

Universidade Federal de Santa Catarina
Programa de Pós-graduação em
Engenharia de Produção

**JOGO DE EMPRESAS LÍDER: MODELAGEM DE UM
SISTEMA BASEADO EM AGENTES UTILIZANDO A
LINGUAGEM UML**

MARCOS VIRGILIO DA COSTA

Dissertação apresentada ao
Programa de Pós-graduação em
Engenharia de Produção da
Universidade Federal de Santa Catarina
como requisito parcial para obtenção
do título de Mestre em
Engenharia de Produção

Florianópolis
2004

Marcos Virgilio da Costa

**JOGO DE EMPRESAS LÍDER: MODELAGEM DE UM
SISTEMA BASEADO EM AGENTES UTILIZANDO A
LINGUAGEM UML**

Esta dissertação foi julgada e aprovada para a obtenção do título de **Mestre em Engenharia de Produção** no **Programa de Pós-graduação em Engenharia de Produção** da
Universidade Federal de Santa Catarina
Florianópolis, 29 de Outubro de 2004.

Prof. Edson Pacheco Paladini Dr.
Coordenador do Curso

BANCA EXAMINADORA

Prof. Bruno Hartmut Kopittke, Dr.
Orientador

Prof. Jorge Hermenegildo, Dr.

Prof. Luiz Ricardo Uriarte, Dr.

Aos meus pais Marcos Sandoval e Juçara Virgili, por seu apoio, carinho e dedicação incondicional. Ao meu irmão Marcos Eduardo, por sua amizade e companheirismo.

Agradecimentos

Ao meu orientador Bruno Hartmut Kopittke, por sua confiança, apoio e orientação.

Aos professores Jorge Hermenegildo e Luiz Ricardo Uriarte, pelas críticas construtivas e sugestões que contribuíram significativamente para a realização desta dissertação.

Ao meu mentor e amigo Maurício Capobianco Lopes, por ter me guiado no início da minha carreira e acreditado no meu potencial profissional e acadêmico.

"Education is an admirable thing. But it is well to remember from time to time that nothing that is worth knowing can be taught". (Oscar Wilde)

Sumário

Lista de Figuras	3
Lista de Tabelas	4
Resumo	5
Abstract	6
Capítulo 1 – Introdução	7
1.1 Definição do problema	9
1.2 Objetivos	9
1.3 Metodologia	10
1.4 Escopo	11
1.5 Estrutura	12
Capítulo 2 – Revisão Teórica.....	14
2.1 Os Jogos de Empresas	14
2.2 O Jogo Líder.....	15
2.2.1 O modelo matemático.....	17
2.2.2 O sistema atual	19
2.3 Agentes.....	22
2.3.1 Sociedades de agentes	23
2.3.2 Classificação e Taxonomia.....	24
2.3.3 O meio ambiente.....	25
2.3.4 Sistemas que utilizam sociedades de agentes	26
2.4 A importância da modelagem de um sistema.....	27
2.5 A linguagem UML - <i>Unified Modeling Language</i>	29
2.5.1 Diagrama de Casos de Uso.....	30
2.5.2 Diagrama de Classes.....	30
2.5.3 Diagrama de Seqüência	31
2.5.4 Diagrama de Colaboração	32
2.5.5 Diagrama de Estados e Atividades	33
2.5.6 Diagrama de Pacotes	33
2.5.7 Diagrama de Componentes.....	34
2.6 A ferramenta <i>Rational Rose</i>	34
2.7 Considerações	35
Capítulo 3 – O Sistema proposto LIDER X	36

3.1	Visão geral	36
3.2	Estrutura	38
3.3	Os Agentes Colaboradores	40
3.3.1	A inteligência dos Agentes	42
3.3.2	Considerações sobre os agentes.....	44
3.4	Dinâmica de aplicação	44
3.5	Modelagem.....	46
3.5.1	Diagrama de Classes.....	60
3.6	Considerações	62
Capítulo 4 – Conclusões e Recomendações		63
4.1	Avaliação dos objetivos	63
4.2	Limitações	64
4.3	Sugestões.....	65

Lista de Figuras

Figura 2.1: Dinâmica da Aplicação do Jogo Líder.....	16
Figura 2.2: Modelo Matemático do jogo de empresas Líder - Fonte (NIVEIROS, 1998).....	18
Figura 2.3: Tela de decisões globais do sistema atual.....	20
Figura 2.4: Tela de decisões individuais do sistema atual.....	21
Figura 2.5: Tela de análise dos resultados do sistema atual.....	21
Figura 2.6: Exemplo de diagrama de casos de uso.....	30
Figura 2.7: Exemplo de diagrama de classes.....	31
Figura 2.8: Exemplo de diagrama de seqüência.....	32
Figura 2.9: Exemplo de diagrama de colaboração.....	33
Figura 2.10: Exemplo de diagrama de estado e atividade.....	33
Figura 2.11: Exemplo de diagrama de pacotes.....	34
Figura 2.12: Exemplo de diagrama de componentes.....	34
Figura 3.1: Estrutura para múltiplas aplicações do jogo simultâneas.....	38
Figura 3.2: Estrutura dos Jogos.....	39
Figura 3.3: Estrutura das empresas.....	40
Figura 3.4: Estrutura do agente colaborador.....	41
Figura 3.5: Fluxograma do comportamento do agente colaborador.....	43
Figura 3.6: Dinâmica de aplicação do sistema Líder X.....	45
Figura 3.7: Diagrama de casos de uso do animador.....	46
Figura 3.8: Diagrama de Colaboração - Acessar jogo.....	47
Figura 3.9: Diagrama de Seqüência - Acessar jogo.....	48
Figura 3.10: Diagrama de Colaboração - Avaliar empresas.....	49
Figura 3.11: Diagrama de Seqüência - Avaliar empresas.....	50
Figura 3.12: Diagrama de Colaboração – Definir variáveis do jogo.....	51
Figura 3.13: Diagrama de Seqüência – Definir variáveis do jogo.....	52
Figura 3.14: Diagrama de casos de uso do jogador.....	53
Figura 3.15: Diagrama de Colaboração - Acessar empresa.....	54
Figura 3.16: Diagrama de Seqüência - Acessar empresa.....	55
Figura 3.17: Diagrama de Colaboração - Tomar decisões globais.....	56
Figura 3.18: Diagrama de Colaboração - Consultar o perfil dos colaboradores.....	57
Figura 3.19: Diagrama de Colaboração – Demitir colaborador.....	58
Figura 3.20: Diagrama de Seqüência – Demitir colaborador.....	58
Figura 3.21: Diagrama de Colaboração - Consultar ranking do jogo.....	59
Figura 3.22: Diagrama de Seqüência - Consultar ranking do jogo.....	60
Figura 3.23: Diagrama de domínio.....	61

Lista de Tabelas

Tabela 2.1: Decisões a serem tomadas pelo participante - Fonte (NIVEIROS, 1998).....	17
Tabela 3.1: Comportamento do agente colaborador.....	42

Resumo

Esta pesquisa tem como objetivo o estudo e o desenvolvimento da modelagem de um novo sistema para a aplicação do jogo de empresas Líder, baseado em agentes inteligentes, provendo uma expansão do potencial didático e funcional desta ferramenta de treinamento em gestão de recursos humanos. A utilização de agentes, objetiva tornar o sistema mais próximo do modelo real, intensificando o realismo do ambiente simulado e conseqüentemente a vivência dos participantes do jogo. Para a apresentação deste modelo, adotou-se a linguagem UML - **Unified Modeling Language**.

Palavras-chave: Jogos de Empresas, Agentes Inteligentes, UML, Jogo Líder.

Abstract

The goal of this academic research is the study and development of a new computational system model to business game Líder, based on intelligent Agents, providing a potential didactic and functional increase of this training toll. The use of agent technology has as objective became the system closer the real model, increasing the realism of the simulated environment and the contact the players with the game. The language chosen to present this model was UML - **Unified Modeling Language**.

Key-words: Business Games, Intelligent Agents, UML, Game Líder.

Capítulo 1 – Introdução

Nas últimas quatro décadas, devido ao seu reconhecimento como ferramenta pedagógica, os jogos de empresas foram alvo de grandes aprimoramentos para sua viabilização como instrumento didático. Apesar de os sistemas computacionais que implementam estes jogos terem evoluído tecnologicamente, ainda possuem um grande caminho a percorrer.

Pelas tendências evolutivas, a realidade virtual (RV) imersiva é o grande delineador de um novo horizonte para os jogos e simuladores de maneira geral, por unificar três conceitos: imersão, interação e envolvimento (MORIE, 1994).

A idéia de imersão está ligada com o sentimento de se estar dentro de um ambiente. Normalmente, um sistema imersivo somente pode ser obtido com o uso de dispositivos específicos como capacete de visualização ou por meio de salas com projeções das visões nas paredes, teto e piso. Além do fator visual, existem outros dispositivos ligados com os outros sentidos que também são importantes para o sentimento de imersão, como o som e o tato. A visualização tridimensional por intermédio de um monitor não é considerada imersiva (MORIE, 1994).

O conceito de interação está relacionado com a capacidade de o sistema detectar as ações do usuário e modificar instantaneamente o mundo virtual de acordo as ações realizadas sobre ele, o que se denomina capacidade reativa.

A sensação de envolvimento, por sua vez, está ligada com o grau de motivação para o engajamento de uma pessoa com determinada atividade. Este envolvimento pode ser passivo ou ativo, e realidade virtual tem potencial para estimular ambos, permitindo a exploração de um ambiente virtual e propiciando a interação do usuário com este de uma forma dinâmica (MORIE, 1994).

Com exceção da imersão, que possui como requisito um sistema em RV e dispositivos específicos para sua obtenção, os demais conceitos podem ser atingidos por intermédio de sistemas computacionais tradicionais, ou até mesmo sem a utilização de ferramentas informatizadas.

Alguns sistemas computacionais desenvolvidos especificamente para a aplicação de determinados jogos de empresas já possuem uma parte destes conceitos incorporados, mesmo não sendo baseados em tecnologias de RV. O sistema atual para a aplicação do jogo líder permite que o jogador altere o ambiente do jogo (empresa virtual) de forma interativa por intermédio de determinadas ações (decisões). Estas alterações no ambiente são classificadas em dois níveis:

- **Objetos de interface:** Estas ações resultam em alterações visuais imediatas na interface do sistema.

Ex: Demissão de pessoal - No momento que o usuário confirma a demissão um determinado colaborador, sua referência visual na interface do sistema é removida imediatamente da tela do sistema, sugerindo que o mesmo retirou-se fisicamente do ambiente (empresa virtual).

- **Objetos de controle, processamento e persistência:** Ações que refletem no ambiente mais seu efeito não pode ser visualizado imediatamente a nível de interface.

Ex: Aumento de salário de um colaborador – o resultado desta ação só poderá ser visto por meio de relatórios financeiros ou que mostrem o efeito deste aumento no perfil do colaborador.

Para que um determinado jogo de empresas consiga obter um grau de envolvimento satisfatório dos participantes, é preciso que estes percebam que o sistema é capaz de recriar situações pelas quais eles poderiam passar no mundo real, e que com esta vivência podem aprender a lidar com eventos iguais ou similares, estando assim mais preparados para exercerem suas atividades administrativas (DETTMER, 2001).

Alem de interferir no envolvimento dos participantes, a implementação de um contexto irreal em um jogo de empresas pode comprometer sua validade como ferramenta de ensino empresarial (DETTMER, 2001).

1.1 Definição do problema

O primeiro sistema computacional para aplicação do jogo Líder foi desenvolvido em 1994 no Laboratório de Jogos de Empresas da Universidade Federal de Santa Catarina (UFSC). Sua finalidade era informatizar o processamento das decisões dos jogadores, que eram tomadas por meio de formulários de papel e depois digitadas no sistema. Desenvolvido na época para o sistema operacional MS-DOS, este sistema permitia também a persistência dos dados e a emissão de relatórios dos resultados processados.

De 1994 até hoje, novos sistemas computacionais para o jogo Líder foram desenvolvidos. Desconsiderando aspectos de atualização tecnológica como migração para plataforma Windows e utilização de banco de dados para persistência das informações, a única diferença entre eles e o primogênito, é que o formulário de papel foi substituído por um formulário na interface do sistema, eliminando assim o re-trabalho da digitação das decisões do papel para o sistema a cada período.

Nestes aproximadamente 10 anos, os novos sistemas que implementaram o modelo do jogo Líder foram apenas atualizações tecnológicas do primeiro, agregaram valor ao *software* mais não à ferramenta didática construída. A tecnologia não foi aplicada para intensificar a vivência do jogador e sua forma de interagir com o modelo do jogo, e muito menos para aumentar o grau de realismo da simulação.

A problemática desta pesquisa está centrada em alterar este quadro, em estudar e aplicar tecnologias para modelar um sistema computacional que incremente o grau de realismo do jogo de empresas Líder, proporcionando aos participantes uma maior sensação de interação com uma empresa real.

1.2 Objetivos

Esta pesquisa pretende incrementar o grau de realismo do jogo Líder, atuando especificamente no comportamento dos colaboradores, tanto a nível individual como coletivo, incorporando a eles, aspectos do comportamento humano que sejam relevantes aos objetivos e a dinâmica do jogo, e que sejam facilmente perceptíveis do ângulo do jogador.

Para atingir este objetivo, pretende-se aplicar técnicas de Inteligência Artificial ao contexto social do jogo Líder, visando incorporar determinados aspectos do modelo comportamental humano aos colaboradores virtuais da empresa.

Como objetivos específicos da pesquisa:

- a) Especificar aspectos comportamentais humanos específicos (individuais e ou coletivos) a serem incorporados aos colaboradores que compõem a estrutura do jogo Líder (Inteligência dos agentes);
- b) Especificar de que forma estes aspectos podem ser implementados nos colaboradores, sem alterar sua função matemática dentro da estrutura do jogo Líder descrita em MALDONADO (1990), SALVATIERRA (1990), LOPES (1994) e NIVEIROS (1998);
- c) Rever a dinâmica de aplicação atual do jogo, e adaptá-la ao novo contexto, se necessário;
- d) Apresentar a modelagem do sistema proposto por intermédio da linguagem UML.

1.3 Metodologia

A origem da pesquisa teve como condição básica, a vivência como analista e desenvolvedor de sistemas para aplicação do jogo Líder, e também como co-aplicador do jogo em disciplinas de graduação e pós-graduação das Universidades Regional de Blumenau (FURB) e Federal de Santa Catarina (UFSC) desde 1996.

Durante as aplicações do jogo, ao interagir diretamente com os jogadores, foi identificada a necessidade de retratar melhor o contexto empresarial simulado do jogo dentro do sistema, dar mais vida aos colaboradores, tornar as empresas virtuais mais parecidas com uma empresa real.

A partir desta idéia foram realizados estudos de técnicas de Inteligência Artificial, que poderiam ser aplicadas na estrutura do jogo para viabilizar este aprimoramento do sistema. Neste estudo, o conceito de Agentes Inteligentes pareceu compatível com a estrutura do jogo Líder que é baseada no perfil dos colaboradores que compõem as empresas.

Pela forte relação entre agentes e objetos, foi escolhida uma linguagem orientada a objetos para a modelagem do sistema proposto, a UML. Para suporte a UML, foi utilizada a ferramenta de *software Rational Rose 2003*, que oferece suporte a elaboração de todos os diagramas da linguagem.

O próximo passo foi avaliar a estrutura das versões atuais dos sistemas para a aplicação do Líder, a fim de levantar os requisitos para o novo modelo proposto. As seguintes versões foram consideradas para esta etapa:

- 6x (versão atual), aplicada na Universidade Regional de Blumenau (FURB) e na Universidade Federal de Santa Catarina (UFSC), em disciplinas de graduação e pós-graduação, considerada a versão mais estável e a única que está sendo aplicada efetivamente.
- 8/NT que ainda se encontra em fase de testes. Sua única aplicação até o momento foi em um grupo de estudo na Faculdade de Vitória (FDV). Por este motivo ela não foi considerada nesta pesquisa como versão atual. Foi desenvolvida com o objetivo de ser altamente portátil e viabilizar aplicações do jogo a distância

Após a identificação dos requisitos, com auxílio da ferramenta de *software Rational Rose*, os pontos analisados dos sistemas atuais foram mapeados para os diagramas de casos de uso da linguagem UML, a partir destes, os próximos diagramas foram sendo construídos e assim, sucessivamente o conjunto de diagramas que compõem a modelagem foi definido.

1.4 Escopo

Esta dissertação apresenta a modelagem de um novo sistema de *software* para a aplicação do jogo Líder, desenvolvida na linguagem UML. Para um maior entendimento do sistema apresentado, este volume inclui diagramas ilustrativos e textos descritivos complementares.

É importante mencionar que nem todos os diagramas disponibilizados pela linguagem UML serão adotados para ilustrar este sistema, conforme sugerido pelos próprios criadores da linguagem, apenas o conjunto necessário para o entendimento do mesmo será utilizado.

O entendimento completo do sistema proposto pode ser vislumbrado pela modelagem presente nesta pesquisa, devidamente complementada pelos textos originais que definem o modelo matemático do jogo.

Um dos requisitos para a compreensão e posterior aproveitamento do conteúdo apresentado nesta dissertação, é o entendimento sobre a unidade e independência existente entre o modelo (princípio) matemático do jogo e o sistema computacional que o implementa. O modelo matemático especifica as variáveis que compõem o jogo, quais devem ser manipuladas e ou persistidas, os cálculos que as afetam e a seqüência em que estes devem ser realizados. O sistema computacional é uma mera ferramenta que automatiza a execução dos cálculos, regras e procedimentos especificados, por intermédio de uma lógica programada previamente.

A modelagem apresentada é específica do sistema computacional, não contemplando os cálculos matemáticos do jogo Líder. Este conteúdo precisa ser complementado pelo entendimento do modelo matemático, para que todas as informações necessárias para a implementação de um sistema para aplicação do jogo Líder, baseado no modelo proposto, estejam presentes.

