de seqüência manter turno	103
7.7.4.9	Diagrama de seqüência manter professor	104
7.7.4.10	Diagrama de seqüência manter disciplina.....	104
7.7.4.11	Diagrama de seqüência manter turma	105
7.7.4.12	Diagrama de seqüência manter grade	105
7.7.4.13	Diagrama de seqüência criar genético	106
7.7.4.14	Diagrama de seqüência criar genético	106
7.7.4.15	Diagrama de seqüência visualizar grade	107
7.7.4.16	Diagrama de seqüência desiste de login (A1).....	108
7.7.4.17	Diagrama de seqüência cancelar seleção de instituição (A2).....	109
7.7.4.18	Diagrama de seqüência cancela alterar senha (A3).....	109
7.7.4.19	Diagrama de seqüência editar (A4).....	110
7.7.4.20	Diagrama de seqüência cancelar (A5)	110
7.7.4.21	Diagrama de seqüência excluir (A6)	111
7.7.4.22	Diagrama de seqüência editar restrições (A7)	111
7.7.4.23	Diagrama de seqüência editar tuplas (A8)	112
7.7.4.24	Diagrama de seqüência editar dias (A9)	112
7.7.4.25	Diagrama de seqüência editar horários (A10).....	113
7.7.4.26	Diagrama de seqüência interromper genético (A11).....	114
7.7.5	Diagramas de atividade	114
7.7.5.1	Diagramas de atividade do millenium.....	115
7.7.5.2	Diagrama de atividades salvar/alterar	116
7.7.6	Diagrama de estado do millenium.....	116
7.8.	MODELO ENTIDADE RELACIONAMENTO	117
7.9.	DICIONÁRIO DE DADOS	118
7.9.1	Descrição dos campos das tabelas	118
7.9.1.1	Tabela usuario	118
7.9.1.2	Tabela configuracao.....	119
7.9.1.3	Tabela instituicao	119
7.9.1.4	Tabela curso	119
7.9.1.5	Tabela turno	120
7.9.1.6	Tabela grade	120
7.9.1.7	Tabela dia	121
7.9.1.8	Tabela intervalo.....	121
7.9.1.9	Tabela disciplina.....	122
7.9.1.10	Tabela professor	122
7.9.1.11	Tabela turma	122
7.9.1.12	Tabela tupla	123
7.9.1.13	Tabela restricao.....	124
7.9.1.14	Tabela resultado.....	124
7.9.1.15	Tabela resultadodimensao continua.....	125
7.9.1.16	Tabela resultadofato.....	125
7.9.1.17	Tabela modelo	126
7.9.2	Script para criação das tabelas.....	127
7.9.2.1	Script tabela usuario.....	127
7.9.2.2	Script tabela configuracao.....	127
7.9.2.3	Script tabela instituicao	128
7.9.2.4	Script tabela curso.....	128
7.9.2.5	Script tabela turno	129

7.9.2.6	Script tabela grade	129
7.9.2.7	Script tabela dia.....	130
7.9.2.8	Script tabela intervalo.....	131
7.9.2.9	Script tabela disciplina.....	131
7.9.2.10	Script tabela professor	132
7.9.2.11	Script tabela turma	132
7.9.2.12	Script tabela tupla.....	133
7.9.2.13	Script tabela restricao.....	134
7.9.2.14	Script tabela resultado.....	135
7.9.2.15	Script tabela resultadodimensao	136
7.9.2.16	Script tabela resultadofato.....	136
7.9.2.17	Script tabela modelo.....	138
8.	RECURSOS.....	139
8.1.	HARDWARES UTILIZADOS	139
8.1.1	Microcomputadores utilizados.....	139
8.2.	SOFTWARES UTILIZADOS.....	140
8.2.1	Eclipse	140
8.2.2	Netbeans.....	141
8.2.3	Jude.....	141
8.2.4	TortoiseSVN.....	141
8.2.5	Derby	142
8.2.6	JasperReport	142
8.2.7	DBDesigner 4.....	142
8.3.	OUTROS SOFTWARES UTILIZADOS	143
8.3.1	MS-Word.....	143
8.3.2	MS-PowerPoint.....	143
9.	CRONOGRAMA	144
9.1.	CALENDÁRIO DAS ATIVIDADES	144
9.2.	DIAGRAMA DE GANTT	145
9.2.1	Composição básica do projeto.....	145
9.2.2	Composição do planejamento - WBS	145
9.2.3	Composição da elaboração - WBS	146
9.2.4	Composição da construção - WBS	146
9.2.5	Diagrama de gantt aplicado ao projeto	147
9.3.	WORKFLOW DE TESTE	148
9.3.1	Teste 1: criação de tabelas.....	148
9.3.2	Teste 2: permissões do usuário.....	149
9.3.3	Teste 3: troca de senha do usuário.....	150
9.3.4	Teste 4: manutenção de usuário.....	151
9.3.5	Teste 5: manutenção de instituição.....	152
9.3.6	Teste 6: seleção de instituição.....	153
9.3.7	Teste 7: manutenção do curso.....	154
9.3.8	Teste 8: manutenção do turno.....	155
9.3.9	Teste 9: manutenção da grade horária	156
9.3.10	Teste 10: manutenção do professor.....	157
9.3.11	Teste 11: manutenção da disciplina.....	158
9.3.12	Teste 12: manutenção da turma.....	159
9.3.13	Teste 13: manutenção da tupla.....	160
9.3.14	Teste 14: manutenção da restrição.....	161
9.3.15	Teste 15: manutenção do dia.....	162

9.3.16	Teste 16: manutenção dos horários.....	163
9.3.17	Teste 17: gerar grade horária.....	164
9.3.18	Teste 18: uso de recursos.....	165
10.	TESTES COMPUTACIONAIS	166
10.1.	ESTUDO DE CASO 1	166
10.2.	ESTUDO DE CASO 2	168
10.3.	CONCLUSÃO DOS ESTUDOS DE CASOS	170
11.	CONCLUSÃO	171
	REFERÊNCIAS BIBLIOGRÁFICAS.....	173
	DOCUMENTOS CONSULTADOS	176
	APÊNDICE	177

1. TEMA

Este trabalho tem por tema tratar da geração de Grade Horária em instituições de ensino por meio da abordagem de algoritmos evolutivos para o encontro de soluções de boa qualidade. É descrito também o uso de metodologia e tecnologias empregadas, ministradas ou não no Curso de Tecnologia em Informática da UFPR, para atestar o conhecimento dos estudantes na sua área de formação. Na busca de uma melhor solução para essa demanda foi desenvolvido um aplicativo desktop baseado em algoritmo genético chamado Millenium.

2. PROBLEMA

A construção da grade horária para instituições educacionais é um problema clássico. Em instituições de ensino, uma grade horária é algo que influi de forma notável na vida de todo o corpo docente, discente e dos funcionários da instituição. Uma vez elaborada e implementada essa grade, ela será válida durante todo o período letivo, o que fará, por várias vezes, com que alunos e professores tenham que se adaptar a ela, já que não necessariamente atende a todos os interesses e disponibilidades dos envolvidos.

A elaboração do horário escolar em instituições de ensino é realizada geralmente de forma manual, conforme figura 1, exigindo um considerável tempo para sua conclusão. Isso ocorre, principalmente, porque é necessário respeitar um conjunto grande de restrições para que o horário obtido satisfaça as necessidades dos envolvidos: os professores e a instituição.



FIGURA 1 – MODELO DO PROBLEMA DE GRADE HORÁRIA MANUAL

FONTE: O autor (2007)

Neste contexto, as aulas de uma instituição são o conjunto de reuniões entre professores, disciplinas e turmas num conjunto de períodos de tempo. Portanto, o problema de construção de grades horárias pode ser visto como relações entre conjuntos de professores, disciplinas, alunos, períodos de tempo e recursos. Estas relações estão sujeitas ainda a uma série de restrições, cujas combinações aumentam a complexidade do problema.

A representação do problema mediante um software, conforme idéia da figura 2, possibilita a exclusão da pessoa geradora da informação manual. Os dados são acolhidos manualmente por meio de questionário, mas processado por uma

máquina. O processo torna-se mais ágil e com resultados melhores na confecção da grade horária escolar.



FIGURA 2 – MODELO DO PROBLEMA DE GRADE HORÁRIA MANUAL

FONTE: O autor (2007)

O meio utilizado para efetuar o processamento foi implementar o programa utilizando conceitos de algoritmo genético para encontrar uma solução viável para o problema da geração de grades horárias, dentro de uma série de combinações e restrições existentes. Esse conceito nada mais é do que uma técnica de busca heurística que faz uma analogia com as teorias de evolução natural e genética.

O processo de busca de uma solução é feita mediante operadores genéticos que fazem o papel da evolução natural, seguindo limites definidos que são chamados de parâmetros genéticos. As restrições do problema são transformados em funções que avaliam cada geração de indivíduos, possibilitando ou não a evolução dos mesmos.

O aplicativo será composto pelos componentes: algoritmo genético, responsável pela implementação das operações genéticas (seleção, elitismo, mutação e cruzamento) e também pelo processo como um todo (início/fim); base de dados, local onde é armazenado o conhecimento necessário para processamento do problema em questão; lista de disciplinas, professores e suas correspondências; restrições de dias e horários para cada aula; número de dias na semana e preferências de dias e horários de cada professor.

3. DELIMITAÇÃO

Esta parte do projeto preocupa-se em apresentar de forma mais detalhada os itens relacionados ao Millenium de acordo com o seu público, sua importância, sua limitação e seu risco. Nesse sentido é apresentada a tabela 1 para dar uma dimensão do universo disponível para o uso da aplicação.

TABELA 1 - ESCOLA ESTADUAL NO PARANÁ POR MODALIDADE

Totais de Escolas do Estado por Ensino

Data SABI: 08/12/2007 12:14

Núcleo Regional de Educação	Total de Escolas	Modalidades de Ensino										
		Educ Infantil	Ensino Fundamental		Ensino Médio	Educação de Jovens e Adultos - Presencial			Educ Especial	Educação de Jovens e Adultos - CEEBJA		
			1/4 Série	5/8 Série		Fase I 1/4	Fase II 5/8	Ens Médio		Fase I 1/4	Fase II 5/8	Ens Médio
Total do Estado	2108	14	143	1846	1240	0	112	139	106	85	85	71

FONTE: Replica SAE e SABI (2007)

3.1. PÚBLICO-ALVO DO TRABALHO

Durante este trabalho o público-alvo da pesquisa foi as Escolas Estaduais do Paraná, focando as instituições do Ensino Fundamental do período da 5ª à 8ª série e o Ensino Médio. A escolha foi feita por razões de complexidades apresentadas nessa faixa do ensino, principalmente com relação à diversidade de disciplinas, número de turmas e de professores. Especificidades importantes para o desenvolvimento do projeto.

3.2. IMPORTÂNCIA DO TRABALHO

A importância do trabalho fica evidente na Tabela 1 em virtude do alto número de instituições que pertencem ao público, uma vez que esse número é superior pelo menos a 1846 escolas. Não é possível precisar o número exato, pois o Governo do Paraná não divulgou uma relação com a quantidade desse público, já que na Tabela 1 um colégio do Ensino Fundamental pode também ter Ensino Médio gerando duplicidade de informações. Porém pela tabela podemos identificar que o número de escolas estaduais é de 2108 até o presente momento.

Esse universo tornar-se importante uma vez que ele gera uma fatia interessante no mercado que pode gerar receitas, porém muito monopolizada quando levada em consideração à participação de soluções para o problema de Grade Horária. A maioria das instituições utiliza, quando não fazem manualmente, um aplicativo disponibilizado por uma empresa. Ela cobra uma licença anualmente no valor mínimo de R\$ 440,00 para o uso do aplicativo, podendo variar de acordo com a complexidade da instituição. Fazendo uma operação básica nesses números temos um montante de R\$ 812.240,00 para 1846 escolas. Vale salientar que a prática de pirataria é comum também nesse meio, porém é um valor que não pode ser desconsiderado mesmo que na prática seja inferior ao calculado.

Não somente com relação ao valor econômico, mas também a questões operacionais e sociais esse trabalho é benéfico. A economia de um valor como esse significa na prática para o governo estadual a alocação de receitas em outras áreas, produzindo mesmo que indiretamente outros benefícios para população, como, por exemplo, saúde, alimentação e habitação. Dentro da própria instituição de ensino existem vantagens, como a possibilidade de desprender mão de obra de áreas operacionais e alocando esses recursos na sala de aula, gerando economia de novos funcionários e produzindo principalmente geração de conhecimento.

A importância do projeto é ampla e justifica sua concepção e implementação tanto do ponto de vista econômico, operacional e social.

3.3. LIMITAÇÃO DO TRABALHO

Consideramos, como base para nosso trabalho, o problema encontrado em escolas públicas do Estado do Paraná do ensino Fundamental e Médio. Onde o problema é tratar o agendamento semanal do encontro entre professores e turmas.

Nessas escolas a grade é dividida em turnos (manhã, tarde e noite), cada turno com n dias de aula por semana (geralmente cinco de segunda à sexta) e cada dia possuindo um número fixo de aulas.

Outra característica das instituições de ensino da rede pública é de que seus professores podem lecionar para um conjunto de turmas mais de uma disciplina. As aulas para as turmas ocorrem em uma única sala, sendo esta determinada no início do ano letivo, essas turmas estão sempre disponíveis em seu turno e precisam ter sua grade totalmente preenchida.

Não é levada em consideração, no modelo proposto, a associação de turma-sala, deslocamento entre sedes entre outras características que são aplicadas a instituições de ensino da rede particular ou do ensino superior.

3.4. RISCO DO TRABALHO

Foram levantados os seguintes riscos relacionados ao desenvolvimento deste trabalho conforme tabela 2:

TABELA 2 – RELAÇÃO DE RISCO DO TRABALHO *continua*

Nr	Descrição	Classificação	Estratégia	Resposta Proposta
1	Desistência de algum integrante da equipe	Alta	Aceitar	Redefinir atividades
2	Atraso no cronograma	Média	Prevenir	Reuniões semanais para conferir as atividades

RELAÇÃO DE RISCO DO TRABALHO conclusão

Nr	Descrição	Classificação	Estratégia	Resposta Proposta
3	Dificuldade no manuseio de alguma ferramenta utilizada.	Média	Mitigar	Sanar dúvidas com professores. Participar de treinamentos e cursos de capacitação.
4	Problemas com equipamentos	Média	Aceitar	Manutenção, manter backup das informações.
5	Perda de dados	Alta	Prevenir	- Realizar backup ao final de cada tarefa

FONTE: O autor (2007).

4. JUSTIFICATIVA

A educação faz parte de qualquer plano de governo de um País. Porém nos últimos anos o enfoque tem sido maior, com a valorização do conhecimento para o processo de desenvolvimento da população.

A informática tem um papel de facilitar o acesso à educação, privilegiando meios de acessos e criando mecanismos para facilitar o cotidiano dos profissionais e estudantes inseridos nessa área. Pensando nisso este projeto foi desenvolvido com o intuito de oferecer uma opção de agilizar o processo de grades horárias nas escolas e desprender pessoas para outras atividades, principalmente relacionadas com o ensino. A idéia é que um professor ou um diretor não perca o seu tempo com a distribuição de horários, mas priorize seu tempo em ensinar.

O problema da distribuição de horários conhecidos na literatura como *Timetabling*, consiste em criar uma agenda distribuindo o tempo de cada profissional por diferentes atividades e em diferentes locais por um período de tempo pré-fixado (no caso de horários escolares, normalmente este período é de uma semana). Esta distribuição deve ser feita de modo a satisfazer um conjunto de restrições.

A solução manual para o problema, além de ser demorada, pode ocasionar grades horárias não ideais, onde uma ou mais restrições podem não ser atendidas.

Já existem no mercado hoje outros sistemas que se dedicam a solucionar este problema, mas em geral trata-se de sistemas proprietários, que exigem a compra ou aluguel de licenças.

5. OBJETIVOS

Os objetivos do presente trabalho desdobram-se em um de caráter geral e quatro específicos.

5.1. OBJETIVO GERAL

Criar um produto de software capaz de atender à necessidade de geração de grade horária utilizando-se de algoritmos genéticos, que utilizam conceitos do princípio de seleção natural proposto por *Darwin*, aliado aos princípios da genética propostos por *Mendel*, para abordar uma série ampla de problemas, especialmente os de otimização.

5.2. OBJETIVOS ESPECÍFICOS

Os objetivos específicos do presente trabalho são:

- Analisar o funcionamento das instituições de ensino, através de entrevistas, para levantar os problemas encontrados por elas no momento da geração da grade horária.
- Verificar como é o funcionamento de sistemas já existentes no mercado, seus pontos fortes e suas falhas.
- Construir uma análise baseada na UML (*Unified Modelling Language*) e desenvolver a solução utilizando a linguagem Java.
- Por fim, apresentam-se as considerações finais deste trabalho frente ao conteúdo pesquisado.

6. REVISÃO DE LITERATURA

Esta seção tem por objetivo fundamentar teoricamente o projeto em pesquisa.

6.1. GERAÇÃO DE HORÁRIO

Para se desenvolver um método eficiente para a Grade Horária é necessário conhecer bem a organização de uma instituição de ensino.

Em uma instituição de ensino, a geração da grade horária deve levar em consideração diversos fatores. Entre eles podemos destacar os seguintes:

1. em quantos períodos a instituição trabalha;
2. quantas turmas existem na instituição;
3. quais disciplinas são ministradas, com quais turmas elas se relacionam e suas peculiaridades, como por exemplo, limitações de horários a serem ministradas;
4. disponibilidade dos professores e suas preferências;
5. deve-se levar em consideração o número de aulas por período e o número de dias da semana.

Segundo GERALDO FILHO (2001, p. 87)

Essas aulas são ministradas por um conjunto de professores que trabalham na escola. Cada professor tem seu próprio número de aulas. Muitas vezes os professores trabalham em mais de uma escola e em cada uma delas ministram um diferente número de aulas. Uma escola pode ter professores que trabalhem em um único período e professores que trabalhem em mais de um período.

As principais características das instituições de ensino público no Brasil são que todas as salas disponíveis são utilizadas, os alunos têm aulas em todos os horários do período em que estão matriculados, os professores ministram aulas para mais de uma turma em um determinado período e podem ministrar mais de uma

disciplina e conflitos de simultaneidade podem ocorrer tanto com turmas, disciplinas quanto com professores.

Para que o algoritmo genético garanta a viabilidade da solução temos que considerar uma intensidade para as restrições presentes na elaboração da Grade Horária, isto é aquela que venham gerar conflitos de simultaneidade. As restrições podem ser classificadas em:

Leves: aquelas ligadas às preferências dos professores como dias e horários, as penalizações para essas restrições são baixas, pois não afetam a viabilidade da solução.

Média: aquelas de complexidade maior que buscam soluções como distribuição uniforme das aulas, visam evitar aulas vazias durante o período para essas restrições as penalidades são médias ou altas.

Severas: essas influenciam diretamente na viabilidade da solução, pois estão ligadas as disponibilidades da solução cuidando de regras como disponibilidade de professores, capacidade de salas, carga horária, etc. Para essas restrições a penalidade é altíssima e em alguns casos segundo TOMAZELA (2003, p.28) “[...] servindo como Agente Filtrador [...]” descartando soluções não viáveis.

6.2. INTRODUÇÃO A ALGORITMO GENÉTICO

Algoritmos genéticos são algoritmos de otimização global, baseados nos mecanismos de seleção natural e da genética Goldberg (1989). Apesar de aleatórios, eles exploram informações históricas para encontrar novos pontos de busca onde são esperados melhores desempenhos. Isto é feito através de processos iterativos, onde cada iteração é chamada de geração.

Algoritmos Genéticos são muito eficientes para busca de soluções ótimas, ou aproximadamente ótimas, em uma grande variedade de problemas, pois não impõem muitas das limitações encontradas nos métodos de busca tradicionais. São capazes de identificar e explorar fatores ambientais e convergir para soluções ótimas, ou aproximadamente ótimas em níveis globais.

Os algoritmos genéticos empregam uma terminologia originada da teoria da evolução natural e da genética. Um indivíduo da população é representado por um único cromossomo, contendo a codificação (genótipo) de um candidato à solução do problema (fenótipo). Um cromossomo é usualmente implementado na forma de um vetor (lista de atributos), em que cada componente é conhecido como gene. Os possíveis valores que um determinado gene pode assumir são denominados alelos.

O processo de evolução executado por um algoritmo genético corresponde a um processo de busca em um espaço de soluções potenciais para alcançar o objetivo proposto.

6.2.1 Definição de algoritmo genético

De acordo com os autores a seguir, os algoritmos genéticos têm as seguintes definições:

Goldberg define algoritmo genético como:

...um algoritmo de procura baseado nos mecanismos de seleção natural e genética natural. Ele combina a sobrevivência feita por uma função de avaliação entre uma cadeia de caracteres com uma estrutura de informações mudadas aleatoriamente, para formar um algoritmo de procura com algum talento inovador, o mesmo de uma procura de um ser humano. Em toda geração, um novo conjunto de criaturas artificiais (cadeia de caracteres) é criado usando bits e pedaços do teste de avaliação da geração anterior; ocasionalmente uma parte nova é testada. Enquanto aleatório, algoritmos genéticos não são nenhum passeio simples sem destino. A procura mais eficiente das informações anteriores para especular os pontos da nova procura resultam e um aumento na sua performance.

Já Tanomaru define algoritmos genéticos como sendo:

...métodos computacionais de busca baseados nos mecanismos de evolução natural e na genética. Em algoritmos genéticos, uma população de possíveis soluções para o problema em questão evolui de acordo com operadores probabilísticos concebidos a partir de metáforas biológicas, de modo que há uma tendência de que, na média, os indivíduos representem soluções cada vez melhores à medida que o processo evolutivo continua.

6.2.2 Características do algoritmo genético

De um modo geral, os algoritmos genéticos têm as seguintes características:

- Operam numa população (conjunto) de pontos, e não a partir de um ponto isolado;
- Operam num espaço de soluções codificadas, e não no espaço de busca diretamente;
- Necessitam somente de informação sobre o valor de uma função objetivo para cada membro da população, e não requerem derivadas ou qualquer outro tipo de conhecimento;
- Usam transições probabilísticas, e não regras determinísticas.

6.2.3 Componentes do algoritmo genético

Um Algoritmo Genético deve conter os seguintes componentes:

- Uma representação genética para soluções candidatas ou potenciais (processo de codificação);
- Uma população inicial, gerada aleatoriamente;
- Função de avaliação que faz o papel da pressão ambiental, classificando as soluções de acordo com sua adaptação ao meio;
- Operadores genéticos;
- Valores para os parâmetros genéticos (tamanho da população, probabilidade de aplicação dos operadores, entre outros).

6.2.4 Funcionamento do algoritmo genético

Os algoritmos genéticos básicos funcionam da seguinte forma:

1. Determinar uma população inicial de cromossomos.
2. Avaliar cada indivíduo da população através do cálculo de sua aptidão, utilizando a função de avaliação.
3. Selecionar os cromossomos que produzirão descendentes para a próxima geração, em função dos seus desempenhos apurados no passo anterior. (etapa de seleção).
4. Gerar descendentes através da aplicação das operações de cruzamento e mutação sobre os cromossomos selecionados no passo anterior, criando a próxima geração. (etapa de reprodução).
5. Se o critério de parada for satisfeito, termina e retorna o melhor cromossomo até então gerado, senão volta ao passo 2.

A Figura. 3 mostra um diagrama da execução de um algoritmo genético padrão.

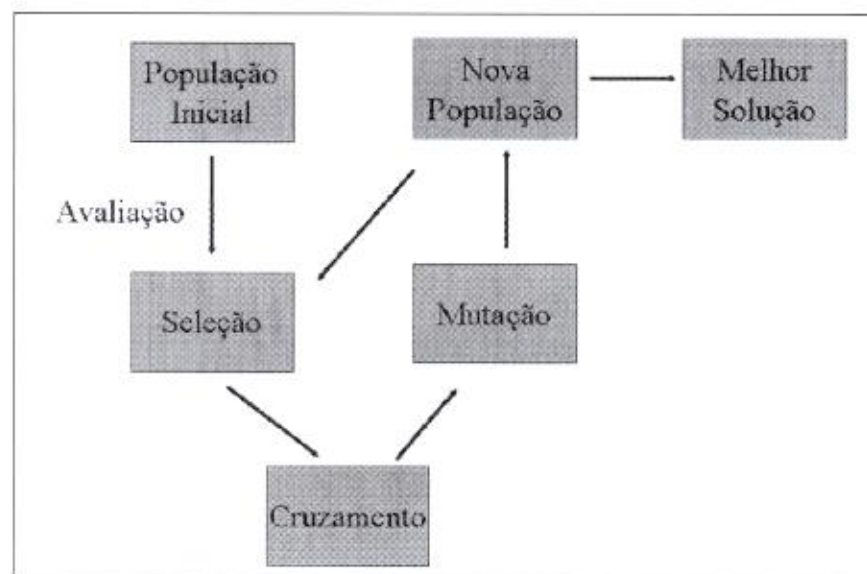


FIGURA 3 – DIAGRAMA DE EXECUÇÃO DE UM ALGORITMO GENÉTICO

FONTE: O autor (2007)

Durante cada geração, os princípios de seleção e reprodução são aplicados a uma população de candidatos que pode variar, dependendo da complexidade do problema e dos recursos computacionais disponíveis. Através da seleção, se determinam quais indivíduos conseguirão reproduzir, gerando um número determinado de descendentes para a próxima geração, com uma probabilidade determinada pelo seu índice de aptidão. Em outras palavras, os indivíduos com maior adaptação relativa têm maiores chances de se reproduzir.

6.2.4.1 Codificação do indivíduo

O indivíduo contém uma grade de horários, onde cada célula contém o código da turma com o professor responsável pela mesma. Cada indivíduo também contém o valor de quantidade de aulas, turno e disciplinas ministradas.

6.2.4.2 Sorteio e aleatoriedade

Para garantir resultados corretos e precisos em problemas onde é utilizada a Computação Evolucionária como solução, é utilizada a geração de números aleatórios.

6.2.4.3 Base de dados

A base de dados da ferramenta para gerar a grade horária, consiste de

tabelas como seguem:

- Dados das Disciplinas: Contém as informações referentes às disciplinas, tais como: abreviação e descrição.
- Dados dos Professores: Contém as informações referentes aos professores, tais como: abreviação e descrição.
- Dados das Turmas: Contém as informações referentes às turmas, tais como: abreviação e descrição.
- Dados das Tuplas: Trata-se do relacionamento do professor com a disciplina em relação à turma. Contém as informações referentes aos professores, tais como: abreviação, descrição e quantidade.
- Dados das Restrições: Contém as informações referentes aos professores, tais como: nome, descrição e restrições com relação ao horário e aos dias da semana.

6.2.4.4 Criação da população inicial

Conforme descrito em seções anteriores: a criação da população inicial é de extrema importância para o desempenho do Algoritmo Genético. Na criação da população inicial são gerados somente indivíduos factíveis. Ao final da criação da população inicial, a restrição número de aulas por semana de cada disciplina está satisfeita, entretanto, no decorrer do processamento, novamente esta verificação deverá ser feita, tendo seu resultado influência sobre a aptidão, já que com a aplicação dos operadores genéticos as grades horárias serão modificadas.

6.2.5 Operadores genéticos

Nesta seção, são apresentados os principais aspectos relacionados aos

operadores genéticos, como: seleção, elitismo, cruzamento, mutação e evolução.

6.2.5.1 Seleção

O mecanismo de seleção em algoritmos genéticos emula os processos de reprodução assexuada e seleção natural. Em geral, gera-se uma população temporária de N indivíduos extraídos com probabilidade proporcional à adequabilidade relativa de cada indivíduo na população, ou seja, um indivíduo apto a aparecer na nova população. Neste processo, indivíduos com baixa adequabilidade terão alta probabilidade de desaparecerem da população, ou seja, serem extintos. Já os indivíduos mais adequados terão grandes chances de sobreviverem.

6.2.5.2 Elitismo

Elitismo é um processo que garante que um percentual dos melhores indivíduos de uma população não seja perdido. Desta forma os melhores indivíduos de uma população são reproduzidos na população seguinte.

6.2.5.3 Cruzamento

O processo de recombinação é um processo sexuado - ou seja, envolve mais de um indivíduo - que emula o fenômeno de cruzamento, ou seja, a troca de fragmentos entre pares de cromossomos. Na forma mais simples, trata-se um processo aleatório que ocorre com probabilidade fixa, que deve ser especificada

pelo usuário. A figura 4 apresenta um modelo de cruzamento.

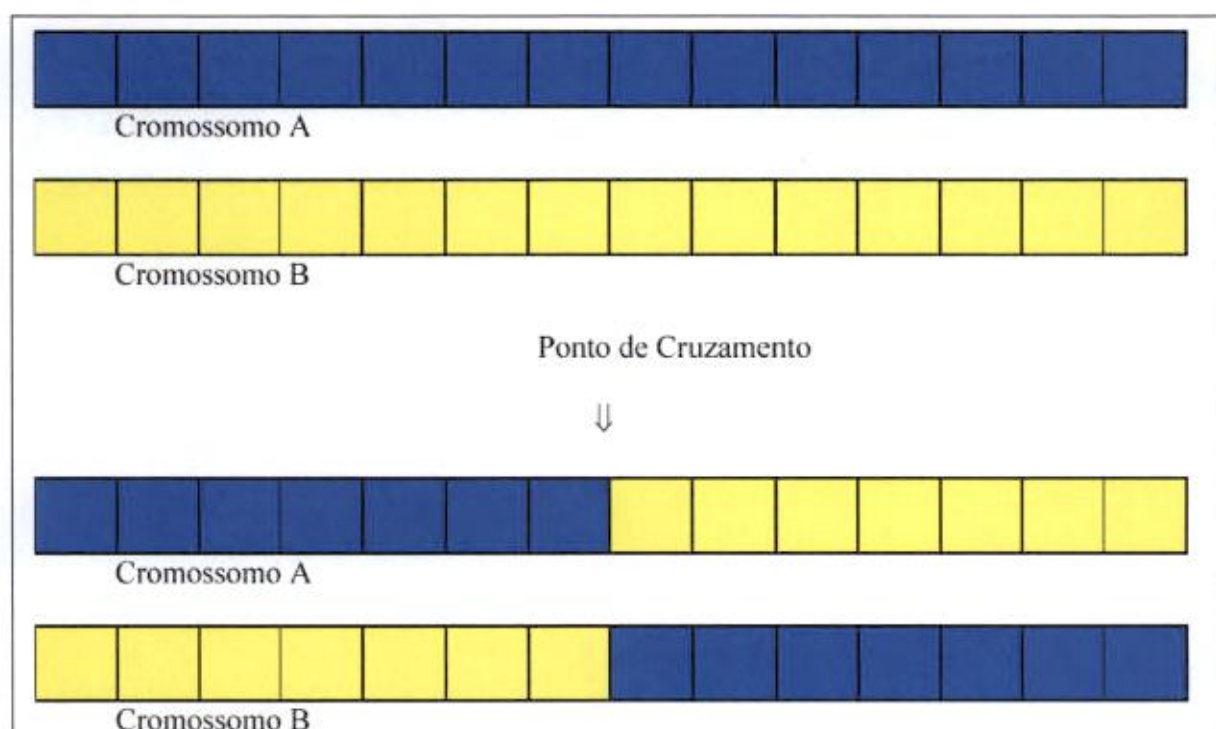


FIGURA 4 – EXEMPLO DE CRUZAMENTO DE UM PONTO.

FONTE: O autor (2007)

6.2.5.4 Mutação

O processo de mutação em algoritmos genéticos é equivalente à busca aleatória.

Basicamente, seleciona-se uma posição num cromossomo e muda-se o valor do gene correspondente aleatoriamente para um outro alelo possível.

O processo é geralmente controlado por um parâmetro fixo, que indica a probabilidade de um gene sofrer mutação.

A mutação tem como objetivo apenas trocar os alelos de posição no indivíduo. Para cada indivíduo de cada espécie é sorteada a posição de forma aleatória e o conteúdo da posição é trocado.

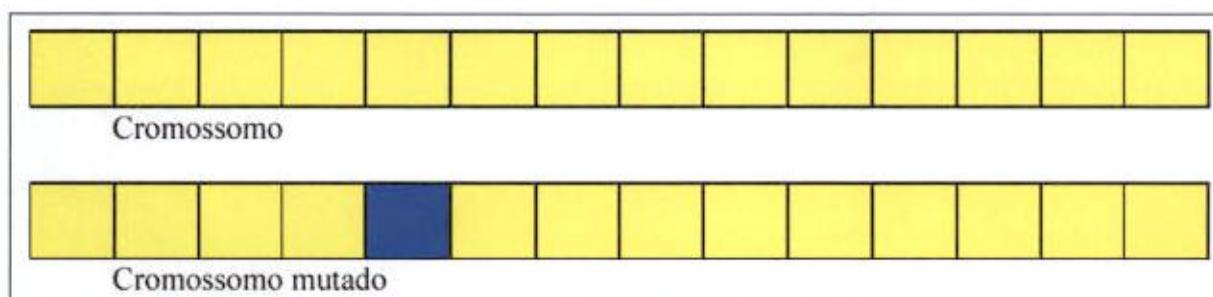


FIGURA 5 – EXEMPLO DE MUTAÇÃO

FONTE: O autor (2007)

6.2.5.5 Evolução

Para realizar a avaliação, primeiramente é necessário obter um modelo de domínio na qual todos os indivíduos de todas as espécies serão avaliados. O modelo constituirá de um conjunto com um indivíduo de cada espécie com maior valor da aptidão. Na avaliação, cada indivíduo é inserido nesse modelo de domínio no lugar do indivíduo que pertence a sua espécie. A avaliação é realizada com um indivíduo de cada vez. Na avaliação, o indivíduo será penalizado caso existam conflitos de horário entre períodos diferentes. Essa penalização irá afetar diretamente o valor de aptidão, fazendo assim, com que o processo evolutivo do indivíduo seja alterado. A verificação é realizada em todas as gerações criadas.

7. METODOLOGIA

7.1. O MODELO GENÉTICO

Segue, nesta seção, o detalhamento do modelo de Algoritmo Genético utilizado no Millenium.

7.1.1. Funcionamento Geral

O mecanismo de execução do Algoritmo Genético centraliza-se nas classes Genetico e Contexto. Na classe Contexto são definidos os parâmetros do algoritmo genético, bem como o objeto "Turno" que será processado. A classe Genetico recebe um objeto "Contexto" e realiza a sua execução.

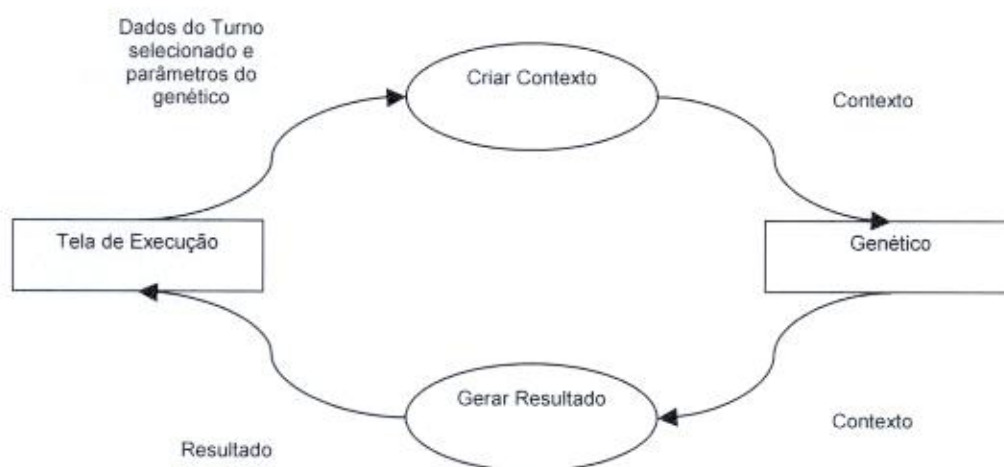


FIGURA 6 – FLUXO DE DADOS ENTRE A TELA DE EXECUÇÃO E O GENÉTICO

FONTE: O autor (2007)

7.1.2. A classe genético

A classe genético contém basicamente por uma População e métodos para trabalhar com ela. Seguindo o princípio dos Algoritmos Genéticos, a execução inicia-se criando uma população com elementos aleatórios. Após essa etapa o sistema começa a realizar iterações até que uma solução ideal seja encontrada ou até que o usuário interrompa a execução. Uma iteração nada mais é do que o ato de gerar uma nova população a partir da atual.

```
genetico.iniciar();  
enquanto( MelhorSolucaoNãoÉIdeal E UsuarioNãoInterrompeu ) {  
    genetico.evoluir();  
}
```

FIGURA 7 – PSEUDO-CÓDIGO REPRESENTANDO O LOOP PRINCIPAL DE EXECUÇÃO DO GENÉTICO

FONTE: O autor (2007)

7.1.3. A classe populacao

Uma população é composta por um determinado número de Cromossomos. Cada cromossomo representa um potencial solução para o problema da grade horária para o turno escolhido. O número de cromossomos que uma população contém é definido nas opções de parametrização do genético.

7.1.4. A classe cromossomo

Um cromossomo é uma distribuição das aulas de cada turma dentro da sua respectiva grade horária. Como pode ocorrer de, mesmo dentro do mesmo turno, duas turmas terem aulas em horários diferentes, essa distribuição é armazenada

dentro do cromossomo em Tabelas, sendo que cada tabela representa uma grade horária do turno escolhido pelo usuário.

O cromossomo possui uma propriedade chamada "*Fitness*" que indica a qualidade da solução levando em consideração as restrições especificadas pelo usuário. Quanto menor a *fitness*, melhor é a solução. Considera-se uma solução ideal quando o sistema consegue produzir um cromossomo com *fitness* igual a 0 (zero).

A estrutura interna do cromossomo foi criada de forma a facilitar o cálculo da *fitness*. Este cálculo inclui diversas verificações, sendo a mais importante delas a verificação de Colisões. Uma colisão ocorre quando em um mesmo cromossomo, um professor está escalado para dar aula em duas turmas diferentes no mesmo horário. Para facilitar a verificação destas colisões, além de permitir um controle das aulas por turma o cromossomo armazena internamente dois tipos de tabelas: professor x horário e turma x horário.