Nesta dissertação não será detalhado o modelo matemático do jogo Líder, que pode ser encontrado na íntegra em MALDONADO, SALVATIERRA (1990), LOPES (1994) e NIVEIROS (1998).

Nenhum protótipo de interface é disponibilizado, uma vez que a modelagem de um sistema é uma especificação formal e que pode ser desenvolvida em qualquer linguagem ou plataforma computacional.

1.5 Estrutura

A dissertação esta dividida em quatro capítulos.

O primeiro capítulo permite uma contextualização sobre o trabalho, apresentando a justificativa para seu desenvolvimento, o problema, seus objetivos e seu escopo.

No segundo capítulo é apresentada a revisão teórica que inclui um descritivo do jogo de empresas Líder, como ferramenta de treinamento em gestão de recursos humanos, o

sistema computacional atualmente utilizado para sua aplicação, um posicionamento sobre as tecnologias aplicadas ao sistema modelado (agentes inteligentes e sociedades de agentes) e apresenta a linguagem UML, descrevendo seus principais diagramas e a ferramenta *Rational Rose*, utilizada como suporte a linguagem.

O terceiro capítulo descreve a estrutura do sistema proposto e disponibiliza a modelagem desenvolvida na linguagem UML.

O quarto capítulo finaliza a dissertação apresentando as conclusões, limitações e sugestões para o aprimoramento da pesquisa.

Capítulo 2 – Revisão Teórica

Este capítulo apresenta os jogos de empresas como estratégia para desenvolver a aprendizagem gerencial, dando ênfase ao Líder que modela a ciência comportamental (sociologia e psicologia) por meio de uma ciência exata (matemática) implementada em um sistema computacional, as tecnologias utilizadas no desenvolvimento da pesquisa (agentes e sociedades de agentes) e a linguagem UML utilizada para a elaboração da modelagem do sistema.

O conteúdo apresentado neste capítulo é de extrema relevância para o entendimento da pesquisa, não somente por expor um descritivo e um exemplo de cada diagrama da UML utilizado para a modelagem apresentada no capítulo 4, mas também por contextualizar o próprio jogo Líder.

2.1 Os Jogos de Empresas

Há mais de 40 anos os jogos de empresas em computadores conquistaram com êxito o seu espaço incontestável entre os métodos de ensino. A grande vantagem dos jogos de empresas como instrumento didático e de treinamento é a sua capacidade de acelerar o tempo real, sendo que por intermédio da simulação se comprimem, em poucos dias, vários anos de experiência, oferecendo assim, um preparo a funcionários e estudantes para suas atividades profissionais futuras (NIVEIROS, 1998).

Como método de ensino, tem-se observado que nenhum outro permite simular situações de decisões tão interessantes, com tamanha participação e interesse dos alunos. Segundo estudo mencionado por NIVEIROS (1998), 90% das equipes participantes do Líder considerou a experiência positiva e com elevado nível de comprometimento.

A aplicação de jogos de empresas como instrumento de treinamento exige, dos participantes, a prática da arte do planejamento, bem como exerce e desenvolve a habilidade da tomada de decisão em nível de alta e média gerência/ administração (LOPES, 1994).

Um histórico sobre a evolução dos jogos de empresas, bem como suas características e vantagens podem ser vistas em MALDONADO (1990), SALVATIERRA (1990), LOPES (1994) e NIVEIROS (1998).

2.2 O Jogo Líder

O jogo de empresas Líder apresenta, como principal finalidade, propiciar aos participantes, um ambiente empresarial hipotético para estimular o aprendizado ou o aprimoramento das habilidades gerenciais nos recursos humanos, fundamentalmente no que diz respeito à motivação e liderança.

Este jogo de empresas surgiu preenchendo a lacuna existente na área de desenvolvimento de recursos humanos utilizando-se dos recursos da informática. Os simuladores disponíveis no mercado nacional nesta área, antes do Líder, eram de caráter manual e eram aplicados pro intermédio de formulários e tabelas. Eram simuladores simples e concentravam-se num só aspecto e, possivelmente por serem manuais e não computacionais, mostravam a escassez de variáveis interagindo nos modelos (NIVEIROS, 1998).

O objetivo do Líder consiste em oferecer a oportunidade de experimentar a aplicação prática da teoria comportamental. Sua intenção é aplicar, principalmente, os conceitos da teoria e técnicas da Liderança Situacional descritas por Hersey e Blanchard, a hierarquia das necessidades de Maslow e a teoria de motivação e higiene de Herzberg, transformando-os de teóricos e descritivos em práticos e prescritivos (NIVEIROS, 1998).

A aplicação do jogo de empresas Líder não se resume apenas na utilização de um sistema computacional. Normalmente uma entidade externa ao sistema, um coordenador denominado Animador, geralmente o professor da disciplina, apresenta o sistema, sua finalidade e a forma correta de utilização.

Após uma apresentação e conceituação do simulador aos participantes (processo não computacional), os animadores definem as empresas fictícias, escolhendo seu quadro de colaboradores, também fictícios, previamente definidos dentro do sistema. Os colaboradores são dispostos em um organograma hierárquico empresarial.

O jogo, deste ponto em diante, consiste basicamente em tomadas de decisões sucessivas sobre os colaboradores que lhe são subordinados. Estas decisões são informadas no sistema computacional que, após a simulação, fornece as conseqüências das mesmas aos participantes. Os resultados servir-lhes-ão de base para a tomada de novas decisões. (figura 2.1).

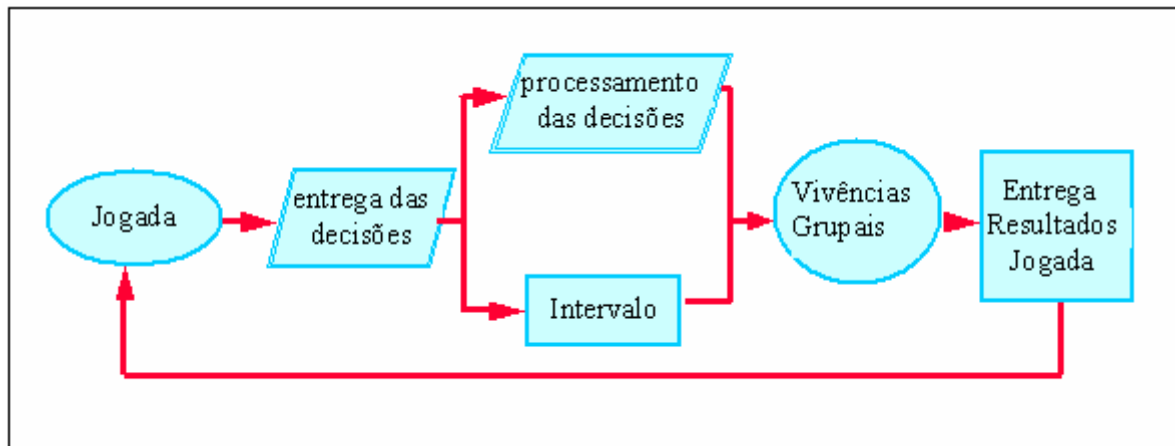


Figura 2.1: Dinâmica da Aplicação do Jogo Líder.

No jogo, assim como na prática, o clima organizacional depende de um conjunto de variáveis, consideradas entradas do sistema. Estes fatores influenciam o comportamento das pessoas, provocando motivação entre os colaboradores, diferentes níveis de satisfação e produtividade, e estimulando-os para produzirem o resultado final em termos de eficiência (NIVEIROS, 1998).

As variáveis dependentes constituem o clima organizacional da empresa. Na medida em que as variáveis de entrada produzirem influência positiva nas variáveis dependentes, maior será a eficiência dos colaboradores, quer dizer a capacidade dos mesmos de fazer a tarefa corretamente. Assim como, quanto mais influência negativa as variáveis dependentes apresentarem em conseqüência das variáveis de entrada, menor será a eficiência dos colaboradores (LOPES, 1994).

Este processo define um comportamento para os colaboradores, e conseqüentemente um comportamento para a empresa. Se os colaboradores forem menos eficientes ou estiverem menos motivados, a produção da empresa tende a diminuir, abalando, conseqüentemente, a situação financeira da mesma.

Os participantes podem criar e desenvolver climas organizacionais, modificando seu estilo gerencial, na forma de administrar pessoas, no projeto de trabalho, no sistema de motivação, no treinamento de sua equipe, no seu estilo de liderança, no sistema de remuneração, etc. Os participantes decidem, a partir de um amplo leque de possibilidades (incentivos), por aqueles que julgarem mais capazes de atender às diversas necessidades de seus colaboradores. Determinam ainda, níveis de desempenho de produção desejáveis e os estilos de poder e de liderança (LOPES, 1994), (NIVEIROS, 1998) (Tabela 2.1).

DECISÕES SOBRE A EMPRESA EM NÍVEL GLOBAL	DECISÕES SOBRE OS FUNCIONÁRIOS EM NÍVEL INDIVIDUAL
Alimentação Melhoria ambiental e ergonômica Consultoria de design para o trabalho Lanches Intervalos de descanso Plano de saúde Redução no horário de trabalho Reuniões informais Promoções esportivas por funcionário	Metas de produção Estilo de liderança Base de poder Locação de pessoal Treinamento antes de uma promoção Treinamento específico para um trabalho Treinamento de Liderança Relatório do perfil das necessidades e maturidades Aumento Salarial Prêmio

Tabela 2.1: Decisões a serem tomadas pelo participante - Fonte (NIVEIROS, 1998).

2.2.1 O modelo matemático

Este item apresenta a estrutura do modelo matemático do jogo de empresas Líder, ou seja, na forma em que os diversos cálculos estão interligados e hierarquicamente posicionados para determinar a produção real individual de cada colaborador pertencente à empresa.

O modelo atualmente em vigor difere bastante do definido nos trabalhos de SALVATIERRA (1990) e MALDONADO (1990), mentores intelectuais da primeira versão do jogo de empresas Líder. As sucessivas melhorias e aperfeiçoamentos propostos por LOPES (1994) e NIVEIROS (1998) aos modelos originais, resultaram na seguinte estrutura para o modelo, definido na figura 2.2.

É importante ressaltar que cada fator, representado na estrutura da figura 2.2, representa um determinado conjunto de cálculos e procedimentos descritos no modelo matemático do Líder. Em NIVEIROS (1998), existem diagramas que representam detalhadamente alguns destes fatores, provendo um entendimento mais detalhado sobre o cálculo de cada fator.

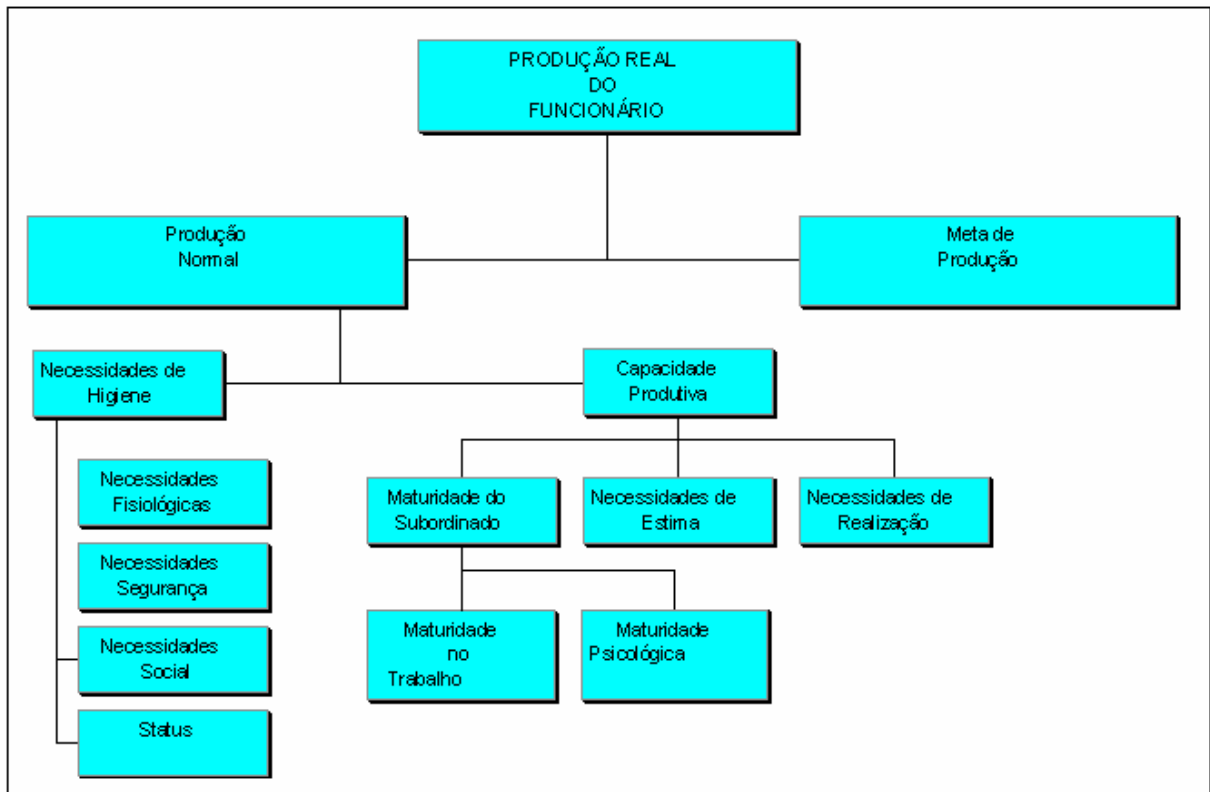


Figura 2.2: Modelo Matemático do jogo de empresas Líder - Fonte (NIVEIROS, 1998)..

A estrutura da figura 2.2 representa graficamente a hierarquia dos fatores que influenciam no cálculo da produção final do colaborador, ou seja, cada fator calculado é utilizado diretamente no cálculo do fator posicionado hierarquicamente acima, e assim sucessivamente. Esta estrutura ilustra hierarquicamente o impacto de um fator no cálculo de outros fatores que determinam o perfil dos colaboradores do Líder. A seguir é apresentado um exemplo de como é feito o cálculo da produção normal de um colaborador.

A produção normal é definida em função de dois fatores:

1. A capacidade produtiva que está diretamente relacionada com os níveis de maturidade psicológica e do trabalho, e o grau de necessidade motivacional (estima e realização) do funcionário em questão;

2. O grau de satisfação das necessidades de higiene (fisiológicas, segurança, social e status).

Assim: $PN = NH \times CP$

Onde:

- PN é a produção normal do funcionário,
- NH é o grau de necessidades de higiene, e;
- CP é a capacidade produtiva do funcionário.

É importante destacar que nem todos os cálculos são de simples multiplicações. A descrição detalhada sobre o modelo matemático pode ser encontrada em MALDONADO (1990), SALVATIERRA (1990), LOPES (1994) e NIVEIROS (1998).

2.2.2 O sistema atual

Desde sua primeira especificação, por meio de fluxogramas, idealizada por SALVATIERRA e MALDONADO (1990), o Líder sofreu uma série de alterações, algumas destas necessárias para acompanharem as evoluções no modelo matemático.

A primeira implementação em linguagem computacional foi proposta e implementada na Universidade Federal de Santa Catarina por LOPES (1994). Esta primeira versão do sistema, desenvolvida em linguagem computacional Pascal, utilizava arquivos binários para armazenamento das informações relevantes à simulação, permitindo, assim, o salvamento do estado das empresas para uma eventual retomada de contexto, possibilitando a continuidade da simulação. Esta versão não era destinada aos participantes do jogo, os quais tomavam decisões por intermédio de uma ficha preenchida manualmente em papel. O sistema era utilizado pelo animador, o qual introduzia as informações dos participantes no sistema para obter o processamento das decisões, e conseqüentemente avaliar os resultados.

Pesquisas realizadas na Universidade Regional de Blumenau, no sentido de melhorar o sistema atualmente em vigor, transportaram-no para a plataforma Windows, utilizando o ambiente de desenvolvimento Delphi, a linguagem computacional Object Pascal e substituindo os arquivos binários por Bancos de Dados.

Diferente da primeira, esta versão foi desenvolvida para ser utilizada tanto pelo Animador (responsável pela aplicação do jogo) quanto pelos participantes do jogo, ou seja, estes últimos informam suas decisões e verificam os resultados diretamente no sistema, e o Animador só precisa processar as decisões e gerenciar a liberação do próximo período decisório.

Assim como o primeiro, este sistema é composto basicamente por três telas:

- a) Tomada de decisões globais (figura 2.3), onde o participante seleciona quais opções irá aplicar, dentre as disponíveis, a todos os colaboradores da empresa. Para aplicar o item desejado, basta selecioná-lo, pressionando o botão direito do mouse sobre o mesmo, deixando o quadrado branco com um sinal semelhante ao da opção “Job Design” para Setor A;

Figura 2.3: Tela de decisões globais do sistema atual

- b) Tomada de decisões individuais, onde o participante pode aplicar as opções disponíveis (figura 2.4) a cada colaborador individualmente. Neste caso o nome

de cada colaborador é apresentado, seguido de algumas informações atuais (idade, salário, produção) que não podem ser alteradas pelo usuário, e outras que definem as decisões sobre ele (novo cargo, novo setor, aumento de salário, etc);

Globais		Individuais									
Nome	Idade	Salário	Prod.	Novo Cargo	Novo Setor	Aum. Salário	Meta Prod.	Estilo	Poder	Prêmio	
ALMEIDA	48	500	1,27	Inspetor	-	0	0	1	1	0	
BEATRIZ	38	300	0,99	Supervisor	-	0	0	1	1	0	
CLAUDIO	38	200	0,79	Chefe	A	0	0	1	1	0	
DARIO	38	200	0,62	Chefe	A	0	0	1	1	0	
GERALDO	28	100	93	Operário	A	0	69	1	1	0	
HERMES	28	100	56	Operário	A	0	33	1	1	0	
LUIS	28	100	102	Operário	A	0	82	1	1	0	
TADEU	28	100	76	Operário	B	0	53	1	1	0	
VALDO	28	100	34	Operário	B	0	12	1	1	0	
ZINHO	28	100	84	Operário	B	0	74	1	1	0	

Figura 2.4: Tela de decisões individuais do sistema atual

- c) Análise dos resultados, (figura 2.5) que permite uma verificação dos efeitos decorrentes das decisões tomadas no período anterior. Nesta área podem ser verificados o perfil dos funcionários, suas situações perturbadoras e o lucro da empresa. Os resultados são obtidos a partir do processamento das decisões.

Decisões		Relatórios													
Perfil		Perturbadoras										Lucro			
Nome	Idade	Cargo	Setor	Salário	Prod. Real	FIS	SEG	SOC	EST	REA	PSI	TRA	Receita	Despesas	Lucro
ALMEIDA	48	Inspetor	-	500	0,93	-	-	-	-	-	-	-	R\$ 0	R\$ 500	(R\$ 500)
BEATRIZ	38	Supervisor	-	300	0,99	-	-	-	-	-	-	-	R\$ 0	R\$ 300	(R\$ 300)
CLAUDIO	38	Chefe	A	200	0,79	-	-	-	-	-	-	-	R\$ 0	R\$ 200	(R\$ 200)
DARIO	38	Chefe	B	200	0,62	-	-	-	-	-	-	-	R\$ 0	R\$ 200	(R\$ 200)
GERALDO	28	Operário	A	100	93	-	-	-	-	-	-	-	R\$ 326	R\$ 100	R\$ 226
HERMES	28	Operário	A	100	56	-	-	-	-	-	-	-	R\$ 196	R\$ 100	R\$ 96
LUIS	28	Operário	A	100	102	-	-	-	-	-	-	-	R\$ 357	R\$ 100	R\$ 257
TADEU	28	Operário	B	100	76	-	-	-	-	-	-	-	R\$ 266	R\$ 100	R\$ 166
VALDO	28	Operário	B	100	34	-	-	-	-	-	-	-	R\$ 119	R\$ 100	R\$ 19
ZINHO	28	Operário	B	100	84	-	-	-	-	-	-	-	R\$ 294	R\$ 100	R\$ 194

Figura 2.5: Tela de análise dos resultados do sistema atual

2.3 Agentes

Neste tópico são expostas algumas definições de agentes inteligentes, com o objetivo de expor uma idéia do que são agentes inteligentes, e o que se pretende ao utilizar tais estruturas em um determinado sistema.

Para Russell e Norvig (1995, página 33), “um agente é algo que pode ser visto como uma estrutura que monitora seu ambiente por intermédio de sensores e age sobre ele por meio de atuadores”.

O agente de Maes (1995, página 108) “é um sistema computacional imerso em um ambiente dinâmico complexo, capaz de atuar neste ambiente de forma autônoma de acordo com suas metas, e realizar tarefas para as quais foi programado”. Pattie Maes, do Laboratório de Mídia de MIT, é um dos pioneiros de pesquisa de agentes. Ela acrescenta um elemento crucial à definição um agente: agentes têm que agir de forma autônoma para atingir suas metas. Também restringe o ambiente a complexo e dinâmico.

Smith (1994) define agente como uma entidade de software persistente dedicada para um propósito específico. Persistente distingue agentes de sub-rotinas; agentes têm suas próprias idéias de como efetuar tarefas.

Os agentes de Hayes-Roth (1995) executam três funções continuamente: percebem as condições dinâmicas no ambiente; atuam para alterar condições no ambiente; e pensam para interpretar suas percepções, resolver os problemas, tirar conclusões, e determinarem suas ações. Barbara Hayes-Roth do Laboratório de Sistemas de Conhecimento de Stanford insiste que os agentes argumentem durante o processo de seleção de ação. Se argumentando é interpretado amplamente, a arquitetura de agente dela permite ações reflexivas como também planejou ações.

Wooldridge e Jennings (1998) definem um agente como um sistema de computador baseado em *hardware* ou *software* que desfruta as propriedades de: autonomia, capacidade social, reatividade e pró-atividade. Este conceito amplia o conceito de agentes para entidades de *hardware*, considerando além de sua autonomia e pró-atividade, sua capacidade social.

Conforme muitas das definições mencionam, os agentes possuem uma relação forte com o meio em que estão imersos, uma outra linha da IA, considera os agentes de maneira coletiva, como sociedades. As sociedades de agentes são idealizações que consideram tanto os aspectos e comportamentos individuais quanto à interação entre diversos agentes, como determinantes da inteligência de um determinado sistema.

2.3.1 Sociedades de agentes

Uma sociedade de agentes pode ser definida, numa perspectiva muito abrangente, como uma tripla (**A, M, L**) com um conjunto de agentes (**A**) que compartilham um conjunto de meios (**M**) e um conjunto de linguagens (**L**) que permitem comunicação entre eles. Não existe, nesta definição, nenhuma necessidade de se estabelecer alguma restrição como a homogeneidade de conhecimento, estrutura ou habilidades. Uma sociedade de agentes pode ser estabelecida com agentes que compartilhem uma única linguagem, pela qual estabeleçam uma forma de comunicação.

Os agentes podem ser classificados em homogêneos e heterogêneos. Os agentes homogêneos possuem as mesmas habilidades, conhecimentos e a mesma estrutura. No mundo real o que mais observamos, entretanto, são agentes heterogêneos. Portanto, qualquer processo de interação entre agentes, que modele algum aspecto do mundo real, deve considerar agentes heterogêneos.

Moulin & Chaib-Draa classificam a heterogeneidade em sociedades de agentes em três níveis:

- **baixa heterogeneidade** no caso que os agentes difiram apenas nos recursos disponíveis;
- **média heterogeneidade** no caso que os agentes difiram nos métodos de resolução de problemas;
- **alta heterogeneidade** no caso que os agentes compartilham apenas a linguagem de interação.

Esta noção de heterogeneidade pode ser entendida como uma medida do grau de diferença estrutural dos agentes em uma sociedade (JENNINGS, 1998).

2.3.2 Classificação e Taxonomia

Em um contexto social de agentes, onde a heterogeneidade pode estar presente, definir algum tipo de classificação é uma forma de facilitar a compreensão de quais são as possibilidades e limites desta tecnologia. Uma possível classificação dos agentes permite, por exemplo, estabelecer que agentes possam ser ou não mais adequados para determinadas tarefas ou objetivos.

A classificação de agentes é geralmente tomada como uma tipificação baseada em um conjunto de características observáveis. Vários autores as fazem, considerando características diferentes:

Wooldridge e Jennings (1997) em sua definição de agentes listam as seguintes características:

- **reatividade** é perceber o ambiente e responder de maneira oportuna (*timely fashion*) a mudanças que acontecerem nele;
- **autonomia** é a capacidade de operar sem intervenção direta de humanos ou outros e ter alguma espécie de controle sobre suas ações e estado interno;
- **habilidade social** é a capacidade de interagir com outros agentes via algum tipo de linguagem de comunicação de agentes;
- **pró-atividade** é a capacidade de exibir comportamentos baseados em objetivos.

Brustoloni apud (JENNINGS, 1998) propõe uma classificação dos agentes em três tipos:

- **agentes reguladores** que reagem a estímulos e com um conjunto de conhecimentos e ações pré-determinados;
- **agentes planejadores** que podem estabelecer planos para atingir as suas metas;

- **agentes adaptativos** que, além da capacidade de planejar, têm condições de adquirir novos conhecimentos sobre o domínio onde atuam e aumentar a sua gama de habilidades.

Russel e Norvig (1995) classificam os programas de agentes em quatro tipos:

- **agentes reflexos simples**, que reagem ao estado do mundo apenas a partir de um conjunto de regras (condição-ação) pré-determinadas;
- **agentes que acompanham o mundo** são agentes que além de um conjunto de regras, guardam algum estado interno que dado o estado do mundo, influencia a ação a tomar;
- **agentes baseados em metas** são aqueles capazes de armazenar algum tipo de meta em sua memória e estabelecer as suas ações a partir de algum tipo de planejamento que lhe permite buscar uma ação ou conjunto de ações para alcançar as suas metas;
- **agentes baseados em utilidade** são aqueles que agem no sentido de aumentar o seu grau de felicidade ou satisfação. Mais que metas, os agentes procuram maximizar alguma função utilidade explícita. As decisões dos agentes levarão sempre em conta como poderão estar a sua satisfação dada uma determinada trajetória de evolução do mundo e uma escolha de ação feita por eles.

Algumas das classificações mencionadas anteriormente, apresentam um elemento em comum, a tipificação dos agentes a partir de determinadas características qualitativas. Estas classificações geralmente apresentam alguma forma de ordenação que vai dos agentes mais simples (os agentes **reguladores**, **reativos** ou **reflexos**), até os mais complexos (agentes **adaptativos**, **baseados em utilidade e sociais**).

2.3.3 O meio ambiente

Quanto ao meio ambiente em que o agente estará inserido, é importante que se definam os tipos que podem existir, pois isto interfere diretamente na complexidade do agente em questão.

Norvig e Russel (1995) definem as propriedades do meio ambiente como:

- **inacessível X acessível:** um ambiente é acessível ao agente se seus aparatos são capazes de perceber o estado completo deste ambiente;
- **determinístico X não-determinístico:** um ambiente determinístico é aquele cujo próximo estado é completamente determinado considerando o estado atual e as ações selecionadas pelo agente;
- **episódico X não-episódico:** num ambiente episódico, as ações dos agentes são divididas em episódios. Cada episódio consiste em o agente perceber e agir, e a qualidade da ação depende do próprio episódio, não dependendo do que aconteceu em episódios anteriores;
- **estático X dinâmico:** um ambiente é dinâmico para o agente se seu estado pode mudar enquanto o agente está deliberando;
- **discreto X contínuo:** um ambiente é discreto se houver um número limitado de percepções e ações claras e distintas.

2.3.4 Sistemas que utilizam sociedades de agentes

Jennings (1998), cita exemplos de sistemas baseados em sociedades de agentes, utilizados para controle de tráfego aéreo, tráfego de veículos, controles de produção, gerenciamento de documentos e serviços financeiros. Um destes exemplos é o OASIS, sistema de controle de tráfego aéreo, que modela inclusive considerações sobre fenômenos meteorológicos e trajetórias de pouso onde haja maximização na economia de combustível das aeronaves.

Outro exemplo é o sistema SWARMM, sistema utilizado pelo *Royal Australian Air Force* no treinamento em técnicas de combate aéreo, inclui uma análise sobre a técnica utilizada pelo piloto durante o treinamento e avalia o desempenho dos mesmos.

2.4 A importância da modelagem de um sistema

BOOCH, RUMBAUGH e JACOBSON (1998) exemplificam a importância de modelar da seguinte maneira. Se você deseja construir uma casa de cachorro, pode começar com uma pilha de madeira, pregos e algumas ferramentas básicas, como um martelo, uma serra e uma fita métrica. Em algumas horas você provavelmente terá uma casa de cachorro que é razoavelmente funcional. Contanto que seja grande bastante e não chova dentro, seu cachorro vai ficar contente. Se não funcionar, você sempre pode tirar um fim de semana e recomeçar do zero, ou adquirir um cachorro menos exigente.

Agora imagine que você quiser construir uma casa para sua família, também é possível começar com uma pilha de madeira, pregos e algumas ferramentas básicas, mas tenha em mente que vai levar muito mais tempo que construir a casinha de cachorro, e lembre-se também, que a família certamente é mais exigente que o cachorro. Neste caso, a menos que já tenha alguma experiência, será mais bem sucedido fazendo algum planejamento antes de dar a primeira martelada, ou pelo menos, fazendo alguns esboços de como quer a casa, para poder ter uma idéia do que é pretendido. Se seu desejo é construir uma casa de qualidade que satisfaz as necessidades de sua família, outros fatores precisam ser considerados, quantidade de cômodos e outros pontos importantes como sistemas elétricos e hidráulicos. Dados estes planos, é possível começar a fazer estimativas razoáveis de tempo e materiais que este trabalho vai requerer. Embora seja possível construir uma casa sozinho, você achará muito mais eficiente trabalhar com ajuda de outras pessoas e possivelmente adquirindo muitos materiais já prontos. Contanto que você fique dentro dos limites de tempo e dinheiro, e obtenha o resultado pretendido, sua família provavelmente ficará satisfeita. Caso não tenha que trabalhar fora durante a semana, se o seu projeto não der certo, sempre é possível recomeçar, ou claro, procurar uma família menos exigente.

Entretanto, se planeja construir um grande edifício comercial, seria infinitamente estúpido começar com uma pilha de madeira, pregos e algumas ferramentas básicas. Porque neste caso, é necessário atender as necessidades de uma grande quantidade de pessoas, e provavelmente investir muito dinheiro. Dependendo do seu relacionamento com estas pessoas, elas vão querer participar das definições da obra, como tamanho, forma, estilo, cor, etc. É quase certo, que estas pessoas vão querer alterar suas próprias definições anteriores de forma aleatória, mesmo depois que inicie a construção do edifício. Neste caso a única opção é

um planejamento extenso, porque o custo de fracasso é alto. Neste caso, existe um grupo onde todos precisam entender da mesma forma as definições, inclusive para que se comuniquem em uma linguagem única. Contudo que você adquira as pessoas certas e as ferramentas certas e supervisione ativamente o processo de transformar um conceito arquitetônico em realidade, provavelmente o resultado será um edifício que satisfará a todos os envolvidos.

Modelar é uma técnica de engenharia provada e bem-aceita. São construídos modelos arquitetônicos de casas e edifícios altos para ajudar os usuários a visualizarem o produto final. Podemos construir até mesmo modelos matemáticos para analisar os efeitos de ventos ou terremotos em nossos edifícios (BOOCH, RUMBAUGH e JACOBSON, 1998).

O conceito de modelar, não é só uma parte da engenharia civil, seria inconcebível desenvolver uma nova aeronave ou um automóvel sem primeiro testá-lo por meio de moldes em computadores, ou por intermédio de modelos sólidos e submetidos a túneis de vento, por exemplo. Na indústria do cinema, são utilizados os *storyboards* como uma forma de modelar, nos campos de sociologia, economia e administração, existem modelos para validar teorias e para minimizar os riscos de investimentos.

Um modelo é uma simplificação de realidade, provê um raio-x de um sistema. Modelos podem contemplar planos detalhados e amplos, todo sistema pode ser descrito de aspectos diferentes, e por intermédio de modelos diferentes. Um modelo pode ser estrutural, enquanto enfatizando a organização do sistema, ou comportamental quando enfatizando a dinâmica do sistema (BOOCH, RUMBAUGH e JACOBSON, 1998).

Há uma razão fundamental para modelar, construímos modelos para entendermos melhor os sistemas que desenvolvemos. Os modelos nos ajudam a:

- visualizar um sistema como é ou como queremos que seja.
- especificar a estrutura ou comportamento de um sistema.
- nos guiar ou mapear para a construção de um sistema.
- documentar decisões que tomamos sobre o sistema.

Modelar não é só para sistemas grandes, até mesmo o *software* equivalente a uma casa de cachorro pode ter benefícios se modelado. Porém, definitivamente quanto maior e mais

complexo o sistema, mais importante se torna sua modelagem, por uma razão muito simples, construímos modelos de sistemas complexos porque não podemos compreender tal sistema em sua totalidade, há limites à habilidade humana para entender a complexidade.

Por modelar, estreitamos o problema que estamos estudando focalizando um só aspecto de cada vez. Isto é essencialmente a aproximação de "dividir para conquistar", o que Edsger Dijkstra queria dizer com “ataque um problema complexo dividindo isto em uma série de problemas menores que você pode resolver”.

2.5 A linguagem UML - *Unified Modeling Language*

A utilização de uma metodologia orientada a objetos é uma consequência da ampla relação existente entre agentes e objetos, defendida por quase todos os autores relacionados anteriormente como Norvig (1995) e Wooldridge (1998).

Dentre as diversas tecnologias para a modelagem de sistemas orientados a objetos existentes, a adotada neste trabalho é a UML (FURLAN, 1998). A UML oferece mecanismos de suporte a modelos estáticos, dinâmicos e funcionais, uma vez que todos os sistemas possuem uma estrutura estática e um comportamento dinâmico.

A UML (*Unified Modeling Language*) é uma linguagem para especificação, documentação, visualização e desenvolvimento de sistemas orientados a objetos. Sintetiza os principais métodos existentes, sendo considerada uma das linguagens mais expressivas para modelagem de sistemas orientados a objetos. Por meio de seus diagramas é possível representar sistemas de *softwares* sob diversas perspectivas de visualização. Facilita a comunicação de todas as pessoas envolvidas no processo de desenvolvimento de um sistema - gerentes, coordenadores, analistas, desenvolvedores - por apresentar um vocabulário de fácil entendimento. É uma linguagem de diagramação ou notação e não um método de desenvolvimento, o que significa que ele não diz para você o que fazer primeiro e em seguida ou como desenhar seu sistema, mas ele lhe auxilia a visualizar seu desenho e a comunicação entre os objetos. A UML é controlada pelo Grupo de Gerenciamento de Objeto (*Object Management Group* - OMG) e é um padrão da indústria para descrever graficamente *software* (FURLAN, 1998).

2.5.1 Diagrama de Casos de Uso

Um caso de uso descreve um objetivo que um ator externo ao sistema tem com o sistema. Um ator pode ser um elemento humano ou não que interage com o sistema. O ator se encontra fora do escopo de atuação do sistema, e o conjunto dos casos de uso, formam o escopo do sistema. Opcionalmente uma linha separa os atores dos casos de uso, representando a fronteira do sistema.

O diagrama de caso de uso representa graficamente esta interação, definindo o contexto do sistema. Os atores são representados por representações simplificadas de uma figura humana, enquanto os casos de uso são elipses contendo cada uma o nome de um caso de uso. Os atores se comunicam com os casos de uso, que é representado por uma linha unindo os dois elementos. Uma seta pode, opcionalmente, representar o fluxo principal de informação nesta interação e ajudar a leitura do caso de uso (FURLAN, 1998).