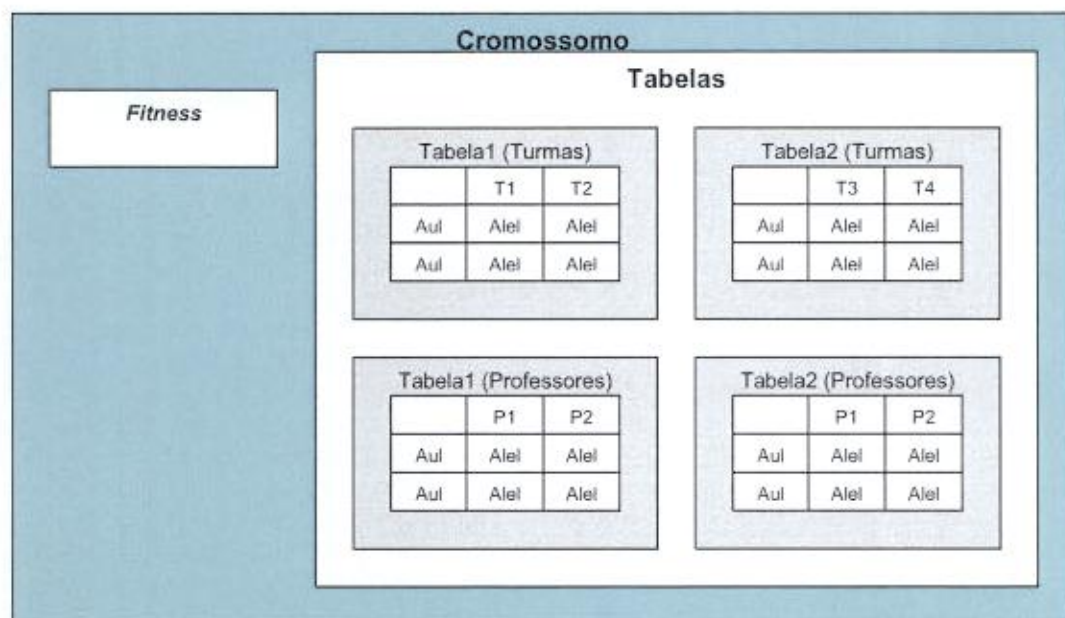


FIGURA 8 – ESTRUTURA DO CROMOSSOMO

FONTE: O autor (2007)

7.1.5. O molde

Os cromossomos são construídos através de um Molde. Este molde é gerado durante a inicialização da classe Genetico e é armazenado dentro do Contexto, de forma que fique disponível para todos os cromossomos que precisem ser criados.

O molde contém informações sobre a estrutura que o cromossomo deve conter, como a lista das tabelas, linhas e colunas de cada uma das tabelas e também a lista de aulas a serem distribuídas.

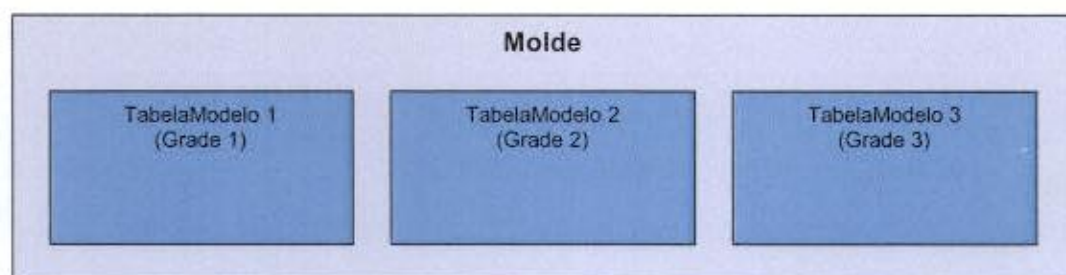


FIGURA 9 – ESTRUTURA DO MOLDE DO CROMOSSOMO

FONTE: O autor (2007)

As colunas das tabelas do molde representam as turmas. Apesar de serem poucos os casos onde um mesmo turno tenha mais de uma grade horária, turmas com grades horárias distintas acabam sendo armazenadas em tabelas distintas nos cromossomos gerados a partir do molde.

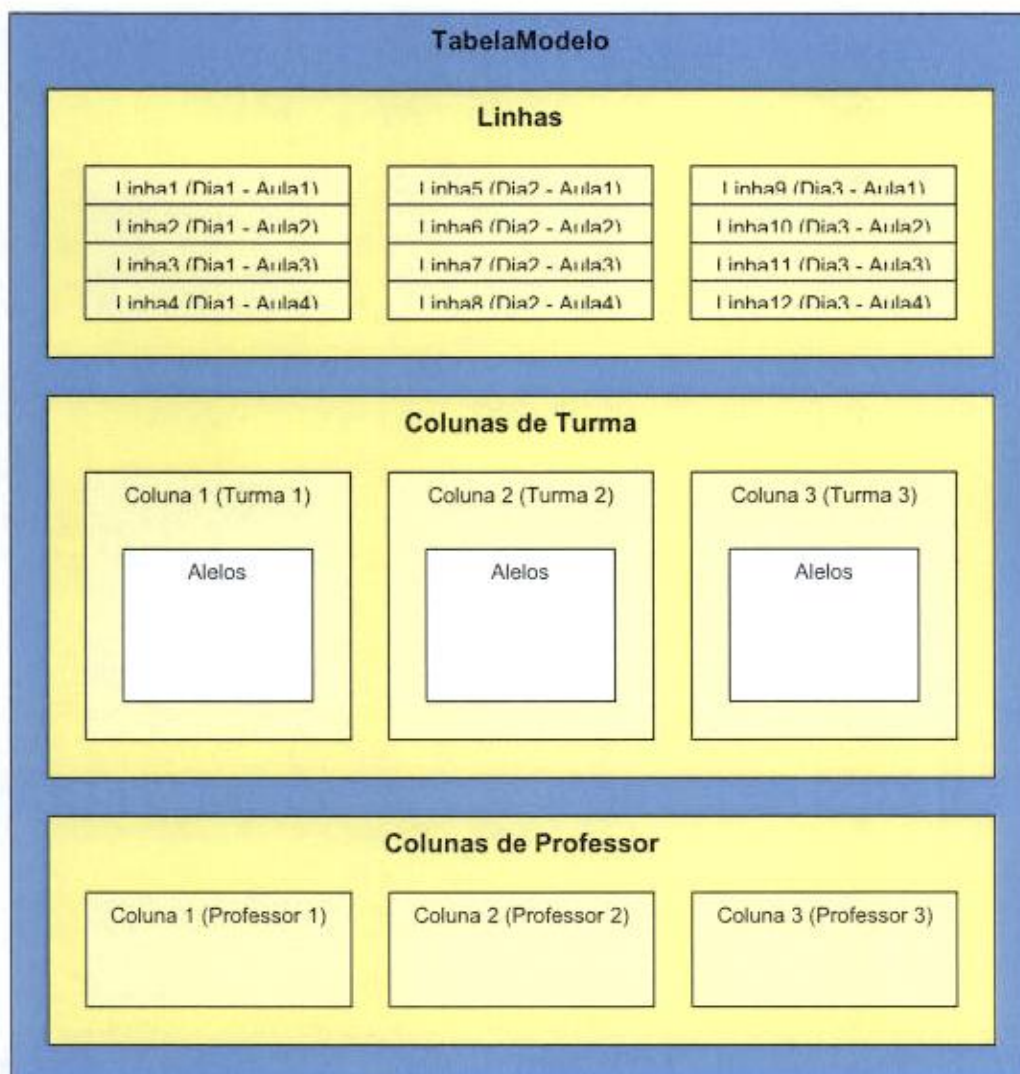


FIGURA 10 – ESTRUTURA DE UMA “TABELAMODELO”

FONTE: O autor (2007)

Observe pela figura 10 que uma TabelaModelo possui dois conjuntos de colunas: as de turma e as de professor. A razão para isso é que cada TabelaModelo dará origem a duas tabelas dentro do cromossomo. Uma delas representando a distribuição de aulas por turma e a outra por professor.

Dentro de cada Coluna de Turma o molde existe uma lista de Alelos. Um alelo representa uma aula de um professor, em determinada disciplina para uma determinada turma. Uma tupla pode dar origem a um ou mais alelos (a quantidade de alelos gerada pela tupla é determinada pela propriedade Quantidade do objeto Tupla).

Como cada Coluna de Turma representa uma turma, os alelos gerados pelas tuplas dessa turma serão distribuídos apenas dentro da sua respectiva coluna.

Assim, alelos da turma "A" ficarão na coluna da turma "A", os alelos da turma "B" ficarão na coluna da turma "B", e assim por diante. Dentro do cromossomo, cada alelo só poderá ser distribuído pelas linhas da sua própria coluna. Assim, evita-se que as aulas das turmas se misturem.

Dentro do cromossomo, sempre que um alelo é posicionado numa célula da tabela de distribuição por turma, o mesmo alelo é imediatamente posicionado na célula equivalente da tabela de distribuição por professor. Apesar de se tratar praticamente da mesma informação, o armazenamento do objeto de duas formas diferentes agiliza exponencialmente a velocidade do cálculo da *fitness*.

As linhas das tabelas do cromossomo representam os horários da semana toda. Assim, para cada horário de cada dia, tem uma linha correspondente nos dois tipos de tabela do cromossomo (a tabela de distribuição por turma e a tabela de distribuição por professor).

7.1.6. O cálculo da "fitness"

Como citado anteriormente, a *fitness* representa a viabilidade de um determinado cromossomo como solução para o problema da distribuição de horários. Quanto menor este valor, melhor é a solução.

O primeiro passo no cálculo da *fitness* é a verificação de colisões de professores. Para isto, o sistema verifica se numa mesma linha de cada tabela de turma consta o mesmo professor mais de uma vez.

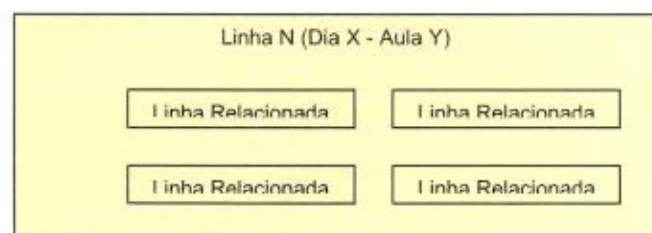


FIGURA 11 – LINHA DO CROMOSSOMO COM LINHAS RELACIONADAS

FONTE: O autor (2007)

O passo a seguinte a verificar são as colisões de horário entre grades horárias distintas. Podemos ter duas turmas, cada uma relacionada a uma grade horária, e um mesmo professor dando aulas para as duas turmas. Nessa situação, pode ocorrer que os horários de aula das duas grades horárias não sejam totalmente coincidentes (por exemplo, a primeira aula da grade "A" pode começar meia hora antes da primeira aula da grade "B").

Por esta razão, o objeto Linha pode possuir uma ou mais Linhas Relacionadas, que são referências para linhas de outras TabelasModelo que representam um conflito de horário com a linha atual.

Assim, caso um alelo de um professor esteja posicionado numa determinada linha, nenhuma das linhas relacionadas poderá ter um alelo do mesmo professor, caso contrário o sistema irá considerar o fato como uma Colisão e a *fitness* do cromossomo sofrerá penalização.

Outro ponto de verificação do algoritmo de *fitness* são as restrições de horário. O Millenium permite que restrições sejam definidas por professor, disciplina, turma ou tupla. Na prática, as tuplas acabam "herdando" as restrições definidas seus respectivos professores, disciplinas e turmas, e cada alelo recebe a lista total das restrições da tupla, incluindo as "herdadas".

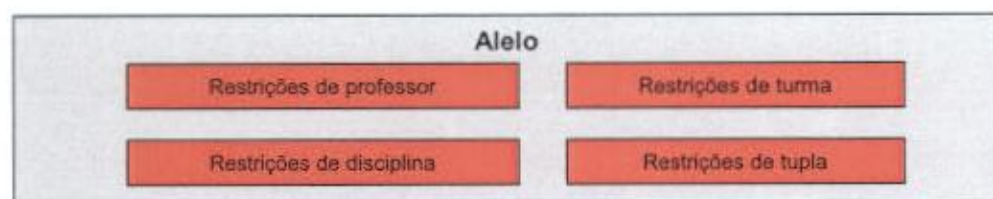


FIGURA 12 – O ALELO HERDA AS RESTRIÇÕES DE TODOS OS ELEMENTOS QUE O COMPÕEM

FONTE: O autor (2007)

A verificação é simples: o sistema percorre todos os alelos que possuam pelo menos uma restrição e faz a validação se aquele alelo deveria ou não estar naquele horário, penalizando a *fitness* se for o caso.

Dentro do cromossomo, as tabelas de distribuição por professor são utilizadas principalmente pela restrição conhecida como Janelas de Professores. Esta restrição pode favorecer, se o usuário assim definir, soluções onde os professores fiquem com menos "janelas", ou seja, horários sem aula entre dois

horários com aula. Para fazer esta verificação, o sistema varre as tabelas de distribuição por professor em busca de "janelas" e penaliza a *fitness* para cada janela encontrada.

Verificação similar é efetuada por uma outra restrição: a de "Janelas de Turmas". O princípio de execução é o mesmo, mas esta verificação utiliza as tabelas de verificação por turmas ao invés das por professor.

Outras restrições que o usuário pode definir são o limite do número de aulas da mesma tupla por dia e o favorecimento de soluções onde aulas da mesma tupla sejam dadas sempre na seqüência para a mesma turma.

7.1.7. O ciclo de vida do algoritmo genético

Durante a execução normal, a primeira população é montada gerando cromossomos vazios e depois distribuindo aleatoriamente os alelos pelas colunas. Logo a seguir, é calculada a *fitness* de cada cromossomo e a população é ordenada pelo valor da *fitness* em ordem crescente.

Enquanto nenhuma condição de parada for atendida, o sistema irá realizar iterações. Cada iteração se baseia em gerar uma nova população a partir da população anterior. São definidas duas condições de parada. A primeira é quando o sistema encontra uma solução perfeita, o que ocorre quando a *fitness* do primeiro elemento da população é igual a 0 (zero). A segunda condição de parada é quando o usuário escolhe a opção de interromper a operação.

O primeiro cromossomo da população é sempre considerado como a "solução atual", por ser sempre o que tem o maior número de restrições atendidas.

Depois que o usuário interrompe a operação, ele pode salvar a solução atual e depois tem a opção de voltar a executar, dessa forma, o sistema volta novamente a fazer iterações em um "loop" com as mesmas condições de parada anteriores.

O Millenium permite também que o usuário cancele a execução e depois a reinicie através de um resultado salvo. No entanto, o Millenium salva apenas a "solução atual", os demais cromossomos da população são descartados quando a execução é cancelada. Por isso, ao carregar uma solução salva, o sistema terá

apenas um cromossomo para trabalhar. Neste caso, o sistema preenche o restante da população com cromossomos distribuídos aleatoriamente antes de começar a fazer iterações.

7.1.8. O funcionamento da "iteração"

Uma iteração é o processo de, a partir de uma população existente, gerar uma nova população, potencialmente melhor que a antecessora e com o mesmo número de elementos. Para tanto, são realizadas as operações de Elitização, Cruzamento e Mutação.

A quantidade de cromossomos produzidos por cada uma destas operações é definida nas parametrizações do genético, juntamente com o tamanho da população. O mais comum é ter uma quantidade grande de cruzamentos (85% a 95%), uma quantidade bem menor de mutações (de 5% a 15%) e um único elemento é elitizado.

Criação de uma nova População



GRÁFICO 1 – PROPORÇÃO GERALMENTE USADA ENTRE OS CROMOSSOMOS CRIADOS DE UMA NOVA POPULAÇÃO

FONTE: O autor (2007)

7.1.8.1 Sorteio

Para a realização de cruzamentos e mutações, o sistema utiliza um mecanismo especial para sortear o elemento a ser cruzado ou a sofrer mutação.

O sistema sorteia um número aleatório entre 0 e 1 e o eleva a uma determinada potência, chamada de Coeficiente de Sorteio. O resultado é multiplicado pelo tamanho da população.

$$\text{sorteado} = [(\text{numeroaleatório})^{\text{coeficiente}}] * \text{tamanho da população}$$

FIGURA 13 – FÓRMULA USADA NO SORTEIO DE CROMOSSOMOS

FONTE: O autor (2007)

O coeficiente de sorteio é um número entre 1 e 2, definido nas parametrizações do genético. Valores comuns para este coeficiente são 1,05 e 1,1.

O resultado da fórmula é arredondado para um número inteiro, que irá indicar a posição dentro da população onde está o cromossomo sorteado.

O uso desta fórmula faz com que o cromossomo sorteado tenha uma maior probabilidade de estar entre os primeiros da população. Como os elementos sorteados serão usados para cruzamento e mutação, esse fato faz com que a próxima população tenha tendência a ser melhor que a população atual, uma vez que foi construída dando preferência a elementos melhores.

Fórmula de Sorteio do Millenium

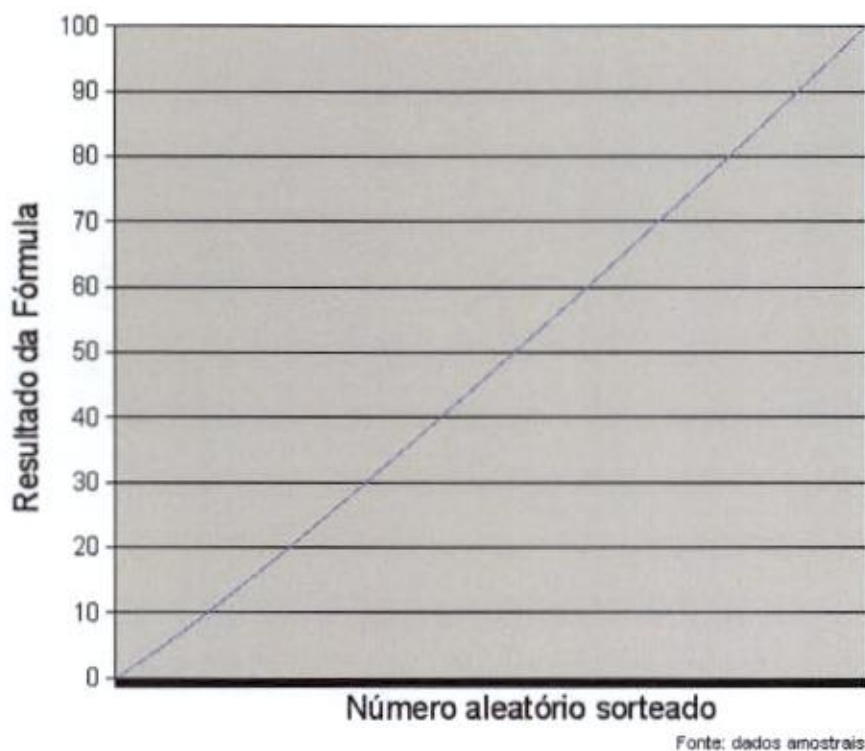


GRÁFICO 2 – COMPORTAMENTO DA FÓRMULA DE SORTEIO DO CROMOSSOMO
 FONTE: O autor (2007)

O gráfico 2 mostra como a fórmula se comporta para 1000 números aleatórios sorteados entre 0 e 1, com tamanho da população 100 e coeficiente 1,1. É possível perceber uma ligeira curvatura na base da linha, o que indica que um número maior de números sorteados resultou num índice menor.

O gráfico também demonstra claramente o quanto é sutil esta tendência a ser sorteado o menor número. Isto é muito importante para que não ocorra um fenômeno conhecido como "domínio da população", situação na qual uma população acaba ficando com várias cópias idênticas da mesma solução, o que prejudica enormemente a obtenção de uma solução.

7.1.8.2 Cruzamento

O cruzamento consiste nas seguintes etapas:

1. Sortear dois cromossomos, segundo o método descrito na seção anterior;
2. Escolher aleatoriamente um ponto de corte;
3. O novo cromossomo gerado deverá ser composto pela primeira metade do cromossomo 1 e pela segunda metade do cromossomo 2.

Como a estrutura do cromossomo não é linear, o ponto de corte acaba sendo um conjunto de valores:

- Tabela onde será feito o corte;
- Coluna onde será feito o corte;
- Linha onde será feito o corte.

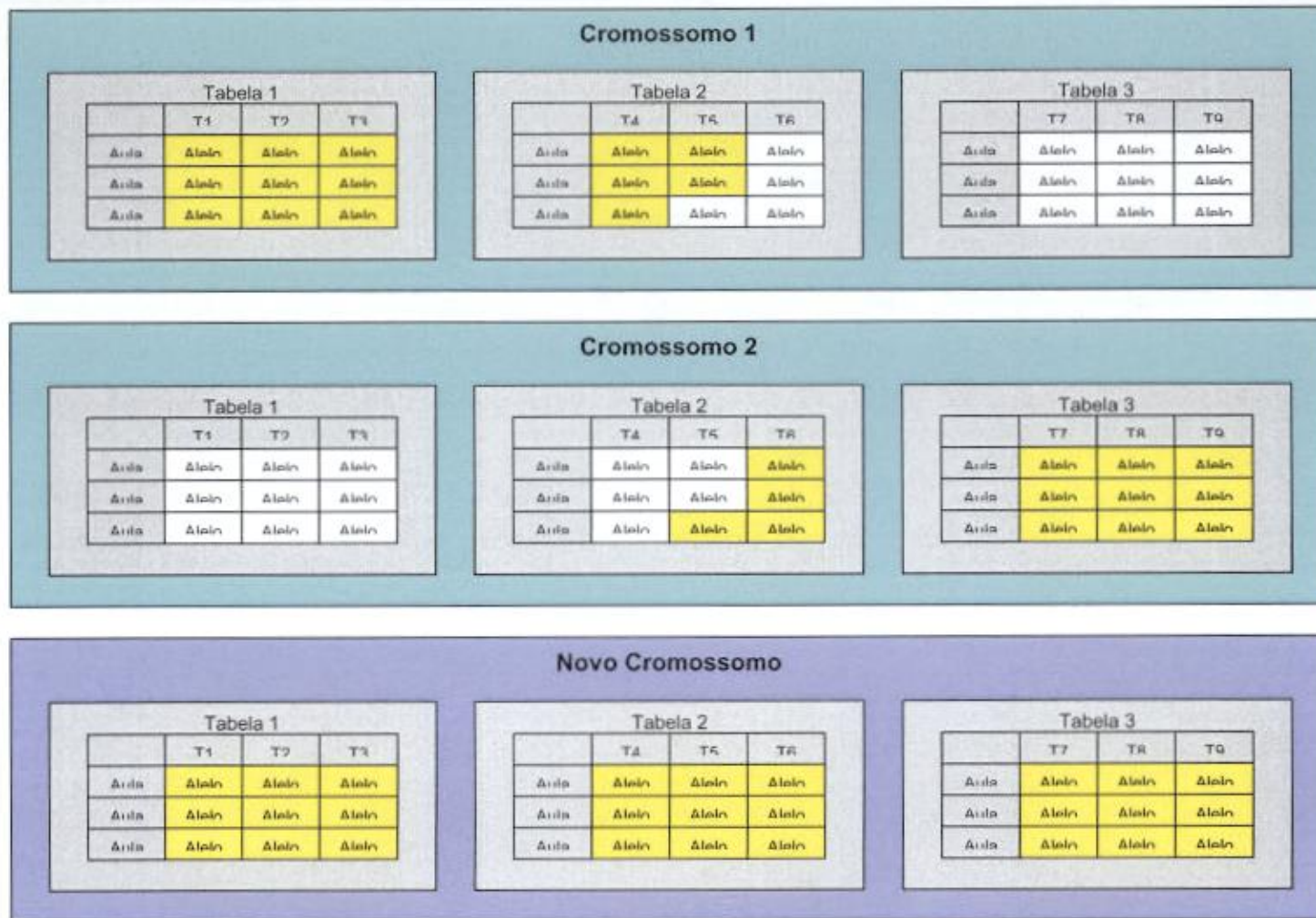


FIGURA 14 – PONTO DE CORTE DOS CROMOSSOMOS 1 E 2 E ALELOS COPIADOS PARA O NOVO CROMOSSOMO
 FONTE: O autor (2007)

As tabelas anteriores à do ponto de corte serão copiadas do cromossomo 1 para o novo cromossomo. Da mesma forma, as tabelas posteriores à do ponto de corte serão copiadas do cromossomo 2 para o novo cromossomo.

Na tabela do ponto de corte, as colunas anteriores à do ponto de corte serão copiadas do cromossomo 1 para o novo cromossomo. Da mesma forma, as colunas posteriores à do ponto de corte serão copiadas do cromossomo 2 para o novo cromossomo.

Na coluna do ponto de corte, os alelos das linhas anteriores e o da linha do ponto de corte serão copiadas do cromossomo 1 para o novo cromossomo. Os alelos das linhas posteriores serão obtidos do cromossomo 2, o detalhe é que nessa lista de alelos do cromossomo 2 podem existir alelos que já estão posicionados, vindos do cromossomo 1.

A estratégia usada neste caso é esta: o sistema tentará inserir no horário o alelo do cromossomo 2. Se o mesmo já existir, o sistema o descarta e tenta inserir, no mesmo horário, o próximo alelo do cromossomo 2, e assim por diante. Após terminar de processar todos os alelos do cromossomo 2, o sistema distribui os alelos que ficaram faltando.

7.1.8.3 Mutaç o

O mecanismo de mutaç o consiste em sortear um cromossomo aleat rio e a partir dele criar um novo, invertendo as posiç es de dois alelos da mesma turma.

Para isso o sistema faz uma c pia do cromossomo e, nesta c pia, sorteia aleatoriamente uma tabela e depois uma coluna desta tabela.

Nesta coluna, o sistema sorteia dois alelos distintos e inverte a posiç o de ambos.

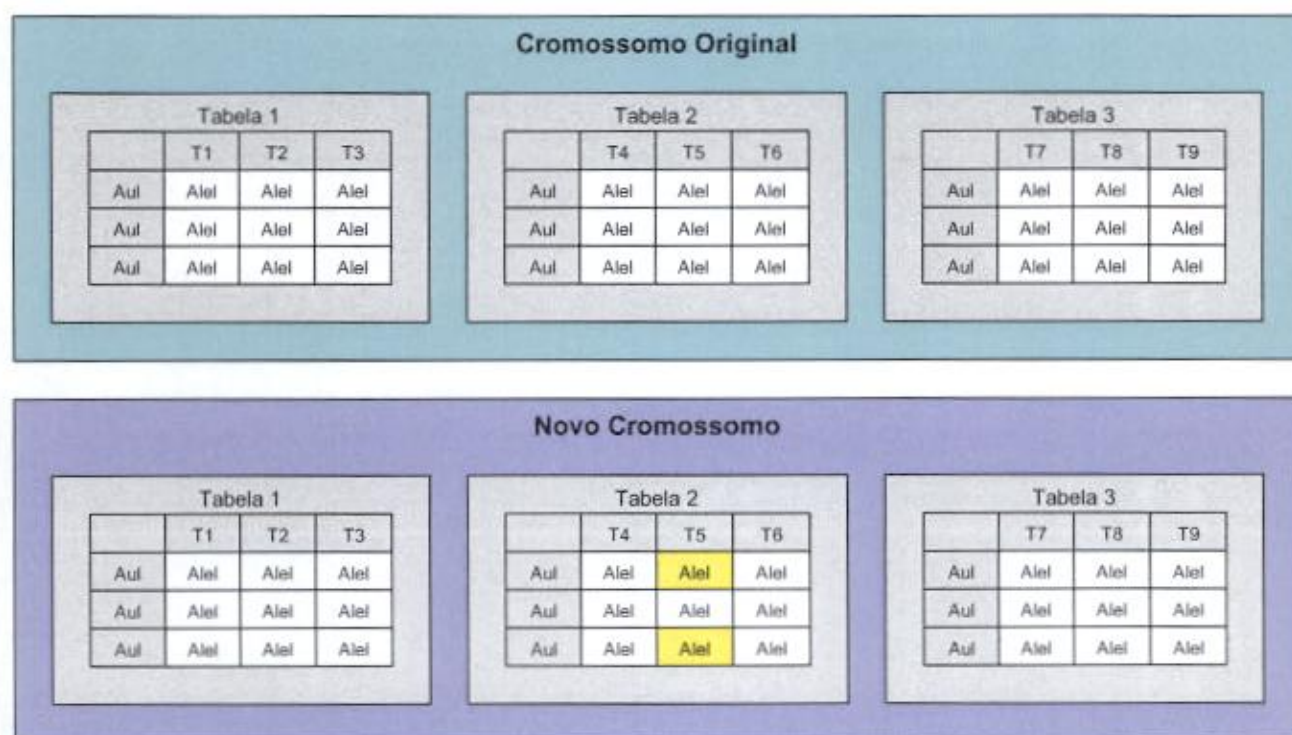


FIGURA 15 – NOVO CROMOSSOMO CRIADO POR MUTAÇÃO

FONTE: O autor (2007)

7.2. ANÁLISE DE REQUISITOS

O sistema tem o intuito de disponibilizar a geração de horário de forma local. Podendo manter ainda um histórico de soluções disponíveis de horários.

Os limites do sistema foram definidos com base em uma idéia primária de composição de horários por meio de preferências e restrições. A equipe decidiu usar o máximo dos conhecimentos adquiridos durante o curso, então procuramos definir o escopo do projeto envolvendo as áreas de: programação de baixo nível, gerenciamento de banco de dados e linguagem UML.

Após análise inicial o grupo, com o auxílio do orientador, definiram os requisitos principais para que o sistema fosse completo e a altura de um Trabalho de Conclusão de Curso de Graduação. Os principais requisitos do sistema são:

- a) Controlar permissões de usuário utilizando três níveis: administração, edição e consulta;
- b) Permitir cadastrar cursos, professores, disciplinas, turmas e aulas;

- c) Permitir criar uma grade horária customizada, onde o usuário cadastre os dias e horários individualmente;
- d) Permitir a criação de uma ou mais instituições, de forma que cada instituição possa cadastrar seus dados e gerar a grade independente da outra;
- e) Permitir gerar automaticamente a distribuição das aulas pela grade de forma a não ocorrerem colisões;
- f) Permitir que o usuário interaja com a geração da grade, visualizando o progresso da operação e o melhor resultado atual;
- g) Permitir que sejam geradas e armazenadas várias soluções diferentes;
- h) Permitir que as soluções geradas sejam consultadas sem problemas e sem alterações, mesmo que tenha havido mudanças nos dados que foram usados para gerar a solução;
- i) Permitir a definição de restrições de horário para professores, disciplinas e turmas;
- j) Permitir restrições adicionais, como número máximo de aulas por dia e evitar janelas.

7.3. DESCRIÇÃO DAS REGRAS DO NEGÓCIO

Esta seção tem por objetivo descrever as regras do negócio.

7.3.1 Efetuar login

Todo o acesso às funcionalidades será protegido por um sistema de *login*.

Sendo assim antes de executar qualquer consulta ou operação, usuário deve-se certificar que seu nome e sua senha estão devidamente cadastrados no sistema. Então será necessário que no início de suas atividades o usuário digite seu nome no campo "usuário" e sua senha pessoal dentro de "senha". Confirme a operação acionando o botão "OK". Caso seja o primeiro acesso o usuário deverá cadastrar os campos mencionados anteriormente e o sistema efetuará o *login*.

Tendo feito a identificação inicial o usuário poderá cadastrar uma instituição nova, ou caso já tenha, selecionará uma instituição para continuar no sistema.

7.3.2 Selecionar instituição

Após efetuar o *login* e obter acesso ao sistema será apresentado ao usuário a tela de seleção de instituição, através dela o usuário selecionará na lista uma instituição, já cadastrada anteriormente, da qual ele deseja manipular os dados.

7.3.3 Manter instituição

Ao clicar no botão Editar da tela de seleção de instituição será apresentada a tela de manutenção de instituição. Através dela o usuário poderá cadastrar uma nova instituição, excluir ou manter os dados de instituições já cadastradas assim como visualizar suas principais informações, como por exemplo, quantidade de cursos, turnos, turmas, professor cadastrados. Para cadastrar uma nova instituição deverão existir os seguintes campos: Abreviação e Descrição.

7.3.4 Definir senha

O administrador quando cadastrar um usuário ele definirá apenas um *login* para o novo usuário, ficando a cargo do usuário a definição de sua senha quando acessar o sistema pela primeira vez.

Ao clicar no menu Alterar senha será apresentada a tela de definição de senha, através dela o usuário poderá definir sua senha ou alterar sua senha. Para cadastrar uma senha deverão existir os seguintes campos: Senha antiga, Senha nova e confirmação.

Quando o usuário optar por alterar sua senha o sistema deve impedir que ele cadastre sua senha antiga como se fosse uma senha nova.

7.3.5 Manter usuário

Ao clicar no menu Usuários da tela principal será apresentada a tela de manutenção de usuário. Através dela o administrador poderá cadastrar um novo usuário, excluir ou manter os dados dos usuários já cadastrados. Para cadastrar um novo usuário deverão existir os seguintes campos: *Login*, Nome completo, e Permissões subdivididas em Apenas consulta, Consulta e Edição e Administrador.

O sistema deverá garantir que sempre existirá um usuário com permissão de administrador, para isso ele criará um "administrador" *default* quando o sistema for utilizado pela primeira vez e impossibilitará a exclusão do usuário com permissão de administrador quando ele for o único existente no sistema.

7.3.6 Manter curso

Ao clicar no botão Cursos da tela principal será apresentada a tela de manutenção de curso.

Através dela o usuário poderá cadastrar um novo curso, excluir ou manter os dados dos cursos já cadastrados Para cadastrar um novo curso deverá existir os seguintes campos: Abreviação, Descrição.

7.3.7 Manter turno

Ao clicar no botão Turnos da tela principal será apresentada a tela de manutenção de turno.

Através dela o usuário poderá cadastrar um novo turno, excluir ou manter os dados dos turnos já cadastrados Para cadastrar um novo turno deverá existir os seguintes campos: Abreviação, Descrição.

7.3.8 Manter professor

Ao clicar no botão Professores da tela principal será apresentada a tela de manutenção de professor.

Através dela o usuário poderá cadastrar um novo professor, excluir ou manter os dados dos professores já cadastrados, assim como possibilitar a edição das restrições dos professores e visualização de suas restrições e tuplas, quando existir.

Para cadastrar um novo professor deverão existir os seguintes campos: Abreviação ou Apelido, Nome completo.

Deverá existir, também, o botão "Editar Restrições", que só será visível após

a inclusão do professor na base de dados. Esse botão possibilitará ao usuário ter acesso a tela de edição de restrições.

7.3.9 Manter disciplina

Ao clicar no botão Disciplinas da tela principal será apresentada a tela de manutenção de disciplinas.

Através dela o usuário poderá cadastrar uma nova disciplina, excluir ou manter os dados das disciplinas já cadastradas, assim como possibilitar a edição das restrições das disciplinas e visualização de suas restrições e tuplas, quando existir.

Para cadastrar uma nova disciplina deverá existir os seguintes campos: Abreviação, Descrição.

Deverá existir, também, o botão Editar Restrições, que só será visível após a inclusão da disciplina na base de dados. Esse botão possibilitará ao usuário ter acesso a tela de edição de restrições.

7.3.10 Manter turmas

Ao clicar no botão Turmas da tela principal será apresentada a tela de manutenção de turma.

Através dela o usuário poderá cadastrar uma nova turma, excluir ou manter os dados das turmas já cadastradas, assim como, editar e visualizar suas tuplas e restrições.

Para cadastrar uma nova turma deverão existir os seguintes campos: Abreviação, Descrição, Curso e Grade horária.

Deverão existir, também, os botões "Editar tuplas.." e "Editar restrições" esses botões só serão visíveis após a inclusão da turma na base de dados. Esses

botões possibilitarão ao usuário ter acesso as telas de edição de tuplas e restrições.

7.3.11 Manter grade

Ao clicar no botão Grades Horárias da tela principal será apresentada a tela de manutenção de grade horária. Através dela o usuário poderá cadastrar uma nova grade, excluir ou manter os dados das grades já cadastradas, assim como editar os dias e horários pertencentes à grade horária.

Para cadastrar uma nova grade deverão existir os seguintes campos: Abreviação, Descrição, Turno.

Deverão existir, também, os botões Dias e Horários esses botões só serão visíveis após a inclusão da grade na base de dados. Esses botões possibilitarão ao usuário ter acesso as telas de edição dos dias e horários.

7.3.12 Edição de restrições

Após confirmar a inclusão de um professor, turma ou disciplina o botão Editar Restrições passa a ser visível ao usuário.

Ao clicar nesse botão o usuário obtém acesso a tela de edição de restrições que possibilitará ao usuário definir os seguintes tópicos:

- 1 impedir, alocar ou favorecer que um professor ministre aula em um dia ou em um intervalo de tempo específico.
- 2 impedir, alocar ou favorecer que uma disciplina seja lecionada em um dia ou em um intervalo de tempo específico.
- 3 impedir, alocar ou favorecer que uma turma tenha aula em um dia ou em um intervalo de tempo específico.

Para cadastrar uma nova restrição deverão existir os seguintes campos.

1. Um grupo de Período com as seguintes opções:

- O dia todo
 - Intervalo de tempo (De:, Até)
 - Antes do intervalo
 - Após o intervalo
2. Um grupo de Dias da semana com um checkbox para cada dia da semana.
 3. Os botões "Marcar todos" e "Desmarcar todos" para possibilitar ao usuário selecionar ou desmarcar todos os dias da semana.
 4. Um grupo Restrição com os seguintes checkbox:
 - Impedira aulas no período
 - Desfavorecer aulas no período
 - Favorecer aulas no período
 - Forçar aulas a serem no período

7.3.13 Editar dias

Ao clicar no botão Dias da tela da grade horária será apresentada a tela de manutenção de dias. Através dela o usuário poderá cadastrar um novo dia, excluir ou manter os dados dos dias já cadastrados.

Para cadastrar um novo dia deverão existir os seguintes campos: Abreviação, Descrição, Dias da semana.

7.3.14 Editar horários

Ao clicar no botão Horários da tela da grade horária será apresentada a tela de manutenção de horário. Através dela o usuário poderá cadastrar um novo horário, excluir ou manter os dados dos horários já cadastrados.

Para cadastrar um novo dia deverão existir os seguintes campos: Nome, Observação, Horário de início, Duração e um checkbox (Terá aula nesse período).

7.3.15 Editar tuplas

Ao clicar no botão Editar tuplas da tela de edição de turmas será apresentada a tela de edição de tuplas. Através dela o usuário poderá cadastrar as aulas, ou seja, o relacionamento entre professor, disciplina e turma. Através dela será possível, também, excluir ou manter os dados das tuplas já cadastradas e dividir a turma para assistir aulas com professores diferentes no mesmo horário.

Para cadastrar uma nova tupla deverá existir os seguintes campos: Turma, Professor, Disciplina, Aulas por semana, um *checkbox* Dividir turmas entre 2 professores, Professor secundário, Disciplina secundária.

7.3.16 Gerar grade

Ao clicar no botão Geração da tela principal será apresentada a tela de geração de horário. Através dela o usuário poderá gerar uma nova grade horária.

Para gerar uma nova grade horária a tela de geração será dividida em seis grupos, são eles:

- Tipo de execução com as opções: Criar novo resultado, Usar resultado existente como base.
- Opções de Execução com os seguintes campos: Número máximo de iterações, Tempo máximo de execução.
- Turno contendo uma lista dos turnos cadastrados.
- Restrições com as opções: Forçar aulas da mesma tupla serem

seqüenciais, Máximo de aulas da mesma tupla por dia.

- Janelas de Turmas coma as opções: Ignorar, Mandar para o inicio, Mandar para o final, Início ou Final.
- Janelas de Professores com as opções: Ignorar, Mandar para o inicio, Mandar para o final, Início ou Final.

Deverá conter também um botão "Iniciar" que só estará editável após o usuário selecionar o turno para criação da grade horária.

7.4. DIAGRAMA DE CASO DE USO DO SISTEMA

São utilizados para modelar as interações entre o sistema e seus atores.

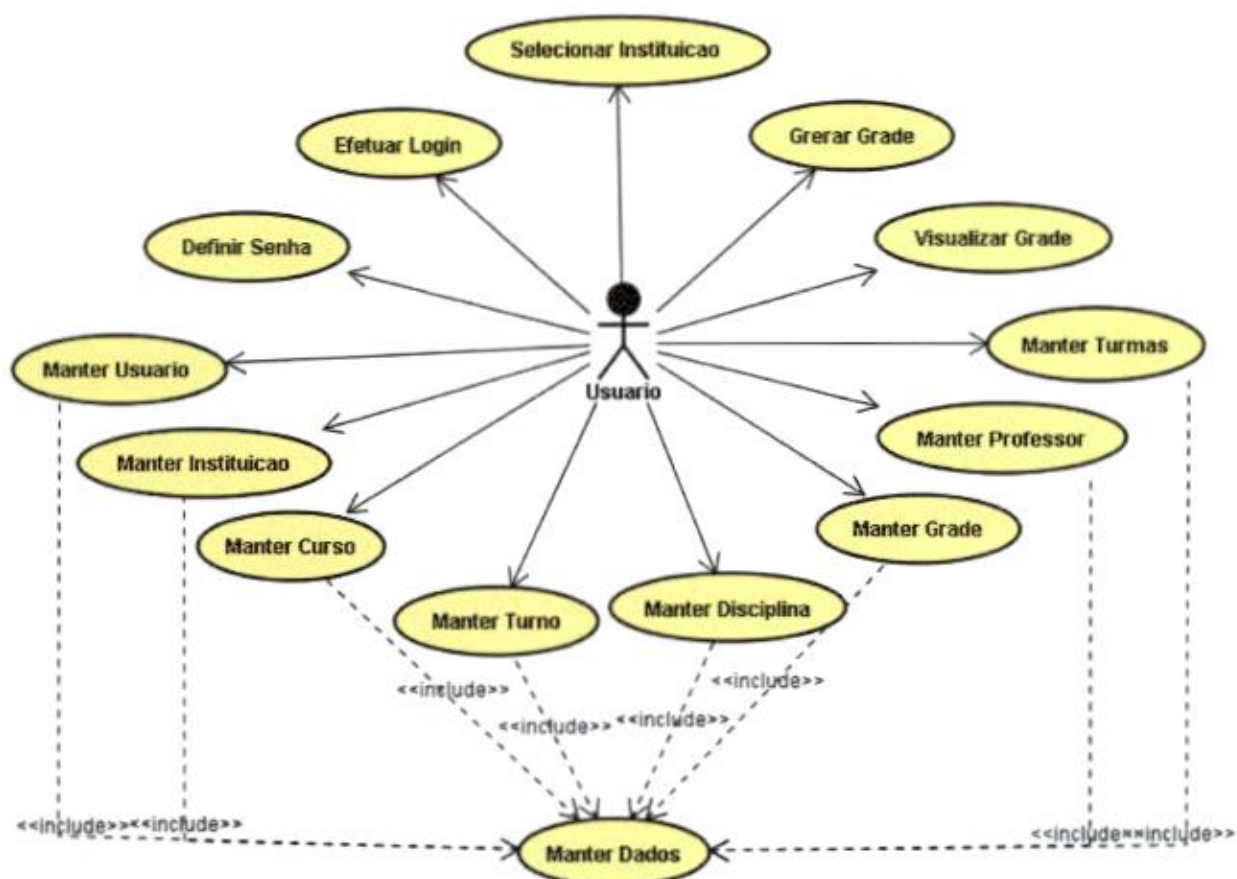


FIGURA 16 – CASO DE USO DO SISTEMA MILLENIUM
 FONTE: O AUTOR (2007)

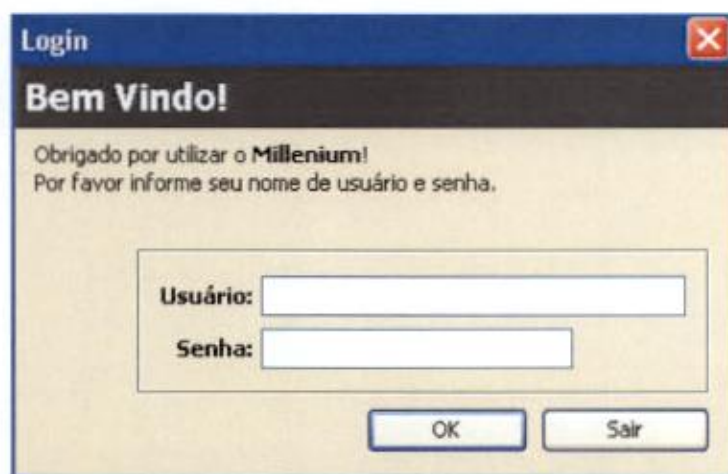
7.5. DESCRIÇÃO DOS CASOS DE USO DO SISTEMA MILLENIUM

Esta seção tem por objetivo especificar o comportamento dos casos de usos identificados, descrevendo de maneira textual seu fluxos principal, exceção e alternativo e relaciona-los com suas respectivas interfaces.

7.5.1 Caso de uso 1: efetuar login

7.5.1.1 Tela de login

Todo acesso ao sistema é protegido, esta tela é apresentada assim que o sistema é iniciado e tem por finalidade autorizar o acesso.



Login

Bem Vindo!

Obrigado por utilizar o **Millenium!**
Por favor informe seu nome de usuário e senha.

Usuário:

Senha:

OK Sair

FIGURA – TELA DE LOGIN

FONTE: O AUTOR (2007)

7.5.1.2 Caso de uso 1: efetuar login

Autor: Equipe

Ator: Usuário.

Regra de Negócio:

R1: Quando utilizado pela primeira vez o sistema informa que criou um usuário default "administrador"

R2: Sistema verifica se existe instituição cadastrada, se não houver exibe a tela de edição de instituição.

Descrição: Este Caso de Uso descreve obter acesso ao sistema.

Pré-Condições: Usuário deve estar cadastrado pelo administrador.

Pós-Condição: Ter acesso ao sistema.

Fluxo de eventos principal:

- a) Usuário inicia o sistema
- b) Sistema exibe tela de login (R1)
- c) Usuário informa "Usuário" e "Senha" e clica no botão "OK" (A1)
- d) Sistema confere os dados e devolve a tela de seleção de instituição (E1) (R2)
- e) Caso de uso finalizado

Fluxo de exceção:

E1: Usuário insere dados incorretos

- a) Sistema exibe mensagem de erro
- b) Usuário clica no botão "OK" da mensagem de erro
- c) Sistema fecha mensagem e devolve tela de login
- d) Retorna ao passo c do fluxo principal

A1: Usuário desiste de abrir o sistema

- a) Usuário clica no botão "Sair"
- b) Sistema é encerrado
- c) Caso de uso finalizado

7.5.2 Caso de uso 2: selecionar instituição

7.5.2.1 Tela de seleção de instituição

O Millenium possibilita ao profissional da área pedagógica administrar grades horárias de diversas instituições em uma única base de dados, para isso é

apresentada uma interface que permite ao usuário, assim efetua o login, selecionar a instituição que ele deseja trabalhar.

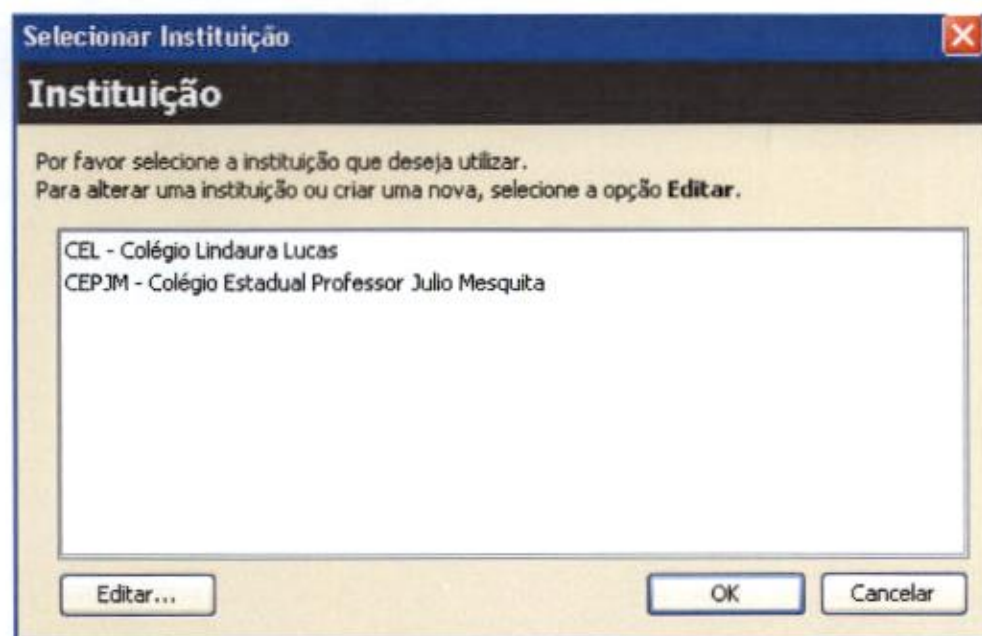


FIGURA 17 – TELA DE SELEÇÃO DE INSTITUIÇÃO

FONTE: O AUTOR (2007)

7.5.2.2 Caso de uso 2: selecionar instituição

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como selecionar uma instituição.

Pré-Condições: Ter efetuado Login

Pós-Condição: Obter acesso a tela principal do sistema.

Fluxo de eventos principal:

- Usuário seleciona uma instituição na lista e clica no botão "OK" (A2)
- Sistema exibe tela principal do sistema
- Caso de uso finalizado

Fluxo Alternativo:

A2: Usuário desiste da operação

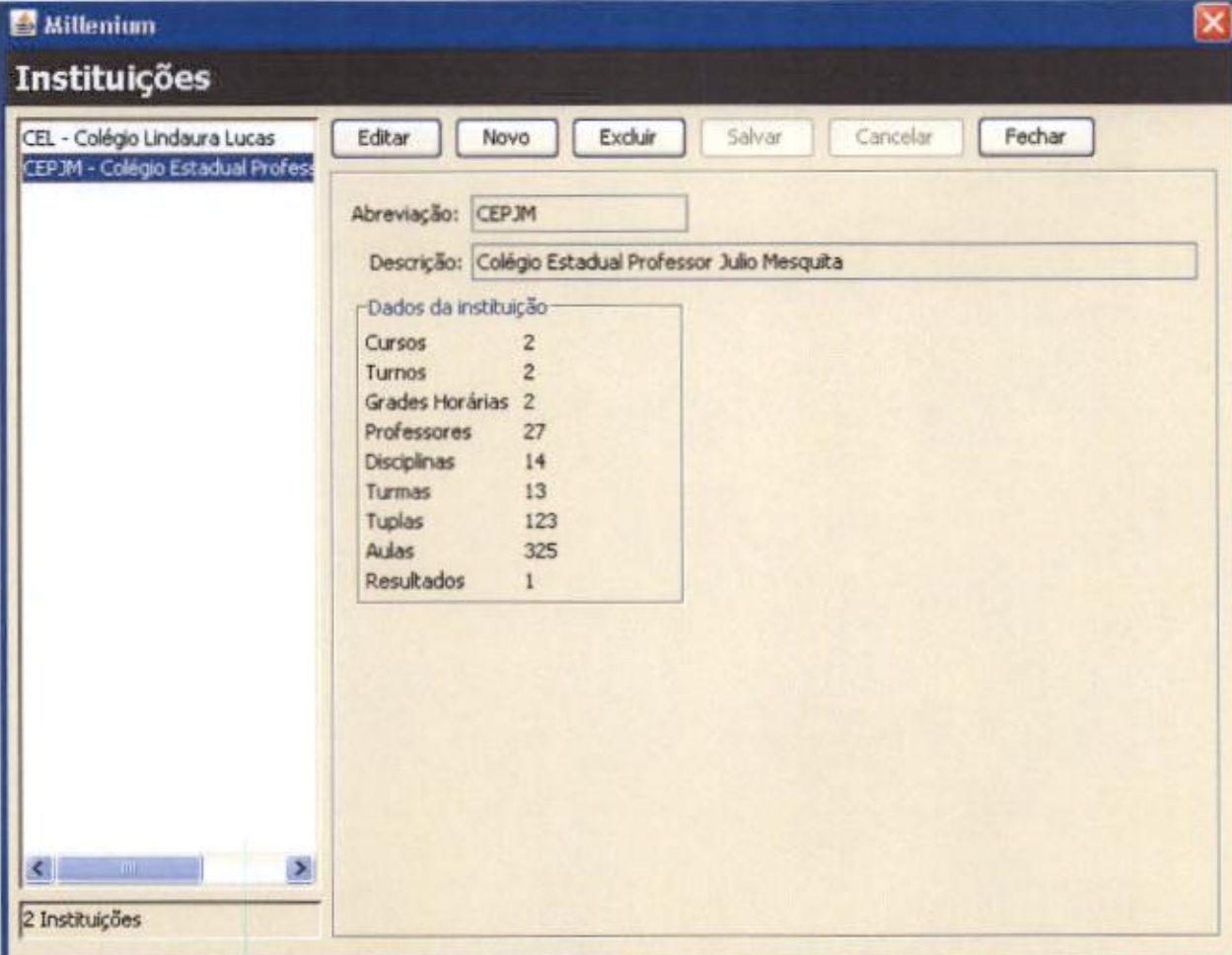
- Usuário clica no botão "Cancelar"
- Sistema retorna a tela de login

c) Executa o fluxo principal do caso de uso Efetuar Login

7.5.3 Caso de uso 3: manter instituição

7.5.3.1 Tela manter instituição

Ao usuário do Millenium é permitido cadastrar diversas instituições, através dessa tela pode-se gerenciar as instituições cadastradas.



Abreviação: CEPJM

Descrição: Colégio Estadual Professor Julio Mesquita

Dados da instituição

Cursos	2
Turnos	2
Grades Horárias	2
Professores	27
Disciplinas	14
Turmas	13
Tuplas	123
Aulas	325
Resultados	1

2 Instituições

FIGURA 18 – TELA DE CADASTRO DE INSTITUIÇÃO

FONTE: O AUTOR (2007)

7.5.3.2 Caso de uso 3: manter instituição

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como manter os dados de uma instituição no sistema.

Pré-Condições: Efetuado Login com permissão de administrador ou edição.

Pós-Condição: Novo registro cadastrado.

Fluxo de eventos principal:

- a) Usuário clica no botão "Escolher outra..."
- b) Sistema exibe tela de edição de instituição
- c) Usuário clica no botão "Editar..." (A2) – pág. 58
- d) Include (Manter Dados) - pág 63
- e) Usuário clica no botão "Fechar"
- f) Sistema encerra tela de edição de instituição
- g) Caso de uso finalizado

7.5.4 Caso de uso 4: Definir Senha

7.5.4.1 Tela principal

Esta é a tela principal do sistema que dá acesso as demais funcionalidades do Millenium.

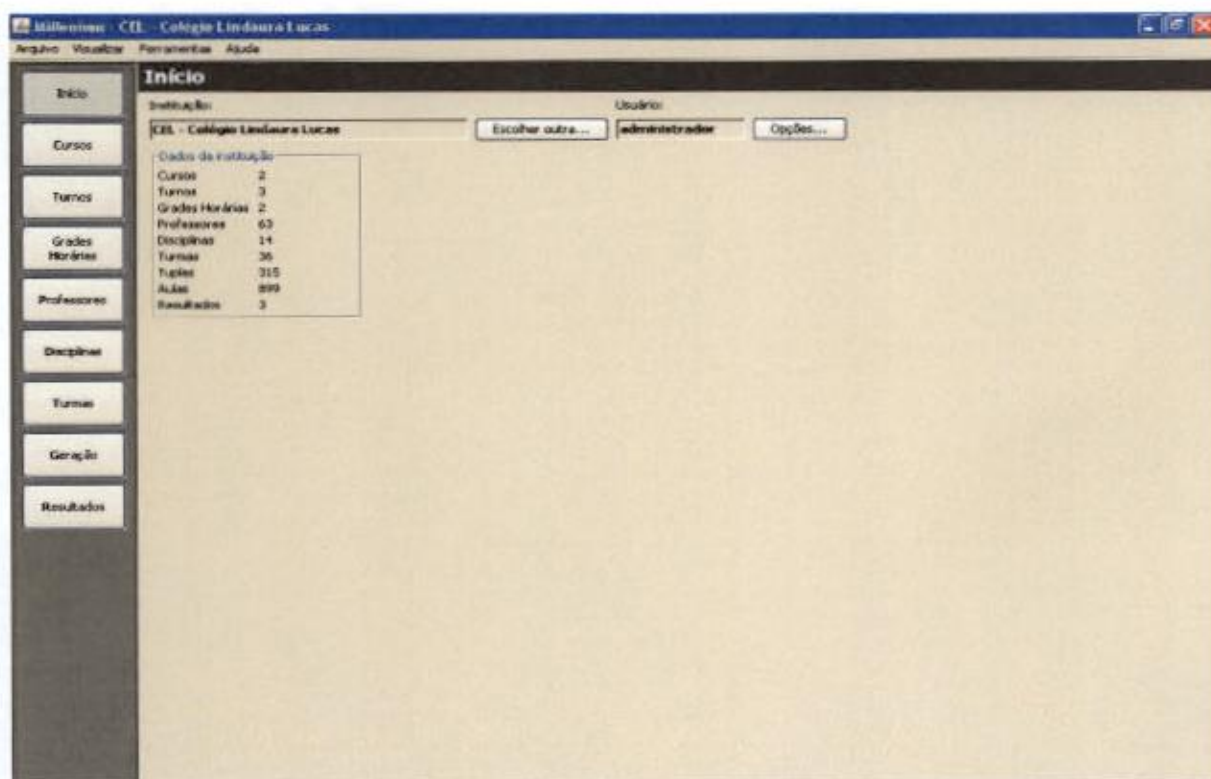


FIGURA 19 – TELA PRINCIPAL

FONTE: O AUTOR (2007)

7.5.4.2 Tela definição/alteração de senha

Esta opção existe para aumentar o nível de segurança, o usuário pode a qualquer momento alterar sua senha.

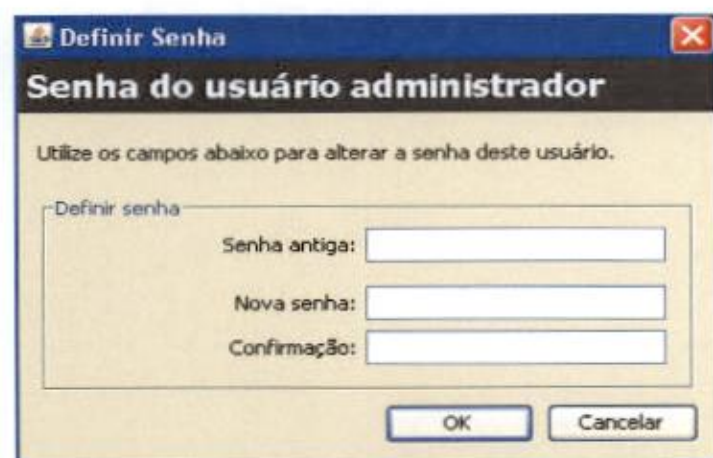


FIGURA 20 – TELA DE ALTERAÇÃO DE SENHA

FONTE: O AUTOR (2007)

7.5.4.3 Caso de uso 4: definir senha

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como alterar a senha do usuário.

Pré-Condições: Efetuado login.

Pós-Condição: Novo senha cadastrada.

Fluxo de eventos principal:

- a) Seleciona "Ferramentas/Alterar senha..." no menu
- b) Sistema exibe tela de edição de senha
- c) Usuário preenche os campos e clica no botão "OK" (A3)
- d) Sistema insere os dados
- e) Caso de uso finalizado

Fluxo de exceção:

E2: Usuário insere dado errado

- a) Sistema exibe mensagem de erro
- b) Usuário clica no botão "OK" da mensagem de erro
- c) Sistema fecha mensagem
- d) Usuário retorna ao passo (c) do fluxo principal

Fluxo Alternativo:

A3: Usuário desiste da operação

- a) Usuário clica no botão "Cancelar"
- b) Sistema encerra tela de edição de senha e retorna a tela principal
- c) Caso de uso finalizado

7.5.5 Caso de uso 5: manter dados

7.5.5.1 Caso de uso 5: manter dados

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como manter os dados no sistema.

Pré-Condições: Sistema deve estar aberto, possuir permissão de administrador ou edição.

Pós-Condição: Novo registro cadastrado.

Regra de Negócio

R3: Sistema deve verificar se o cadastro já existe

Fluxo de eventos principal:

- a) Usuário clica no botão "Novo" (A4) (A6) – pág 64
- b) Sistema torna campos editáveis
- c) Usuário preenche os campos
- d) Usuário clica no botão "Salvar" (A5) – pág. 64
- e) Sistema insere os dados (E2) (R3)
- f) Caso de uso finalizado

Fluxo de exceção:

E2: usuário insere dados já cadastrados

- a) Sistema exibe mensagem de erro informando que os dados já existem
- b) Usuário clica no botão "OK" da mensagem de erro
- c) Retorna ao passo (b) do Fluxo Principal

Fluxo Alternativo

A4: Usuário seleciona um registro na lista

- a) Usuário clica no botão "Editar"
- b) Sistema torna os campos editáveis
- c) Usuário altera os campos desejáveis
- d) Usuário clica no botão "Salvar"
- e) Sistema atualiza os dados no banco
- f) Caso de uso finalizado

A5: Usuário desiste da operação

- a) Usuário clica no botão "Cancelar"
- b) Sistema limpa os campos e os tornam não editáveis
- c) Caso de uso finalizado

A6: Usuário deseja excluir um registro selecionado

- a) Usuário clica no botão excluir
- b) Sistema solicita confirmação de exclusão
- c) Usuário clica no botão "OK"
- d) Sistema exclui registro
- e) Caso de uso finalizado

7.5.6 Caso de uso 6: Manter Usuario

7.5.6.1 Tela manter usuário

O administrador conta com o recurso de cadastrar e manter os usuários do sistema, dando a eles diferentes níveis de acesso ao sistema.

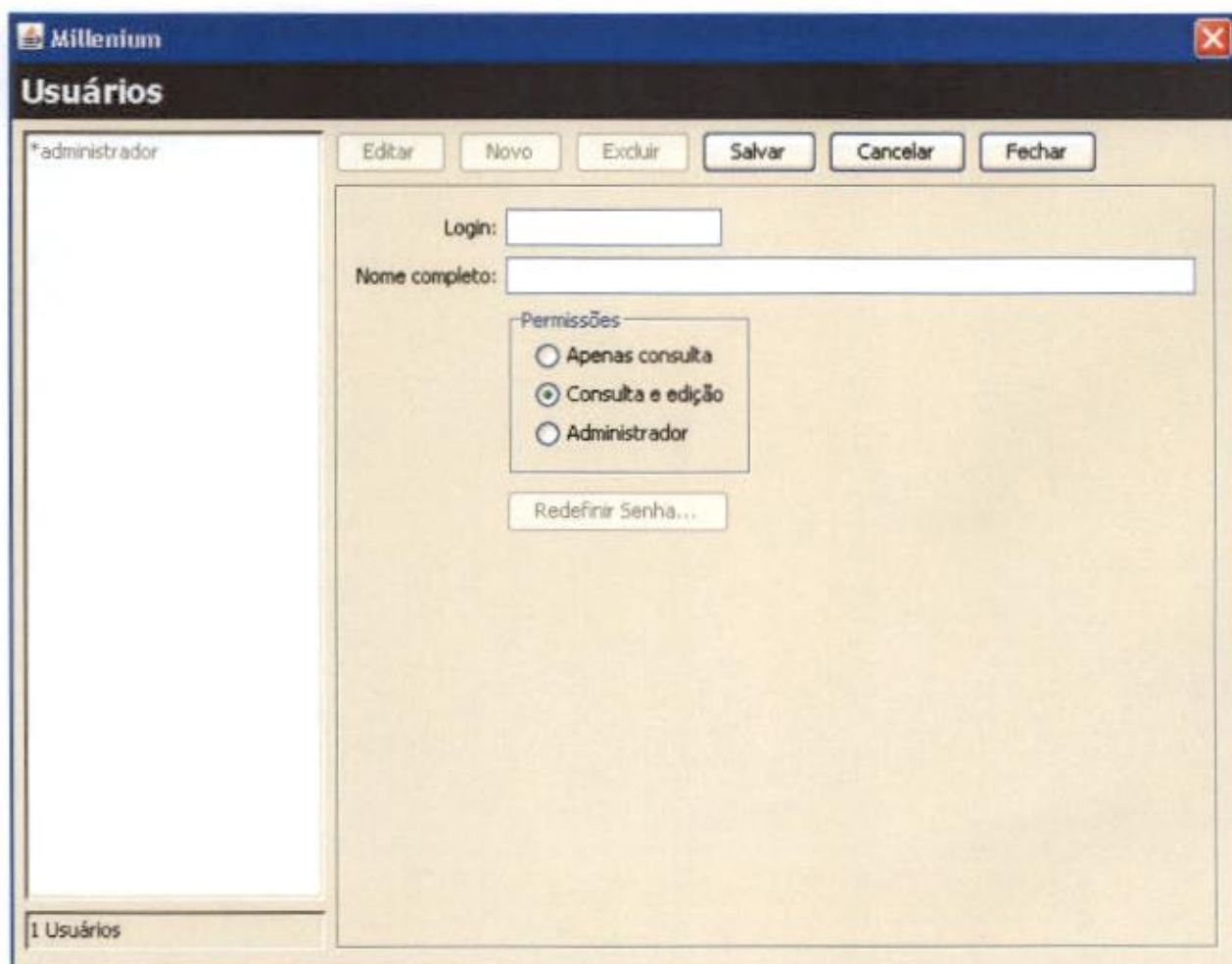


FIGURA 21 – TELA DE MANUTENÇÃO DE USUÁRIO

FONTE: O AUTOR (2007)

7.5.6.2 Caso de uso 6: manter usuario

Autor: Equipe

Ator: Usuário.

Regra de Negócio

R4: Sistema deve verificar se existe outro usuário cadastrado

Descrição: Este Caso de Uso descreve como manter os dados do usuário no sistema.

Pré-Condições: Sistema deve estar aberto, possuir permissão de administrador.

Pós-Condição: Novo registro cadastrado.

Fluxo de eventos principal:

- a) Usuário escolhe a opção "Ferramentas/Usuários" no menu

- b) Sistema exibe tela de cadastro de usuários
- c) Inclui (Manter Dados) - pág 63
- d) Usuário clica no botão "Fechar"
- e) Sistema encerra a tela de cadastro de usuário e retorna a tela principal
- f) Caso de uso finalizado

Fluxo Alternativo

- a) A4 – pág. 63 (R4)

7.5.7 Caso de uso 7: Manter Curso

7.5.7.1 Tela manter curso

Essa opção existe para que o usuário possa realizar a manutenção dos dados dos diferentes cursos existentes na instituição.

Cursos

Ens Fund - Ensino Fundamental
Ens Med - Ensino Medio

Editar Novo Excluir Salvar Cancelar

Abreviação:

Descrição:

2 Cursos

FIGURA 22 – TELA DE MANUTENÇÃO DE CURSOS

FONTE: O AUTOR (2007)

7.5.7.2 Caso de uso 7: manter curso

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como manter os dados do curso no sistema.

Pré-Condições: Sistema deve estar aberto, possuir permissão de administrador ou edição.

Pós-Condição: Novo registro cadastrado.

Fluxo de eventos principal:

- Usuário clica no botão “Cursos”
- Sistema exibe painel de edição de curso
- Include (Manter Dados) - pág 63

d) Caso de uso finalizado

7.5.8 Caso de uso 8: Manter Turno

7.5.8.1 Tela manter turno

Essa opção existe para que o usuário possa realizar a manutenção dos dados dos diferentes turnos existentes na instituição e relacionar a grade horária com um período do dia.

A interface de usuário para a manutenção de turnos, intitulada "Turnos". No topo, há uma barra de navegação com os botões "Editar", "Novo", "Excluir", "Salvar" e "Cancelar". À esquerda, há uma lista de turnos existentes: "Man - Manhã", "Not - Noturno" e "Tarde - Tarde". Abaixo desta lista, um contador indica "3 Turnos". À direita, há um formulário para a criação ou edição de um turno, com campos para "Abreviação:" e "Descrição:".

FIGURA 23 – TELA DE MANUTENÇÃO DOS TURNOS

FONTE: O AUTOR (2007)

7.5.8.2 Caso de uso 8: manter turno

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como manter os dados do turno no sistema.

Pré-Condições: Sistema deve estar aberto, possuir permissão de administrador ou edição.

Pós-Condição: Novo registro cadastrado.

Fluxo de eventos principal:

- a) Usuário clica no botão "Turnos"
- b) Sistema exibe painel de edição de turno
- c) Include (Manter Dados) - pág 63
- d) Caso de uso finalizado

7.5.9 Caso de uso 9: Manter Professor

7.5.9.1 Tela manter professor

Essa opção existe para que o usuário possa realizar a manutenção dos dados dos professores que lecionam na instituição.

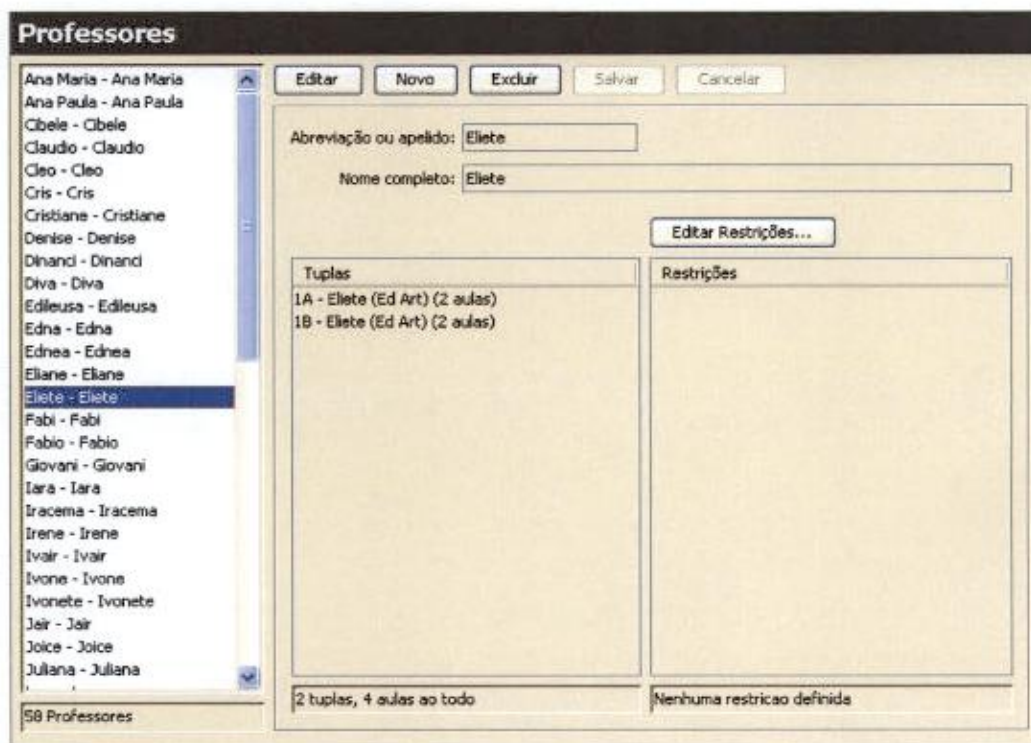


FIGURA 24 – TELA DE MANUTENÇÃO DE PROFESSOR

FONTE: O AUTOR (2007)

7.5.9.2 Caso de uso 9: manter professor

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como manter os dados do professor no sistema.

Pré-Condições: Sistema deve estar aberto, possuir permissão de administrador ou edição.

Pós-Condição: Novo registro cadastrado.

Fluxo de eventos principal:

- a) Usuário clica no botão “Professores”
- b) Sistema exibe painel de edição do professor
- c) Include (Manter Dados) - pág 63
- d) Sistema torna visível o botão “Editar Restrições...”
- e) Caso de uso finalizado (A7)

Fluxo Alternativo:

A7: Usuário deseja editar restrições

- a) Usuário clica no botão "Editar Restrições"
- b) Sistema abre tela de edição de restrições
- c) Inclui (Manter Dados) - pág 63
- d) Usuário clica no botão "Fechar"
- e) Sistema encerra a tela de cadastro de restrições e retorna ao painel de edição
- f) Caso de uso finalizado

7.5.10 Caso de uso 10: manter disciplina

7.5.10.1 Tela manter disciplina

Essa opção existe para que o usuário possa realizar a manutenção dos dados das disciplinas ministradas pela instituição.

FIGURA 25 – TELA DE MANUTENÇÃO DE DISCIPLINA

FONTE: O AUTOR (2007)

7.5.10.2 Caso de uso 10: manter disciplina

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como manter os dados das disciplinas no sistema.

Pré-Condições: Sistema deve estar aberto, possuir permissão de administrador ou edição.

Pós-Condição: Novo registro cadastrado.

Fluxo de eventos principal:

- a) Usuário clica no botão "Disciplinas"
- b) Sistema exibe painel de edição de disciplina

- c) Include (Manter Dados) - pág 63
- d) Caso de uso finalizado (A7) – pág. 70

7.5.11 Caso de uso 11: manter turmas

7.5.11.1 Tela manter turmas

Essa opção existe para que o usuário possa realizar a manutenção dos dados das turmas existentes na instituição.

Turmas

Editar Novo Excluir Salvar Cancelar

Abreviação: 5G

Descrição: 5G

Curso: Ens Fund - Ensino Fundamental

Grade horária: GTarde - Grade Tarde

Editar Tuplas... Editar Restrições...

Tuplas	Restrições
5G - Luciene (Port) (4 aulas)	
5G - Marcia (Ingles) (2 aulas)	
5G - Marluci (Ed Art) (2 aulas)	
5G - Eliane (Ed Fis) (3 aulas)	
5G - M ^a Clara (Mat) (4 aulas)	
5G - Irene (Cien) (3 aulas)	
5G - Juliana (Hist) (3 aulas)	
5G - Ednea (Geo) (3 aulas)	
5G - M ^a Odete (Ens. Rel) (1 aulas)	

36 Turmas

9 tuplas, 25 aulas ao todo

Nenhuma restricao definida

FIGURA 26 – TELA DE MANUTENÇÃO DE TURMAS

FONTE: O AUTOR (2007)

7.5.11.2 Caso de uso 11: manter turmas

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como manter os dados das turmas no sistema.

Pré-Condições: Sistema deve estar aberto, possuir permissão de administrador ou edição.

Pós-Condição: Novo registro cadastrado.

Fluxo de eventos principal:

- a) Usuário clica no botão "Turmas"
- b) Sistema exibe painel de edição de turmas
- c) Inclui (Manter Dados) - pág 63
- d) Caso de uso finalizado (A7) – pág. 70 (A8) – pág. 74

7.5.11.3 Tela de edição de aulas

Essa opção existe para que o usuário possa realizar o relacionamento entre turma, professor, disciplina e a quantidade de aula que ele leciona na semana para aquela turma.

Millenium

Aulas da Turma 5A

Novo Editar Excluir Salvar Cancelar Fechar

Turma:

Aula normal

Professor: Disciplina:

Aulas por semana: Dividir a turma entre 2 professores

Divisão de turma

Professor secundário: Disciplina secundária:

9 tuplas, 25 aulas ao todo

FIGURA 27 – TELA DE EDIÇÃO DE AULAS

FONTE: O AUTOR (2007)

Fluxo alternativo:

A8: Usuário deseja editar Aulas

- Usuário clica no botão "Editar Aulas..."
- Sistema abre tela de edição de aulas
- Inclui (Manter Dados)
- Usuário clica no botão "Fechar"
- Sistema encerra a tela de edição de tuplas e retorna ao painel de edição de turmas
- Caso de uso encerrado

7.5.12 Caso de uso 12: manter grade

7.5.12.1 Tela manter grade

Essa opção existe para que o usuário possa realizar a manutenção dos dados das grades existentes na instituição.

Grades Horárias

GM - GradeManha
 GTarde - Grade Tarde

Editar Novo Excluir Salvar Cancelar

Abreviação: GTarde

Descrição: Grade Tarde

Turno: Tarde - Tarde

Segunda	Terça	Quarta	Quinta	Sexta
13:30 - 14:20 A...	13:30 - 14:20 A...	13:30 - 14:20 A...	13:30 - 14:20 A...	13:30 - 14:20 A...
14:20 - 15:10 A...	14:20 - 15:10 A...	14:20 - 15:10 A...	14:20 - 15:10 A...	14:20 - 15:10 A...
15:10 - 15:20 In...	15:10 - 15:20 In...	15:10 - 15:20 In...	15:10 - 15:20 In...	15:10 - 15:20 In...
15:20 - 16:10 A...	15:20 - 16:10 A...	15:20 - 16:10 A...	15:20 - 16:10 A...	15:20 - 16:10 A...
16:10 - 17:00 A...	16:10 - 17:00 A...	16:10 - 17:00 A...	16:10 - 17:00 A...	16:10 - 17:00 A...
17:00 - 17:50 A...	17:00 - 17:50 A...	17:00 - 17:50 A...	17:00 - 17:50 A...	17:00 - 17:50 A...

Dias... Horários...

2 Grades Horárias

FIGURA 28 – TELA DE MANUTENÇÃO DE GRADE HORÁRIA

FONTE: O AUTOR (2007)

7.5.12.2 Caso de uso 12: manter grade

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como manter os dados da grade horária no sistema.

Pré-Condições: Sistema deve estar aberto, possuir permissão de administrador ou edição.

Pós-Condição: Novo registro cadastrado.

Fluxo de eventos principal:

- a) Usuário clica no botão "Grades Horárias"
- b) Sistema exibe painel de edição de grade horária
- c) Inclui (Manter Dados) - pág 63
- d) Sistema torna visível a tabela de dias e seus horários e os botões "Dias..." e "Horários..."
- e) Caso de uso finalizado (A9) - pág. 77 (A10) – pág. 78

7.5.12.3 Tela de edição dos dias

Essa opção existe para que o usuário possa realizar a manutenção dos dados dos dias das grades horárias, pois cada grade pode trabalhar com diferentes dias.