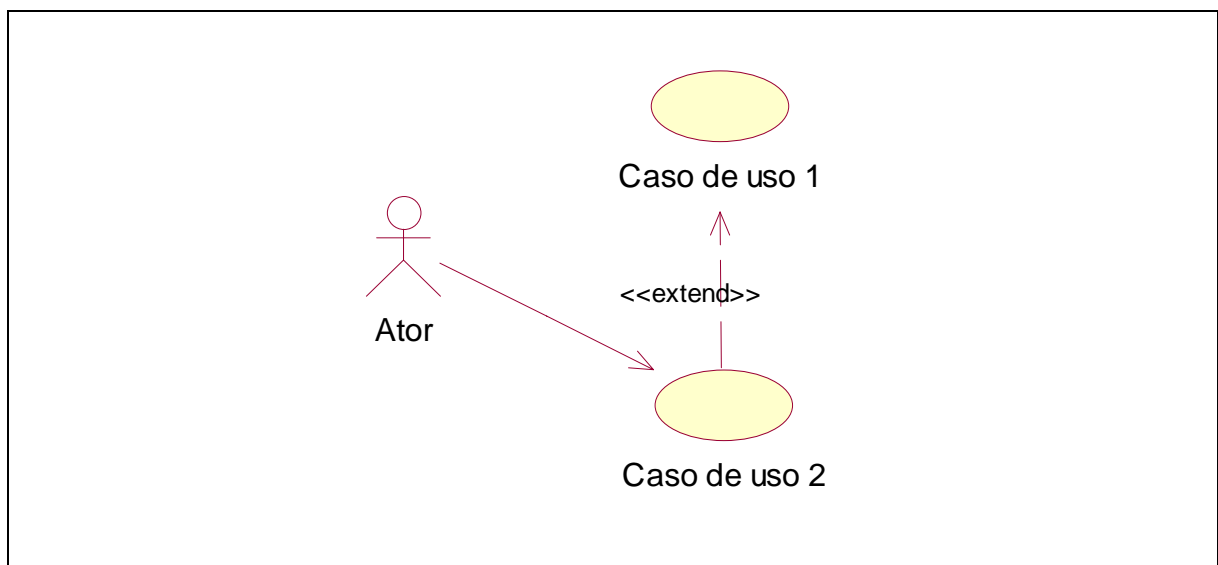


Figura 2.6: Exemplo de diagrama de casos de uso.

2.5.2 Diagrama de Classes

O diagrama de classes descreve as classes que formam a estrutura do sistema e suas relações. As classes possuem além de um nome, os atributos e as operações que desempenham para o sistema. Uma relação indica um tipo de dependência entre as classes,

essa dependência pode ser forte ou fraca, mas todas indicam que as classes relacionadas cooperam de alguma forma para cumprir um objetivo para o sistema (FURLAN, 1998).

Sendo uma linguagem de descrição, a UML permite diferentes níveis de abstração aos diagramas, dependendo da etapa do desenvolvimento do sistema em que se encontram. Assim, os diagramas de classe podem exibir nas fases iniciais da análise apenas o nome das classes, e em uma fase seguinte os atributos e operações. Finalmente, em uma fase avançada do projeto, pode exibir os tipos dos atributos, a visibilidade, a multiplicidade das relações e diversas restrições. Existem elementos na UML para todas estas representações.

O diagrama de classes, ao final do processo de modelagem, pode ser traduzido em uma estrutura de código que servirá de base para a implementação do sistema. Observa-se, no entanto, que não existe no diagrama de classes uma informação sobre os algoritmos que serão utilizados para as implementações das operações definidas nas classes. Neste diagrama, não se pode também precisar a dinâmica do sistema, porque não há elementos sobre o processo ou a seqüência de processamento das operações (FURLAN, 1998).

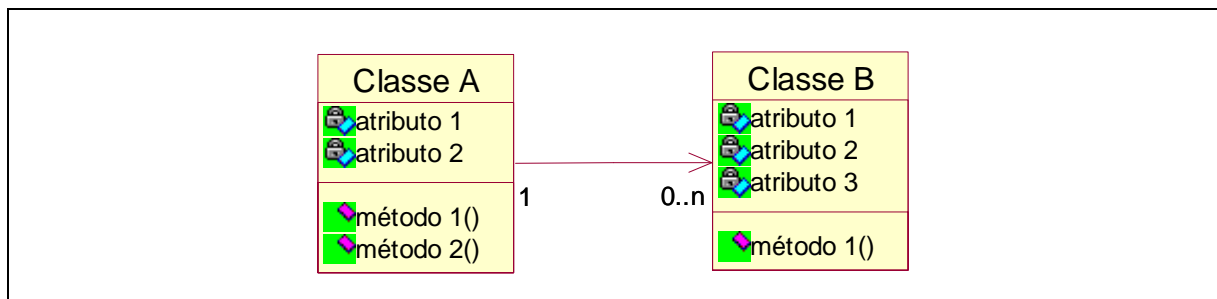


Figura 2.7: Exemplo de diagrama de classes.

2.5.3 Diagrama de Seqüência

Os casos de uso, descritos anteriormente, representam conjunto de cenários que descrevem os diferentes processos que o sistema atende. O diagrama de seqüência permite modelar estes processos ilustrando a troca de mensagens (eventos) entre os objetos do sistema. Os objetos são representados por linhas verticais e as mensagens como setas que partem do objeto que invoca um outro objeto. As setas pode ser cheias para indicar uma mensagem de chamada ou tracejadas para indicar uma mensagem de retorno. Devem ser

desenhados tantos diagramas de seqüência quanto cenários foram levantados no diagrama de casos de uso.

Cada mensagem no diagrama de seqüência corresponde a uma operação no diagrama de classes. Como as mensagens são operações invocadas, elas devem estar presentes nos objetos de destino, que são ativadas pelas mensagens oriundas dos objetos de origem (FURLAN, 1998).

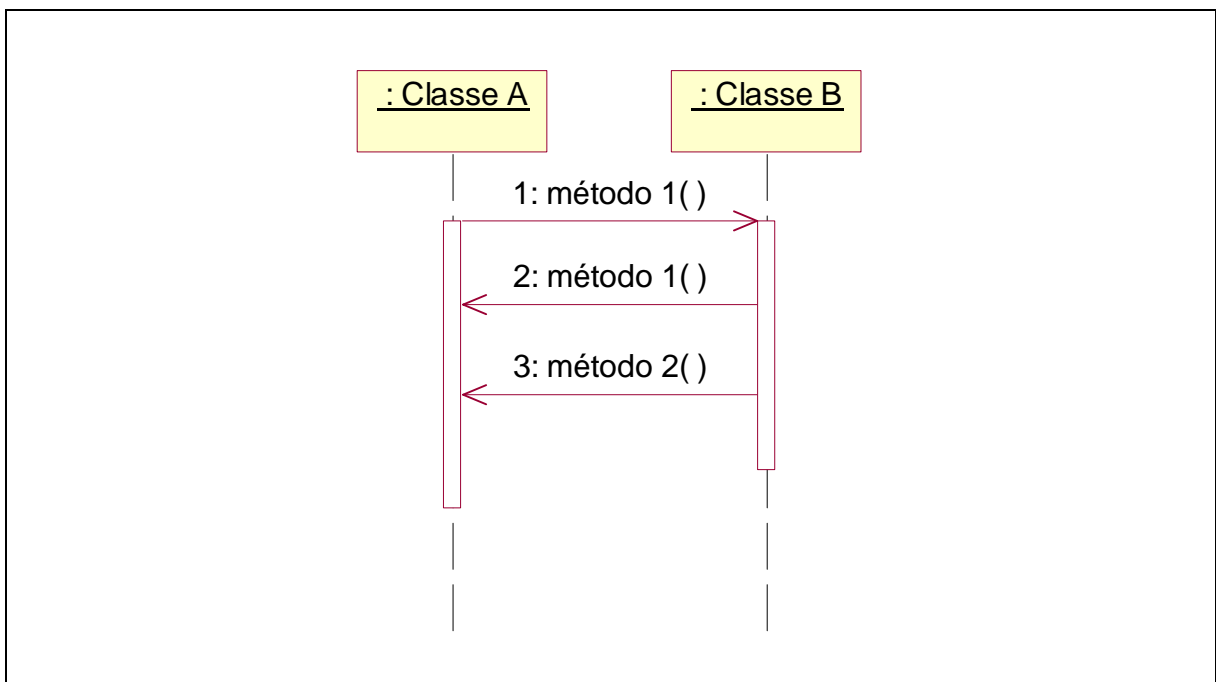


Figura 2.8: Exemplo de diagrama de seqüência.

2.5.4 Diagrama de Colaboração

Os diagramas de colaboração possuem, essencialmente, a mesma informação que um diagrama de seqüência, mas que é apresentada de uma outra forma. Este diagrama também mostra as mensagens sendo trocadas entre as classes, mas agora em um diagrama onde são apresentados os relacionamentos entre as classes, servindo de caminho para as mensagens. Os diagramas de colaboração também servem para descrever os cenários identificados pelos casos de uso, e podem ser traçados em conjunto com o diagrama de classes (FURLAN, 1998).

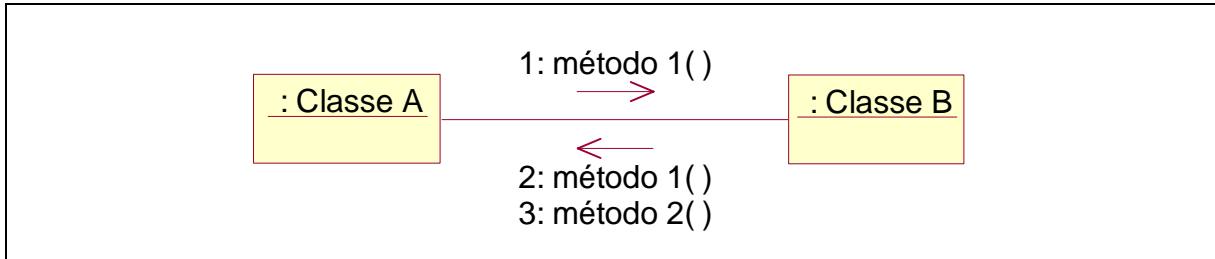


Figura 2.9: Exemplo de diagrama de colaboração.

2.5.5 Diagrama de Estados e Atividades

Os diagramas de transição de estados mostram a dinâmica interna de uma classe. Apenas os eventos e estados de uma única classe são apresentados neste diagrama. Entende-se por eventos os fatos que ocorrem em uma classe, provocados por elementos externos (mensagens) ou internos como condições internas da classe que provocam uma troca de estado. Uma classe pode ter vários estados, caracterizados por situações em que a classe se encontra. O diagrama de estados pode possuir ainda, estados especiais como o estado inicial e o estado final (FURLAN, 1998).

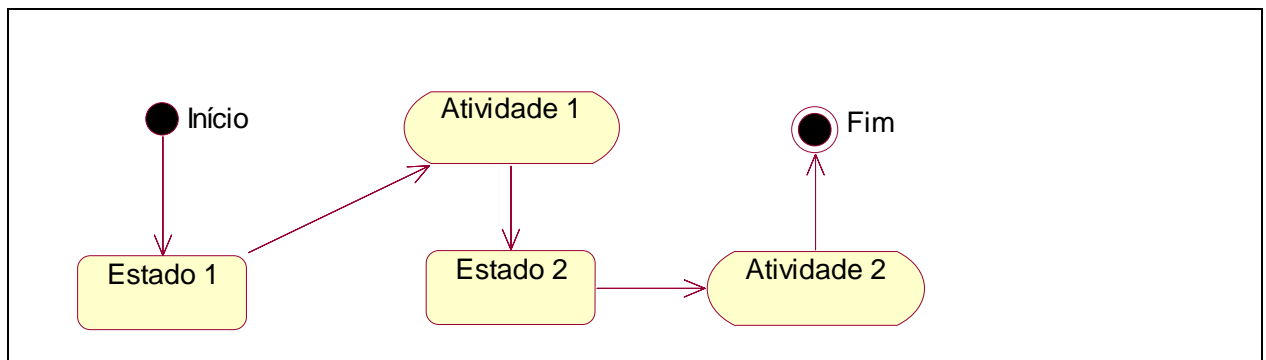


Figura 2.10: Exemplo de diagrama de estado e atividade.

2.5.6 Diagrama de Pacotes

Em muitos casos, um único diagrama de classes pode ser exageradamente grande para representar todo o sistema. Assim, é conveniente utilizar-se de um elemento para organizar os subsistemas do modelo. Para isto existem os diagramas de pacote ou componentes. Um pacote representa um grupo de classes (ou outros elementos) que se relaciona com outros pacotes por

intermédio de uma relação de dependência. Um diagrama de pacotes pode ser utilizado em qualquer fase do processo de modelagem e visa organizar os modelos (FURLAN, 1998).

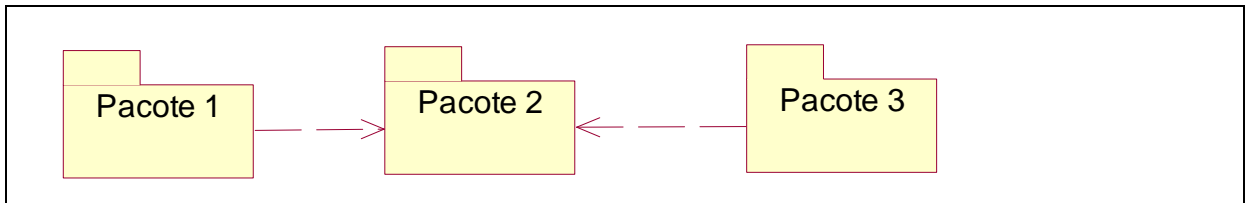


Figura 2.11: Exemplo de diagrama de pacotes.

2.5.7 Diagrama de Componentes

Os diagramas de componentes mostram os elementos reutilizáveis de *software* e sua interdependência. Um componente é formado por um conjunto de classes que se encontram implementadas nele. Um componente, assim como as classes que ele possui, dependem funcionalmente das classes de outro componente. O diagrama de componentes mostra esta dependência. No diagrama de componentes também é possível mostrar a configuração de um sistema de software mostrando, graficamente, a dependência entre os diversos arquivos que compõem o sistema (FURLAN, 1998).

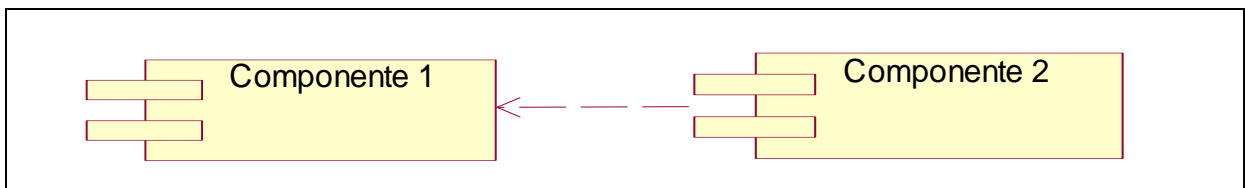


Figura 2.12: Exemplo de diagrama de componentes.

2.6 A ferramenta *Rational Rose*

A UML oferece uma grande diversidade de notações, permitindo que determinados artefatos sejam representados de diferentes perspectivas, para diferentes objetivos, durante o ciclo de vida de um projeto. O Rational Rose oferece suporte a maioria destas notações, permitindo assim a especificação e documentação de projetos de sistemas, incorporando a UML como meio para facilitar a apresentação de semânticas em desenhos. Pode ser definido

resumidamente como uma solução de software para modelagem visual que permite a criação, análise, projeto, visualização, alteração e manipulação de componentes. Com ele é possível descrever graficamente a visualização geral do comportamento de um sistema.

2.7 Considerações

Este capítulo apresentou o jogo de empresas líder, uma síntese das tecnologias aplicadas no desenvolvimento desta pesquisa e a linguagem UML, utilizada para a apresentação da modelagem do sistema proposto.

Capítulo 3 – O Sistema proposto LIDER X

Neste capítulo é apresentada a estrutura do sistema por intermédio de textos e ilustrações, a nova dinâmica de aplicação e os diagramas da UML que compõem a modelagem. O conteúdo apresentado neste capítulo, além da modelagem, tem como objetivo contribuir para o entendimento global do sistema, e viabilizar um entendimento, mínimo, para os leitores que não estão familiarizados com a linguagem UML.

3.1 Visão geral

A proposta do sistema LIDER X mantém o princípio matemático do jogo Líder inalterado. Suas implementações são realizadas em uma outra esfera, que envolve as “dinâmicas” e “eventos” que contextualizam o jogo. Conceitualmente, o Líder possui um comportamento único que é regido pelo seu modelo matemático, este comportamento é ativado no momento em que são processadas as decisões dos jogadores. Durante o processamento as decisões (globais e individuais) são consideradas como variáveis de entrada para os cálculos que definem os valores de todas as outras variáveis do jogo, como as que compõem o perfil de todos os colaboradores, e da própria empresa processada. Este procedimento permanece inalterado no sistema proposto.

Pode-se afirmar que no sistema atual que implementa o jogo Líder versão 6x, os colaboradores possuem uma postura passiva, onde ficam estáticos aguardando por seu processamento. No LIDER X é proposto um modelo de comportamento pró-ativo, onde os colaboradores possuem iniciativa para realizarem determinadas ações. Para este texto entende-se por comportamento qualquer atitude, ou conjunto de atitudes, realizadas por um ou mais indivíduos em um determinado ambiente.

Para o escopo do jogo Líder, as seguintes atitudes foram escolhidas para serem implementadas no sistema, como parte do comportamento dos colaboradores:

1. Pedido de demissão por parte do colaborador para ir trabalhar em uma outra empresa de uma mesma aplicação do jogo;

2. Ausência aleatória durante o expediente por motivos particulares;
3. Capacidade para perceber alterações no ambiente e autonomia para alterar seu estado (perfil) com base em suas percepções.

Na dinâmica atual de aplicação do jogo Líder, para que o estado (perfil) de um colaborador seja alterado, é necessário que sejam processadas as decisões de todos os outros que compõem o quadro funcional da empresa, ou seja, o jogador não pode tomar decisões sobre um único colaborador e visualizar o resultado imediatamente. Para a implementação do modelo proposto, a atual dinâmica de aplicação do jogo precisa ser adequada.