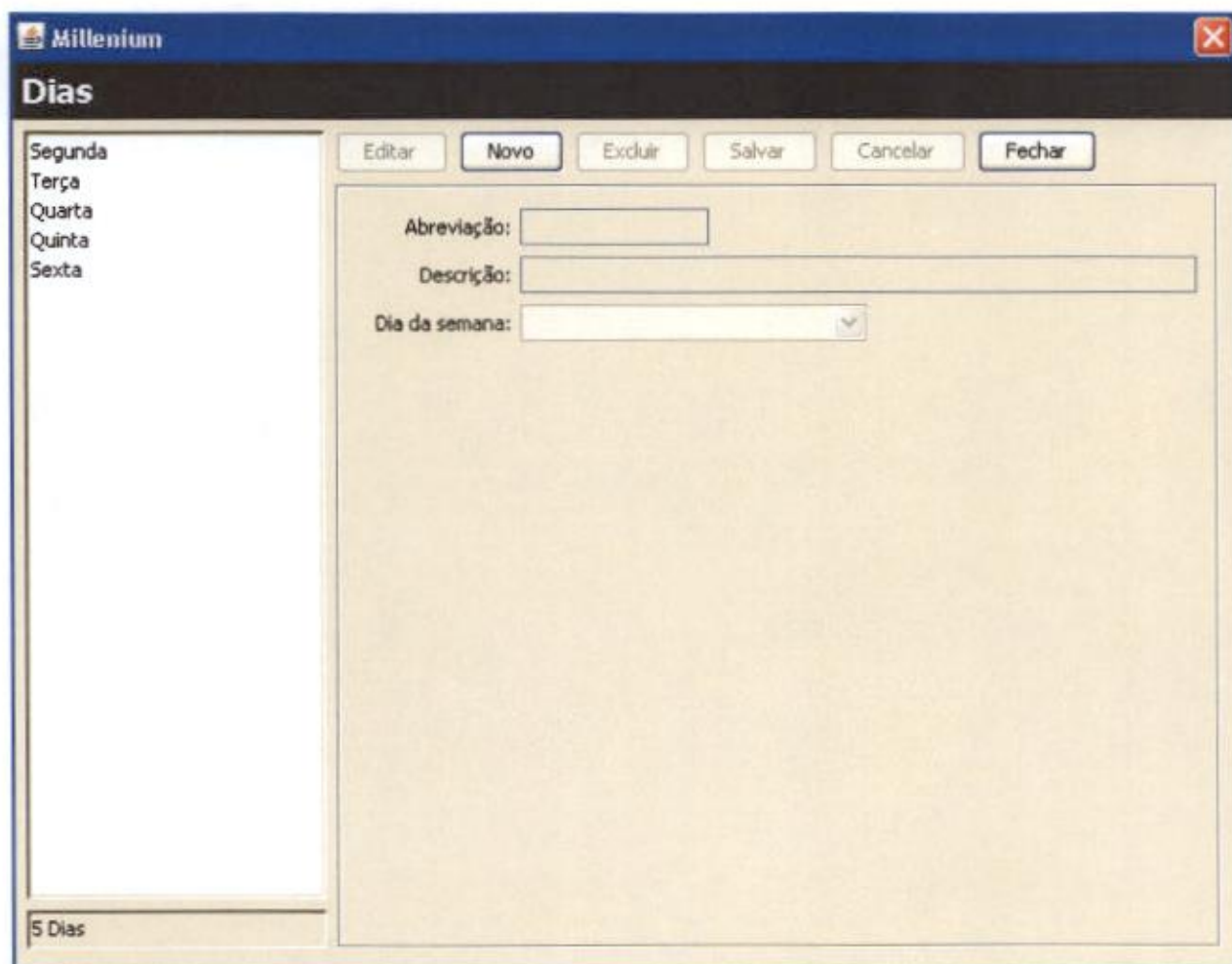


FIGURA 29 – TELA DE EDIÇÃO DOS DIAS

FONTE: O AUTOR (2007)

Fluxo Alternativo:

A9: Usuário deseja editar os dias da semana

- a) Usuário clica no botão "Dias.."
- b) Sistema abre tela de edição de dias
- c) Include (Manter Dados)
- d) Usuário clica no botão "Fechar"
- e) Sistema encerra tela de edição de dias e retorna ao painel de edição de grade horária
- f) Caso de uso finalizado

7.5.12.4 Tela edição dos horários

Essa opção existe para que o usuário possa realizar a manutenção dos horários, pois as aulas podem ter durações diferentes, os dias da grade também podem funcionar em horários alternados.

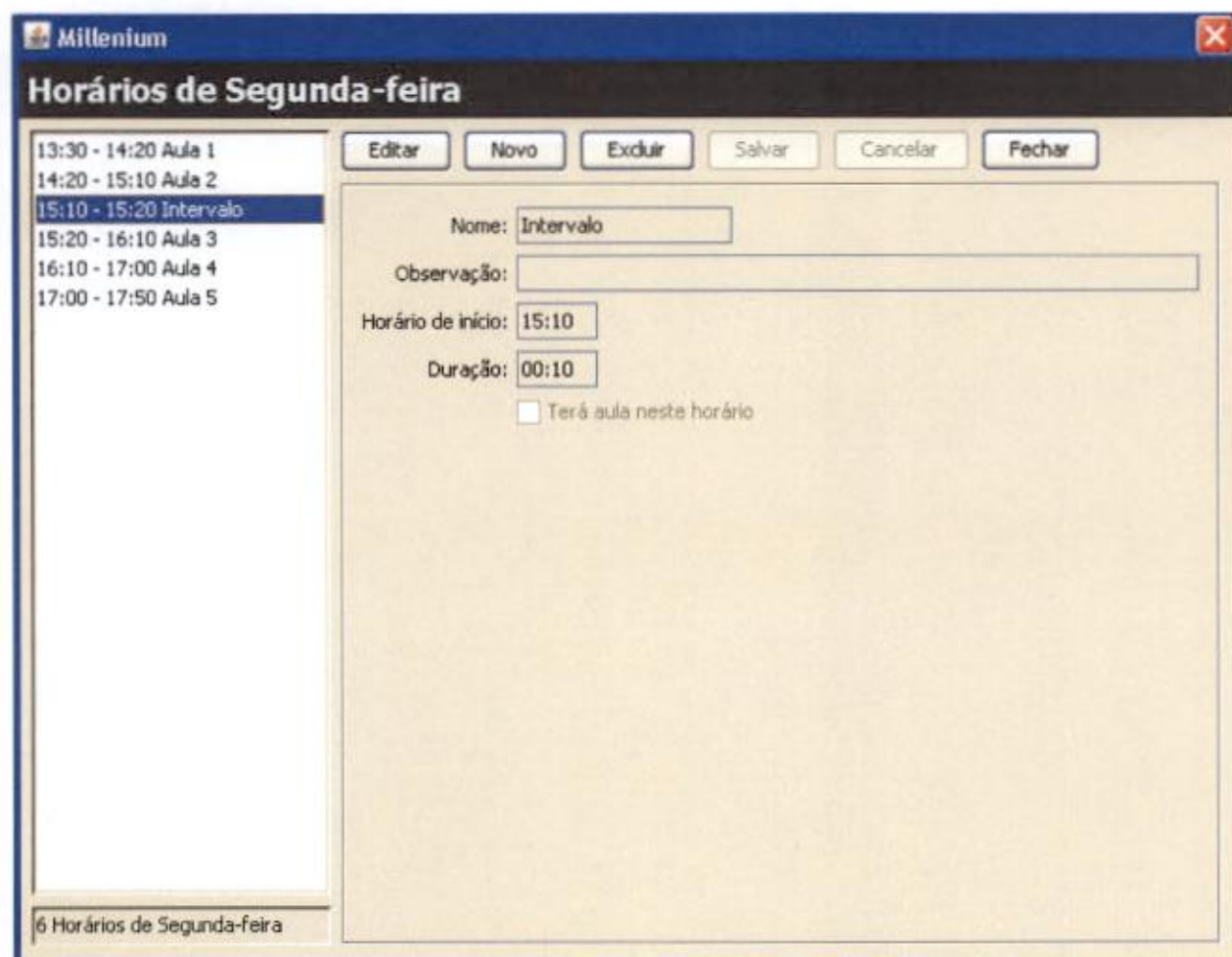


FIGURA 30 – TELA DE EDIÇÃO DE HORÁRIOS

FONTE: O AUTOR (2007)

A10: Usuário deseja editar os horários

- a) Usuário seleciona um dia da semana
- b) Sistema torna o botão "Horários..." editável
- c) Usuário clica no botão "Horários..."
- d) Sistema abre tela de edição de horários
- e) Include (Manter Dados) - pág 63
- f) Usuário clica no botão "Fechar"

- g) Sistema encerra tela de edição de horário e retorna ao painel de edição de grade
- h) Caso de uso finalizado

7.5.13 Caso de uso 13: gerar grade

7.5.13.1 Tela gerar grade

Essa opção existe para que o usuário possa criar a tabela de horário, nesta fase o sistema irá alocar os professores nas turmas em um determinado horário, respeitando as restrições cadastradas.

Geração

Tipo de execução

- Criar novo resultado
- Usar resultado existente como base

Opções de Execução

Número máximo de iterações (0 = ilimitado):

Tempo máximo de execução (00:00 = ilimitado): (horas:minutos)

Turno

- Man - Manhã
- Not - Noturno
- Tarde - Tarde

Restrições

Forçar aulas da mesma tupla a serem sequenciais

Máximo de aulas da mesma tupla por dia (0 = ilimitado):

Janelas de Turmas

- Ignorar
- Mandar para o início
- Mandar para o final
- Início ou final

Janelas de Professores

- Ignorar
- Mandar para o início
- Mandar para o final
- Início ou final

Iniciar

FIGURA 31 – TELA DE GERÇÃO DE GRADE HORÁRIA

FONTE: O AUTOR (2007)

7.5.13.1.1 Tela de execução do genético

Essa opção existe para que o usuário possa acompanhar o andamento da criação da grade e interromper a qualquer momento para análise do resultado produzido.

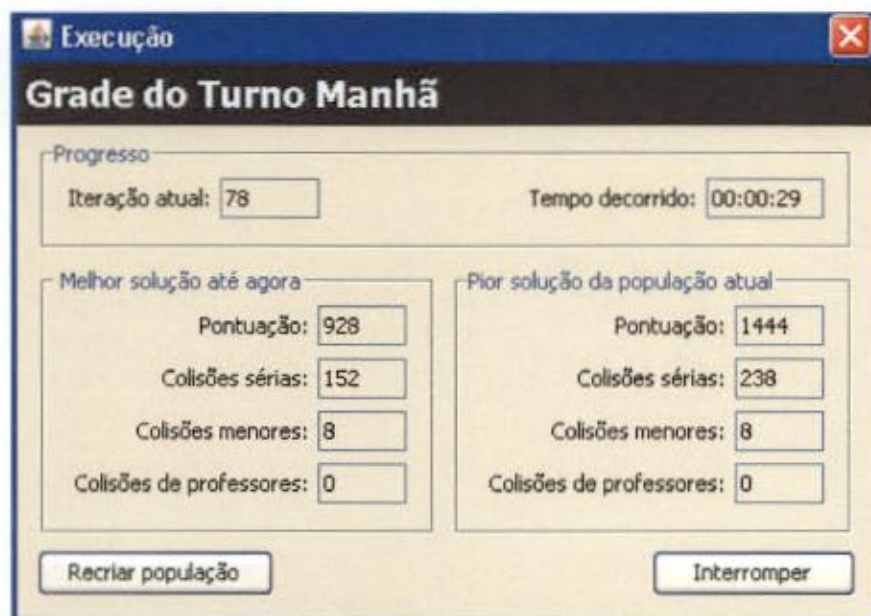


FIGURA 32 – TELA DE EXECUÇÃO DO GENÉTICO

FONTE: O AUTOR (2007)

7.5.13.1.2 Tela de visualização do resultado

Essa opção existe para que o usuário possa visualizar o resultado criado em diferentes ângulos, através dela ele tem a opção de salvar ou descartar a solução apresentada.

Execução Interrompida

Iteração: 7 Tempo: 00:00:02 Colisões: 186 / 10 / 0

Modelo: *Turmas Horizontal - Mod... Editor...

Tabela: Man

	1A	1B	2A	3A	6A	7A	7B	7C	7D	7E	7F	7G	8A	8B	8C	8D	8E	8F
Seq...	Patri...	Div...	Paul...	Jair...	Ros...	Gov...	Cleo...	Gov...	Ana...	Toni...	Luan...	Toni...	Nice...	Simo...	Edn...	Simo...	Clau...	Simo...
Seq...	Div...	Silv...	Silav...	Jair...	Pen...	Lud...	Cleo...	Ros...	Ivair...	Toni...	Nadi...	MP C...	Nice...	Simo...	Clau...	Nice...	Patri...	Edn...
Seq...	Elet...	Cris...	Silav...	Lara...	Marl...	Ivair...	Ros...	Cleo...	MP C...	Lind...	Fabi...	Toni...	Jair...	Iren...	Pedr...	Pedr...	Simo...	Edn...
Seq...	Patri...	Lara...	Paul...	Patri...	Lud...	Gov...	Pedr...	Marl...	Ivair...	Wim...	Toni...	MP C...	Div...	Toni...	Clau...	Edn...	Pedr...	Silv...
Seq...	Edn...	Elet...	Cris...	Ivair...	Marl...	Marl...	--	Gov...	MP C...	Ana...	Luan...	Div...	Toni...	Toni...	Clau...	Pedr...	Iren...	Patri...
Terç...	Div...	Silav...	Fabi...	Silv...	Pen...	Marl...	Ros...	Ivair...	Pen...	MP C...	Fabi...	Ana...	Simo...	Iren...	Silv...	Clau...	Pedr...	Edn...
Terç...	Lara...	Div...	Cris...	Silv...	Lud...	Cleo...	Nadi...	MP C...	Ros...	Ana...	Lud...	Luan...	Jair...	Nice...	Patri...	Clau...	Clau...	Silav...
Terç...	Silav...	Patri...	Silav...	Lara...	Nadi...	Ros...	Marl...	MP C...	Ros...	Nadi...	Fabi...	Toni...	Deni...	Jair...	Iren...	Simo...	Silav...	Clau...
Terç...	Lara...	Jair...	Edn...	Ana...	Lind...	Ros...	Lud...	Marl...	Cleo...	Toni...	MP C...	Fabi...	Nice...	Deni...	Edn...	Pedr...	Patri...	Clau...
Terç...	Silv...	Edn...	Lara...	Fabi...	Ros...	Ivair...	Lud...	Nadi...	Ros...	MP C...	Ana...	Nadi...	Simo...	Div...	Nice...	Edn...	Iren...	Iren...
Qua...	Silav...	Elet...	Lara...	Div...	Lind...	Lind...	Marl...	Nadi...	Nadi...	Wim...	Lud...	MP C...	Toni...	Jair...	Silv...	Nice...	Silav...	Silav...
Qua...	Silav...	Lara...	Patri...	Cris...	Toni...	Marl...	Ivair...	Ivair...	MP C...	MP C...	MP C...	Fabi...	Toni...	Div...	Edn...	Patri...	Simo...	Lara...
Qua...	Clau...	Silav...	Ana...	Patri...	Pen...	Lind...	Pedr...	Ivair...	Lud...	MP C...	Ana...	Ana...	Jair...	Fabi...	Patri...	Patri...	Edna...	Iren...
Qua...	Jair...	Clau...	Ana...	Silv...	Lind...	Lind...	Marl...	Marl...	Pen...	Lud...	Toni...	Luan...	Fabi...	Fabi...	Pedr...	Silv...	Iren...	Patri...
Qua...	Jair...	Lara...	Fabi...	Silv...	Marl...	Gov...	Pedr...	Cleo...	Ana...	Toni...	Ana...	Toni...	Luan...	Toni...	Nice...	Nice...	Edna...	Silav...
Quin...	Elet...	Jair...	Lara...	Silv...	Toni...	Cleo...	Gov...	Marl...	Ana...	Fabi...	Luan...	Lud...	Fabi...	Jair...	Iren...	Iren...	Silv...	Lara...
Quin...	Clau...	Cris...	Silav...	Ana...	Cleo...	Marl...	Gov...	Lud...	Ivair...	Nadi...	Fabi...	Ana...	Nice...	Deni...	Iren...	Pedr...	Iren...	Lara...
Quin...	Silav...	Clau...	Ana...	Lara...	Cleo...	Lud...	Ivair...	Ros...	Ivair...	Lind...	Toni...	MP C...	Luan...	Iren...	Nice...	Patri...	Edna...	Iren...
Quin...	Lara...	Silv...	Silv...	Div...	Nadi...	Nadi...	Pedr...	Ros...	Cleo...	Lind...	MP C...	Lud...	Jair...	Fabi...	Patri...	Nice...	Silav...	Lara...
Quin...	Cris...	Edn...	Jair...	Silv...	Pen...	Ros...	Marl...	Ivair...	MP C...	Lud...	Nadi...	Fabi...	Luan...	Div...	Simo...	Iren...	Clau...	Patri...
Sext...	Edn...	Silav...	Jair...	Lara...	Ros...	Nadi...	Cleo...	MP C...	Ana...	Ana...	MP C...	Luan...	Fabi...	Nice...	Pedr...	Iren...	Pedr...	Iren...
Sext...	Cris...	Edn...	Lara...	Fabi...	Lind...	Cleo...	Nadi...	Cleo...	Pen...	Wim...	Toni...	Div...	Div...	Jair...	Simo...	Iren...	Pedr...	Silv...
Sext...	Silv...	Silav...	Silv...	Ana...	Cleo...	Ivair...	Ivair...	Gov...	Nadi...	Ana...	Fabi...	Ana...	Deni...	Iren...	Nice...	Clau...	Silav...	Simo...
Sext...	Lara...	Patri...	Edn...	Ivair...	Marl...	Ivair...	Ros...	Lud...	Lud...	Fabi...	Fabi...	Div...	Luan...	Nice...	Iren...	Edn...	Patri...	Silav...

Salvar Solução Continuar Fechar

FIGURA 33 – TELA RESULTADO

FONTE: O AUTOR (2007)

7.5.13.1.3 Tela salvar resultado

Essa opção existe para que o usuário possa salvar um resultado que ele tenha aprovado ou para servir de base para próxima execução.

Salvar Resultado

Salvar Resultado

Nome:

Descrição:

Resultados existentes

Resultado_1

OK Cancelar

FIGURA 34 – TELA SALVAR RESULTADO

FONTE: O AUTOR (2007)

7.5.13.2 Caso de uso 13: gerar grade

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como gerara a grade horária.

Pré-Condições: Efetuado login, selecionado uma instituição e realizado os casos de uso 7 a 12.

Pós-Condição: Grade horária disponível.

Fluxo de eventos principal:

- a) Usuário clica no botão "Geração"
- b) Sistema exibe painel de execução do genético
- c) Usuário preenche os campos, seleciona um turno e clica no botão iniciar (E3) (E4)
- d) Sistema exibe tela de andamento da execução, encontra a melhor solução e apresenta o resultado ao usuário (A12) – pág. 83
- e) Usuário clica no botão "Salvar Solução" (A12) – pág. 83
- f) Sistema exibe tela "Salvar Resultado"
- g) Usuário preenche campos e clica no botão "OK" (A12) – pág. 83
- h) Sistema encerra a tela e retorna ao painel de execução do genético
- i) Caso de uso finalizado

Fluxo de exceção:

E3: Usuário não cadastrou tuplas para alguma turma

- a) Sistema exibe mensagem de erro
- b) Usuário clica no botão "OK" da mensagem
- c) Sistema encerra mensagem e retorna ao painel de execução do genético
- d) Caso de uso finalizado

E4: Usuário cadastrou restrições que impedem execução do genético

- a) Sistema exibe mensagem de erro
- b) Usuário clica no botão "OK" da mensagem
- c) Sistema encerra mensagem e retorna ao painel de execução do genético
- d) Caso de uso encerrado

Fluxo Alternativo:

A11: Usuário interrompe execução do genético

- Usuário clica no botão "Interromper"
- Sistema apresenta o resultado gerado
- Retorna ao passo (e) do fluxo principal (A12)

A12: Usuário desiste da operação

- Usuário clica no botão "Cancelar"
- Sistema encerra painel de apresentação do resultado e retorna ao painel de execução
- Caso de uso finalizado

7.5.14 Caso de uso 14: visualizar grade

7.5.14.1 Tela visualizar resultado

Essa opção existe para que o usuário possa visualizar um resultado que ele tenha salvo e possibilitar a impressão ou exportação do mesmo para outros formatos.

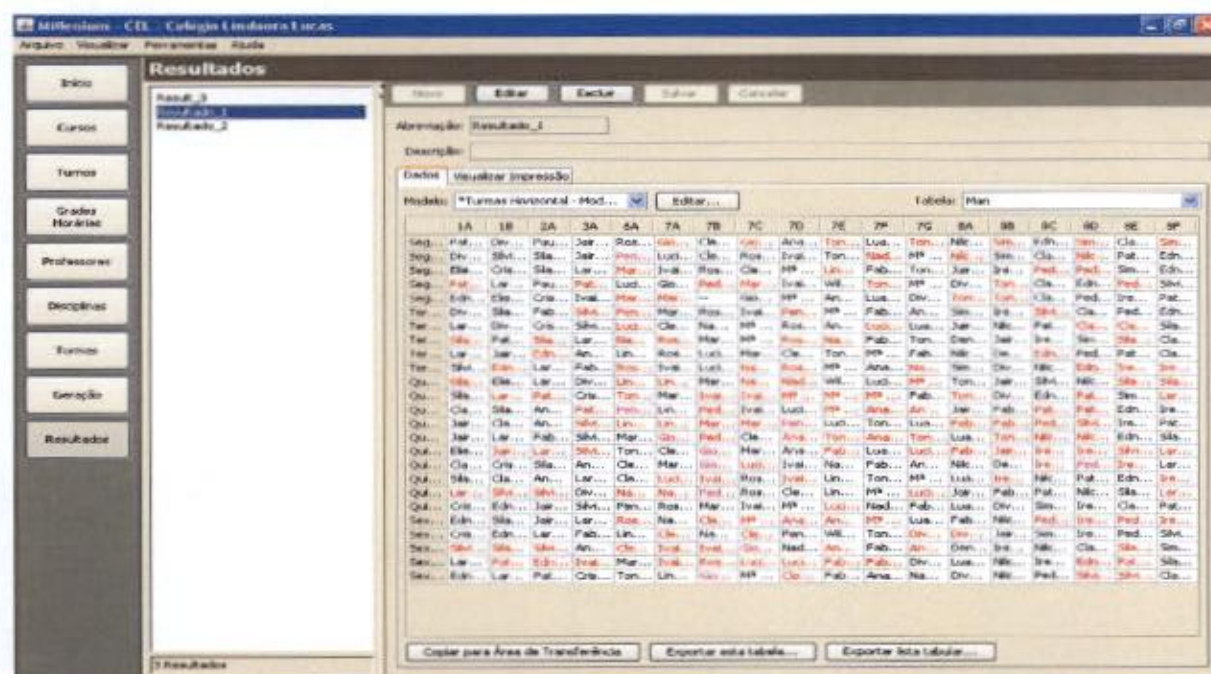


FIGURA 35 – TELA VISUALIZAR RESULTADO

FONTE: O AUTOR (2007)

7.5.14.2 Caso de uso 14: visualizar grade

Autor: Equipe

Ator: Usuário.

Descrição: Este Caso de Uso descreve como visualizar uma grade horária gerada.

Pré-Condições: Efetuado login, existir uma grade salva.

Pós-Condição: Exportação e impressão da grade horária.

Fluxo de eventos principal:

- a) Usuário clica no botão "Resultados"
- b) Sistema exhibe painel de resultado
- c) Usuário seleciona um resultado na lista e o modelo de apresentação
- d) Sistema exhibe modelo selecionado
- e) Caso de uso finalizado

Fluxo Alternativo:

A13: Usuário deseja excluir resultado

- a) Usuário clica no botão "Excluir"
- b) Sistema pede confirmação de exclusão
- c) Usuário clica no botão "OK"
- d) Sistema exclui o registro
- e) Caso de uso finalizado

A14: Usuário deseja editar nome do resultado gerado

- a) Usuário clica no botão "Editar"
- b) Sistema torna os campos editáveis
- c) Usuário altera os campos e clica no botão "Salvar"
- d) Sistema salva as alterações
- e) Caso de uso finalizado

7.6. CENÁRIO

O Sistema de geração de grade horária visa facilitar e tornar mais ágil a

construção e manutenção de horários de instituições de ensino. Ele poderá ser utilizado por qualquer profissional da área pedagógica que deseje automatizar o processo de elaboração de horários escolares dentro do seu ambiente de trabalho.

Um usuário autorizado entra no sistema seleciona uma instituição cadastra cursos, turnos, grades horárias, professores, disciplinas turmas e turmas para a instituição selecionada. O usuário faz os devidos relacionamentos entre professores, disciplinas e turmas com suas respectivas restrições e logo após o sistema tem condições de gerar o horário escolar.

O tempo de resposta ao usuário será proporcional ao grau de complexidade das restrições cadastradas para os professores, disciplinas e turmas.

7.7. DIAGRAMAS DO PROJETO

Esses diagramas são utilizados para possibilitar a visão estática da estrutura dos sistemas com modelagem orientada a objetos (Booch; Rumbauch; Jacobson, 2000, p 90). Ele exibe um conjunto de classes, interfaces e colaborações, bem como o relacionamento entre eles.

A UML (Unified Modeling Language) define três tipos de relacionamentos como sendo os que abrangem os modos mais importantes de colaborações entre os itens e segundo Booch, Rumbauch e Jacobson (2000, p. 60) são eles:

- Dependência que são relacionamentos de utilização
- Generalização: conectam classes generalizadas a outras mais especializadas, relacionamento subclasse/superclasse.
- Associação relacionamentos estruturais entre as instâncias

Para possibilitar uma melhor manipulação e organização do projeto adotamos o mecanismo de pacotes para organizar os elementos do sistema Millennium. Durante o desenvolvimento do projeto foi constatada a necessidade de dividir os elementos em oito pacotes:

- dados: ele contém superclasses necessárias a outros pacotes, arquivo de script SQL, e a classe de conexão com base de dados;
- modelo: nele estão organizados os elementos que abstraem parte do domínio do problema, pois nelas representamos professores, turmas,

disciplinas, etc;

- genético: nele estão organizados os elementos essenciais para a solução do problema;
- relatórios: nele estão organizados os arquivos e classes necessárias para geração dos relatórios do sistema;
- resultado: nele estão as classes necessárias para geração e armazenamento do resultado no banco de dados;
- telas: nele estão os elementos que representam as interfaces do sistema
- útil: nele estão os elementos que prestam serviços a elementos de outros pacotes;
- principal: nele estão os elementos que possibilitarão a execução do sistema;

A partir deste momento será apresentado o diagrama de classes do projeto, neles representaremos seus relacionamentos, sub divididos em pacotes, de generalização, dependências e associação.

7.7.1 Diagrama de pacotes do millenium

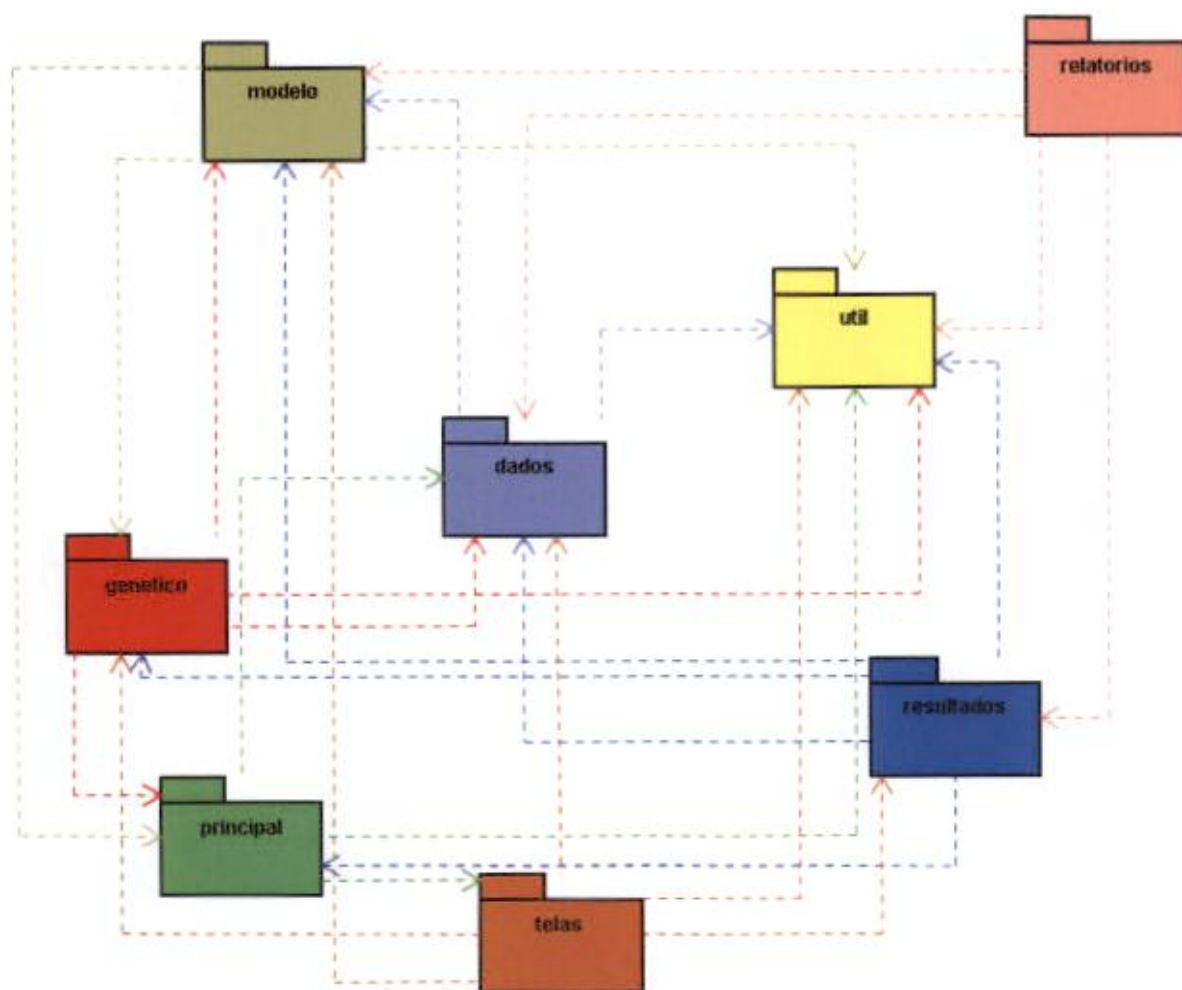


FIGURA 36 – DIAGRAMA DE PACOTES

FONTE: O autor (2007)

7.7.2 Relacionamento de Generalização

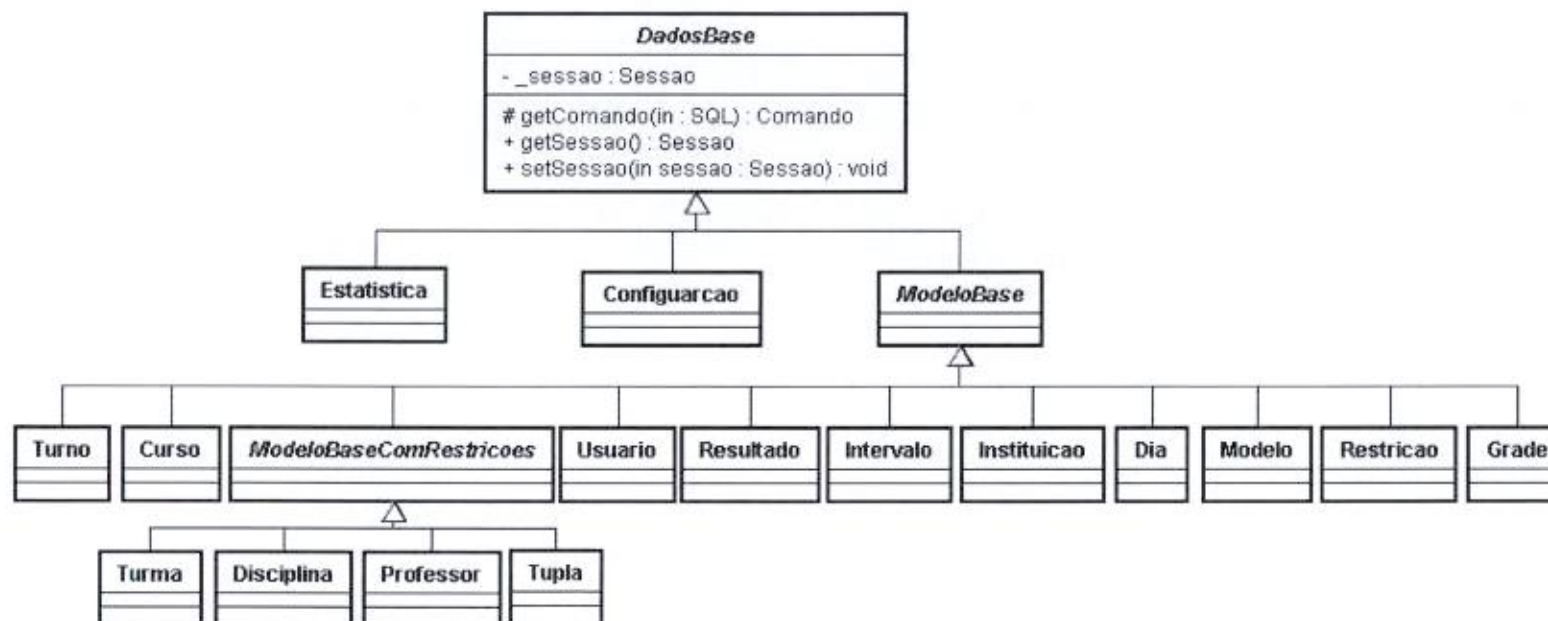


FIGURA 37 – RELACIONAMENTO DE GENERALIZAÇÃO

FONTE: O autor (2007)

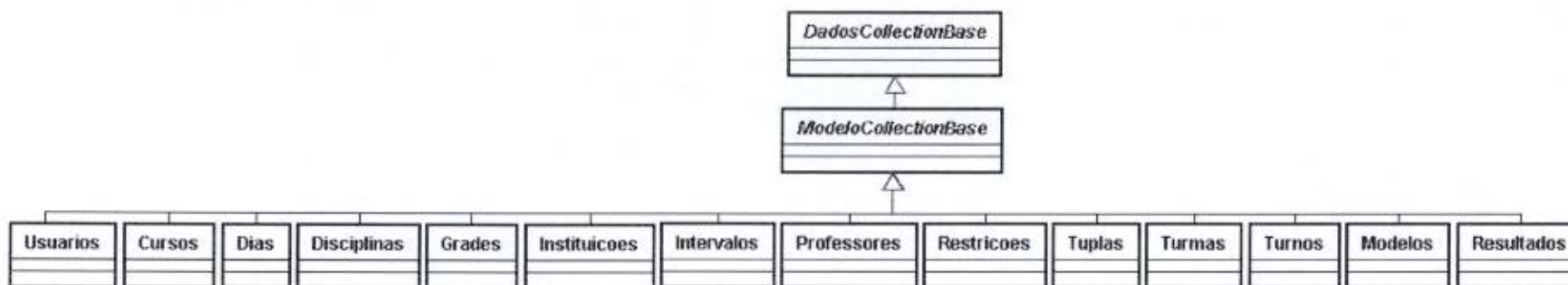


FIGURA 38 – RELACIONAMENTO DE GENERALIZAÇÃO

FONTE: O autor (2007)

7.7.3 Diagrama de classes do pacote

7.7.3.1 Diagrama de classes do pacote dados

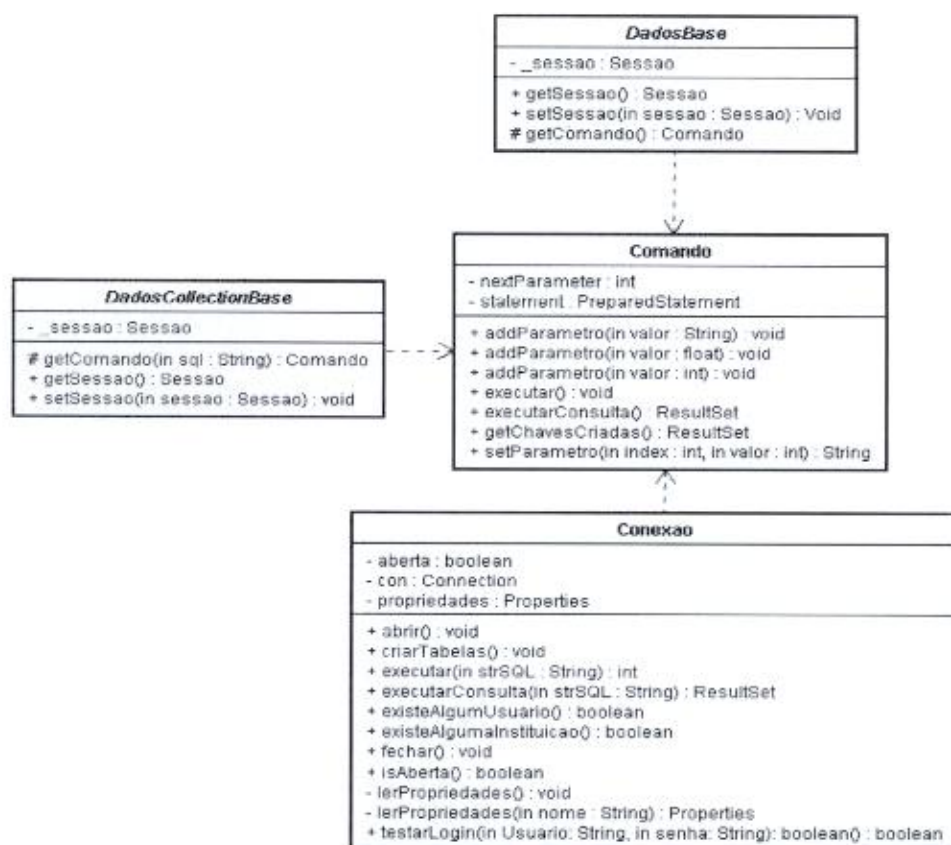


FIGURA 39 – RELACIONAMENTO DE DEPENDÊNCIA

FONTE: O autor (2007)

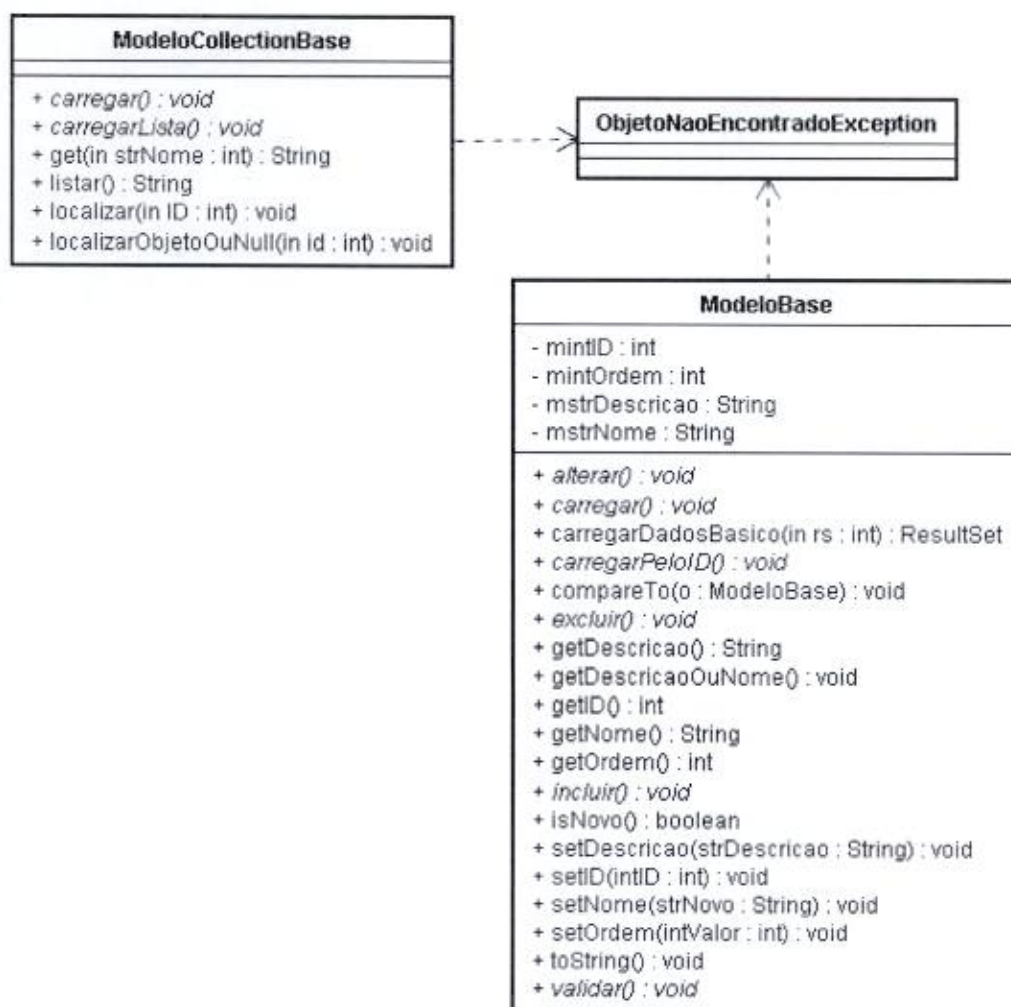


FIGURA 40 – RELACIONAMENTO DE DEPENDÊNCIA

FONTE: O autor (2007)