Conforme apresentada no capítulo 2, a dinâmica original de aplicação do Líder é composta por três etapas: tomada e processamento das decisões (individuais e globais) e análise dos resultados. Cada processamento delimita um período encerrado do jogo. Neste contexto, os jogadores tomam as decisões globais (ao nível de empresa) e individuais (por colaborador) do período, para que em um único processamento sejam feitos os cálculos que determinam os novos perfis para todos colaboradores.

Na dinâmica do LIDER X, cada tomada de decisões individual passa a ser um evento isolado afetando somente o respectivo colaborador envolvido, e o resultado deste processamento individual pode ser visualizado imediatamente após o processamento. No momento em que as decisões globais são tomadas, todos os colaboradores detectam esta alteração na empresa (ambiente), e baseados em sua inteligência, iniciam seus processamentos, pois sabem que estas decisões refletem diretamente em seu perfil.

O conceito de período passa a ser apenas uma referências cronológica para que o animador avalie o desempenho dos participantes. A idéia é que depois de iniciado o jogo, as empresas fictícias e os agentes colaboradores serão instanciados no sistema e permanecem em atividade até o fim do jogo de forma ininterrupta.

Diferente do sistema atual, que é executado apenas no momento em que é utilizado, o LIDER X adota um conceito diferente, permanecendo em execução ininterruptamente durante o período de aplicação do jogo. O sistema fica em execução, funcionando como um servidor, e os jogadores acessam suas respectivas empresas quando julgarem necessário, afinal, eles são os gerentes, e o fato de eles não estarem presentes fisicamente, não faz com que a empresa pare. A idéia é que eles possam “entrar” na empresa e interagir com seus colaboradores no

momento que desejarem (dentro do horário estabelecido pelo animador). Este é um dos principais incrementos para o realismo do jogo, pois permite que os jogadores tenham a opção de acessarem suas empresas e tomarem decisões no momento que desejarem, sobre os colaboradores que desejarem, exatamente como é possível no mundo real.

Se alguns jogadores optarem por deixar a empresa “abandonada”, sem intervir por meio de decisões, os “agentes colaboradores” baseados em sua inteligência e autonomia, realizam seu processamento automaticamente, baseado nas decisões anteriores (ou iniciais se nenhuma tiver sido realizada), e o próprio modelo matemático se encarrega de manter a consistência do seu perfil e sua coerência comportamental.

3.2 Estrutura

Um ponto importante considerado nesta modelagem (que inclusive já é implementado na versão atual do sistema que implementa o Líder), é permitir a criação de várias instâncias de aplicação do jogo simultâneas, para grupos de participantes distintos, como por exemplo, turmas de alunos de períodos ou semestres diferentes de uma mesma instituição educacional ou não. Este tipo de modelo permite ainda a criação de um banco de dados único com o histórico de todas as aplicações do Jogo, gerando uma valiosa base de inferências por parte do animador ou outros interessados. A figura 3.1 ilustra a estrutura descrita acima.

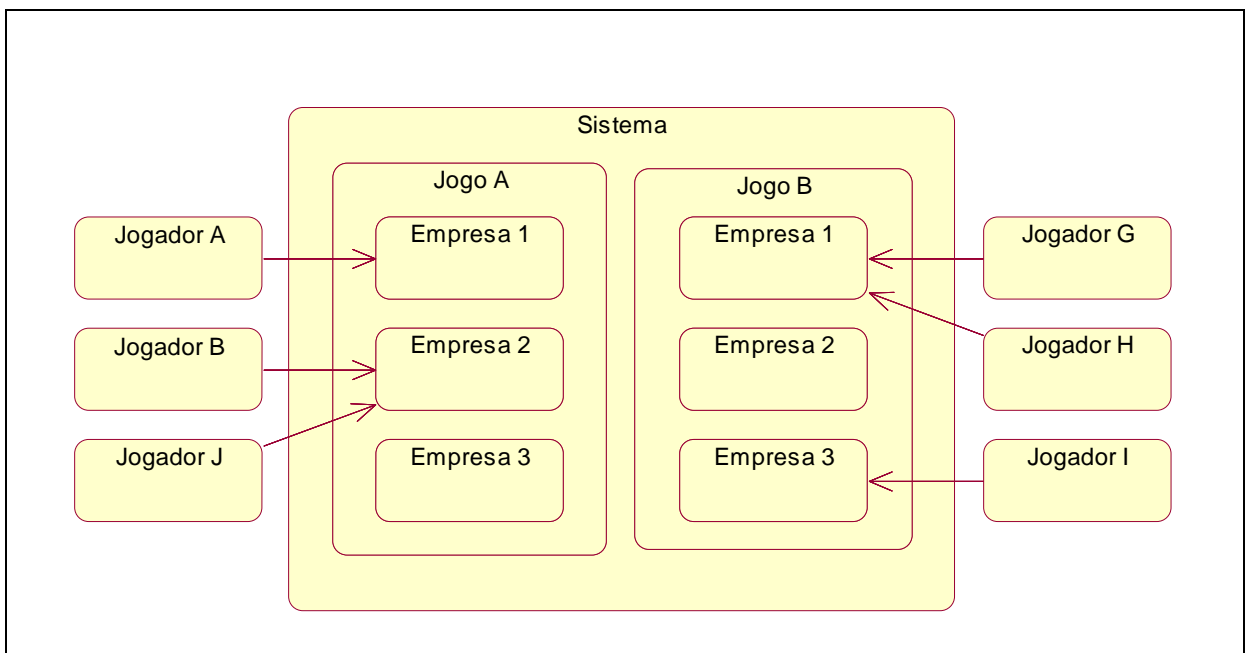


Figura 3.1: Estrutura para múltiplas aplicações do jogo simultâneas.

No mundo real, se observarmos de um ponto de vista simplista, temos um contexto de empresas concorrentes ou não que disputam um determinado mercado comum ou não. O jogo Líder sempre considera as empresas de uma mesma aplicação como concorrentes, mesmo não existindo uma disputa de um mercado consumidor. No modelo atual, toda produção da empresa é vendida incondicionalmente, uma vez que o foco do jogo não é a gestão financeira da empresa, e sim a gestão de recursos humanos.

Cada jogo é formado por um conjunto de empresas e cada empresa pertence a um grupo de participantes da aplicação.

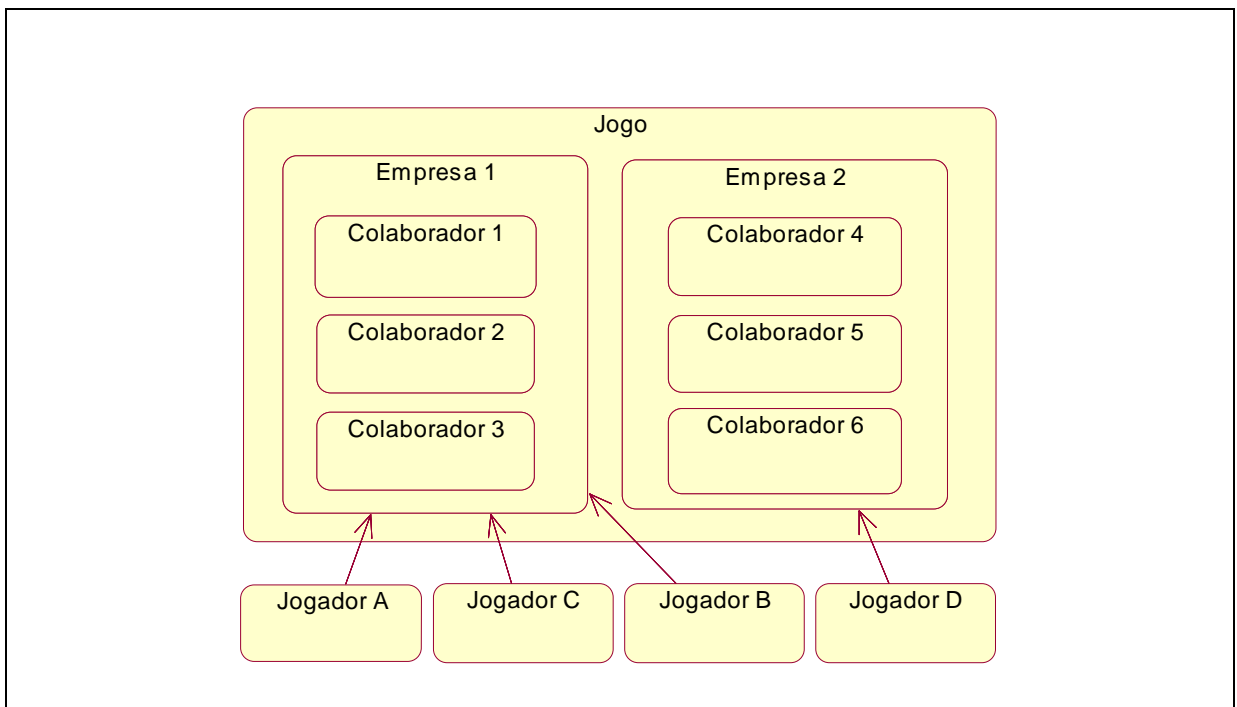


Figura 3.2: Estrutura dos Jogos

Cada empresa é composta basicamente por seus atributos (razão social, código, senha, etc), seu quadro funcional (colaboradores) e seu histórico de decisões globais (decisões iniciais e tomadas pelos jogadores), conforme ilustra a figura 3.3.

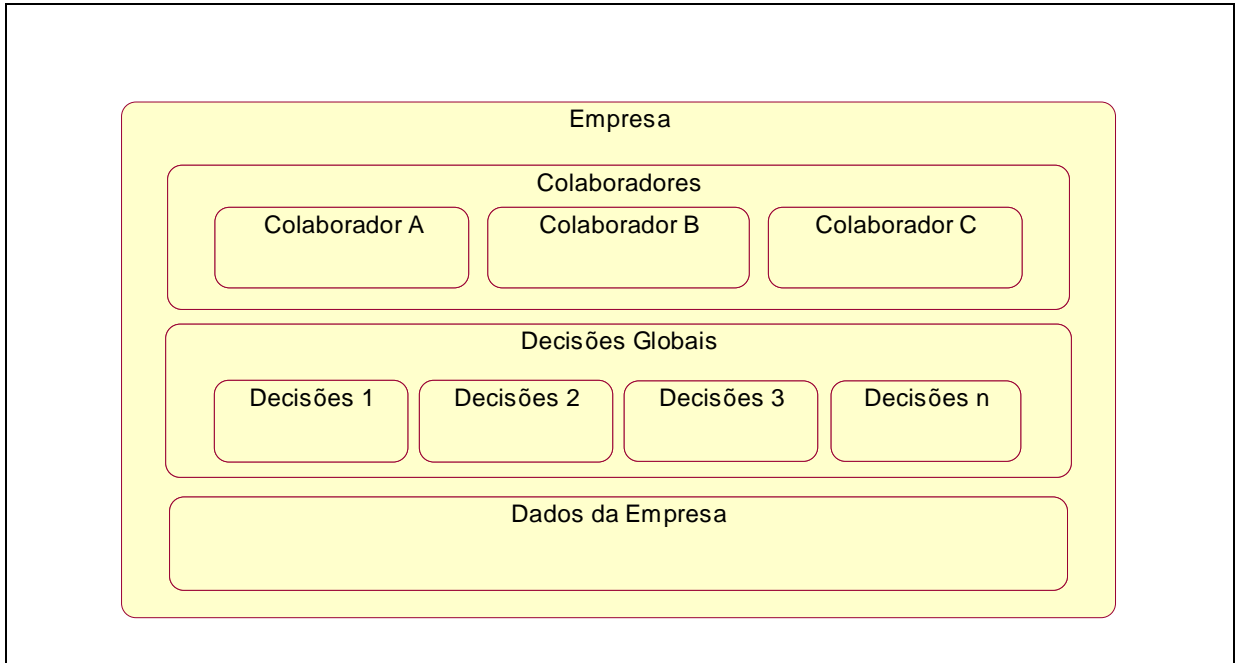


Figura 3.3: Estrutura das empresas.

Cada colaborador possui seu código único de identificação (CPF) dentro do sistema. Entretanto, eles podem ter instâncias distintas em aplicações distintas do jogo.

3.3 Os Agentes Colaboradores

No sistema Líder X, cada colaborador vinculado ao quadro funcional de cada empresa pertencente a qualquer um dos jogos ativos, possui um respectivo agente inteligente reativo em execução dentro do sistema. Os jogadores ou animadores que acessam o sistema, o visualizam em uma determinada interface por meio de um ícone, nome ou código que represente especificamente este colaborador, entretanto, internamente o sistema o relaciona a um agente, um “processo” independente, ou mais especificamente a um objeto instanciado do tipo *thread*.

As *threads* podem ser descritas como tarefas independentes dentro de um mesmo processo que podem ser executadas em paralelo ou de forma intercalada no tempo. O sistema Líder X é uma aplicação descrita por um processo composto de várias *threads*. As *threads* de um mesmo processo compartilham o espaço de endereçamento de memória, arquivos abertos e outros recursos que caracterizam o contexto global de um processo como um todo. Cada *thread*, no entanto, tem seu próprio contexto específico, normalmente caracterizado pelo conjunto de registradores em uso, contador de programas e palavra de status do processador.

Cada agente mantém seu estado (perfil) e seu passado (perfis anteriores) em memória, e utiliza estas informações como base para as suas decisões. Seu perfil é alterado a cada processamento realizado, e o anterior é mantido em sua “memória” cronológica, por uma estrutura de lista encadeada. Para recuperar uma informação, ele só precisa referenciar o índice de sua lista de perfis.

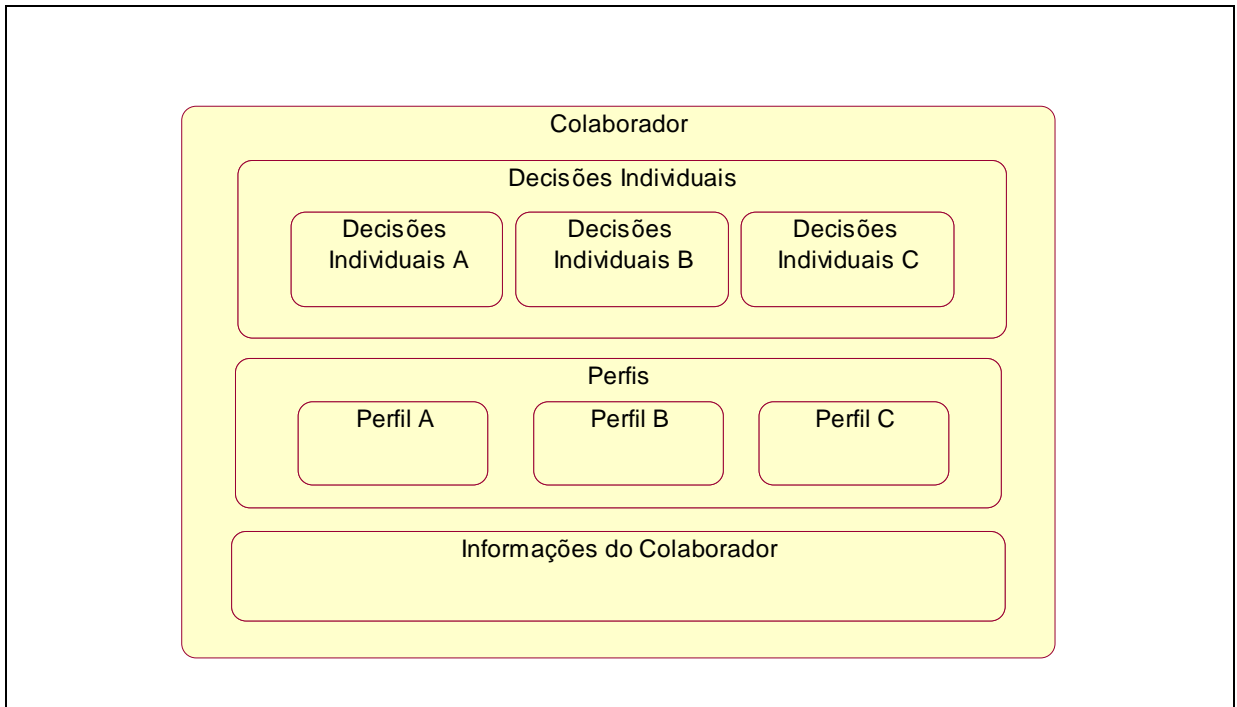


Figura 3.4: Estrutura do agente colaborador.

Como algumas de suas decisões são baseadas em informações que estão contidas no perfil dos outros agentes, e como o próprio conceito de agentes determina que estes possuam a capacidade de comunicarem-se, sua implementação no Líder X permite que estes se comuniquem por mensagens (métodos) intrínsecas da própria classe da qual ele foi instanciado.

A comunicação entre tarefas é grandemente simplificada numa implementação baseada em *threads*, uma vez que neste caso as tarefas compartilham o espaço de endereçamento de memória do processo como um todo que engloba as *threads*, eliminando a necessidade do uso de esquemas especiais, mais restritos e, usualmente, mais lentos de comunicação entre processos providos pelos sistemas operacionais.

3.3.1 A inteligência dos Agentes

Vista de um ângulo simplista, a inteligência dos agentes do sistema Líder X é definida em uma única rotina, que permanece em execução ininterruptamente enquanto o sistema estiver ativo. Um objeto *thread* que implementa o conceito de “processamento independente”, é o responsável por mantê-la em execução. Sua lógica determina quais, de que forma, e em que momento cada função do modelo matemático deve ser acionada.

As seguintes ações foram determinantes do escopo da inteligência do colaborador.

- A. Ausência aleatória durante o expediente por motivos particulares;
- B. Pedido de demissão por parte do colaborador para ir trabalhar em uma outra empresa de uma mesma aplicação do jogo;
- C. Capacidade para perceber alterações no ambiente e autonomia para alterar seu estado (perfil), iniciando seu próprio processamento sem a intervenção dos jogadores, apenas com base em suas percepções.

Uma rotina foi elaborada para determinar as reações do agente, que na verdade é segmentada em sub-rotinas. A ilustração abaixo apresenta a rotina de comportamento do colaborador e a seqüência reativa para cada ação do colaborador.

Item	Seqüência
A	<ol style="list-style-type: none"> 1. Obter a data atual do sistema. 2. SE data atual = data de ausência, iniciar item A, senão iniciar item B.
B	<ol style="list-style-type: none"> 1. Calcular quantos colaboradores da sua empresa possuem o mesmo cargo que ele. 2. Calcular quantos destes possuem um salário maior que o seu, e são menos produtivos. 3. Se mais de 50% dos colaboradores ganham mais e são menos produtivos, pedir demissão, senão continuar.
C	<ol style="list-style-type: none"> 1. Se existem novas decisões globais tomadas, iniciar um processamento do seu perfil e voltar ao início do item A, senão, continuar. 2. Calcular média de processamento dos outros colaboradores. 3. SE sua quantidade de processamentos for três a menos que a média dos outros, processar, e iniciar item A, senão apenas iniciar item A.

Tabela 3.1: Comportamento do agente colaborador.

A coluna “Item” ordenada pelas letras A, B e C, representa cada comportamento dos agentes colaboradores, e a coluna “Seqüência”, enumera a ordem cronológica das interações que compõem a implementação de cada comportamento. Na figura 3.5, a lógica descrita na tabela 3.1 pode ser vista como um diagrama.

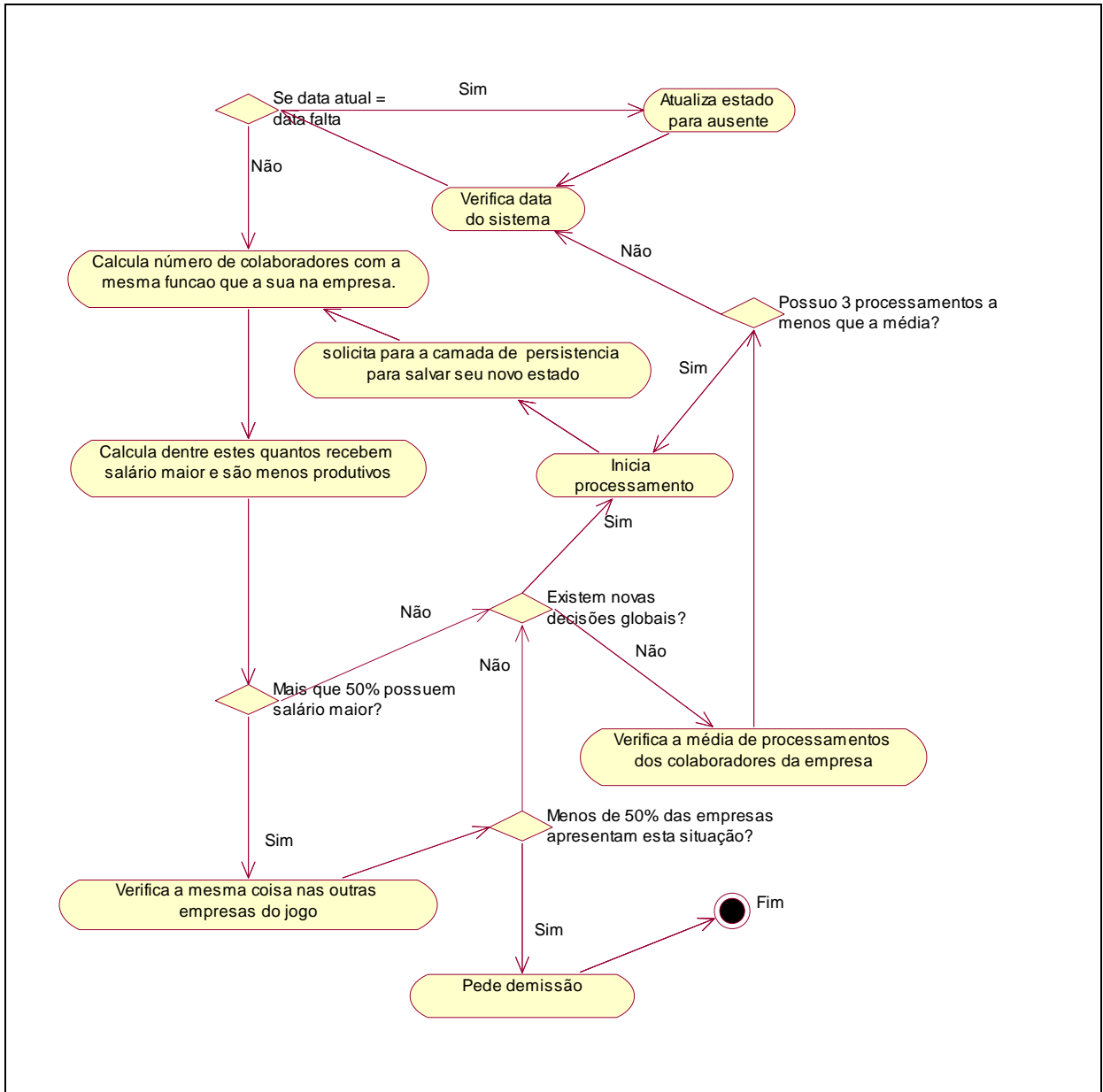


Figura 3.5: Fluxograma do comportamento do agente colaborador.

3.3.2 Considerações sobre os agentes

O sistema Líder X é baseado em uma sociedade de agentes, na qual são considerados tanto os aspectos e comportamentos individuais, quanto à interação entre diversos agentes presentes em um contexto social, para que seja determinada a inteligência do sistema.

Abaixo são feitas as relações entre a referência teórica dos agentes e a forma na qual foram modelados.

- a) Autonomia: justifica-se pelo fato de serem *threads* e possuírem capacidade de se processarem baseados em sua interpretação do ambiente;
- b) Capacidade social: justificada pela capacidade de comunicação entre os agentes, estabelecida via troca direta de mensagens entre os mesmos;
- c) Reatividade: quanto à reação a estímulos do meio ambiente, no caso deste sistema onde o meio é a empresa, os estímulos são as decisões, e as reações são: possível pedido de demissão, ausência em horário de expediente e auto-processamento;
- d) Pró-atividade: os agentes, em determinados momentos buscam analisar alguma possível alteração na sociedade que possa interferir em seu estado, independente de estímulos, investigando se está recebendo menos atenção em relação aos outros agentes, após sua análise, ele decide ou não iniciar seu auto-processamento.

3.4 Dinâmica de aplicação

A modelagem apresentada propõe uma alteração na dinâmica de aplicação do jogo, mais especificamente na etapa de tomada de decisões. Na versão 6x, as decisões globais e individuais são tomadas em um mesmo momento e em uma mesma interface. No Líder X, as decisões individuais e globais são tomadas em momentos diferentes e de forma independente. O jogador agora pode optar por tomar apenas as decisões globais ou individuais em determinados períodos da aplicação do jogo. Após a tomada das decisões globais, é possível

avaliar o reflexo destas em todos os colaboradores da empresa. No caso das decisões individuais, é possível observar os reflexos destas de forma individual, ou seja, do respectivo colaborador alvo das decisões. Conceitualmente, nada impede que o jogador tome apenas decisões individuais ou globais durante determinados períodos do jogo.

Os comportamentos incorporados aos “agentes colaboradores”, não afetaram a dinâmica principal de aplicação do jogo Líder, que continua baseada em três etapas: tomada de decisões, processamento e análise dos resultados. Na figura 3.6, uma ilustração da dinâmica de aplicação do sistema Líder X.

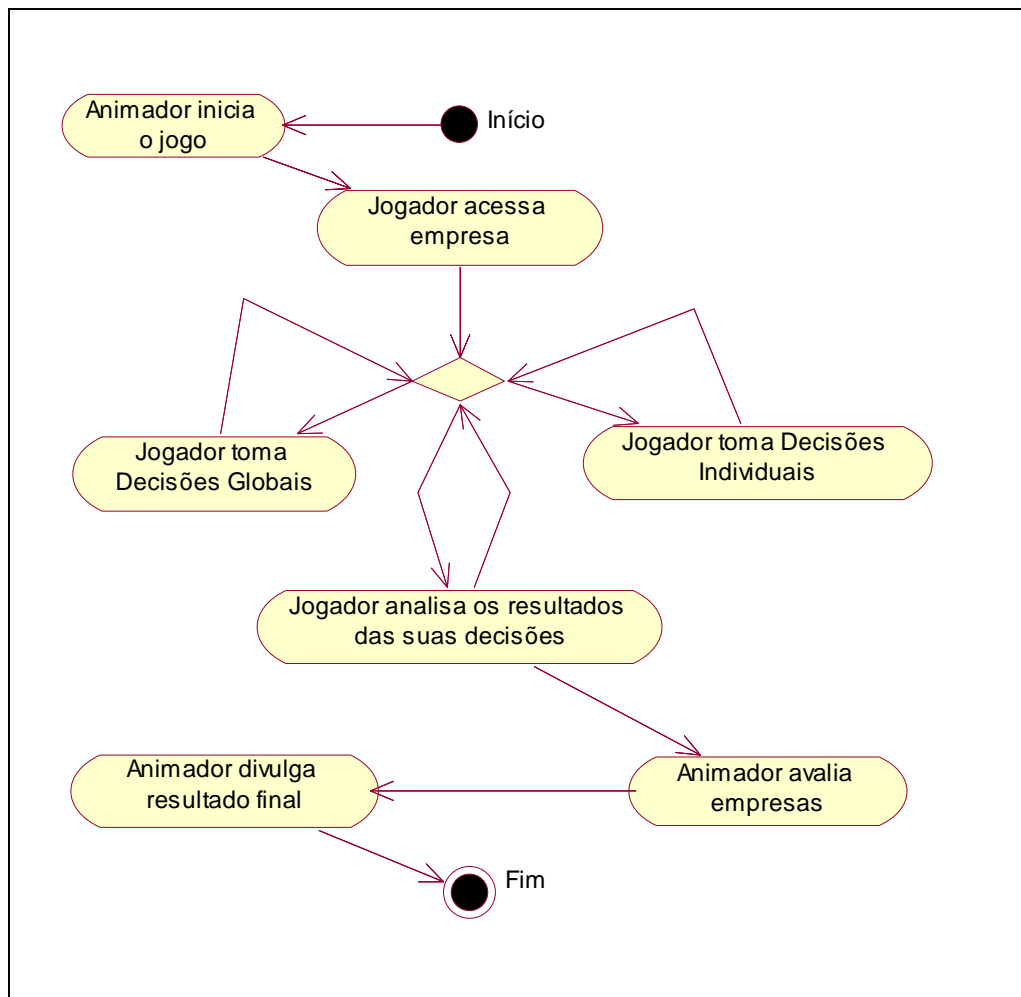


Figura 3.6: Dinâmica de aplicação do sistema Líder X.

3.5 Modelagem

Neste tópicó é apresentada a modelagem do sistema, por intermédio dos diagramas da UML. Os diagramas foram agrupados por caso de uso, e estes divididos em dois grandes grupos: casos de uso do Animador e casos de uso do Jogador.

Os casos de uso foram separados em dois grupos, com objetivo de prover um entendimento sobre as funcionalidades disponibilizadas pelo sistema para cada tipo de usuário que tem acesso a ele (animadores e jogadores).

A figura 3.7 apresenta os casos de uso do Animador.

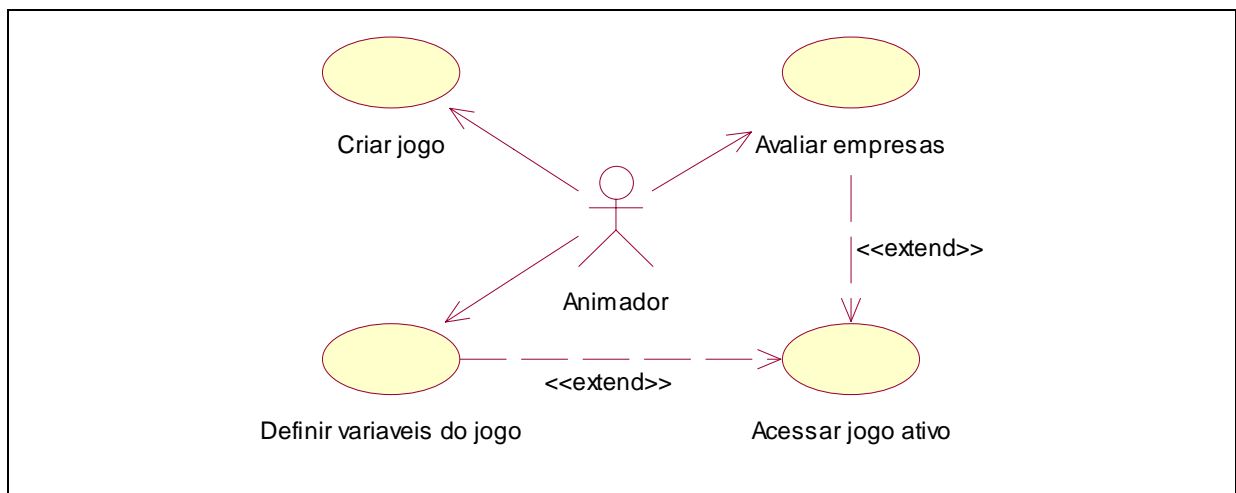


Figura 3.7: Diagrama de casos de uso do animador.

Os **casos de uso** do animador são:

- a) **Criar jogo:** Este é o procedimento para iniciar uma nova aplicação do jogo, consiste em e iniciar e inserir um novo jogo completo no sistema. Envolve internamente a criação de todas as empresas e seus colaboradores, e ao final, a persistência todas as informações criadas em uma base de dados. Para acessar esta funcionalidade do sistema, sugere-se como requisito na implementação, que seja informada uma senha especial de acesso;

Os diagramas de colaboração e seqüência deste caso de uso foram colocados respectivamente como **Apêndices A e B**, devido a seu tamanho.

- b) **Acessar jogo ativo:** Requisito para avaliar empresas e definir as variáveis, consiste em uma validação de acesso aos jogos ativos do sistema, onde o usuário solicita o acesso a um dos jogos ativos no sistema e informa a senha de acesso (cadastrada no momento da criação do jogo pelo usuário). A partir desta premissa, entende-se que ele é o animador do jogo (usuário autorizado), e passa a ter permissão para exercer as funções compatíveis;

A figura 3.8 apresenta o diagrama de colaboração para este caso de uso.

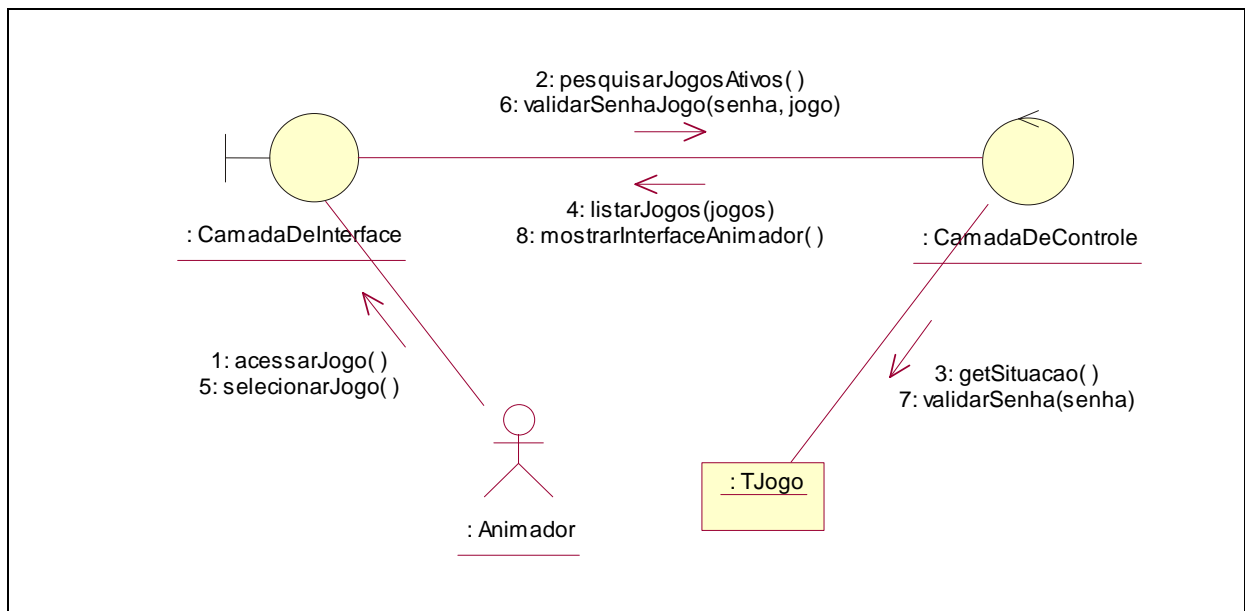


Figura 3.8: Diagrama de Colaboração - Acessar jogo.

A figura 3.9 apresenta o diagrama de seqüência para este caso de uso.