7.7.3.2 Diagrama de classes do pacote genético

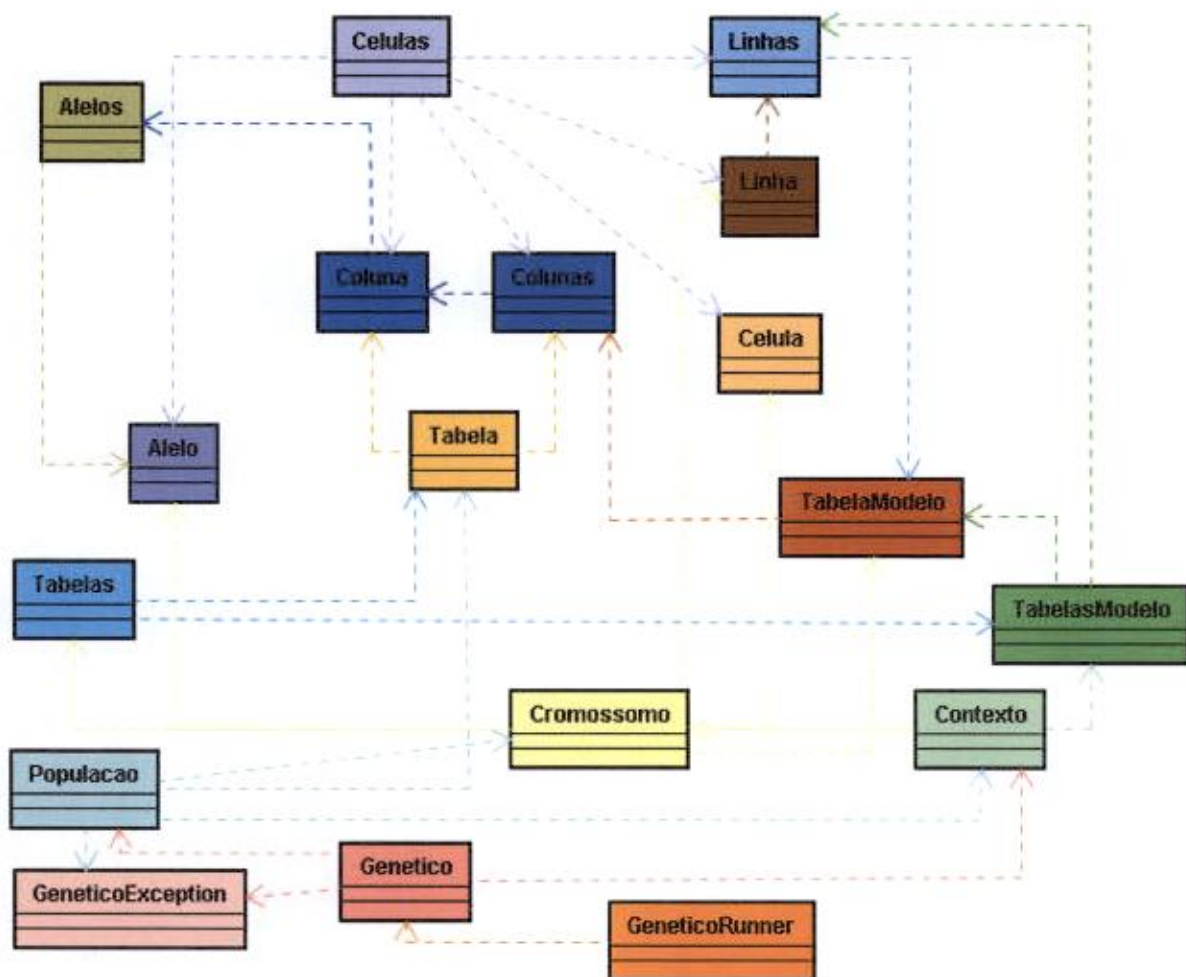


FIGURA 41 – RELACIONAMENTO DE DEPENDÊNCIA

FONTE: O autor (2007)

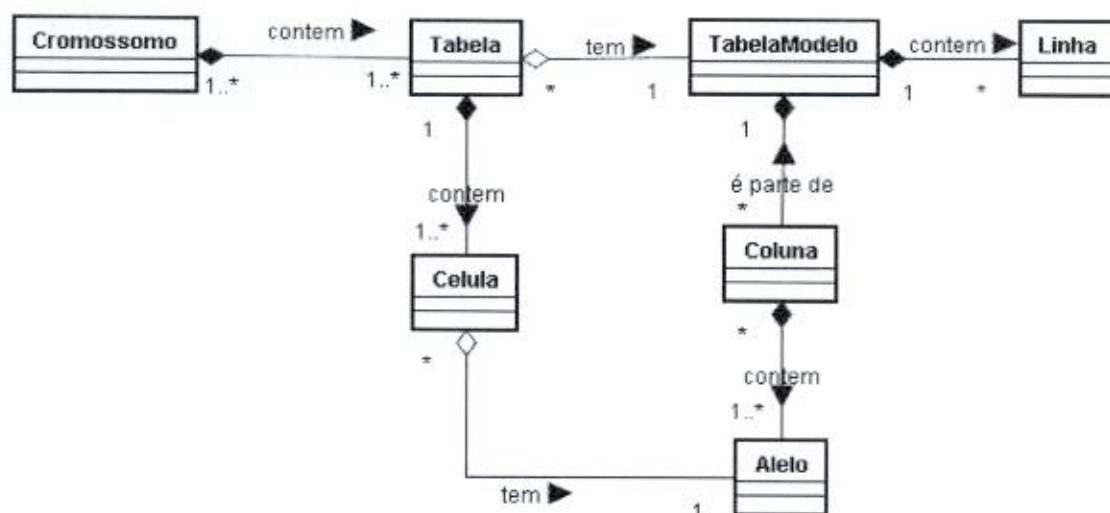


FIGURA 42 – RELACIONAMENTO DE ASSOCIAÇÃO

FONTE: O autor (2007)

7.7.3.3 Diagrama de classes do pacote modelo

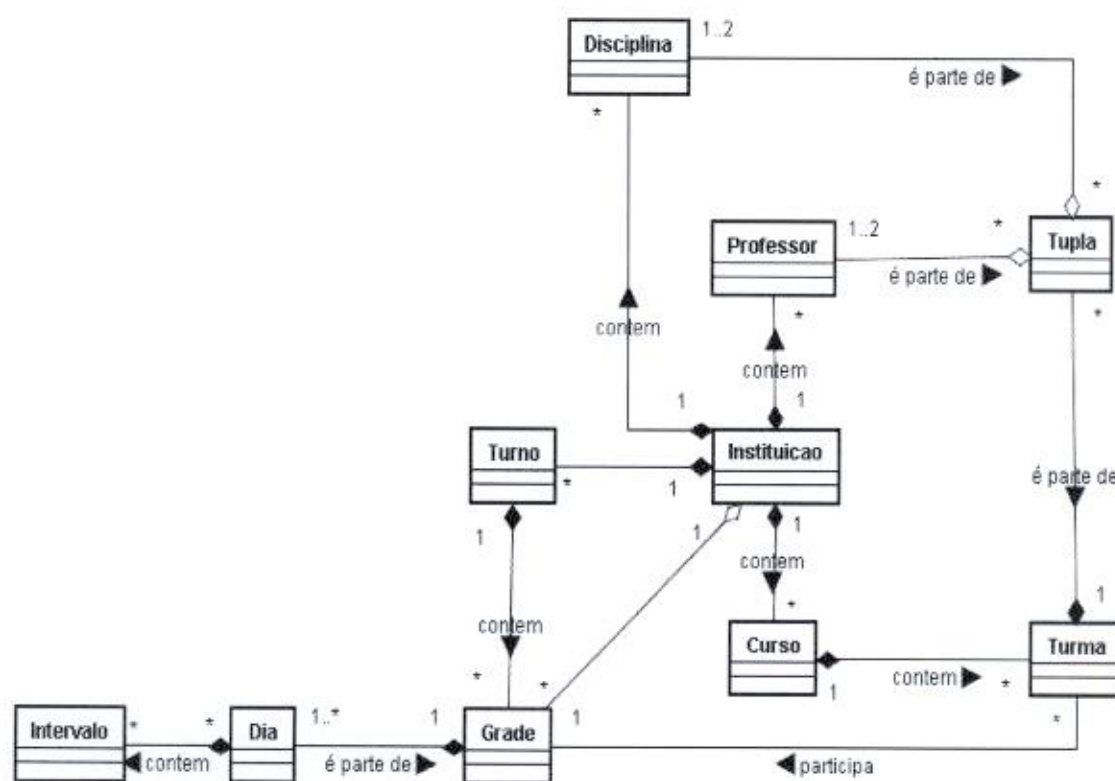


FIGURA 43 – RELACIONAMENTO ASSOCIAÇÃO

FONTE: O autor (2007)

7.7.3.5 Diagrama de classes do pacote resultados

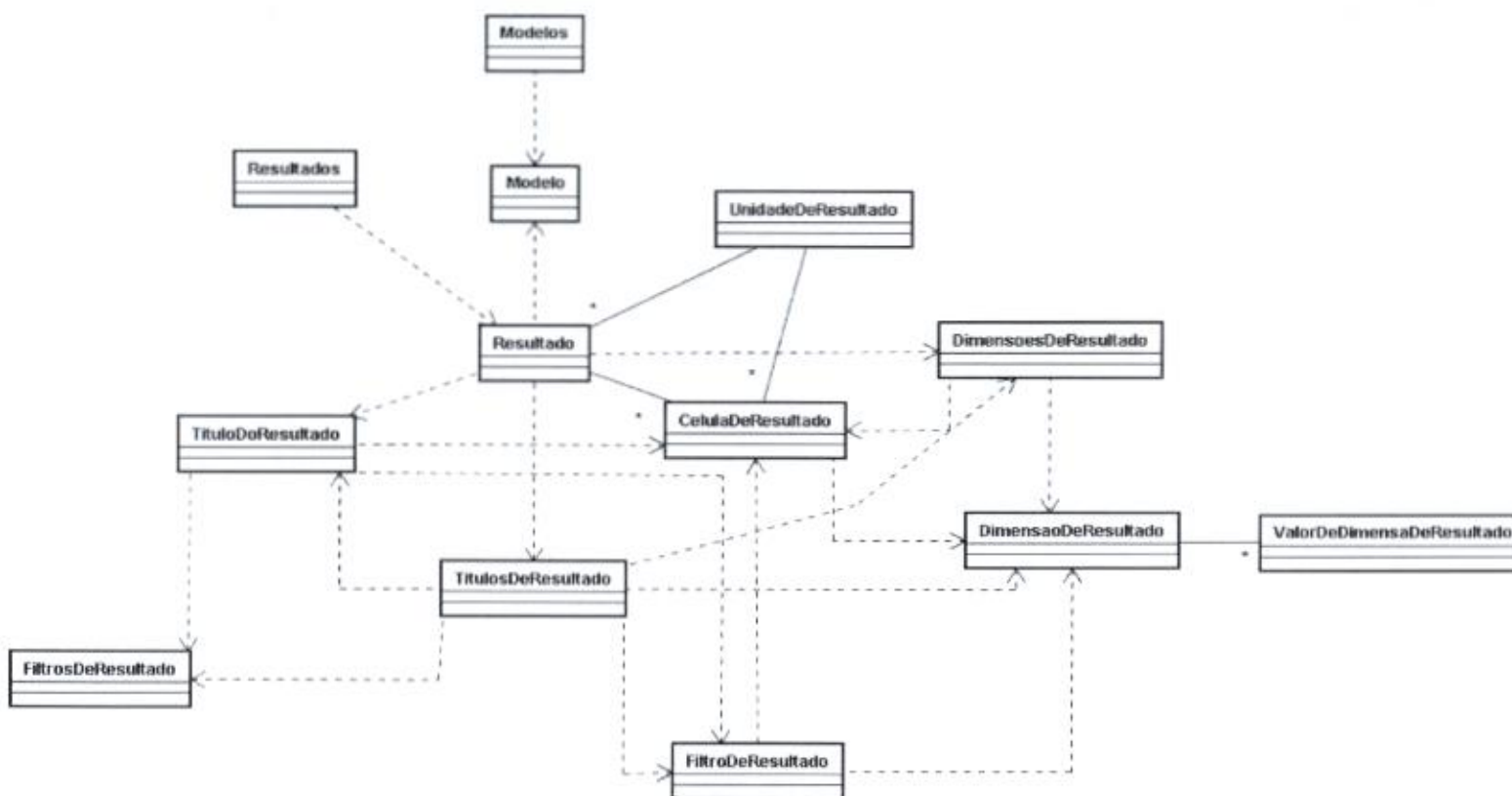


FIGURA 45 – RELACIONAMENTO DE DEPENDÊNCIA/ASSOCIAÇÃO

FONTE: O autor (2007)

7.7.3.6 Diagrama de classes do pacote util

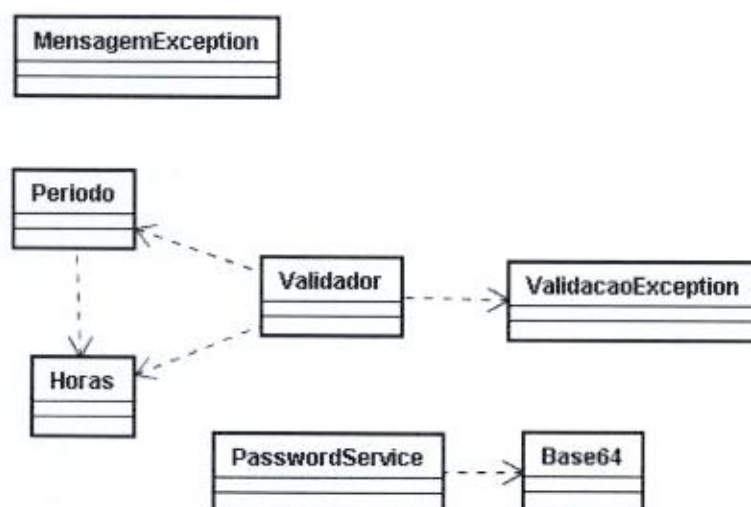


FIGURA 46 – RELACIONAMENTO DE DEPENDÊNCIA

FONTE: O autor (2007)

7.7.3.7 Diagrama de classes do pacote principal

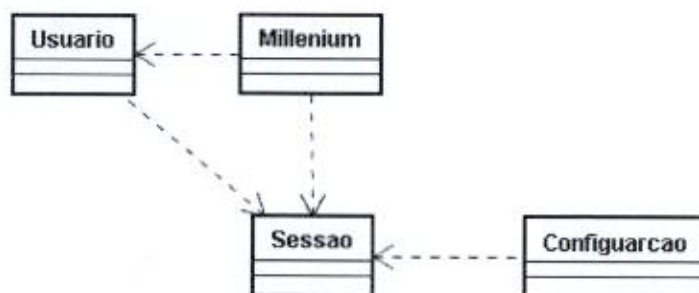


FIGURA 47 – RELACIONAMENTO DE DEPENDÊNCIA

FONTE: O autor (2007)

7.7.4 Diagrama de seqüência

De acordo com Booch, Rumbauch e Jacobson (2000, p. 100) "diagrama de seqüência é um diagrama de interação que dá ênfase à ordenação temporal de mensagens".

Um diagrama de seqüência pode ser modelado em diferentes níveis de abstração, podendo-se ocultar ou exibir seus detalhes, a decisão de qual nível de abstração deve-se utilizar estará relacionada às necessidades dos leitores do Estágio. Esta seção contém todos os diagramas de seqüências utilizados na modelagem do sistema.

7.7.4.1 Diagrama de seqüência efetuar login

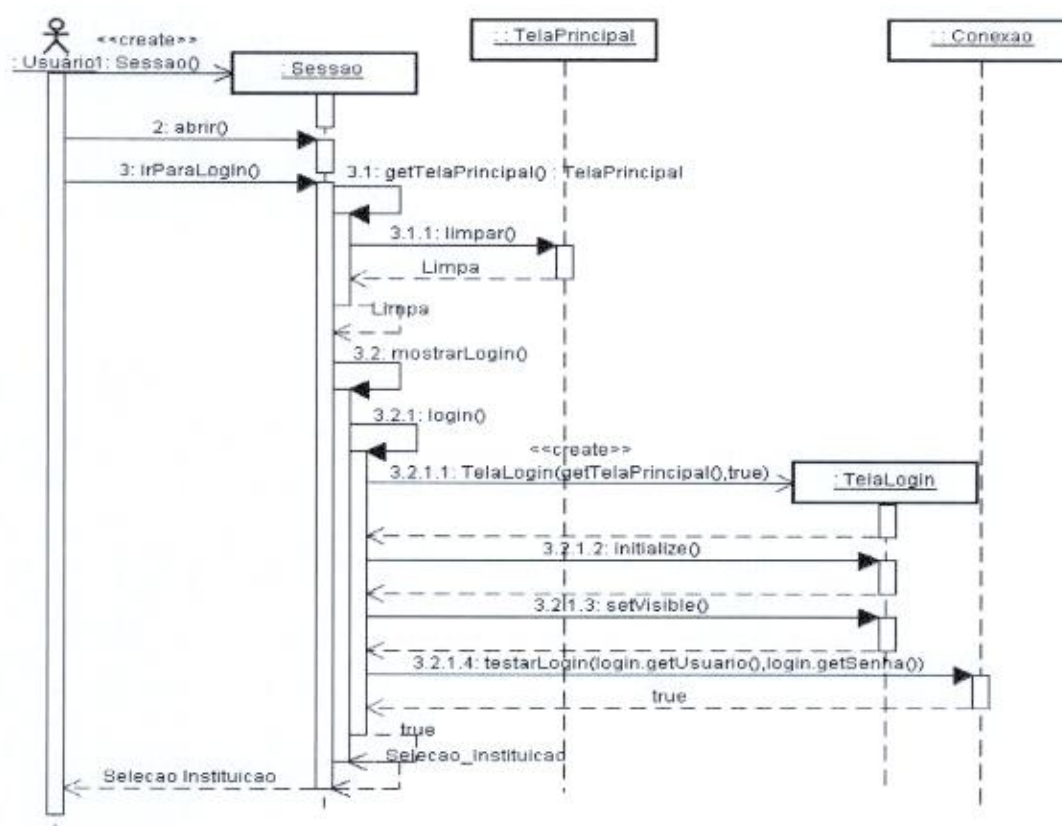


FIGURA 48 – DIAGRAMA DE SEQUENCIA EFETUAR LOGIN

FONTE: O autor (2007)

7.7.4.2 Diagrama de seqüência selecionar instituição

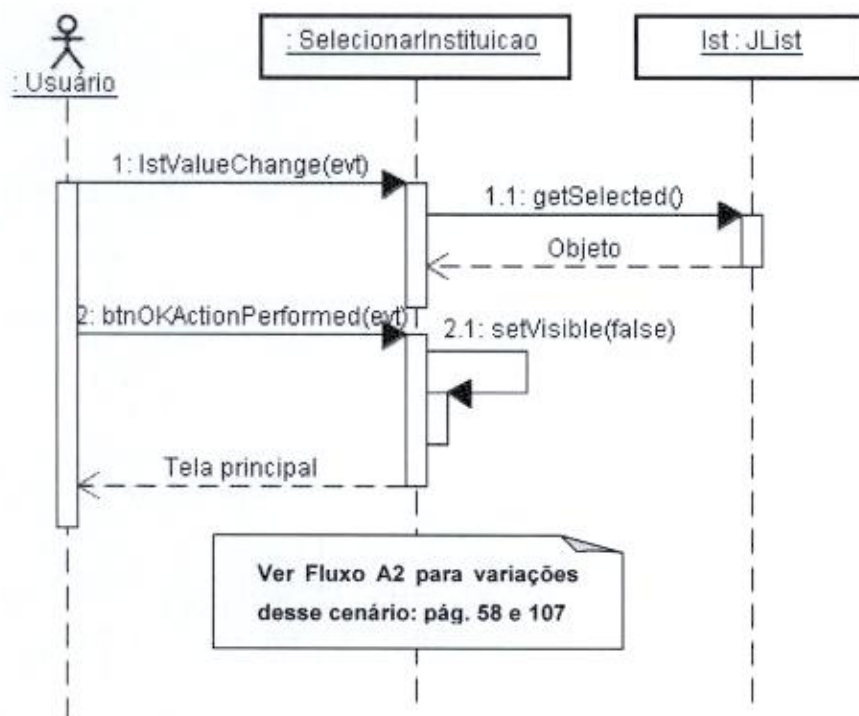


FIGURA 49 – DIAGRAMA DE SEQUENCIA SELECIONAR INSTITUIÇÃO

FONTE: O autor (2007)

7.7.4.3 Diagrama de seqüência manter dados

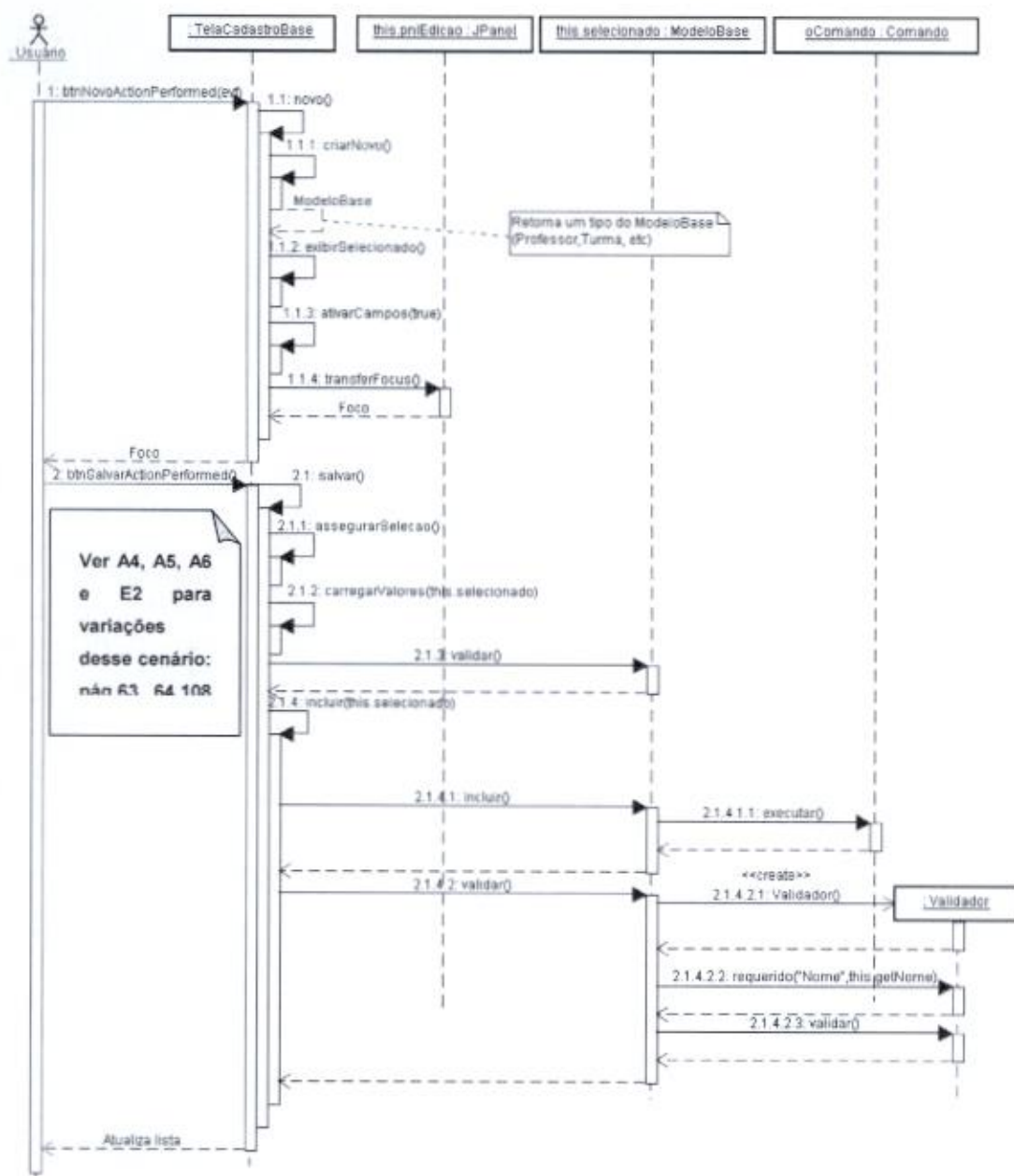


FIGURA 50 – DIAGRAMA DE SEQUENCIA MANTER DADOS

FONTE: O autor (2007)

7.7.4.4 Diagrama de seqüência manter instituição

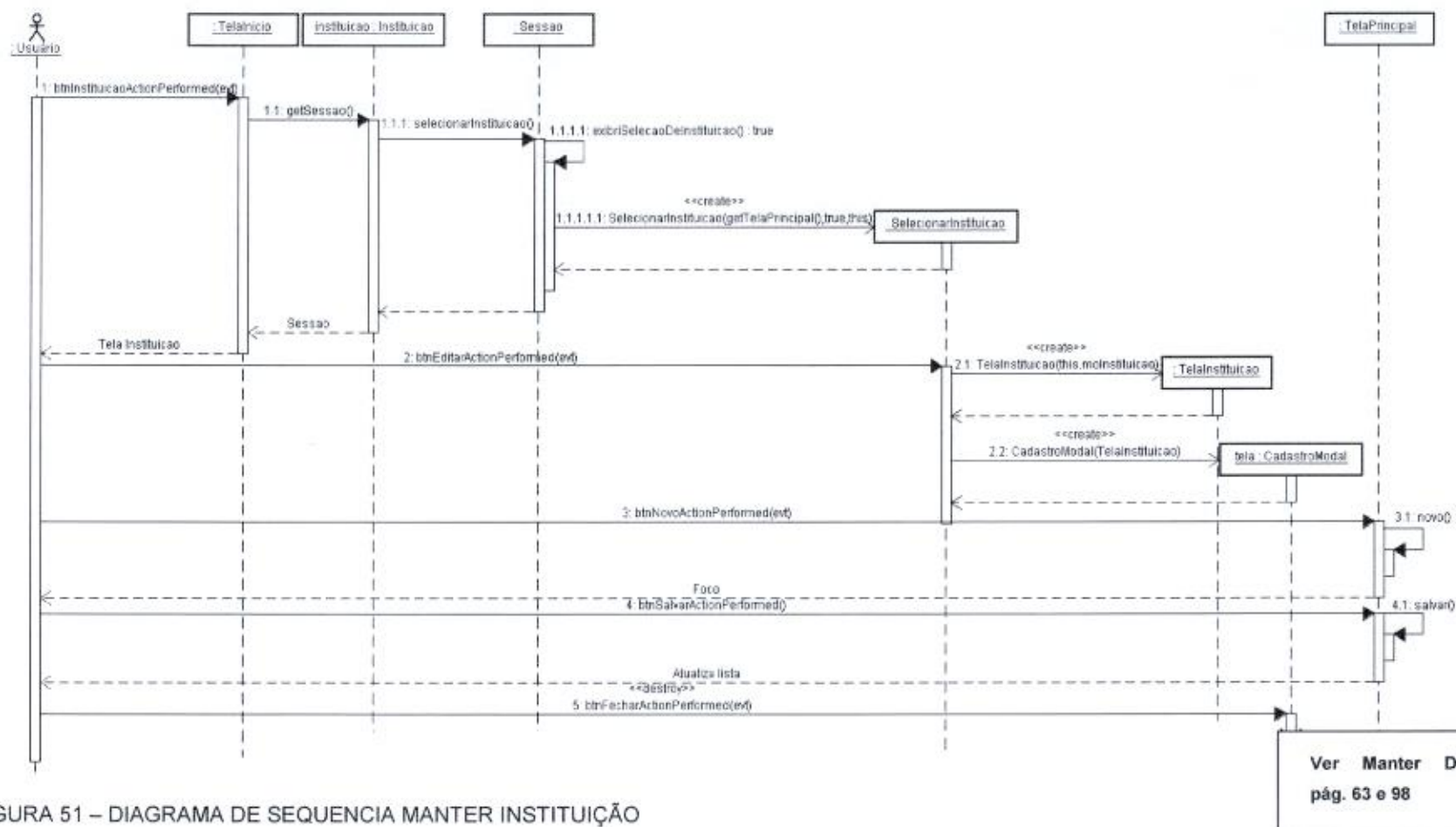


FIGURA 51 – DIAGRAMA DE SEQUENCIA MANTER INSTITUIÇÃO

FONTE: O autor (2007)

7.7.4.5 Diagrama de seqüência definir senha

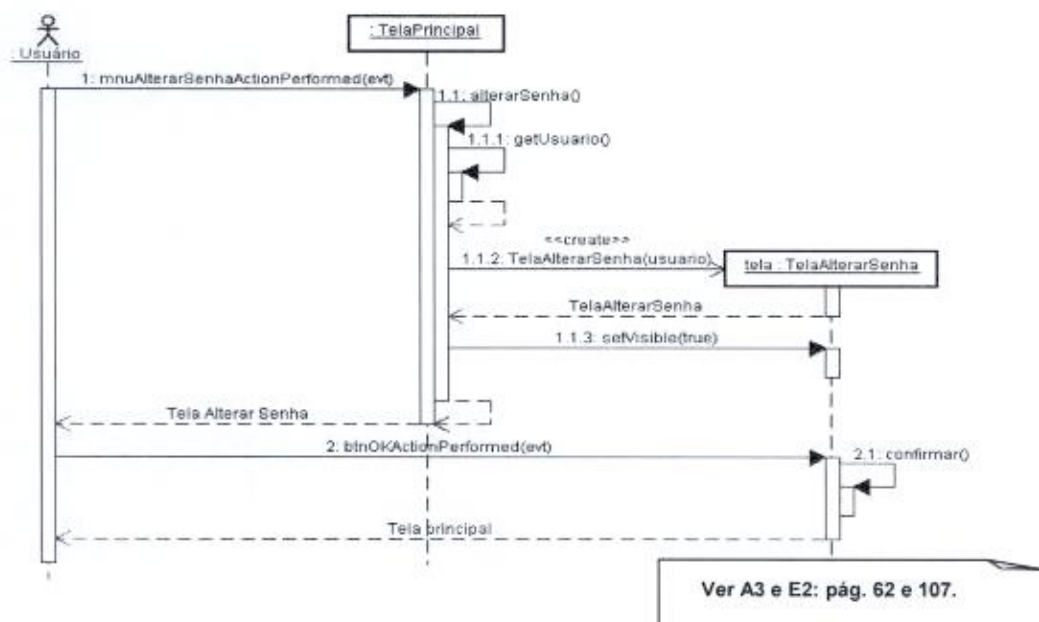


FIGURA 52 – DIAGRAMA DE SEQUENCIA DEFINIR SENHA

FONTE: O autor (2007)

7.7.4.6 Diagrama de seqüência manter usuário

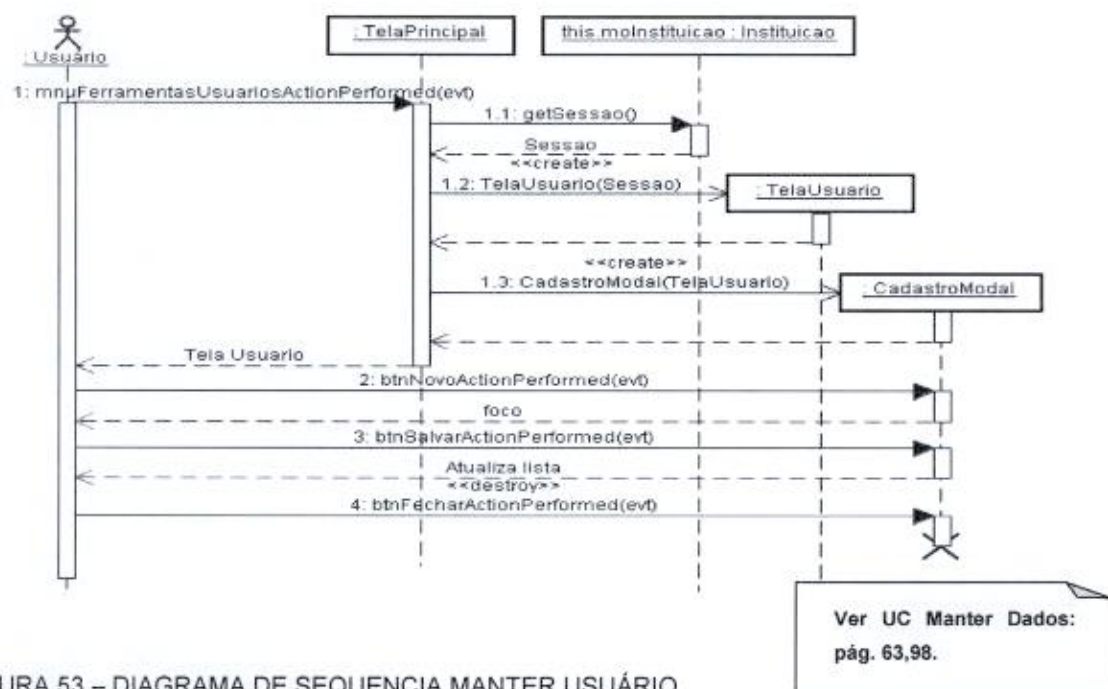


FIGURA 53 – DIAGRAMA DE SEQUENCIA MANTER USUÁRIO

FONTE: O autor (2007)

7.7.4.7 Diagrama de seqüência manter curso

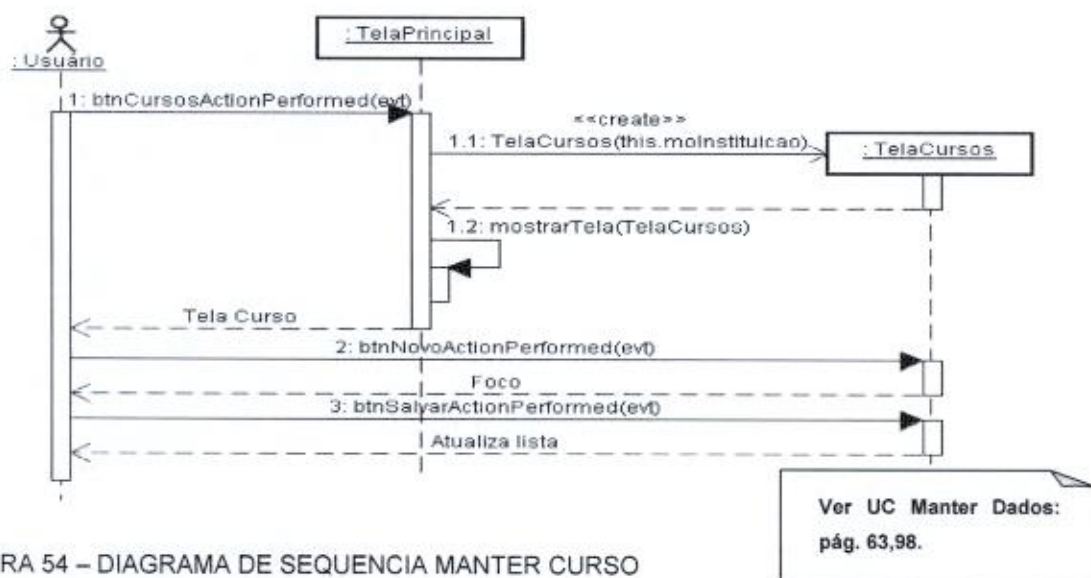


FIGURA 54 – DIAGRAMA DE SEQUENCIA MANTER CURSO

FONTE: O autor (2007)

7.7.4.8 Diagrama de seqüência manter turno

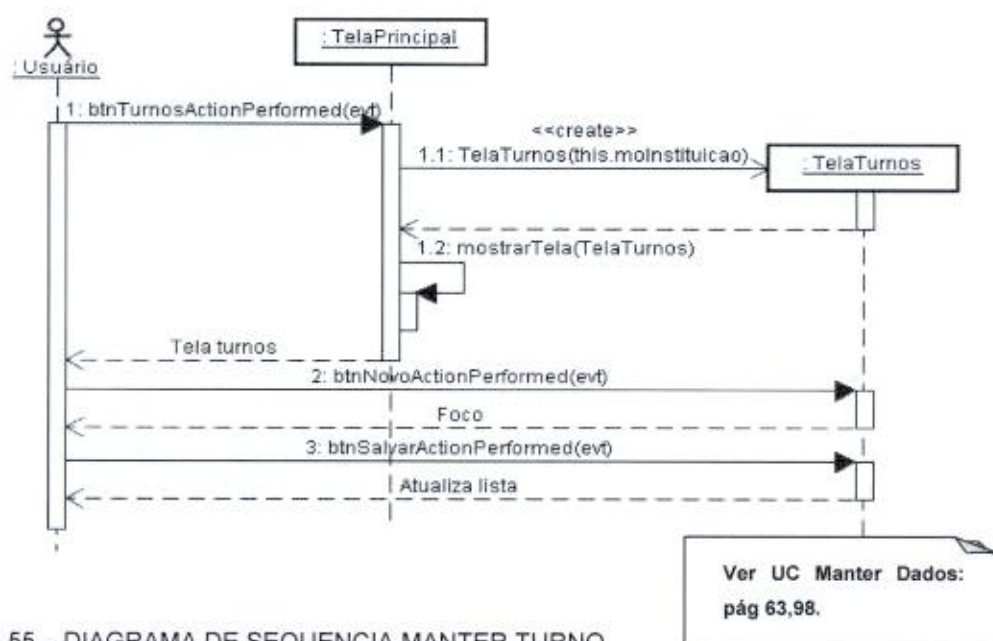


FIGURA 55 – DIAGRAMA DE SEQUENCIA MANTER TURNO

FONTE: O autor (2007)