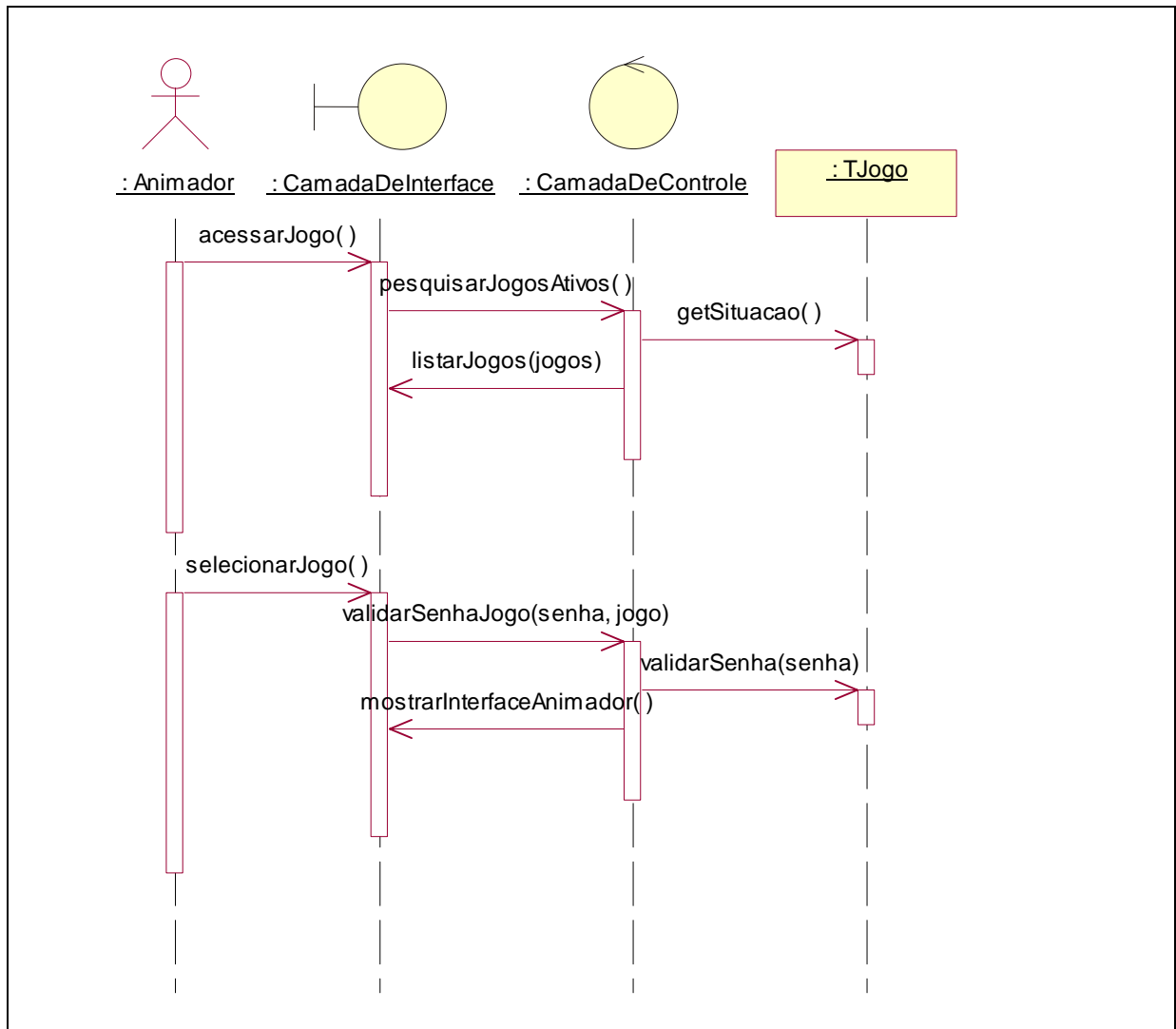


Figura 3.9: Diagrama de Sequência - Acessar jogo.

- c) **Avaliar empresas**: Momento final da aplicação de um jogo, onde o Animador registra o conceito de cada empresa participante, baseado em sua avaliação subjetiva do desempenho das mesmas;

A figura 3.10 apresenta o diagrama de colaboração para este caso de uso.

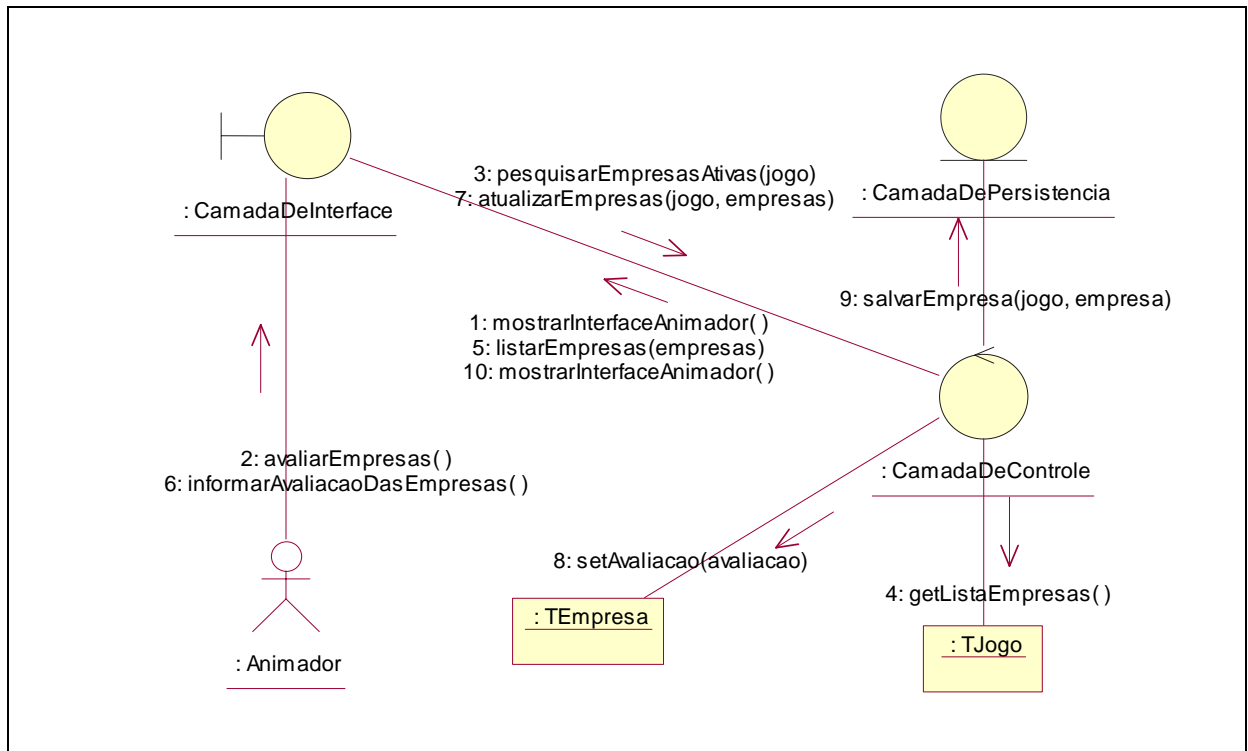


Figura 3.10: Diagrama de Colaboração - Avaliar empresas.

A figura 3.11 apresenta o diagrama de seqüência para este caso de uso.

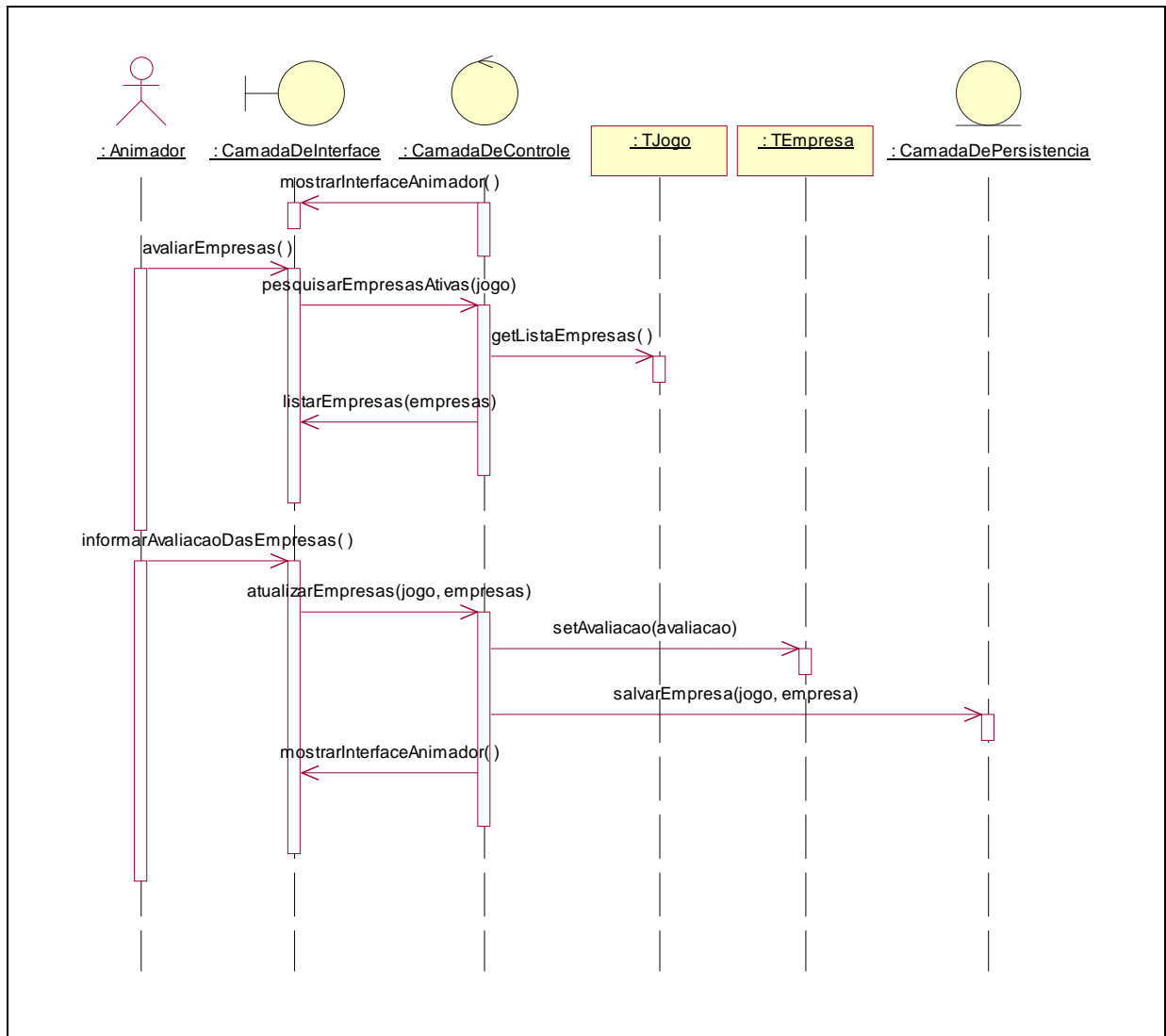


Figura 3.11: Diagrama de Seqüência - Avaliar empresas.

d) **Definir variáveis do jogo:** Procedimento que define o comportamento do jogo, onde são definidos se os seguintes parâmetros ficam ativos:

- Habilitar a demissão voluntária dos colaboradores quando estes estiverem insatisfeitos com a empresa;
- Permitir aos colaboradores iniciarem seu próprio processamento quando julgarem necessário;
- Ativar simulação de faltas ocasionais para os colaboradores;

- Informar o coeficiente numérico de intervenção do animador (valor numérico utilizado em cálculos do modelo matemático do Líder).
- Definir o período de tempo que determina um período para o jogo.

A figura 3.12 apresenta o diagrama de colaboração para este caso de uso.

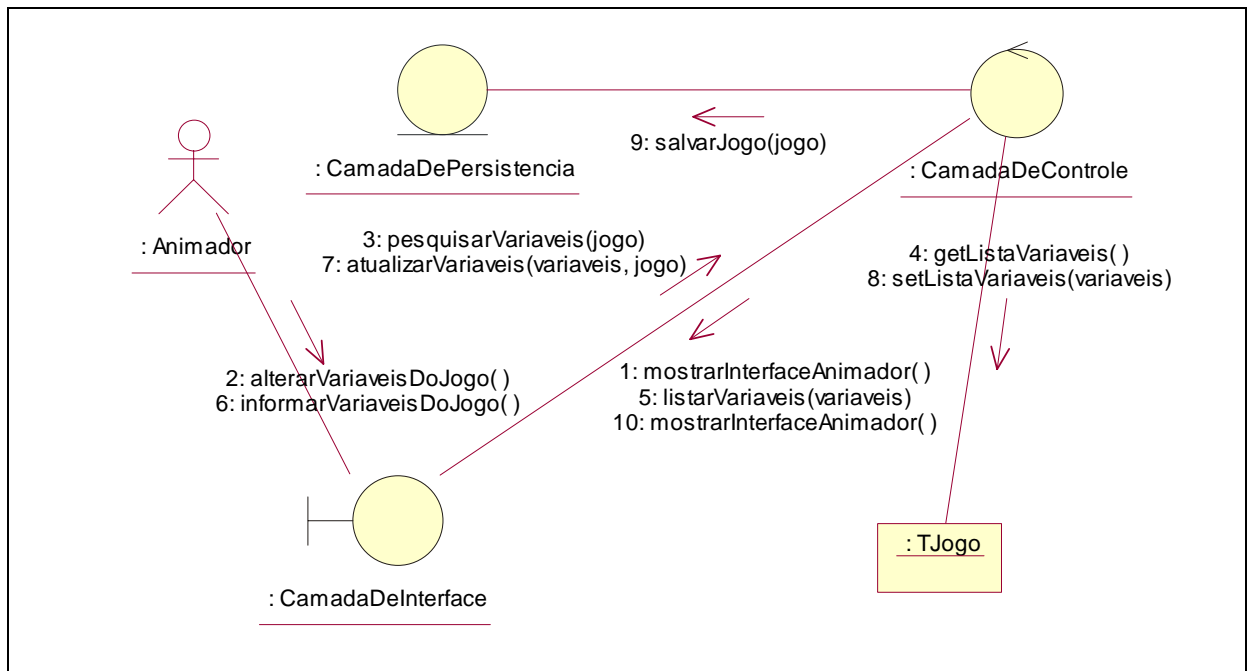


Figura 3.12: Diagrama de Colaboração – Definir variáveis do jogo.

A figura 3.13 apresenta o diagrama de seqüência para este caso de uso.

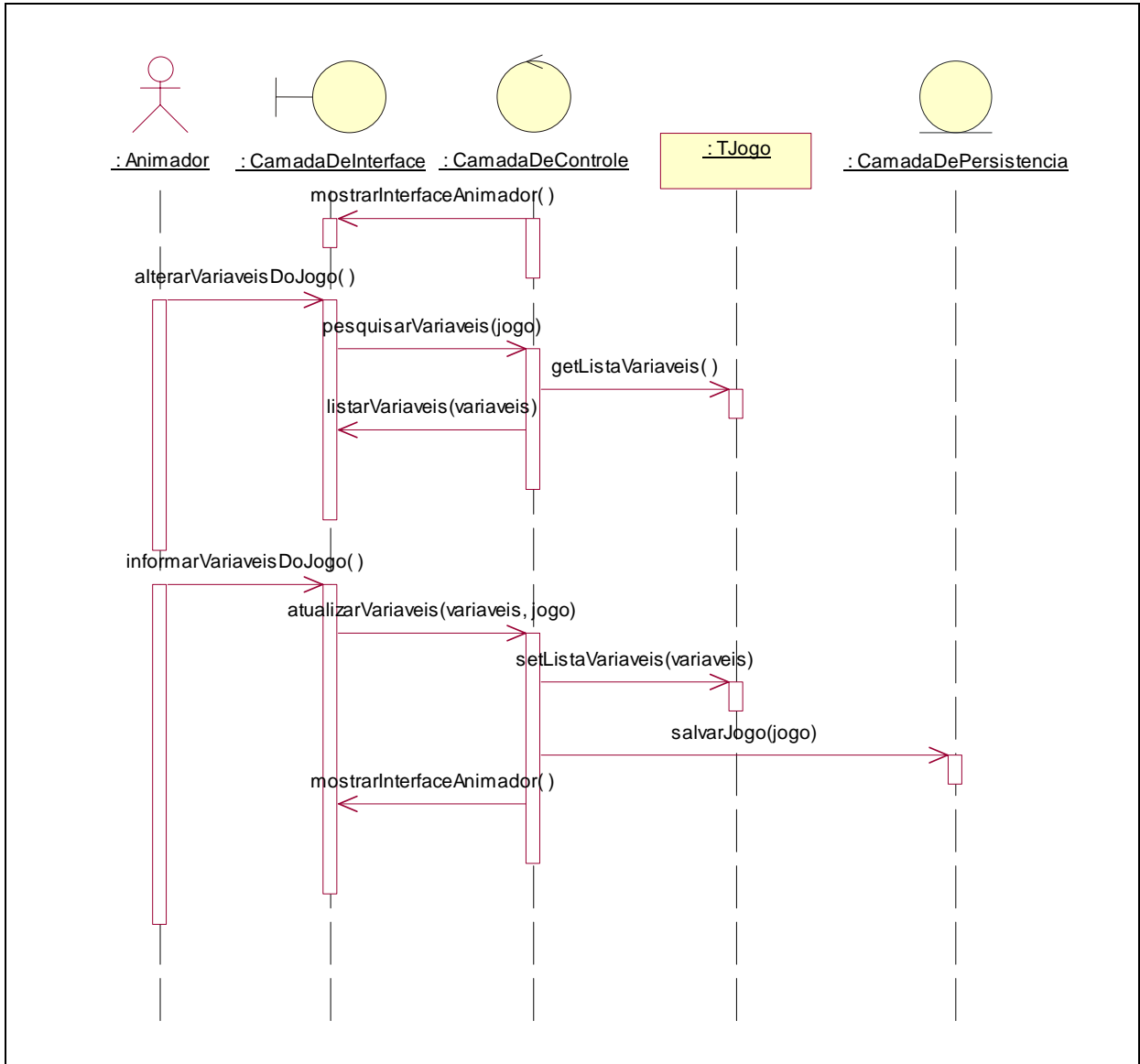


Figura 3.13: Diagrama de Seqüência – Definir variáveis do jogo.

Até este ponto foram apresentados os diagramas que ilustram a modelagem dos casos de uso do Animador. Na seqüência, estão disponibilizados os diagramas para os casos de uso do Jogador.

A figura 3.14 apresenta uma visão geral de todos os casos de uso do Animador.