7.7.4.9 Diagrama de seqüência manter professor

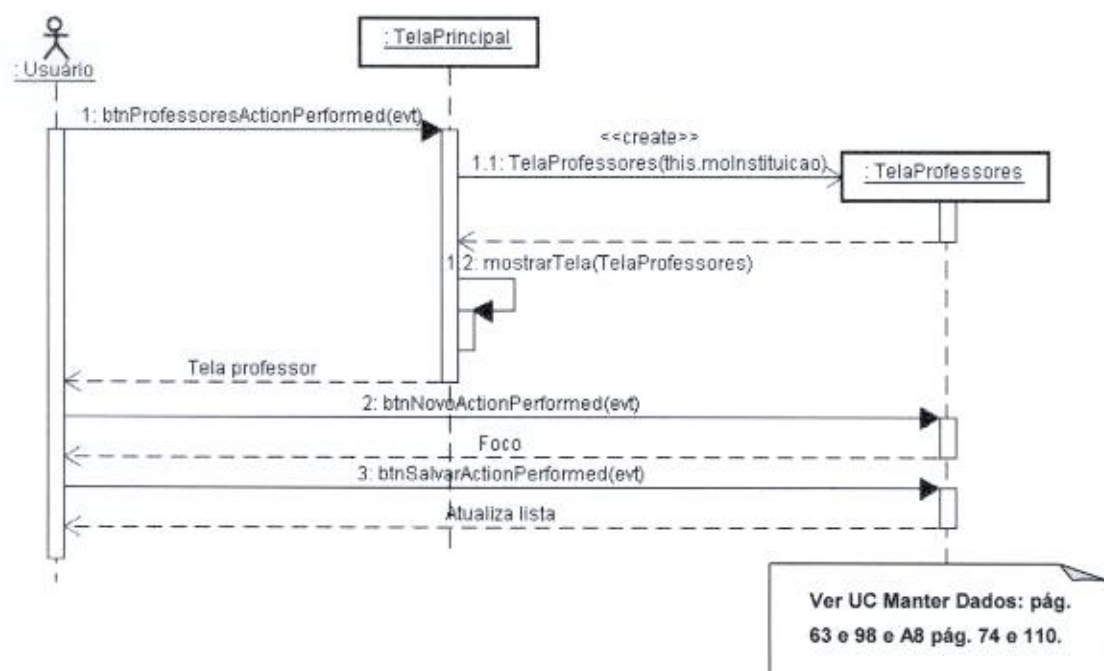


FIGURA 56 – DIAGRAMA DE SEQUENCIA MANTER PROFESSOR

FONTE: O autor (2007)

7.7.4.10 Diagrama de seqüência manter disciplina

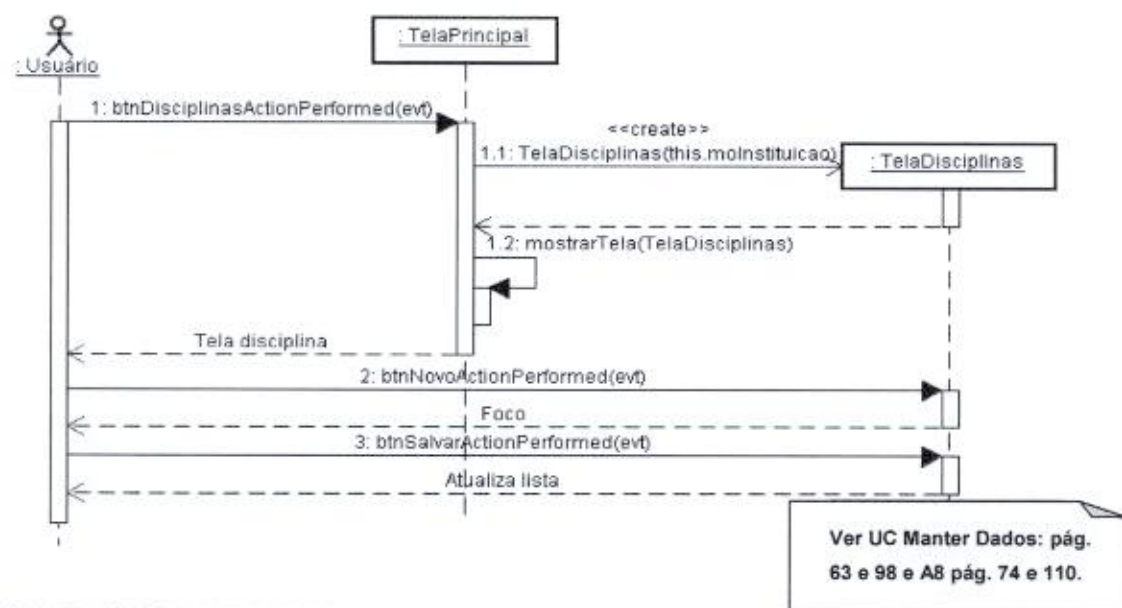


FIGURA 57 – DIAGRAMA DE SEQUENCIA MANTER DISCIPLINAS

FONTE: O autor (2007)

7.7.4.11 Diagrama de seqüência manter turma

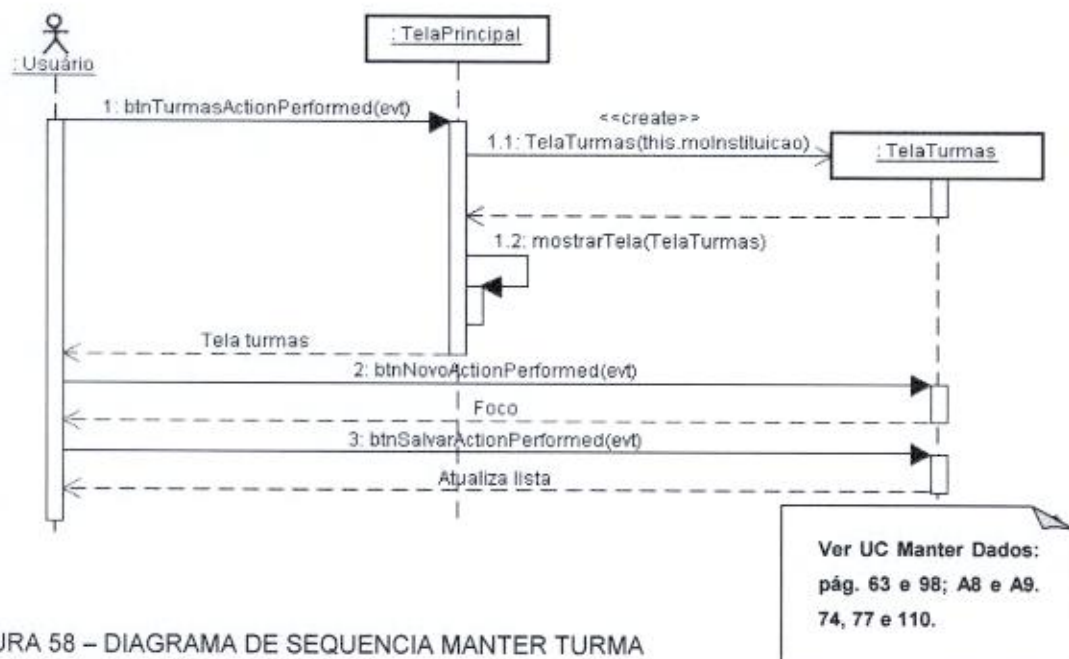


FIGURA 58 – DIAGRAMA DE SEQUENCIA MANTER TURMA

FONTE: O autor (2007)

7.7.4.12 Diagrama de seqüência manter grade

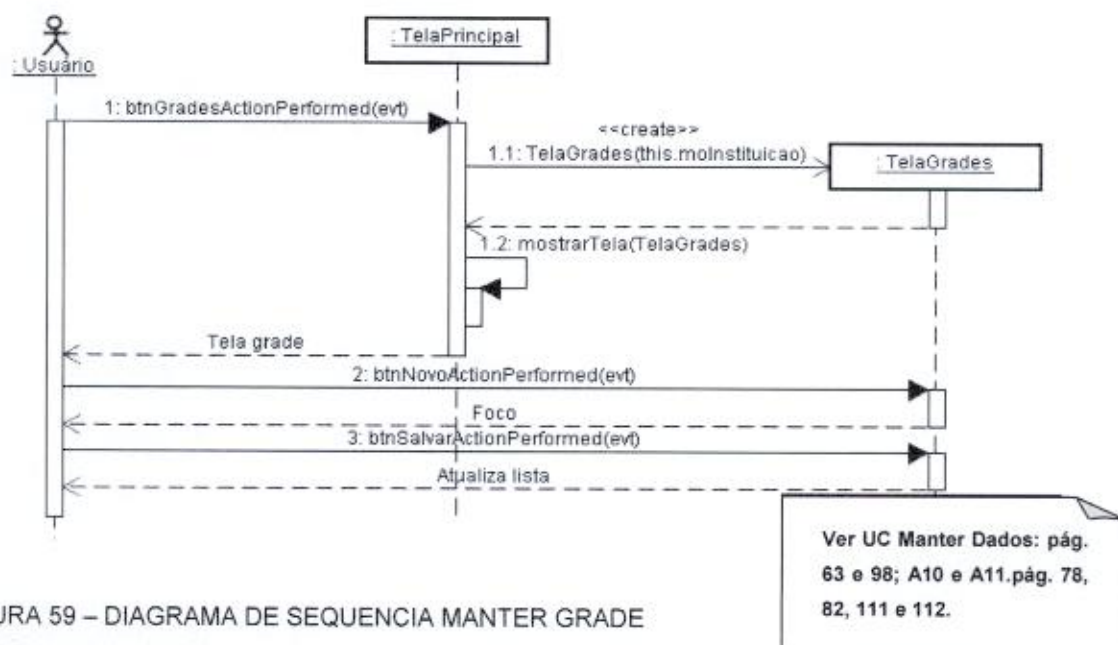


FIGURA 59 – DIAGRAMA DE SEQUENCIA MANTER GRADE

FONTE: O autor (2007)

7.7.4.13 Diagrama de seqüência criar genético

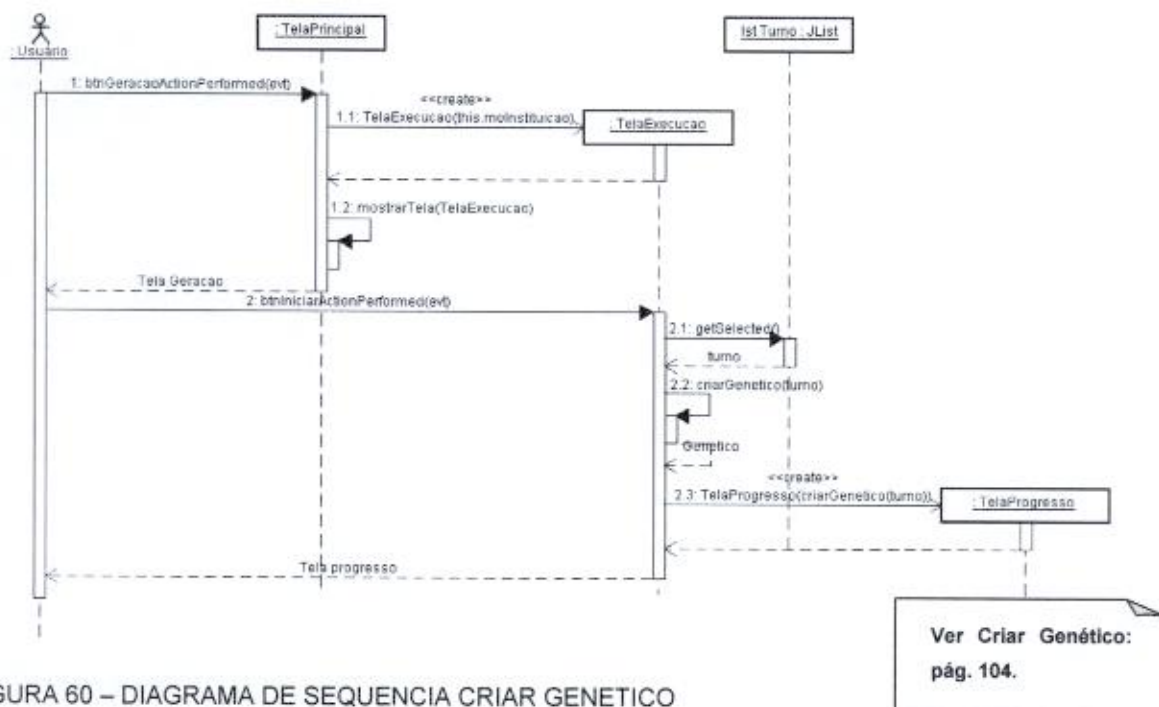


FIGURA 60 – DIAGRAMA DE SEQUENCIA CRIAR GENETICO

FONTE: O autor (2007)

7.7.4.14 Diagrama de seqüência criar genético

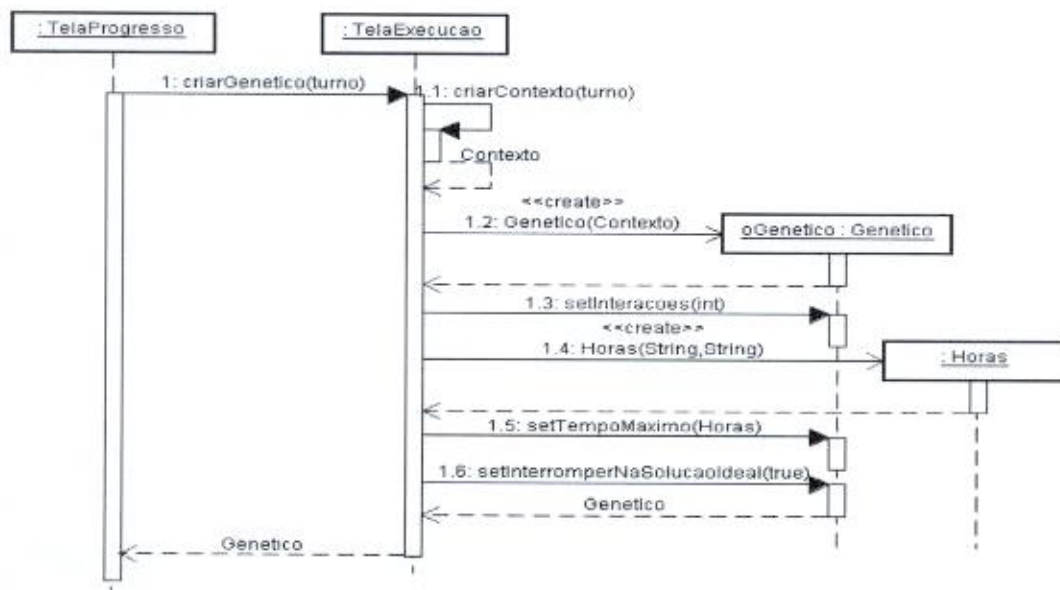


FIGURA 61 – DIAGRAMA DE SEQUENCIA CRIAR GENÉTICO

FONTE: O autor (2007)

7.7.4.15 Diagrama de seqüência visualizar grade

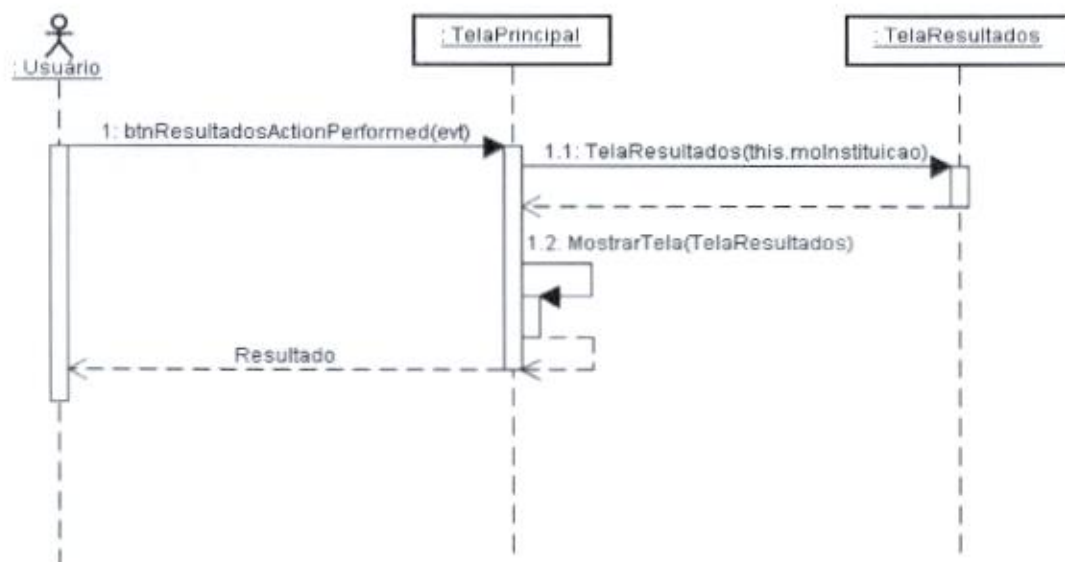


FIGURA 62 – VISUALIZAR GRADE

FONTE: O autor (2007)

7.7.4.16 Diagrama de seqüência desiste de login (A1)

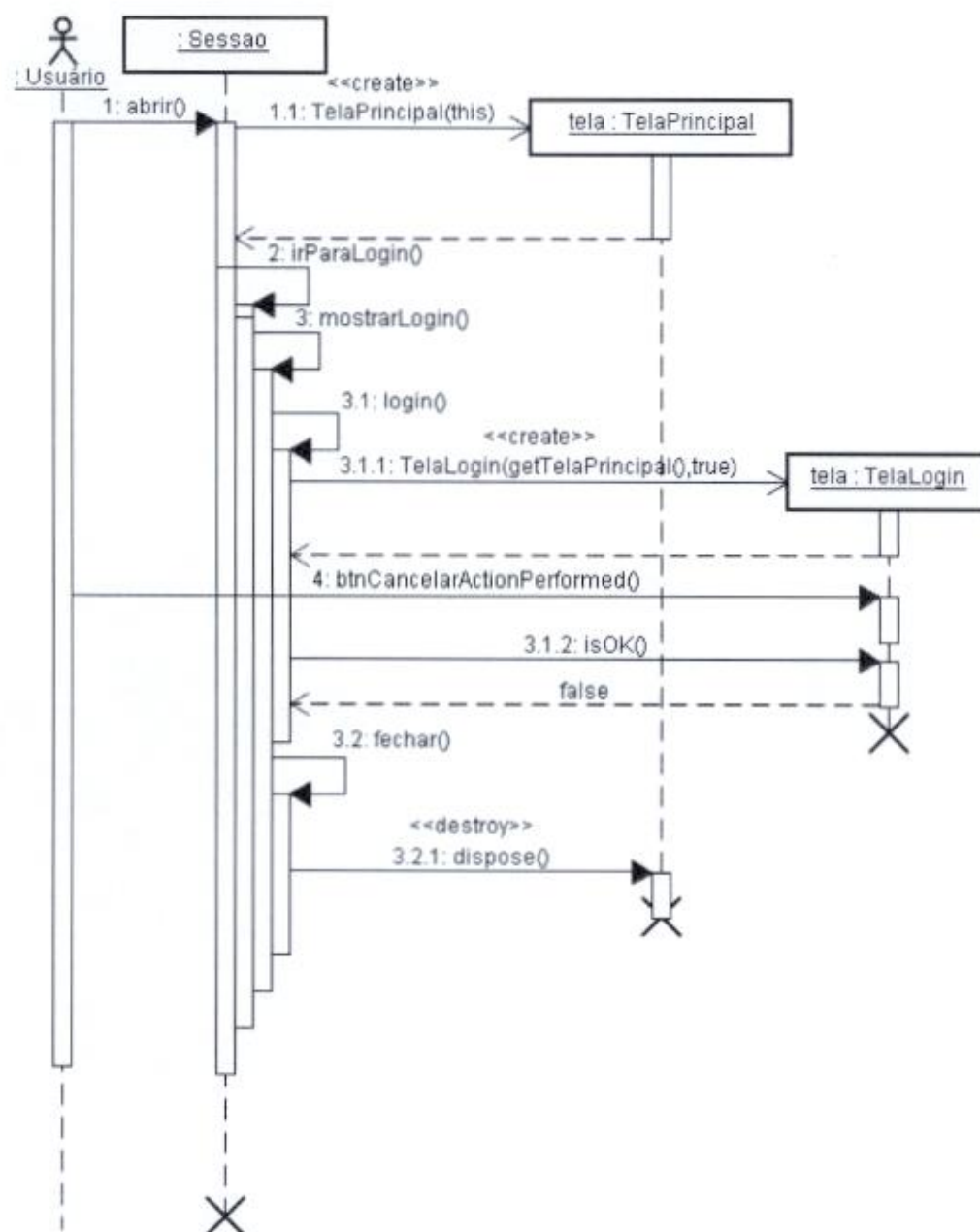


FIGURA 63 – DIAGRAMA DE SEQUENCIA DESISTE DE LOGIN (A1)

FONTE: O autor (2007)

7.7.4.17 Diagrama de seqüência cancelar seleção de instituição (A2)

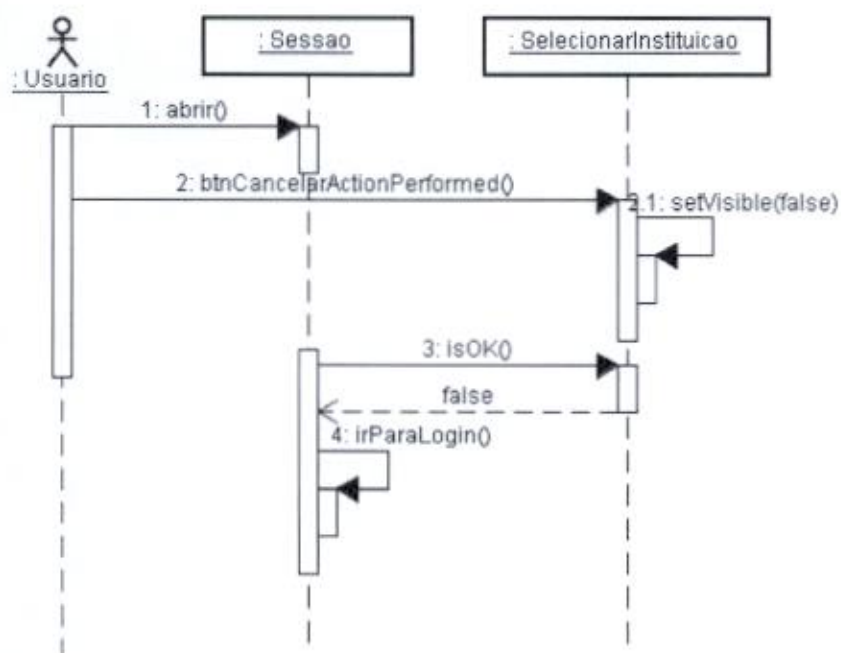


FIGURA 64 – DIAGRAMA DE SEQUENCIA CANCELAR SELEÇÃO DE INSTITUIÇÃO (A2)
 FONTE: O autor (2007)

7.7.4.18 Diagrama de seqüência cancela alterar senha (A3)

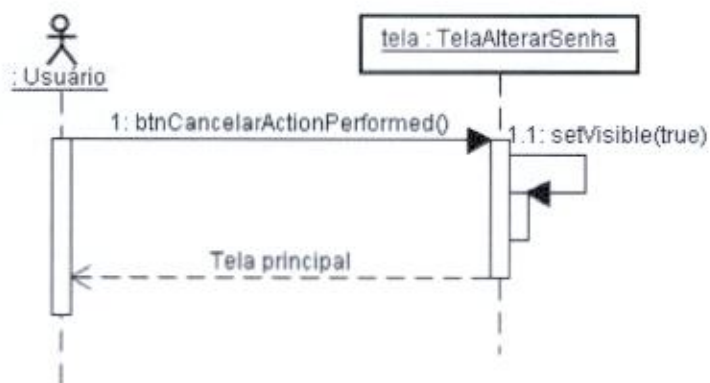


FIGURA 65 – DIAGRAMA DE SEQUENCIA CANCELAR ALTERAR SENHA (A3)
 FONTE: O autor (2007)

7.7.4.19 Diagrama de seqüência editar (A4)

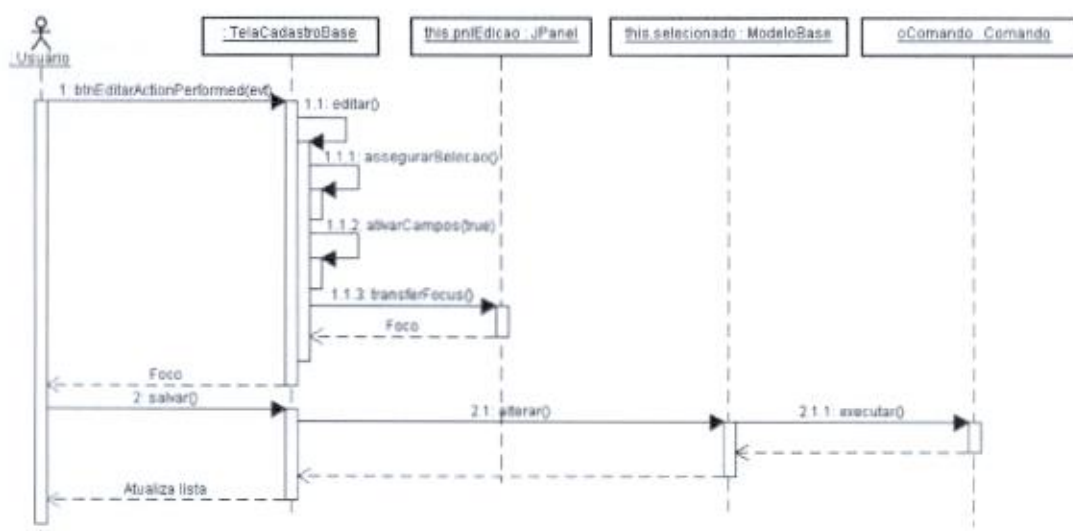


FIGURA 66 – DIAGRAMA DE SEQUENCIA EDITAR (A4)

FONTE: O autor (2007)

7.7.4.20 Diagrama de seqüência cancelar (A5)

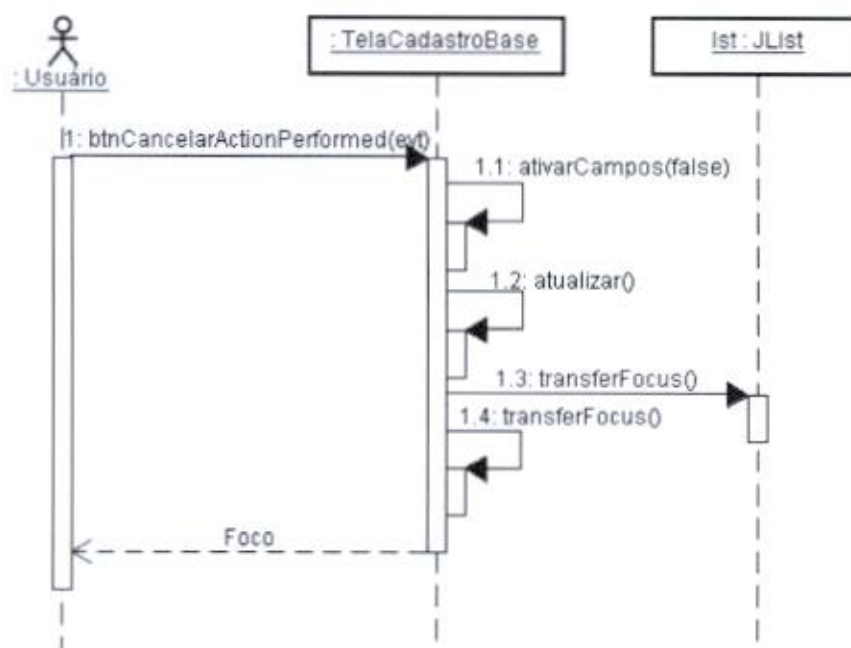


FIGURA 67 – DIAGRAMA DE SEQUENCIA CANCELAR (A5)

FONTE: O autor (2007)

7.7.4.21 Diagrama de seqüência excluir (A6)

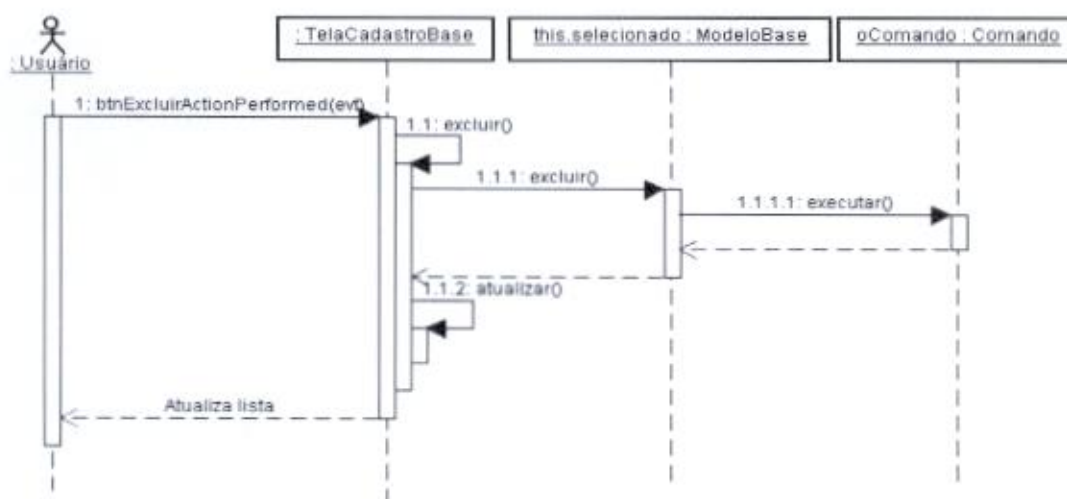


FIGURA 68 – DIAGRAMA DE SEQUENCIA EXCLUIR (A6)

FONTE: O autor (2007)

7.7.4.22 Diagrama de seqüência editar restrições (A7)

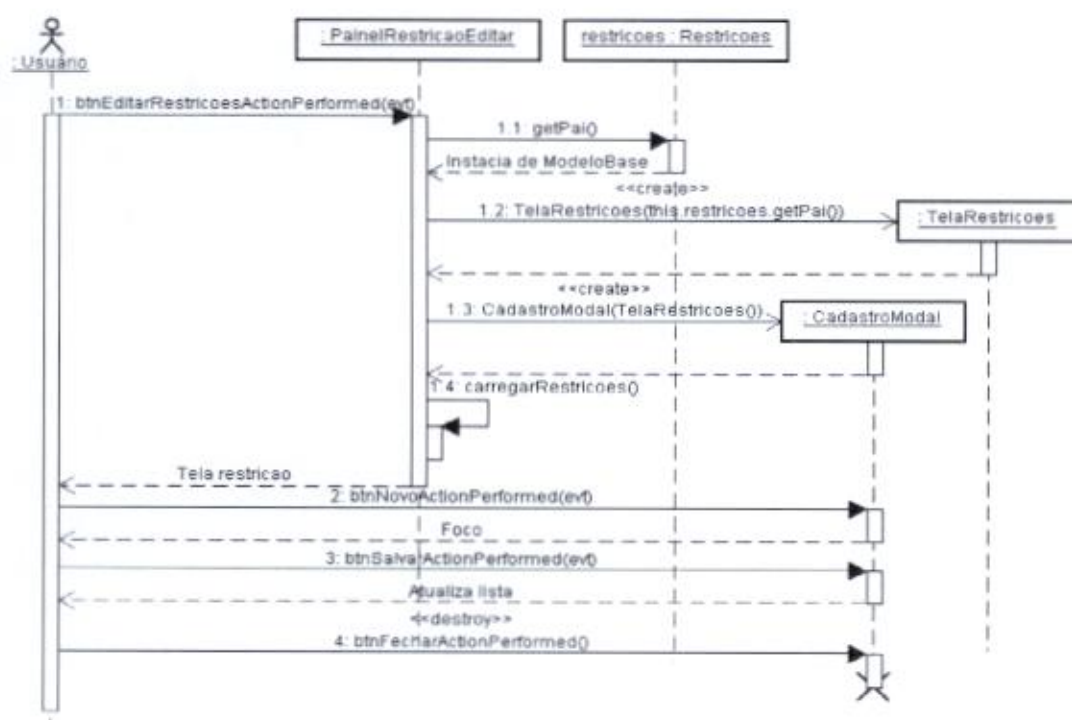


FIGURA 69 – DIAGRAMA DE SEQUENCIA EDITAR RESTRICOES (A7)

FONTE: O autor (2007)

7.7.4.23 Diagrama de seqüência editar tuplas (A8)

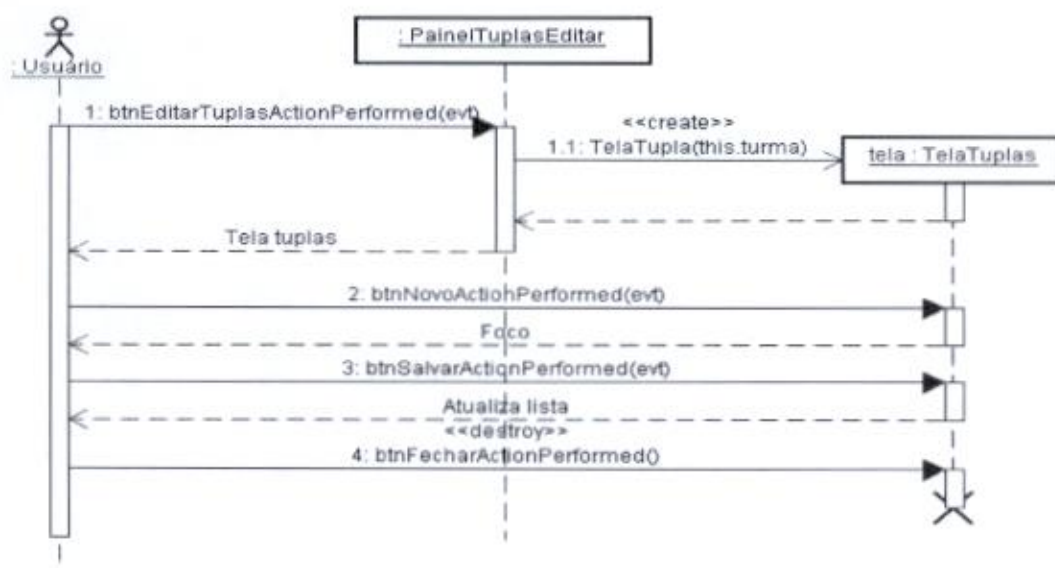


FIGURA 70 – DIAGRAMA DE SEQUENCIA EDITAR TUPLAS (A8)

FONTE: O autor (2007)

7.7.4.24 Diagrama de seqüência editar dias (A9)

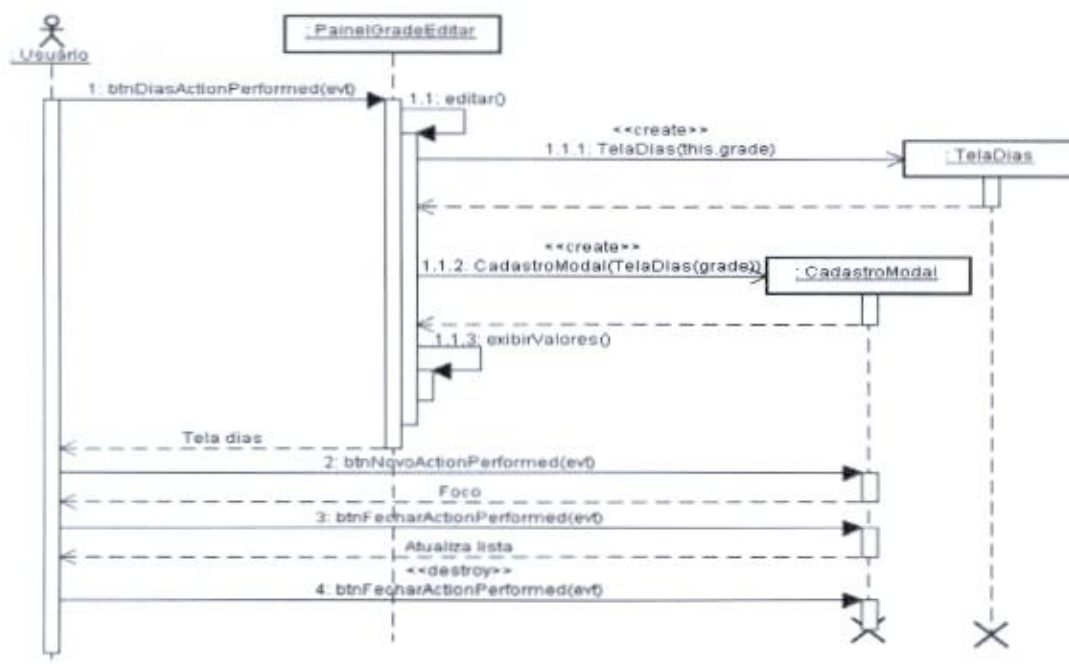


FIGURA 71 – DIAGRAMA DE SEQUENCIA EDITAR DIAS

FONTE: O autor (2007)

7.7.4.25 Diagrama de seqüência editar horários (A10)

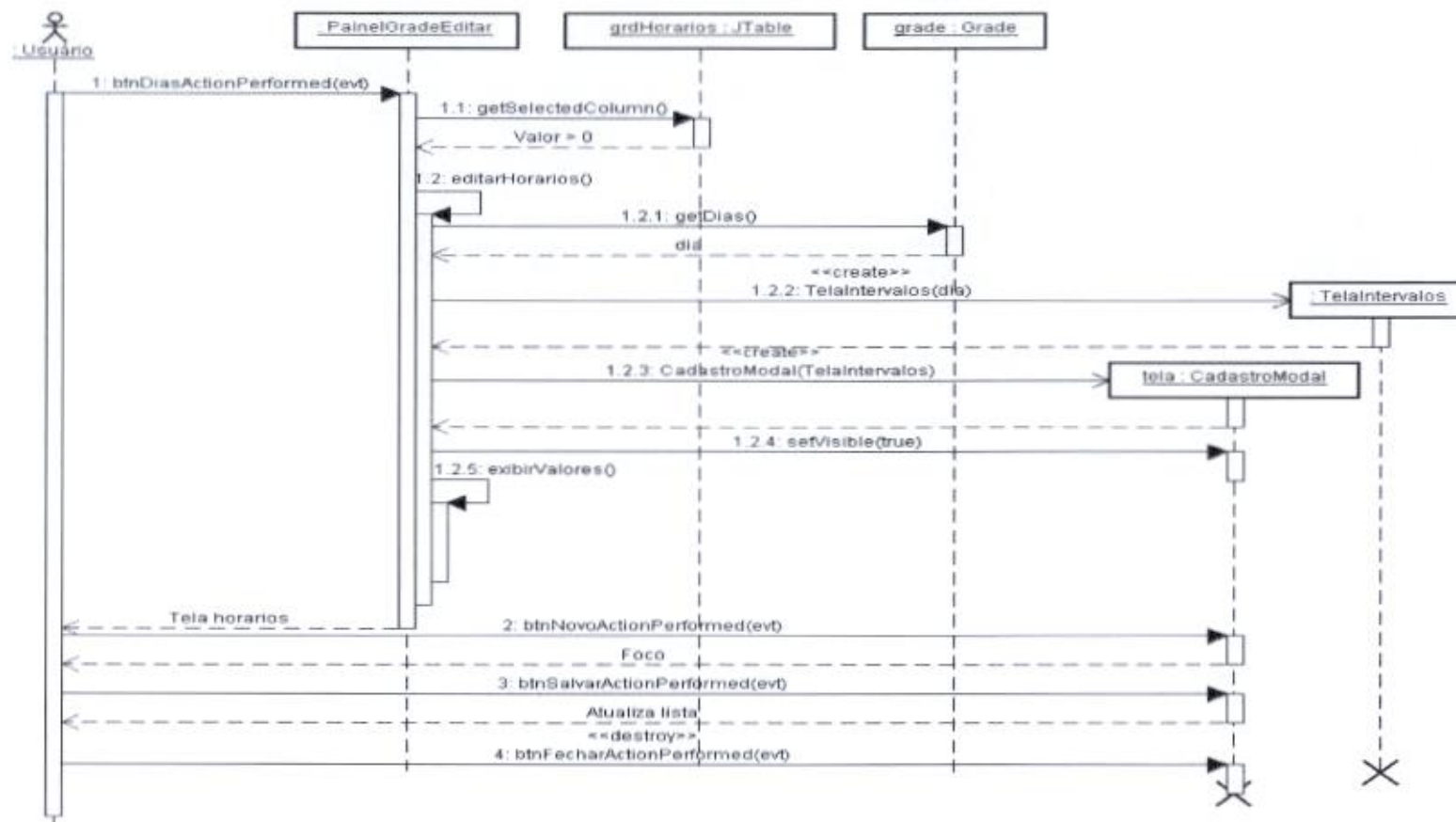


FIGURA 72 – DIAGRAMA DE SEQUENCIA EDITAR HORARIOS (A10)

FONTE: O autor (2007)

7.7.4.26 Diagrama de seqüência interromper genético (A11)

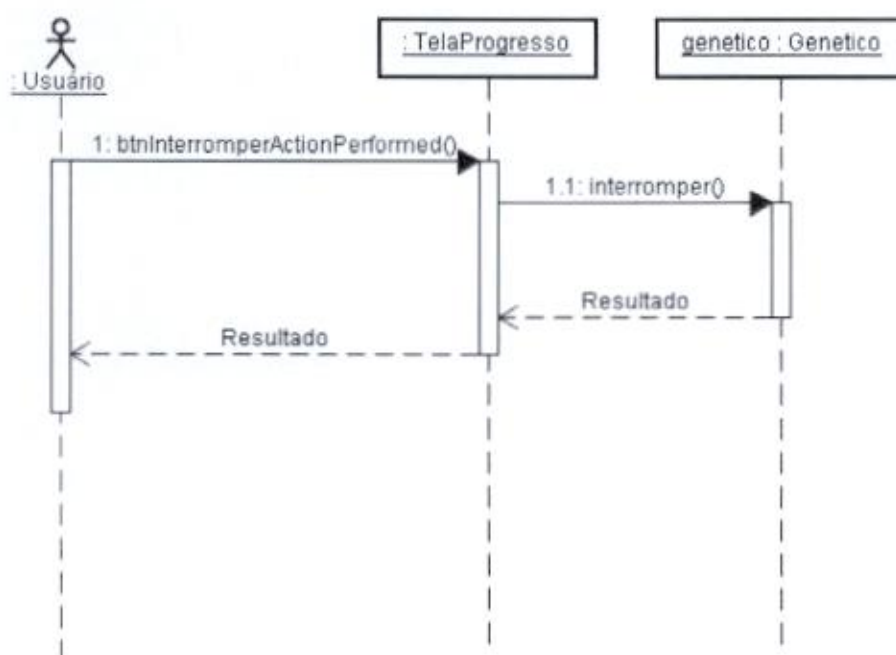


FIGURA 73 – DIAGRAMA DE SEQUENCIA INTERROMPER GENÉTICO (A11)

FONTE: O autor (2007)

7.7.5 Diagramas de atividade

Diagramas de atividades são utilizados para modelar os aspectos dinâmicos de sistemas, esses diagramas é essencialmente um gráfico de fluxo, mostrando o fluxo de controle de uma atividade para outra.

Segundo Booch, Rumbauch e Jacobson (2000, p. 255) eles poderão permanecer isolados para visualizar a dinâmica de uma sociedade de objetos, ou modelar o fluxo de controle de uma operação.

7.7.5.1 Diagramas de atividade do millenium

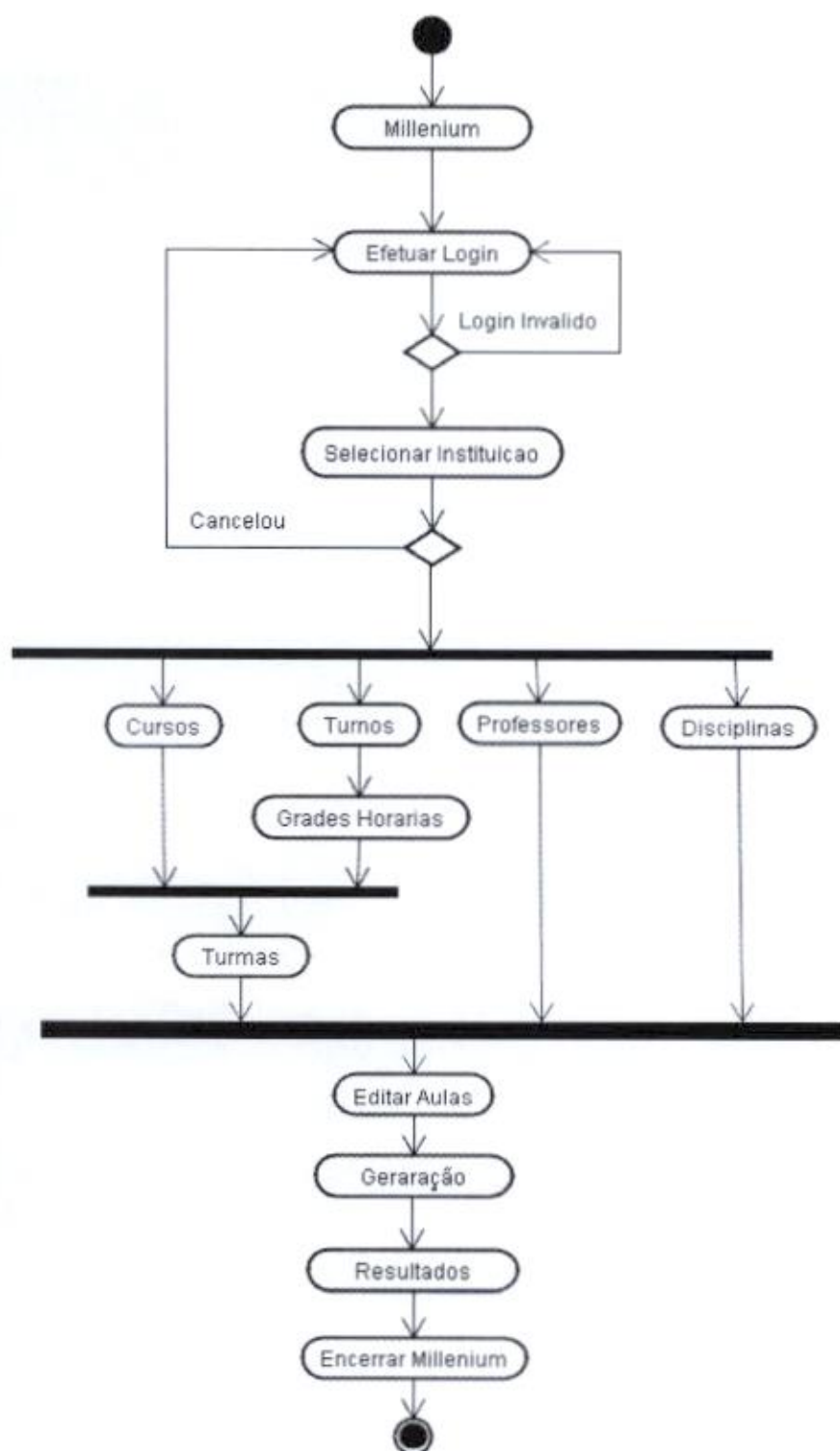


FIGURA 74 – DIAGRAMA DE ATIVIDADES DO MILLENIUM

FONTE: O autor (2007)

7.7.5.2 Diagrama de atividades salvar/alterar

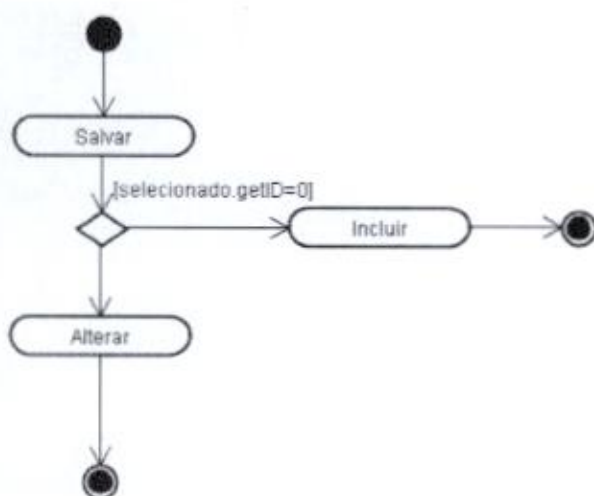


FIGURA 75 – DIAGRAMA DE ATIVIDADES SALVAR/ALTERAR

FONTE: O autor (2007)

7.7.6 Diagrama de estado do millenium

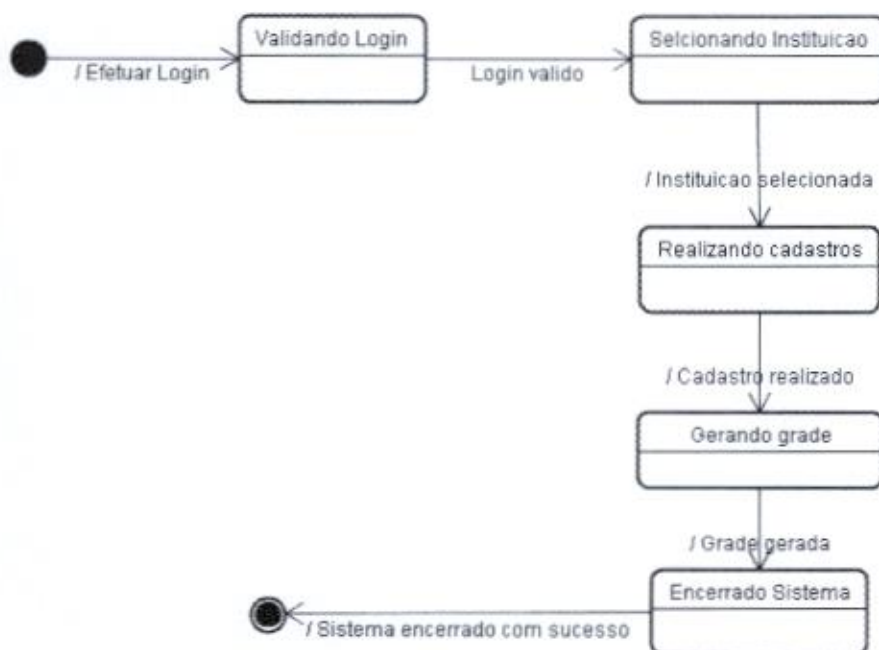


FIGURA 76 – DIAGRAMA DE ESTADO DO MILLENIUM

FONTE: O autor (2007)

7.8 MODELO ENTIDADE RELACIONAMENTO

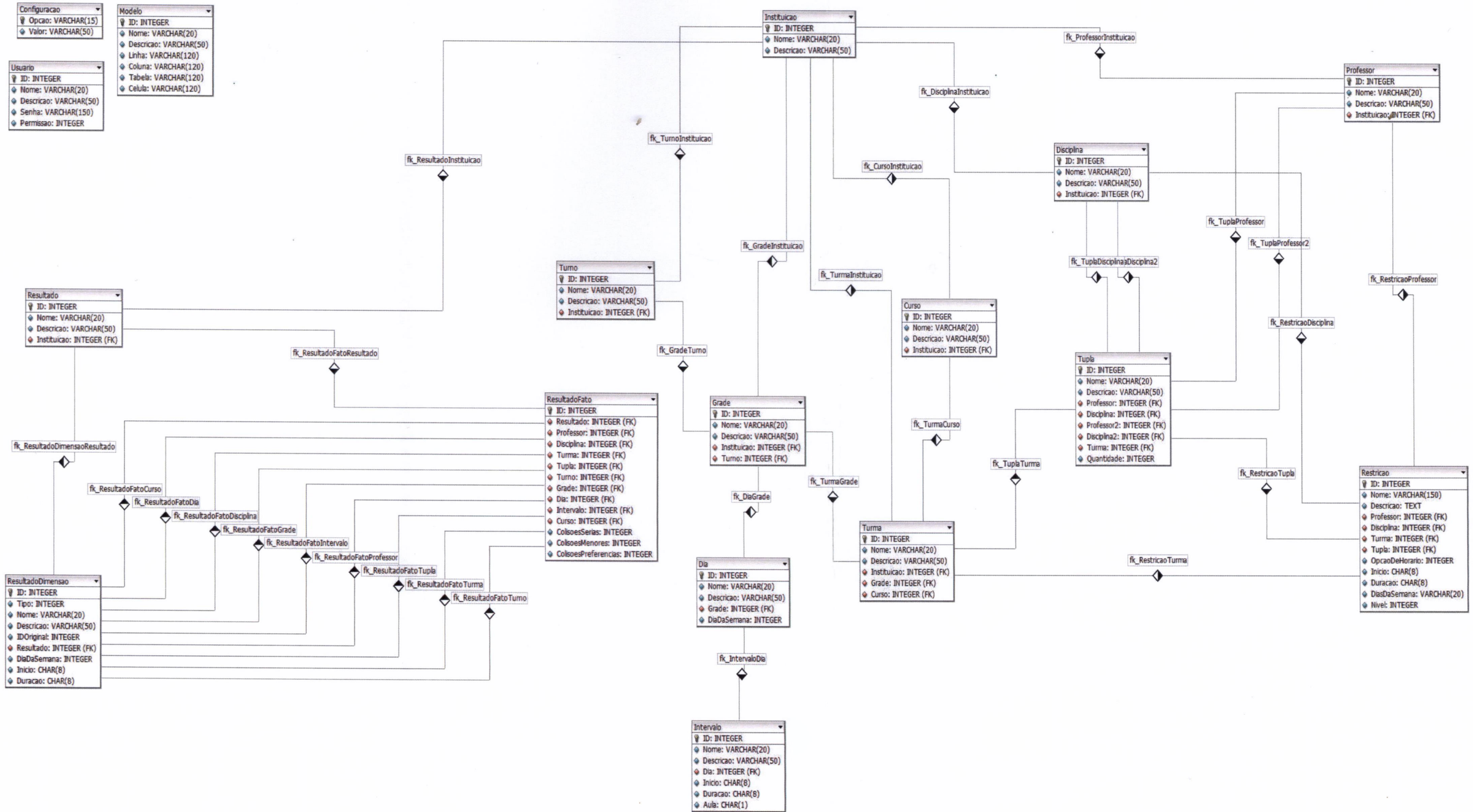


FIGURA 77 – MODELO ENTIDADE RELACIONAMENTO

FONTE: O autor (2007)

7.9. DICIONÁRIO DE DADOS

7.9.1 Descrição dos campos das tabelas

A coluna descrição refere-se ao significado e/ou ao conteúdo de cada campo a fim de facilitar a compreensão dos mesmos. Essa coluna não faz parte das tabelas de fato.

7.9.1.1 Tabela usuario

TABELA 3 – USUARIO

Usuário					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome do usuário
3	Descrição	VARCHAR	(50)	Nulo	Apelido
4	Senha	VARCHAR	(150)	Nulo	Senha do usuário
5	Permissão	INTEGER	-	Nulo	Nível de acesso (1,3 e 5)

FONTE: O autor (2007)

7.9.1.2 Tabela configuracao

TABELA 4 – CONFIGURACAO

Configuração					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	Opção	VARCHAR	(15)	PK	Campo a ser configurado
2	Valor	VARCHAR	(50)	Nulo	Valor do campo

FONTE: O autor (2007)

7.9.1.3 Tabela instituicao

TABELA 5 – INSTITUICAO

Instituição					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome Instituição
3	Descrição	VARCHAR	(50)	Nulo	Abreviação

FONTE: O autor (2007)

7.9.1.4 Tabela curso

TABELA 6 – CURSO

Curso					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome do curso
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Instituicao	INTEGER	-	FK	Instituição do curso

FONTE: O autor (2007)

7.9.1.5 Tabela turno

TABELA 7 – TURNO

Turno					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome do turno
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Instituicao	INTEGER	-	FK	Instituição do turno

FONTE: O autor (2007)

7.9.1.6 Tabela grade

TABELA 8 – GRADE

Grade					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome da grade
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Instituicao	INTEGER	-	FK	Instituição da grade
5	Turno	INTEGER	-	FK	Turno da grade

FONTE: O autor (2007)

7.9.1.7 Tabela dia

TABELA 9 – DIA

Dia					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome do dia
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Grade	INTEGER	-	FK	Grade que o dia pertence
5	DiaDaSemana	INTEGER	-	Não Nulo	Identificador numérico (1 à 7)

FONTE: O autor (2007)

7.9.1.8 Tabela intervalo

TABELA 10 – INTERVALO

Intervalo					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome do intervalo
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Dia	INTEGER	-	FK	Dia que o dia pertence
5	Início	CHAR	(8)	Não Nulo	Hora de início
6	Duração	CHAR	(8)	Não Nulo	Duração do intervalo
7	Aula	CHAR	(1)	Não Nulo	Flag para verificação (0 ou 1)

FONTE: O autor (2007)

7.9.1.9 Tabela disciplina

TABELA 11 – DISCIPLINA

Disciplina					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome da disciplina
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Instituicao	INTEGER	-	FK	Instituição da disciplina

FONTE: O autor (2007)

7.9.1.10 Tabela professor

TABELA 12 – PROFESSOR

Professor					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome do professor
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Instituicao	INTEGER	-	FK	Instituição do professor

FONTE: O autor (2007)

7.9.1.11 Tabela turma

TABELA 13 – TURMA continua

Turma					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único

TABELA 14 – TURMA conclusão

Turma					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
2	Nome	VARCHAR	(20)	Não nulo	Nome da Turma
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Instituicao	INTEGER	-	FK	Instituição da Turma
5	Grade	INTEGER	-	FK	Grade da Turma
6	Curso	INTEGER	-	FK	Curso da Turma

FONTE: O autor (2007)

7.9.1.12 Tabela tupla

TABELA 15 – TUPLA

Tupla					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Nulo	Nome da Tupla
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Professor	INTEGER	-	FK	Professor da tupla
5	Disciplina	INTEGER	-	FK	Disciplina da tupla
6	Professor2	INTEGER	-	FK	Professor da tupla
7	Disciplina2	INTEGER	-	FK	Disciplina da tupla
8	Turma	INTEGER	-	FK	Turma da tupla
9	Quantidade	INTEGER	-	Não nulo	Quantidade de aulas semanais

FONTE: O autor (2007)

7.9.1.13 Tabela restricao

TABELA 16 – RESTRICAO

Restricao					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(150)	Nulo	Nome da restricao
3	Descrição	LONG VARCHAR	(50)	Nulo	Abreviação
4	Professor	INTEGER	-	FK	Restrição do professor
5	Disciplina	INTEGER	-	FK	Restrição da disciplina
6	Turma	INTEGER	-	FK	Restrição da turma
7	Tupla	INTEGER	-	FK	Restrição da tupla
8	OpcaoDeHorario	INTEGER	-	Não nulo	Tipos de restrições (1 a 6)
9	Inicio	CHAR	(8)	Nulo	Início da restrição
10	Duacao	CHAR	(8)	Nulo	Duração da restrição
11	DiasDaSemana	VARCHAR	(20)	Nulo	Dia da semana
12	Nível	INTEGER	-	Não nulo	Preferência ou impossibilidade

FONTE: O autor (2007)

7.9.1.14 Tabela resultado

TABELA 17 – RESULTADO

Resultado					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome do resultado
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Instituicao	INTEGER	-	FK	Instituição do resultado

FONTE: O autor (2007)

7.9.1.15 Tabela resultadodimensao continua

TABELA 18 – RESULTADODIMENSAO

ResultadoDimensao					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Tipo	INT	-	Não nulo	Objeto a ser armazenado
3	Nome	VARCHAR	(20)	Não nulo	Nome do resultado
4	Descrição	VARCHAR	(50)	Nulo	Abreviação
5	IDOriginal	INT	-	Não nulo	Id original do objeto
6	Resultado	INT	-	FK	-
7	DiaDaSemana	INTEGER	-	Nulo	Dia de confecção
8	Inicio	CHAR	(8)	Nulo	Inicio do processamento
9	Duaracao	CHAR	(8)	nulo	Duração do processamento

FONTE: O autor (2007)

7.9.1.16 Tabela resultadofato

TABELA 19 – RESULTADOFATO continua

ResultadoFato					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Resultado	INTEGER	-	FK	-
3	Professor	INTEGER	-	FK	-
4	Disciplina	INTEGER	-	FK	-
5	Turma	INTEGER	-	FK	-
6	Turno	INTEGER	-	FK	-
7	Grade	INTEGER	-	FK	-
8	Dia	INTEGER	-	FK	-
9	Intervalo	INTEGER	-	FK	-

RESULTADOFATO conclusão

ResultadoFato					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
10	Curso	INTEGER	-	FK	-
11	ColisoesSerias	INTEGER	-	Nulo	Numero de colisões e impossibilidades
12	ColisoesMenores	INTEGER	-	Nulo	Numero de desfavorecimentos
13	ColisoesPreferencias	INTEGER	-	Nulo	Outras restrições

FONTE: O autor (2007)

7.9.1.17 Tabela modelo

TABELA 20 – MODELO

Modelo					
ID	CAMPO	TIPO	TAMANHO	CHAVE	DESCRIÇÃO
1	ID	INTEGER	(Auto)	PK	Identificador único
2	Nome	VARCHAR	(20)	Não nulo	Nome do modelo
3	Descrição	VARCHAR	(50)	Nulo	Abreviação
4	Linha	VARCHAR	(120)	Não nulo	Dimensão da linha
5	Coluna	VARCHAR	(120)	Não nulo	Dimensão da coluna
6	Tabela	VARCHAR	(120)	Nulo	Dimensão da tabela
7	Célula	VARCHAR	(120)	Não nulo	Padrão de exibição

FONTE: O autor (2007)

7.9.2 Script para criação das tabelas

7.9.2.1 Script tabela usuario

```
CREATE TABLE Usuario
(
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
  (START WITH 1, INCREMENT BY 1),
  Nome VARCHAR(20) NOT NULL,
  Descricao VARCHAR(50),
  Senha VARCHAR(150),
  Permissao INTEGER,
  CONSTRAINT un_UsuarioNome UNIQUE (Nome)
)
```

```
INSERT INTO Usuario (nome, descrição, senha, permissao) VALUES
('administrador', 'Usuário padrão do sistema', '', 5);
```

7.9.2.2 Script tabela configuracao

```
CREATE TABLE Configuracao
(
  Opcao VARCHAR(15) NOT NULL PRIMARY KEY,
  Valor VARCHAR(50)
)
```

```
INSERT INTO Configuracao(Opcao, Valor) VALUES (?, ?)
```

7.9.2.3 Script tabela instituicao

```
CREATE TABLE Instituicao  
(  
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY  
(START WITH 1, INCREMENT BY 1),  
  Nome VARCHAR(20) NOT NULL,  
  Descricao VARCHAR(50),  
  Local VARCHAR(40),  
  CONSTRAINT un_InstituicaoNome UNIQUE (Nome)  
)  
INSERT INTO Instituicao (Nome, Descricao, Local) VALUES (?, ?, ?);
```

7.9.2.4 Script tabela curso

```
CREATE TABLE Curso  
(  
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY  
(START WITH 1, INCREMENT BY 1),  
  Nome VARCHAR(20) NOT NULL,  
  Descricao VARCHAR(50),  
  Instituicao INTEGER NOT NULL,  
  CONSTRAINT fk_CursoInstituicao FOREIGN KEY (Instituicao)  
  REFERENCES Instituicao (ID) ON UPDATE RESTRICT ON DELETE  
  CASCADE,
```

```
CONSTRAINT un_CursoNome UNIQUE (Instituicao, Nome)
)
INSERT INTO Curso (Nome, Descricao, Instituicao) VALUES (?, ?, ?);
```

7.9.2.5 Script tabela turno

```
CREATE TABLE Turno
(
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
  (START WITH 1, INCREMENT BY 1),
  Nome VARCHAR(20) NOT NULL,
  Descricao VARCHAR(50),
  Instituicao INTEGER NOT NULL ,
  CONSTRAINT fk_TurnoInstituicao FOREIGN KEY (Instituicao)
  REFERENCES Instituicao (ID) ON UPDATE RESTRICT ON DELETE
  RESTRICT,
  CONSTRAINT un_TurnoNome UNIQUE (Instituicao, Nome)
)
INSERT INTO Turno (Instituicao, Nome, Descricao) VALUES (?, ?, ?);
```

7.9.2.6 Script tabela grade

```
CREATE TABLE Grade
(
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
  (START WITH 1, INCREMENT BY 1),
  Nome VARCHAR(20) NOT NULL,
  Descricao VARCHAR(50),
```



```

Instituicao INTEGER NOT NULL,
Turno INTEGER NOT NULL,
CONSTRAINT fk_GradeInstituicao FOREIGN KEY (Instituicao)
REFERENCES Instituicao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT fk_GradeTurno FOREIGN KEY (Turno)
REFERENCES Turno (ID) ON UPDATE RESTRICT ON DELETE RESTRICT,
CONSTRAINT un_GradeNome UNIQUE (Instituicao, Nome)
)

```

```

INSERT INTO Grade (Instituicao, Turno, Nome, Descricao) VALUES (?, ?, ?, ?);

```

7.9.2.7 Script tabela dia

```

CREATE TABLE Dia
(
ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
(START WITH 1, INCREMENT BY 1),
Nome VARCHAR(20) NOT NULL,
Descricao VARCHAR(50),
Grade INTEGER NOT NULL,
DiaDaSemana INTEGER NOT NULL,
CONSTRAINT fk_DiaGrade FOREIGN KEY (Grade)
REFERENCES Grade (ID) ON UPDATE RESTRICT ON DELETE CASCADE,
CONSTRAINT un_DiaNome UNIQUE (Grade, Nome)
)
INSERT INTO Dia (Grade, Nome, Descricao, DiaDaSemana) VALUES (?, ?, ?, ?);

```

7.9.2.8 Script tabela intervalo

```
CREATE TABLE Intervalo
(
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
(START WITH 1, INCREMENT BY 1),
  Nome VARCHAR(20) NOT NULL,
  Descricao VARCHAR(50),
  Dia INTEGER NOT NULL,
  Inicio CHAR(8) NOT NULL,
  Duracao CHAR(8) NOT NULL,
  Aula CHAR(1) NOT NULL,
  CONSTRAINT fk_IntervaloDia FOREIGN KEY (Dia)
  REFERENCES Dia (ID) ON UPDATE RESTRICT ON DELETE CASCADE,
  CONSTRAINT un_IntervaloNome UNIQUE (Dia, Nome)
)
INSERT INTO Intervalo (Dia, Nome, Descricao, Inicio, Duracao, Aula)
VALUES (?, ?, ?, ?, ?, ?);
```

7.9.2.9 Script tabela disciplina

```
CREATE TABLE Disciplina
(
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
(START WITH 1, INCREMENT BY 1),
  Nome VARCHAR(20) NOT NULL,
  Descricao VARCHAR(50),
  Instituicao INTEGER NOT NULL,
  CONSTRAINT fk_DisciplinaInstituicao FOREIGN KEY (Instituicao)
```

```
REFERENCES Instituicao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT un_DisciplinaNome UNIQUE (Instituicao, Nome)
)
INSERT INTO Disciplina (Nome, Descricao, Instituicao) VALUES (?, ?, ?);
```

7.9.2.10 Script tabela professor

```
CREATE TABLE Professor
(
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
(START WITH 1, INCREMENT BY 1),
  Nome VARCHAR(20) NOT NULL,
  Descricao VARCHAR(50),
  Instituicao INTEGER NOT NULL,
  CONSTRAINT fk_ProfessorInstituicao FOREIGN KEY (Instituicao)
REFERENCES Instituicao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT un_ProfessorNome UNIQUE (Instituicao, Nome)
)

INSERT INTO Professor (Nome, Descricao, Instituicao) VALUES (?, ?, ?);
```

7.9.2.11 Script tabela turma

```
CREATE TABLE Turma
(
```



```

ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
(START WITH 1, INCREMENT BY 1),
Nome VARCHAR(20) NOT NULL,
Descricao VARCHAR(50),
Instituicao INTEGER NOT NULL,
Grade INTEGER NOT NULL,
Curso INTEGER NOT NULL,
CONSTRAINT fk_TurmaCurso FOREIGN KEY (Curso)
REFERENCES Curso (ID) ON UPDATE RESTRICT ON DELETE RESTRICT,
CONSTRAINT fk_TurmaGrade FOREIGN KEY (Grade)
REFERENCES Grade (ID) ON UPDATE RESTRICT ON DELETE RESTRICT,
CONSTRAINT fk_TurmaInstituicao FOREIGN KEY (Instituicao)
REFERENCES Instituicao (ID) ON UPDATE RESTRICT ON DELETE
RESTRICT,
CONSTRAINT un_TurmaNome UNIQUE (Instituicao, Nome)
)
INSERT INTO Turma (Nome, Descricao, Instituicao, Curso, Grade)      VALUES
(?, ?, ?, ?, ?);

```

7.9.2.12 Script tabela tupla

```

CREATE TABLE Tupla
(
ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
(START WITH 1, INCREMENT BY 1),
Nome VARCHAR(20),
Descricao VARCHAR(50),
Professor INTEGER NOT NULL,
Disciplina INTEGER NOT NULL,
Professor2 INTEGER,
Disciplina2 INTEGER,

```

```

Turma INTEGER NOT NULL,
Quantidade INTEGER NOT NULL,
CONSTRAINT fk_TuplaDisciplina FOREIGN KEY (Disciplina)
  REFERENCES Disciplina (ID) ON UPDATE RESTRICT ON DELETE CASCADE,
CONSTRAINT fk_TuplaProfessor FOREIGN KEY (Professor)
  REFERENCES Professor (ID) ON UPDATE RESTRICT ON DELETE CASCADE,
CONSTRAINT fk_TuplaDisciplina2 FOREIGN KEY (Disciplina2)
  REFERENCES Disciplina (ID) ON UPDATE RESTRICT ON DELETE CASCADE,
CONSTRAINT fk_TuplaProfessor2 FOREIGN KEY (Professor2)
  REFERENCES Professor (ID) ON UPDATE RESTRICT ON DELETE CASCADE,
CONSTRAINT fk_TuplaTurma FOREIGN KEY (Turma)
  REFERENCES Turma (ID) ON UPDATE RESTRICT ON DELETE CASCADE,
CONSTRAINT un_Tupla UNIQUE (Professor, Disciplina, Turma)
)
INSERT INTO Tupla (Nome, Descricao, Professor, Disciplina, Professor2,
Disciplina2, Turma, Quantidade) VALUES (?, ?, ?, ?, ?, ?, ?, ?);

```

7.9.2.13 Script tabela restricao

```

CREATE TABLE Restricao
(
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
(START WITH 1, INCREMENT BY 1),
  Nome VARCHAR(150),
  Descricao LONG VARCHAR,
  Professor INTEGER,
  Disciplina INTEGER,
  Turma INTEGER,
  Tupla INTEGER,
  OpcaoDeHorario INTEGER NOT NULL,
  Inicio CHAR(8),

```

```

Duracao CHAR(8),
DiasDaSemana VARCHAR(20),
Nivel INTEGER NOT NULL,
CONSTRAINT fk_RestricaoProfessor FOREIGN KEY (Professor)
    REFERENCES Professor (ID) ON UPDATE RESTRICT ON DELETE CASCADE,
CONSTRAINT fk_RestricaoDisciplina FOREIGN KEY (Disciplina)
    REFERENCES Disciplina (ID) ON UPDATE RESTRICT ON DELETE CASCADE,
CONSTRAINT fk_RestricaoTurma FOREIGN KEY (Turma)
    REFERENCES Turma (ID) ON UPDATE RESTRICT ON DELETE CASCADE,
CONSTRAINT fk_RestricaoTupla FOREIGN KEY (Tupla)
    REFERENCES Tupla (ID) ON UPDATE RESTRICT ON DELETE CASCADE
)
INSERT INTO Restricao (Nome, Descricao, OpcaoDeHorario, Inicio, Duracao,
DiasDaSemana, Nivel, Professor, Disciplina, Turma, Tupla)
VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?);

```

7.9.2.14 Script tabela resultado

```

CREATE TABLE Resultado
(
    ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
(START WITH 1, INCREMENT BY 1),
    Nome VARCHAR(20) NOT NULL,
    Descricao VARCHAR(50),
    Instituicao INTEGER NOT NULL,
    CONSTRAINT fk_ResultadoInstituicao FOREIGN KEY (Instituicao)
    REFERENCES Instituicao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
    CONSTRAINT un_ResultadoNome UNIQUE (Instituicao, Nome)
)
INSERT INTO Resultado (Nome, Descricao, Instituicao) VALUES (?, ?, ?);

```


7.9.2.15 Script tabela resultadodimensao

```

CREATE TABLE ResultadoDimensao (
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
  (START WITH 1, INCREMENT BY 1),
  Tipo INT NOT NULL,
  Nome VARCHAR(20),
  Descricao VARCHAR(50),
  IDOriginal INT NOT NULL,
  Resultado INT NOT NULL,
  DiaDaSemana INTEGER,
  Inicio CHAR(8),
  Duracao CHAR(8),
  CONSTRAINT fk_ResultadoDimensaoResultado FOREIGN KEY (Resultado)
  REFERENCES Resultado (ID) ON UPDATE RESTRICT ON DELETE CASCADE
)
INSERT INTO ResultadoDimensao (Tipo, Nome, Descricao, IDOriginal, Resultado,
DiaDaSemana, Inicio, Duracao) VALUES (?, ?, ?, ?, ?, ?, ?, ?);

```

7.9.2.16 Script tabela resultadofato

```

CREATE TABLE ResultadoFato
(
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
  (START WITH 1, INCREMENT BY 1),
  Resultado INTEGER NOT NULL,
  Professor INTEGER NOT NULL,
  Disciplina INTEGER NOT NULL,
  Turma INTEGER NOT NULL,
  Turno INTEGER NOT NULL,

```

Grade INTEGER NOT NULL,
Dia INTEGER NOT NULL,
Intervalo INTEGER NOT NULL,
Curso INTEGER NOT NULL,
ColisoesSerias INTEGER,
ColisoesMenores INTEGER,
ColisoesPreferencias INTEGER,
CONSTRAINT fk_ResultadoFatoResultado FOREIGN KEY (Resultado)
REFERENCES Resultado (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT fk_ResultadoFatoProfessor FOREIGN KEY (Professor)
REFERENCES ResultadoDimensao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT fk_ResultadoFatoDisciplina FOREIGN KEY (Disciplina)
REFERENCES ResultadoDimensao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT fk_ResultadoFatoTurma FOREIGN KEY (Turma)
REFERENCES ResultadoDimensao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT fk_ResultadoFatoTurno FOREIGN KEY (Turno)
REFERENCES ResultadoDimensao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT fk_ResultadoFatoGrade FOREIGN KEY (Grade)
REFERENCES ResultadoDimensao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT fk_ResultadoFatoDia FOREIGN KEY (Dia)
REFERENCES ResultadoDimensao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT fk_ResultadoFatoIntervalo FOREIGN KEY (Intervalo)
REFERENCES ResultadoDimensao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE,
CONSTRAINT fk_ResultadoFatoCurso FOREIGN KEY (Curso)
REFERENCES ResultadoDimensao (ID) ON UPDATE RESTRICT ON DELETE
CASCADE

```

)
INSERT INTO ResultadoFato (Resultado, Professor, Disciplina, Turma,
Turno,Grade, Dia, Intervalo, Curso, ColisoesSerias, ColisoesMenores,
ColisoesPreferenciais)
VALUES (?, ?, ?,?,?,?,?,?,?,?);

```

7.9.2.17 Script tabela modelo

```

CREATE TABLE Modelo
(
  ID INTEGER NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY
(START WITH 1, INCREMENT BY 1),
  Nome VARCHAR(20) NOT NULL,
  Descricao VARCHAR(50),
  Linha VARCHAR(120) NOT NULL,
  Coluna VARCHAR(120) NOT NULL,
  Tabela VARCHAR(120),
  Celula VARCHAR(120) NOT NULL
)
INSERT INTO Modelo (Nome, Descricao, Linha, Coluna, Tabela, Celula) VALUES
(?, ?, ?,?,?,?);

```


8. RECURSOS

Trata de ponto de apoio importante no desenvolvimento do trabalho. Os recursos descritos nessa seção tratam de materiais utilizados desde a modelagem do projeto até a apresentação do mesmo.

8.1. HARDWARES UTILIZADOS

Por tratar-se de uma aplicação *desktop* não existe uma grande lista de hardwares utilizados, fazendo-se necessário apenas o uso de computadores pessoais para o desenvolvimento e teste.

8.1.1 Microcomputadores utilizados

Os hardwares utilizados serão os micro-computadores dos alunos integrantes da equipe com as seguintes configurações:

TABELA 21 – RELAÇÃO DE MICRO-COMPUTADORES UTILIZADOS

Descrição	Usuário	Qtde
Pentium 4 com HyperThreading, 3,06 GHz e 512MB de RAM	Aldinei	1
AMD Athlon™ XP 1700+ 1,1 GHz com 1 G de RAM	Fabio	1
Celeron, 2,66ghz, 256 RAM	Mauro	1

FONTE: O autor (2007)

8.2. SOFTWARES UTILIZADOS

Antes de fazer menção aos softwares utilizados vale destacar que uma das funcionalidades do programa era que o mesmo pudesse rodar em diversas plataformas e, portanto houve a escolha da linguagem Java para desenvolvimento. Muito em função dessa escolha foram utilizado os softwares descritos nessa seção.

8.2.1 Eclipse

Eclipse é um software open source desenvolvido pela *Eclipse Software Foundation*, é um ferramenta baseada em Java, sua IDE contém as ferramentas essenciais, como editor e compilador, para o desenvolvimento de projetos que utilizam a plataforma.

8.2.2 Netbeans

O *Netbeans* é um ambiente integrado de desenvolvimento (IDE) que permite ao programador criar programas utilizando recursos gráficos. Para trabalhar com o *Netbeans* é necessário ter instalado, anteriormente em sua máquina uma das versões do JDK (Java). Em nosso trabalho utilizamos este software para desenvolver as interfaces do projeto.

8.2.3 Jude

Jude Community é uma aplicação open source que usa UML para modelizar o desenho de software de computador. A aplicação corre na maior parte das plataformas uma vez que é implementada em Java. Providencia suporte para quase todos os tipos de diagrama da UML padrão e inclui suporte cognitivo.

Com esse software desenvolvemos toda parte de modelagem referente do Sistema. Ele se mostrou muito eficaz no desenvolvimento dos Casos de Uso, Diagramas de Seqüência, Diagramas de Classes, Diagramas de Atividades e Diagramas de Estado.

8.2.4 TortoiseSVN

O *TortoiseSVN* é uma ferramenta gráfica, que se integra ao Windows Explorer, que facilita o uso do *Subversion* para usuários do Windows, uma vez que o *Subversion* básico é uma ferramenta de comandos diretos de texto puro. Utilizamos este software para gerenciar e controlar as mudanças nos nossos arquivos do projeto.

8.2.5 Derby

O *Derby* é um sistema gerenciador de banco de dados relacional (SGBDR) baseado em Java e SQL. Ele pode ser implementado de várias maneiras diferentes, para nosso projeto optamos por implementá-lo no modo monousuário incorporado-o a nossa aplicação. Dessa forma ele ficará invisível ao usuário final e rodará na mesma máquina virtual do aplicativo dispensando a necessidade de administração.

8.2.6 JasperReport

O *JasperReports* é uma biblioteca Java, ou seja, é um conjunto de pacotes compostos por classes Java, desenvolvido por um fornecedor independente chamado *JasperSoft*. Esta biblioteca tem como finalidade a geração de relatório em aplicações Java.

8.2.7 DBDesigner 4

O *DBDesigner* é um software para criar bases de dados, que integra o desenho, modelação, criação e manutenção num único programa. Este programa é útil para criar tabelas visualmente, em vez de utilizar o código SQL, definir tipos de dados e relações entre tabelas. No nosso projetos utilizamos o *DBDesigner* para construir do Diagrama Entidade Relacional.

8.3. OUTROS SOFTWARES UTILIZADOS

8.3.1 MS-Word

Para a elaboração de textos e de toda a documentação escrita, utilizamos o Microsoft Word. Optamos por esse editor de textos pela sua popularidade e pelo imenso número de ferramentas que supriria todas as necessidades da equipe no decorrer do projeto.

8.3.2 MS-PowerPoint

Elaboramos a apresentação para demonstração do Sistema utilizando os recursos disponíveis no PowerPoint.

9. CRONOGRAMA

9.1. CALENDÁRIO DAS ATIVIDADES

Para desenvolvermos o projeto seguimos o seguinte calendário de atividades:

TABELA 22 – CALENDÁRIO DE ATIVIDADES DO PROJETO

Atividade	Duração (d)	Início	Término
Sistema Millenium	207	12/03/07	13/12/07
Planejamento	37	12/03	18/04
Requisitos	24	12/03	05/04
Levantar Requisitos	15	12/03	27/03
Software Concorrente	15	12/03	27/03
Análise	9	27/3	05/04
Escopo	3	05/04	08/04
Escopo do projeto	2	05/04/	07/04
Escopo do produto	1	08/04	08/04
Estimativas	5	09/04	14/04
Riscos	3	15/04	18/04
Elaboração	60	19/04	18/06
Modelar Negócio	24	19/04	13/05
Levantar UC	10	19/04	29/04
Analisar e Projetar	14	29/04	13/05
Modelar Classes	17	15/05	01/06
Identificar	5	15/05	20/05
Análise e Projeto	10	20/05	30/05
Camadas	10	30/05	09/06
Modelar AG	15	13/05	28/05
Identificar restrições	5	23/05	28/05
Análise e Projeto	15	28/05	12/06
Prototipar	13	01/06	14/06
Construção	107	15/07	30/10
Implementar Sistema	77	15/07	30/09
Documentar	15	30/09	15/10
Testar	15	30/09	15/10
Homologar	15	15/10	30/10
Transição	44	30/10	13/12
Disponibilizar para escolas participantes	40	30/10	09/12
Banca	-	-	13/12
Tempo estimado (dias)	207		

FONTE: O autor (2007)

9.2. DIAGRAMA DE GANTT

Antes de apresentarmos o Diagrama de Gantt desse trabalho convém mostrar a visão desse projeto com relação a conceitos de WBS (Work breakdown structure), que em português quer dizer estrutura analítica do projeto.

9.2.1 Composição básica do projeto

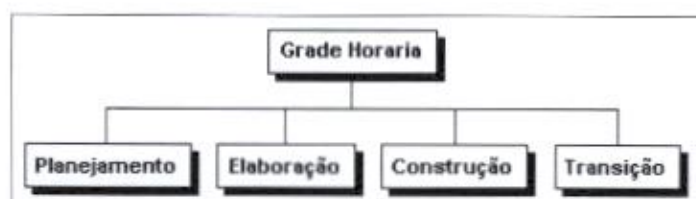


FIGURA 78 – COMPOSIÇÃO BÁSICA DO PROJETO GRADE – WBS

FONTE: O autor (2007)

9.2.2 Composição do planejamento - WBS

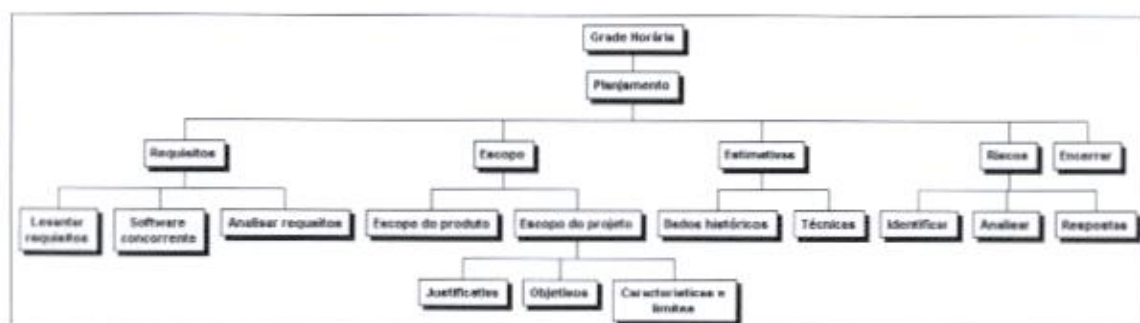


FIGURA 79 – COMPOSIÇÃO DO PLANEJAMENTO – WBS

FONTE: O autor (2007)

9.2.3 Composição da elaboração - WBS

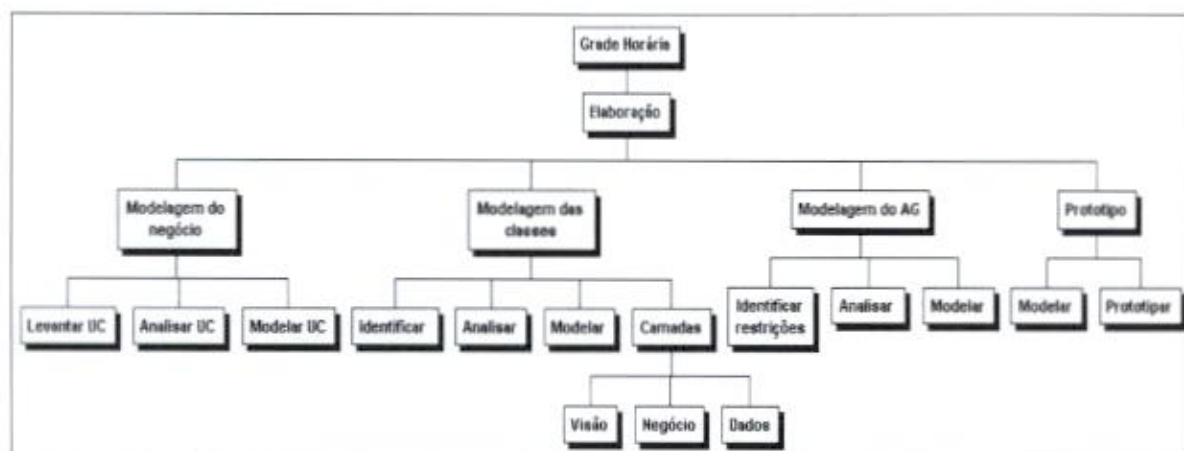


FIGURA 80 – COMPOSIÇÃO DA ELABORAÇÃO – WBS

FONTE: O autor (2007)

9.2.4 Composição da construção - WBS

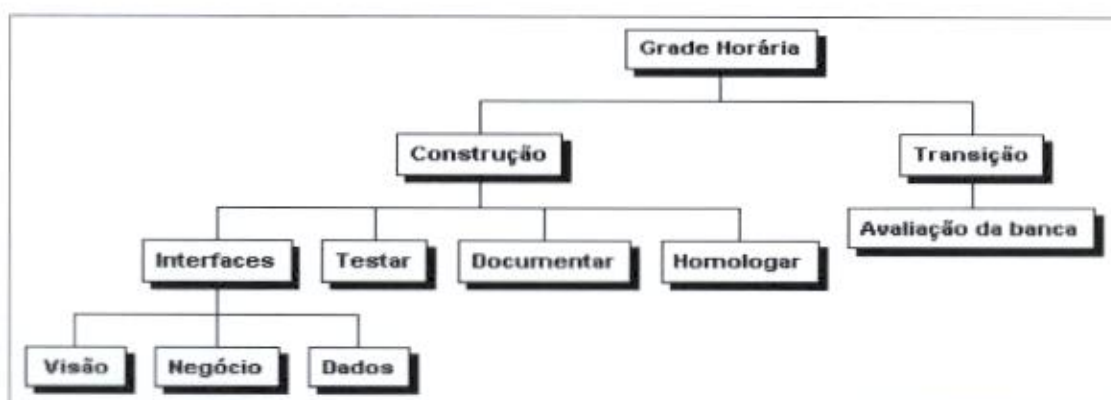


FIGURA 81 – COMPOSIÇÃO DA CONSTRUÇÃO – WBS

FONTE: O autor (2007)

9.2.5 Diagrama de gantt aplicado ao projeto

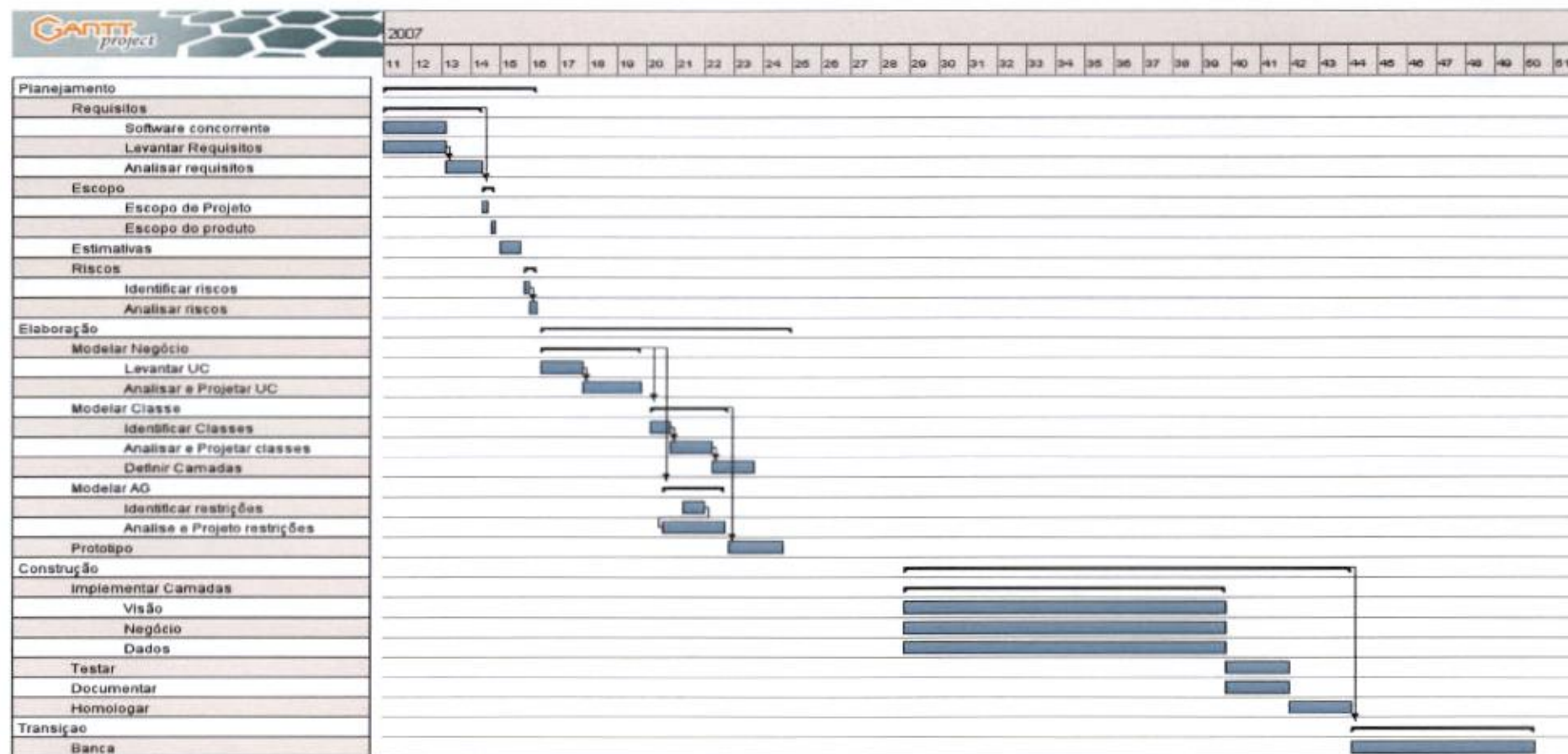


FIGURA 82 – DIAGRAMA DE GANTT

FONTE: O autor (2007)

9.3. WORKFLOW DE TESTE

Nesta seção serão mostrados os workflows de teste feitos durante o processo do projeto. A partir de agora para qualquer sigla WT leia-se workflow de teste.

9.3.1 Teste 1: criação de tabelas

Objetivo: Este procedimento visa verificar se todas as tabelas do banco foram criadas corretamente.

TABELA 23 – WT CRIAÇÃO DE TABELAS

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Gerar todas as tabelas do Millennium	Arquivo com Script SQL para criação de tabelas.	Criação das tabelas: Usuario, Configuração, Instituicao, Curso, Turno, Grade, Dia, Intervalo, Disciplina, Professor, Turma, Tupla, Restrição, Resultado, ResultadoDimensao, ResultadoFato Modelo	Criação realizada com sucesso

FONTE: O autor (2007)

9.3.2 Teste 2: permissões do usuário

Objetivo: Este procedimento visa verificar se a função de permissão de usuário esta funcionando corretamente.

TABELA 24 – WT PERMISSÕES DO USUÁRIO

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Efetuar login com permissão de administrador	Realizar inserção, alteração e exclusão de dados	Gravação, alteração e exclusão do banco dos dados inseridos	Manipulação dos dados realizados com sucesso
Efetuar login com permissão de consulta	Tentar inserir dados	Permitir somente consulta	Somente visualização dos dados
Efetuar login com permissão de edição e consulta	Criar um novo usuário	Permitir manipulação de dados exceto a criação de um novo usuário	Bloqueio na tentativa de criação de um novo usuario

FONTE: O autor (2007)

9.3.3 Teste 3: troca de senha do usuário

Objetivo: Permitir a troca de senha pelo usuário.

TABELA 25 – WT TROCA DE SENHA DE USUÁRIO

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Efetuar troca de senha de um usuário	<ul style="list-style-type: none">- Entrar na tela de troca de senha- Digitar a senha nova- Executar logoff- Efetuar login com a nova senha	Permissão de acesso ao sistema	Acesso permitido

FONTE: O autor (2007)

9.3.4 Teste 4: manutenção de usuário.

Objetivo: Permitir ao usuário com permissão de administrador realizar inclusão, edição e exclusão de um usuário.

TABELA 26 – WT MANUTENÇÃO DE USUÁRIO

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir um novo usuário	Administrador insere um novo usuário	O cadastro deve ser salvo na tabela usuário	Novo usuário cadastrado com sucesso
Editar um usuário	Administrador altera um dado do usuário.	Alteração na tabela usuário	Dado alterado com sucesso
Excluir um usuário do banco	Administrador deve selecionar um usuário e clicar no botão excluir	Usuário deve ser excluído da banco de dados	Usuário excluído com sucesso

FONTE: O autor (2007)

9.3.5 Teste 5: manutenção de instituição.

Objetivo: Permitir ao usuário com permissão de administrador ou edição realizar inclusão, edição e exclusão de uma instituição

TABELA 27 – WT MANUTENÇÃO DE INSTITUIÇÃO.

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir uma nova instituição	usuário insere uma nova instituição	O cadastro deve ser salvo na tabela instituição	Nova instituição cadastrada com sucesso
Editar uma instituição	Usuário altera um dado da instituição.	Alteração na tabela instituição	Dado alterado com sucesso
Excluir uma instituição do banco	Usuário deve selecionar uma instituição e clicar no botão excluir	Instituição deve ser excluída da banco de dados	Instituição excluída com sucesso

FONTE: O autor (2007)

9.3.6 Teste 6: seleção de instituição.

Objetivo: Permitir ao usuário selecionar a instituição a qual ele deseja manipular.

TABELA 28 – WT SELEÇÃO DE INSTITUIÇÃO

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Selecionar uma instituição	- Usuário efetua login no sistema - Usuário seleciona uma instituição na lista e clica no botão OK	Sistema deve carregar todos os dados da instituição selecionada (Professores, Turmas, Cursos, Grades, etc).	Dados da instituição selecionada carregados com sucesso

FONTE: O autor (2007)

9.3.7 Teste 7: manutenção do curso.

Objetivo: Permitir ao usuário com permissão de administrador/edição realizar inclusão, edição e exclusão de um curso.

TABELA 29 – WT MANUTENÇÃO DO CURSO

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir um novo curso	Usuário insere um novo curso	O cadastro deve ser salvo na tabela curso	Novo curso cadastrado com sucesso
Editar um curso	Usuário altera um dado do curso	Alteração na tabela curso	Dado alterado com sucesso
Excluir um curso do banco	Usuário deve selecionar um curso e clicar no botão excluir	Curso deve ser excluído da banco de dados	Curso excluído com sucesso

FONTE: O autor (2007)

9.3.8 Teste 8: manutenção do turno.

Objetivo: Permitir ao usuário com permissão de administrador/edição realizar inclusão, edição e exclusão de um turno.

TABELA 30 – WT MANUTENÇÃO DO TURNO

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir um novo turno	Usuário insere um novo turno	O cadastro deve ser salvo na tabela turno	Novo turno cadastrado com sucesso
Editar um turno	Usuário altera um dado do turno	Alteração na tabela turno	Dado alterado com sucesso
Excluir um turno do banco	Usuário deve selecionar um turno e clicar no botão excluir	Turno deve ser excluído da banco de dados	Turno excluído com sucesso

FONTE: O autor (2007)

9.3.9 Teste 9: manutenção da grade horária.

Objetivo: Permitir ao usuário com permissão de administrador/edição realizar inclusão, edição e exclusão de uma grade horária.

TABELA 31 – WT MANUTENÇÃO DE GRADE HORÁRIA

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir uma nova grade horária	Usuário insere uma nova grade	O cadastro deve ser salvo na tabela grade	Nova grade cadastrada com sucesso
Editar uma grade horária	Usuário altera um dado da grade horária	Alteração na tabela grade horária	Dado alterado com sucesso
Excluir uma grade horária do banco	Usuário deve selecionar uma grade e clicar no botão excluir	Grade deve ser excluída da banco de dados	Grade excluída com sucesso

FONTE: O autor (2007)

9.3.10 Teste 10: manutenção do professor.

Objetivo: Permitir ao usuário com permissão de administrador/edição realizar inclusão, edição e exclusão de um professor.

TABELA 32 – WT MANUTENÇÃO DO PROFESSOR

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir um novo professor	Usuário insere um novo professor	O cadastro deve ser salvo na tabela professor.	Novo professor cadastrado com sucesso
Editar um professor	Usuário altera um dado do professor	Alteração na tabela professor	Dado alterado com sucesso
Excluir um professor do banco	Usuário deve selecionar um professor e clicar no botão excluir	Professor deve ser excluído da banco de dados	Professor excluído com sucesso

FONTE: O autor (2007)

9.3.11 Teste 11: manutenção da disciplina.

Objetivo: Permitir ao usuário com permissão de administrador/edição realizar inclusão, edição e exclusão de uma disciplina.

TABELA 33 – WT MANUTENÇÃO DA DISCIPLINA

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir uma nova disciplina	Usuário insere uma nova disciplina	O cadastro deve ser salvo na tabela disciplina	Nova disciplina cadastrada com sucesso
Editar uma disciplina	Usuário altera um dado de uma disciplina	Alteração na tabela disciplina	Dado alterado com sucesso
Excluir uma disciplina do banco	Usuário deve selecionar uma disciplina e clicar no botão excluir	Disciplina deve ser excluída da banco de dados	Disciplina excluída com sucesso

FONTE: O autor (2007)

9.3.12 Teste 12: manutenção da turma.

Objetivo: Permitir ao usuário com permissão de administrador/edição realizar inclusão, edição e exclusão de uma turma.

TABELA 34 – WT MANUTENÇÃO DA TURMA

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir uma nova turma	Usuário insere uma nova turma	O cadastro deve ser salvo na tabela turma	Nova turma cadastrada com sucesso
Editar uma turma	Usuário altera um dado da turma	Alteração na tabela Turma	Dado alterado com sucesso
Excluir uma turma do banco	Usuário deve selecionar uma turma e clicar no botão excluir	Turma deve ser excluída da banco de dados	Turma excluída com sucesso

FONTE: O autor (2007)

9.3.13 Teste 13: manutenção da tupla.

Objetivo: Permitir ao usuário com permissão de administrador/edição realizar inclusão, edição e exclusão de uma tupla.

TABELA 35 – WT MANUTENÇÃO DA TUPLA

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir uma nova tupla	Usuário insere uma nova tupla	O cadastro deve ser salvo na tabela tupla	Nova tupla cadastrada com sucesso
Editar uma tupla	Usuário altera um dado da tupla	Alteração na tabela tupla	Dado alterado com sucesso
Excluir uma tupla do banco	Usuário deve selecionar uma tupla e clicar no botão excluir	Tupla deve ser excluída da banco de dados	Tupla excluída com sucesso

FONTE: O autor (2007)

9.3.14 Teste 14: manutenção da restrição.

Objetivo: Permitir ao usuário com permissão de administrador/edição realizar inclusão, edição e exclusão de uma restrição.

TABELA 36 – WT MANUTENÇÃO DE RESTRIÇÃO

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir uma nova restrição	Usuário insere uma nova restrição	O cadastro deve ser salvo na tabela restrição	Nova restrição cadastrada com sucesso
Editar uma restrição horária	Usuário altera um dado da restrição	Alteração na tabela restrição	Dado alterado com sucesso
Excluir uma restrição do banco	Usuário deve selecionar uma restrição e clicar no botão excluir	Restrição deve ser excluída da banco de dados	Restrição excluída com sucesso

FONTE: O autor (2007)

9.3.15 Teste 15: manutenção do dia.

Objetivo: Permitir ao usuário com permissão de administrador/edição realizar inclusão, edição e exclusão de uma grade horária.

TABELA 37 – WT MANUTENÇÃO DO DIA

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir um novo dia da semana	Usuário insere um novo dia	O cadastro deve ser salvo na tabela dia	Nova grade cadastrada com sucesso
Editar um dia da semana	Usuário altera um dado da dia	Alteração na tabela dia	Dado alterado com sucesso
Excluir um dia do banco	Usuário deve selecionar um dia e clicar no botão excluir	Dia deve ser excluído da banco de dados	Dia excluído com sucesso

FONTE: O autor (2007)

9.3.16 Teste 16: manutenção dos horários.

Objetivo: Permitir ao usuário com permissão de administrador/edição realizar inclusão, edição e exclusão de um horário.

TABELA 38 – WT MANUTENÇÃO DOS HORÁRIOS

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Incluir um novo horário	Usuário insere um novo horário	O cadastro deve ser salvo na tabela intervalo	Novo horário cadastrado com sucesso
Editar um horário	Usuário altera um dado do horário	Alteração na tabela intervalo	Dado alterado com sucesso
Excluir um horário do banco	Usuário deve selecionar uma horário e clicar no botão excluir	Horário deve ser excluído do banco de dados	Horário excluído com sucesso

FONTE: O autor (2007)

9.3.17 Teste 17: gerar grade horária

Objetivo: Permitir criação da tabela de horários escolar

TABELA 39 – WT GERAR GRADE HORÁRIA

Procedimento	Entrada	Resultado Esperado	Resultado Obtido
Alimentar o banco de dados com as informações da instituição	- Inserir: Instituição, Curso, Turno, Grade, Disciplina, Professor, Turma, Tupla e Restrições	Uma tabela de horários sem colisões de professores	Tabela gerada com sucesso

FONTE: O autor (2007)

9.3.18 Teste 18: uso de recursos.

Objetivo: Verificar como o sistema se comporta durante uma execução prolongada.

TABELA 40 – WT USO DE RECURSOS

Procedimento e Entrada	Resultado Esperado	Resultado Obtido
<ul style="list-style-type: none"> - Alimentar o banco com os dados de uma instituição de tamanho médio (cerca de 14 turmas) - Assegurar de que as restrições inseridas tornem a grade muito difícil ou impossível de resolver. - Ativar todas as restrições opcionais e incluir restrições de horário para a maioria dos professores - Colocar o sistema para gerar a grade e deixar o sistema rodando pelo maior tempo possível (no mínimo 24 horas). Monitorar o uso de processador e memória durante este período. 	<p>O uso de processador e memória deve ter pouca variação, o sistema não deve ficar alocando cada vez mais recursos sem liberar o que ele já alocou.</p>	<p>O sistema rodando por 51 horas e 20 minutos, executou 2035415 iterações, numa velocidade média constante de 11 iterações por segundo consumindo em média 52% da CPU e 62MB de memória RAM durante todo o tempo. Tudo funcionou de acordo com o esperado.</p>

FONTE: O autor (2007)

10. TESTES COMPUTACIONAIS

Esta seção tem por finalidade mostrar o resultado do Millenium por meio de testes em situações reais demonstrando sua performance e comparando resultado com outros aplicativos similares.

Além do Millenium, realizamos testes em mais dois softwares, são eles:

Asc TimeTables: é um software proprietário para plataformas windows, que tem por finalidade criar grade horária escolar, este aplicativo é comercializado em vinte idiomas, entre eles o Português. Para executarmos os testes, adquirimos uma versão demonstrativa através do site <http://www.asctimetables.com>.

O segundo o qual chamaremos de A, pelo motivo de não ter sido autorizado à divulgação do seu nome, é um software proprietário para plataformas windows que tem por finalidade criar grade horária escolar, este aplicativo é de origem nacional sendo utilizado por grande parte das escolas brasileiras, tanto da iniciativa privada quanto às escolas publicas. Para executarmos os testes utilizamos uma versão adquirida por uma das escolas participantes.

10.1. ESTUDO DE CASO 1

Situada em Curitiba-PR, a Escola Estadual Professor Júlio Mesquita possui um total de 27 professores, 13 turmas e 14 disciplinas, embora a escola lecione em 3 turnos, os testes foram feitos somente com os turnos matutino e vespertino.

Seus professores lecionam em mais de uma turma e trabalham para outras instituições de ensino. As turmas têm cinco aulas por dia, somando um total de 25 aulas por semana. A grade do ensino Fundamental e Médio de um mesmo turno funciona em horários diferentes, acrescento um grau de dificuldade para o genético, tendo em vista que um professor pode lecionar tanto para o ensino fundamental quanto para o médio.

Depois de efetuado o cadastro completo da instituição e mandar o aplicativo gerar a grade foram alcançados os seguintes resultados.

TABELA 41 - TESTES REALIZADOS COM O MILLENIUM

Escola Estadual Prof. Julio Mesquita/2007					
Turno	Fitness	Colisões	Desfavorecimento	Iterações	Tempo
Manhã	1056	176	0	140	00:00:25
	606	101	0	1195	00:03:57
	408	68	0	3380	00:11:03
	246	41	0	9344	00:30:33
	162	27*	0	136080	08:56:30
Tarde	750	125	0	7	00:00:05
	216	36	0	1197	00:03:37
	150	25*	0	327	00:10:45
	90	15*	0	11716	00:35:10
	48	8*	0	75900	05:32:33

*Apenas impossibilidades, alocou professor em dias impróprios.

FONTE: O autor (2007)

TABELA 42 - TESTES REALIZADOS COM O ASC TIMETABLES

Escola Estadual Prof. Julio Mesquita/2007		
Turno	Tempo	Resultado apresentado pelo aplicativo
Manhã	5 m	professor e a turma que estavam dificultado a elaboração da grade horária.
		apresentou o resultado gerado até o momento, com quatorze horários em branco.
		solicitou ao usuário reavaliar as informações cadastradas, com objetivo de detectar possíveis erros de cadastro.
Tarde	8 m	professor e a turma que estavam dificultado a elaboração da grade horária.
		apresentou o resultado gerado até o momento, com dez aulas em branco
		solicitou ao usuário reavaliar as informações cadastradas, com objetivo de detectar possíveis erros de cadastro.

FONTE: O autor (2007)

TABELA 43 - TESTES REALIZADOS COM O PROGRAMA A

Escola Estadual Prof. Julio Mesquita/2007		
Turno	Tempo	Resultado apresentado pelo aplicativo
Manhã	5 m	existência de vinte e três pendências e nove sugestões.
		informou ao usuário que só continuasse a execução se não estivesse com pressa e que a partir daquele ponto, provavelmente não haveria melhoras no resultado.
		apresentou o resultado gerado até aquele momento, com vinte e três aulas em branco.
Tarde	8 m	existência de dezoito pendências e seis sugestões.
		informou ao usuário que só continuasse a execução se não estivesse com pressa e que a partir daquele ponto, provavelmente não haveria melhoras no resultado.
		apresentou o resultado gerado até aquele momento, com dezoito aulas em branco.

FONTE: O autor (2007)

10.2. ESTUDO DE CASO 2

Situada no município de São José dos Pinhais - PR, a Escola Estadual Professor Lindaura R. Lucas possui um total de 78 professores, 18 turmas e 14 disciplinas, a escola leciona em 3 turnos.

Seus professores lecionam em mais de uma turma e em mais de um turno. As turmas têm cinco aulas por dia, somando um total de 25 aulas por semana.

Depois de efetuado o cadastro completo da instituição e mandar o aplicativo gerar a grade foram alcançados os seguintes resultados.

TABELA 44 - TESTES REALIZADOS COM O PROGRAMA A

Escola Estadual Profª Lindaura Lucas /2007					
Turno	Fitness	Colisões	Desfavorecimento	Iterações	Tempo (h:m:s)
Manhã	1520	280	25	15	00:00:05
	410	65	12	1322	00:07:26
	292	45	11	2176	00:11:03
	148	20	14	6046	00:30:24
	40	6	4	61321	05:32:27
Tarde	844	135	14	97	00:00:22
	196	28	14	1666	00:06:52
	148	20	14	2646	00:10:48
	68	8*	10	8209	00:32:44
	26	2*	7	21805	01:20:23
	12	2*	0	100203	05:23:06

* Apenas impossibilidades, alocou professor em dias impróprios.

FONTE: O autor (2007)

TABELA 45 - TESTES REALIZADOS COM O ASC TIMETABLES

Escola Estadual Profª Lindaura Lucas /2007		
Turno	Tempo	Resultado apresentado pelo aplicativo
Manhã	9 m	professor e a turma que estavam dificultado a elaboração da grade horária.
		apresentou o resultado gerado até o momento, com vinte horários em branco.
		solicitou ao usuário reavaliar as informações cadastradas, com objetivo de detectar possíveis erros de cadastro.
Tarde	6 m	professor e a turma que estava dificultado a elaboração da grade horária.
		apresentou o resultado gerado até o momento, com dez horários em branco.
		solicitou ao usuário reavaliar as informações cadastradas, com objetivo de detectar possíveis erros de cadastro.

FONTE: O autor (2007)

TABELA 46 - TESTES REALIZADOS COM O PROGRAMA A

Escola Estadual Profª Lindaura Lucas /2007		
Turno	Tempo (m)	Resultado apresentado pelo aplicativo
Manhã	14 m	EXISTÊNCIA DE DEZESSETE PENDÊNCIAS E NOVE SUGESTÕES
		informou ao usuário que só continuasse a execução se não estivesse com pressa e que a partir daquele ponto, provavelmente não haveria melhoras no resultado.
		apresentou o resultado gerado até aquele momento, com dezessete aulas em branco.
Tarde	20 m	existência de nove pendências e cinco sugestões.
		informou ao usuário que só continuasse a execução se não estivesse com pressa e que a partir daquele ponto, provavelmente não haveria melhoras no resultado.
		apresentou o resultado gerado até aquele momento, com nove aulas em branco.

FONTE: O autor (2007)

10.3. CONCLUSÃO DOS ESTUDOS DE CASOS

O Millenium apresenta duas grandes vantagens se comparado aos seus similares: o resultado e a não necessidade de intervenção por parte do usuário.

O melhor resultado propicia menos insatisfação das pessoas e gera um clima melhor para a instituição. Os outros softwares recomendam o usuário reavaliar suas posições em relação à grade gerando intervenções repetitivas e a busca manual de uma parte da solução. A constante comunicação do usuário com o software restringe-o a essa tarefa, impossibilitando de atuar em outras áreas.

A principal desvantagem do Millenium é o tempo de processamento, Ele pode demorar a dar soluções e em muitos casos depois de um certo tempo ele estabiliza na solução, podendo ela não ser viável.

11. CONCLUSÃO

A partir da análise dos resultados obtidos, pode-se concluir que as técnicas implementadas favorecem a obtenção de uma solução satisfatória e eficiente, em sua maioria, para o problema, tornando o início de cada ano letivo menos complicado e mais prático para a instituição.

Em relação à aplicação baseada no estudo de caso, pode-se concluir que a abordagem implementada tem um resultado surpreendente, informatizando e simplificando todas as operações e procedimentos anteriormente realizados de forma manual, utilizando-se métodos complexos da tentativa e erro, adotado na rotina da escola.

Como principal contribuição deste trabalho, pode ser mencionados o estudo e aplicação de técnicas de computação evolutiva (Algoritmos Genéticos), em conjunto com o problema de restrições, vindo como uma solução para o problema de alocação de professores em horários. O ordenamento dos recursos tenta alcançar com sucesso um determinado objetivo respeitando o conjunto de limitações impostas.

Como perspectivas futuras para este trabalho, pode-se sugerir a elaboração de novos operadores genéticos que possam adaptar-se melhor a cada contexto de aplicação e a verificação do desempenho da abordagem com novas restrições. Também podem ser criadas novas funcionalidades de tal forma a atender um número maior de necessidades das instituições. Faz-se necessário a criação de um manual para ser fornecido ao usuário para que o mesmo tenha mais facilidade no manuseio do programa.

Em relação à grade horária obtida como resultado da implementação, pode-se concluir que a abordagem tem um resultado satisfatório, ressaltando, porém que, a adoção da grade horária gerada, será feita pela instituição de ensino. Caso haja necessidade, será feita uma nova execução até uma solução adequada, mesmo que não satisfaça todas as restrições. A decisão de adotar ou não a grade horária gerada pelo Millenium caberá ao responsável designado formalmente para a execução dessa tarefa.

Temos diversos estabelecimentos utilizando o Millenium em caráter de teste

e um deles já confirmou que vai adotar o uso do sistema a partir de 2008. Continuaremos por tempo indeterminado dando suporte aos usuários do sistema em caráter voluntário, ou seja, sem esperar compensação financeira. Pretendemos dar continuidade ao desenvolvimento do sistema, incluindo novas restrições que infelizmente não puderam ser incluídas no escopo do trabalho devido à complexidade e/ou tempo excessivo de análise e desenvolvimento que estas representam.

REFERÊNCIAS BIBLIOGRÁFICAS

BOOCH, G.; RUMABUGH, J.; JACOBSON, I. **UML: guia do usuário**. Tradução: Fábio Freitas. Rio de Janeiro: Elsevier, 2000.

BRAZ JÚNIOR, O. O. **Otimização de horários em instituições de ensino superior através de algoritmos genéticos**. 144 p. Dissertação (Mestre em Engenharia de Produção) – Universidade Federal de Santa Catarina, Florianópolis, 2000.

DEITEL, H. M.; DEITEL, P. J. **JAVA: como programar**. Tradução: Edson Furmankiewicz. 6. ed. São Paulo: Pearson Prentice Hall, 2005.

GOLDBERG, D. E. **Genetic Algorithms in Search, Optimization, and Machine Learning**. Addison-Wesley Publishing Company Inc., 1989.

HAMAWAKI, C. D. L. **Geração automática de grade horária usando algoritmos genéticos: o caso da faculdade de engenharia elétrica da UFU**. 104 p. Dissertação (Mestre em Ciências) - Universidade Federal de Uberlândia, Uberlândia, 2005.

LINDEN, R. **Algoritmos genéticos: uma importante ferramenta da inteligência computacional**. Rio de Janeiro: Brasport, 2006.

QUADROS, M. **Gerência de projetos de software: técnicas e ferramentas**. Santa Catarina: Visual Books, 2002.

RIBEIRO FILHO, G. **Melhoramentos no algoritmo genético construtivo e novas aplicações em problemas de agrupamento**. 120 p. Tese (Doutorado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2001.

SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. **Sistemas de banco de dados**. Tradução: Daniel Vieira. 5. ed. Rio de Janeiro: Elsevier, 2006.

SOUZA, V. L. de. **Programação de grade horária em escolas de ensino fundamental e médio**. 112 p. Dissertação (Mestre em Matemática Aplicada) – Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica, Campinas, 2006.

TANOMARU, J. **Motivação, Fundamentos e Aplicações de Algoritmos Genéticos**. Congresso Brasileiro de Redes Neurais, Curitiba, 1995.

TOMAZELA, M. das G. J. M. **Utilização de algoritmos genéticos para elaboração de grade horária.** Revista científica do IMAPES, Sorocaba, v.4, n. 1, p. 28-34, 2003.

DOCUMENTOS CONSULTADOS

UNIVERSIDADE FEDERAL DO PARANÁ, SISTEMAS DE BIBLIOTECAS. **NORMAS PARA APRESENTAÇÃO DE DOCUMENTOS CIENTIFICOS 2:** teses, dissertações, monografias e outros trabalhos acadêmicos. Curitiba, 2007.

UNIVERSIDADE FEDERAL DO PARANÁ, SISTEMAS DE BIBLIOTECAS. **NORMAS PARA APRESENTAÇÃO DE DOCUMENTOS CIENTIFICOS 3:** citações e notas de rodapé. Curitiba, 2007.

UNIVERSIDADE FEDERAL DO PARANÁ, SISTEMAS DE BIBLIOTECAS. **NORMAS PARA APRESENTAÇÃO DE DOCUMENTOS CIENTIFICOS 4:** referências. Curitiba, 2007

UNIVERSIDADE FEDERAL DO PARANÁ, SISTEMAS DE BIBLIOTECAS. **NORMAS PARA APRESENTAÇÃO DE DOCUMENTOS CIENTIFICOS 9:** redação e editoração. Curitiba, 2007