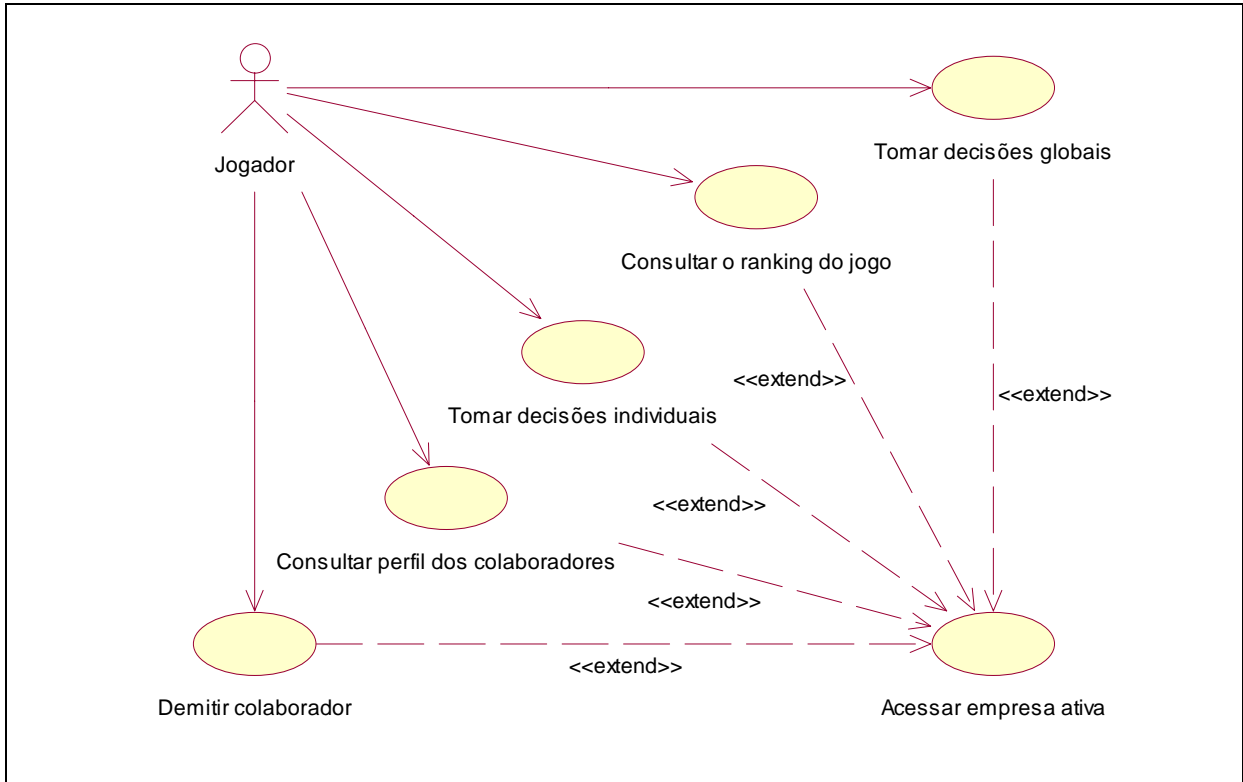


Figura 3.14: Diagrama de casos de uso do jogador.

Os **casos de uso** do jogador são:

- Acessar empresa ativa:** requisito para acesso a quais quer outros casos de uso do jogador, onde é feita a validação de acesso às empresas do jogo. O usuário solicita o acesso a uma das empresas do jogo selecionado e informa a senha de acesso da empresa (gerada automaticamente em sua criação, e entregue a ele pelo animador). A partir desta premissa, entende-se que ele é um jogador cadastrado (usuário autorizado) a atuar nesta empresa;

A figura 3.15 apresenta o diagrama de colaboração para este caso de uso.

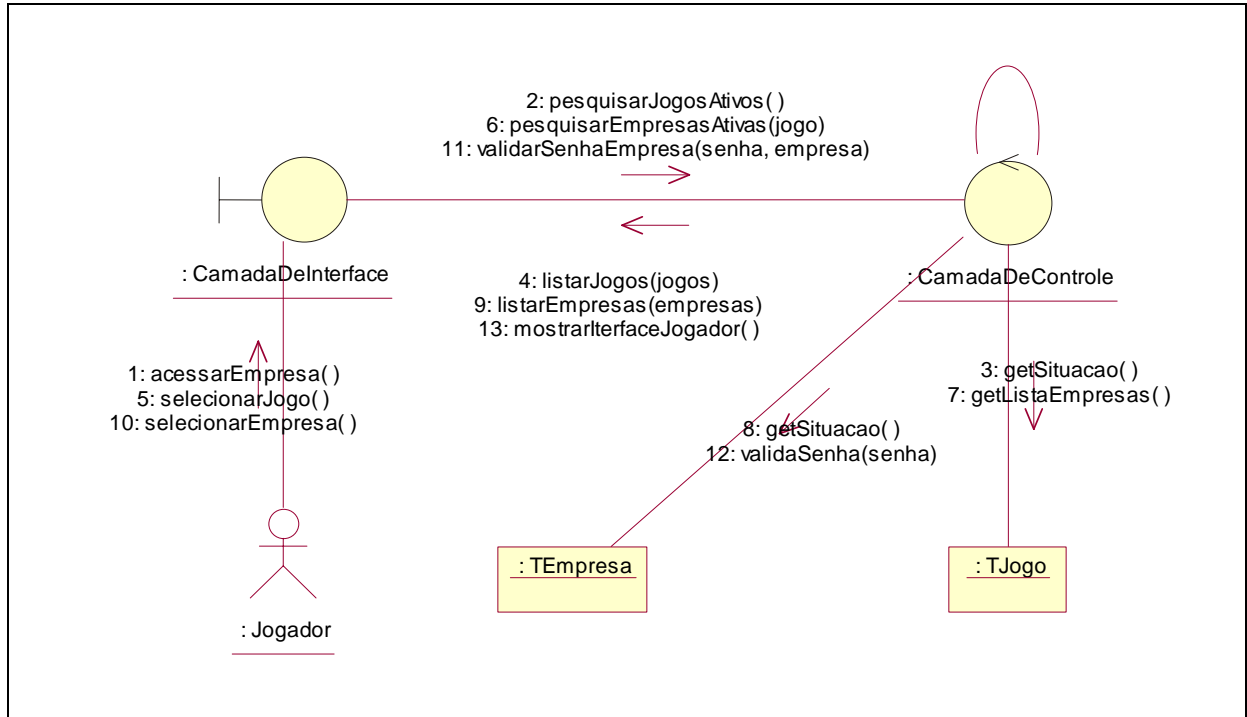


Figura 3.15: Diagrama de Colaboração - Acessar empresa.

A figura 3.16 apresenta o diagrama de seqüência para este caso de uso.

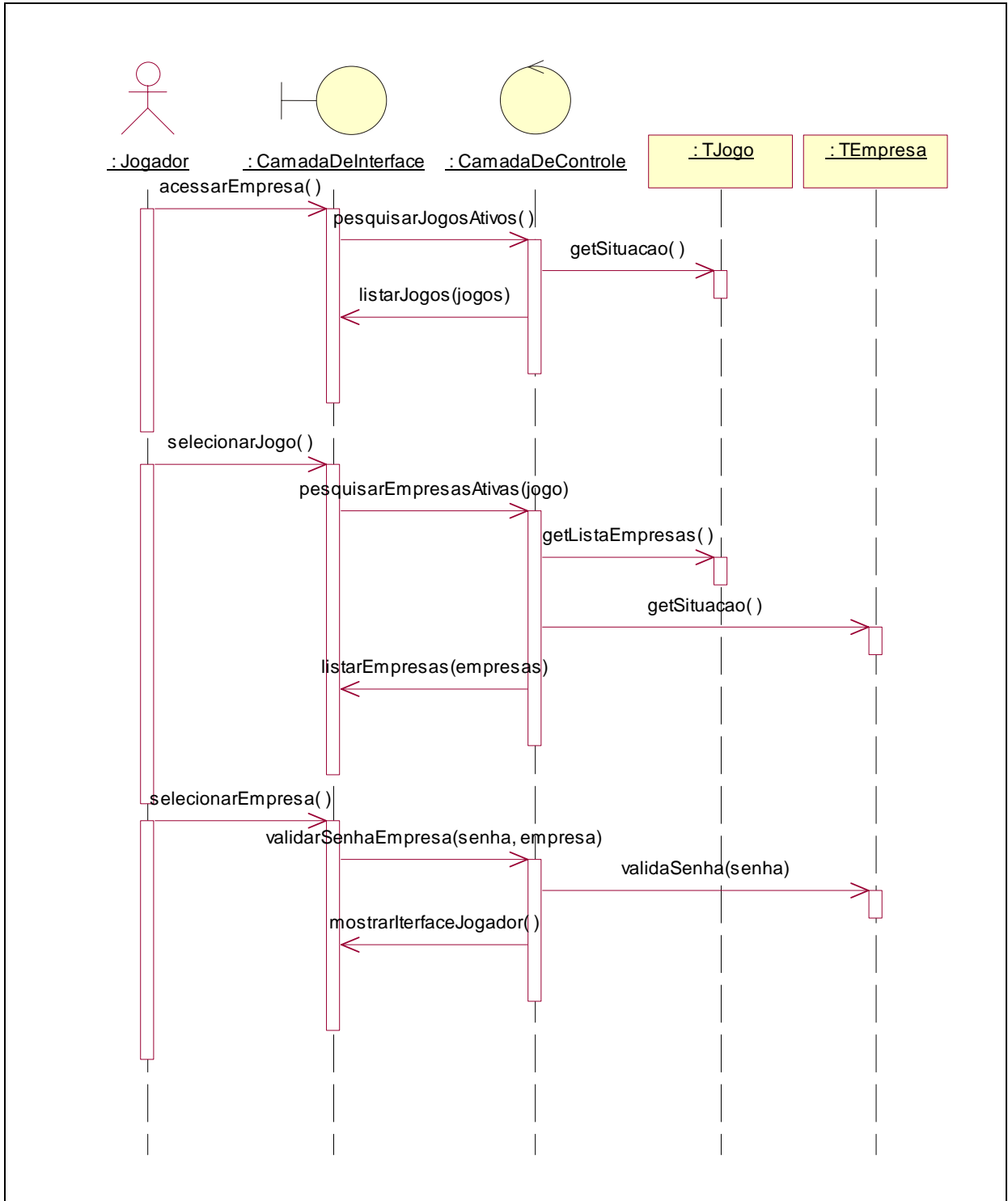


Figura 3.16: Diagrama de Seqüência - Acessar empresa.

- b) **Tomar Decisões Individuais:** onde é exibida a interface para tomada de decisões individuais sobre um determinado colaborador selecionado. Depois de salvar as decisões, o colaborador que sofreu a ação inicia seu processamento para atualizar seu estado, e conseqüentemente o estado da empresa;

Os diagramas de colaboração e seqüência deste caso de uso foram colocados respectivamente como **Apêndices C e D**, devido a seu tamanho.

- c) **Tomar Decisões Globais:** momento onde o participante toma as decisões globais para a empresa. Após salvar tais decisões, todos os colaboradores iniciam seu processamento para atualizarem seus estados baseados nas novas decisões aplicadas;

A figura 3.17 apresenta o diagrama de colaboração para este caso de uso.

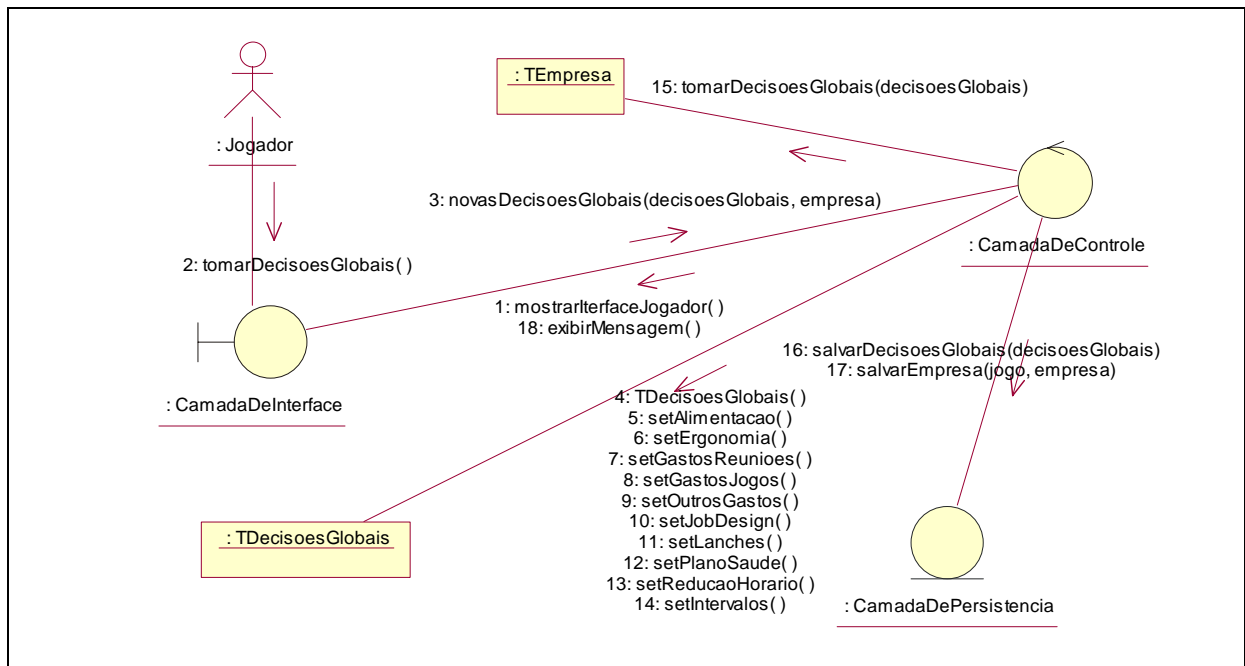


Figura 3.17: Diagrama de Colaboração - Tomar decisões globais.

O diagrama de seqüência deste caso de uso foi colocado como **Apêndice E**, devido a seu tamanho.

- d) **Consultar perfil dos colaboradores:** este caso de uso apresenta na interface do sistema o perfil atual do colaborador selecionado. Este procedimento pode ser executado a qualquer momento pelo jogador, e é uma das funcionalidades disponíveis para análise dos resultados;

A figura 3.18 apresenta o diagrama de colaboração para este caso de uso.

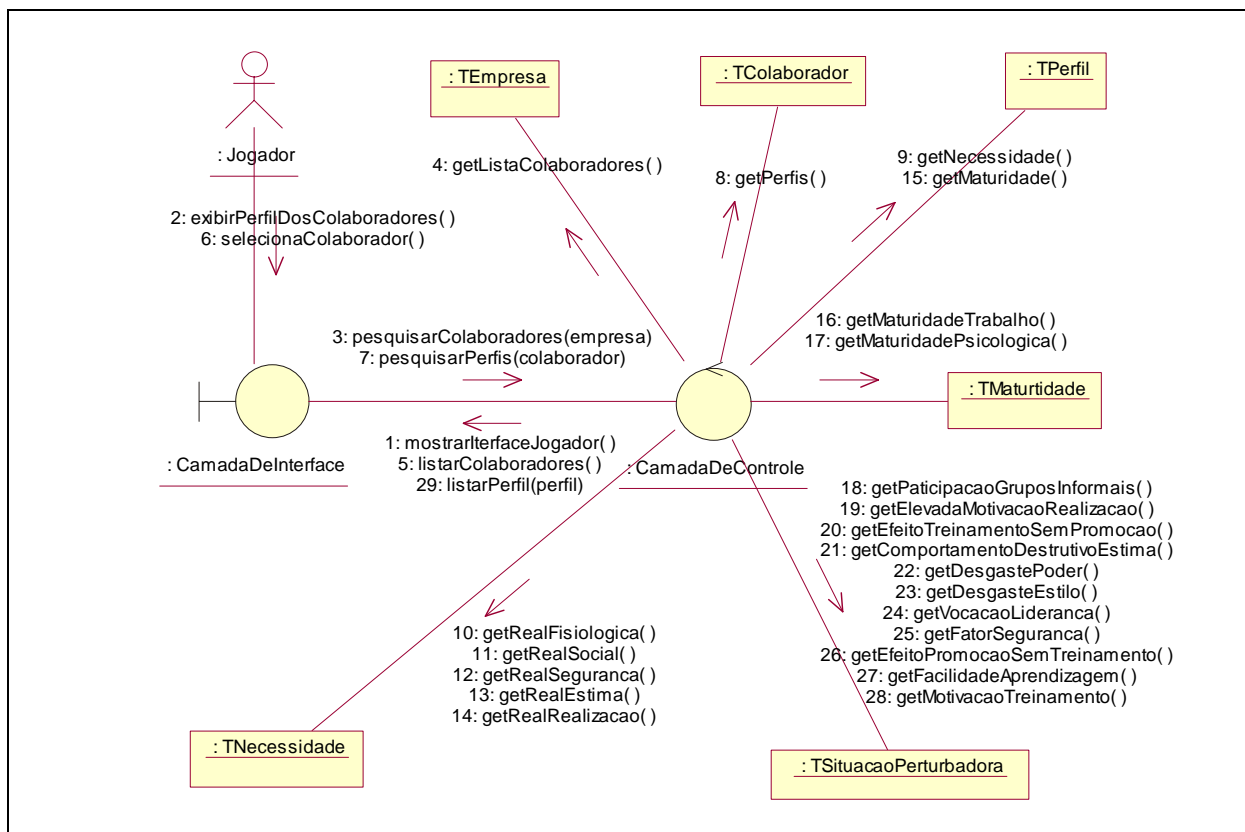


Figura 3.18: Diagrama de Colaboração - Consultar o perfil dos colaboradores.

O diagrama de seqüência deste caso de uso foi colocado como **Apêndice F**, devido a seu tamanho.

- e) **Demitir colaborador:** procedimento que exclui o colaborador selecionado do quadro funcional da empresa, colocando-o novamente na lista de candidatos para uma possível contratação por outra empresa do jogo.

A figura 3.19 apresenta o diagrama de colaboração para este caso de uso.

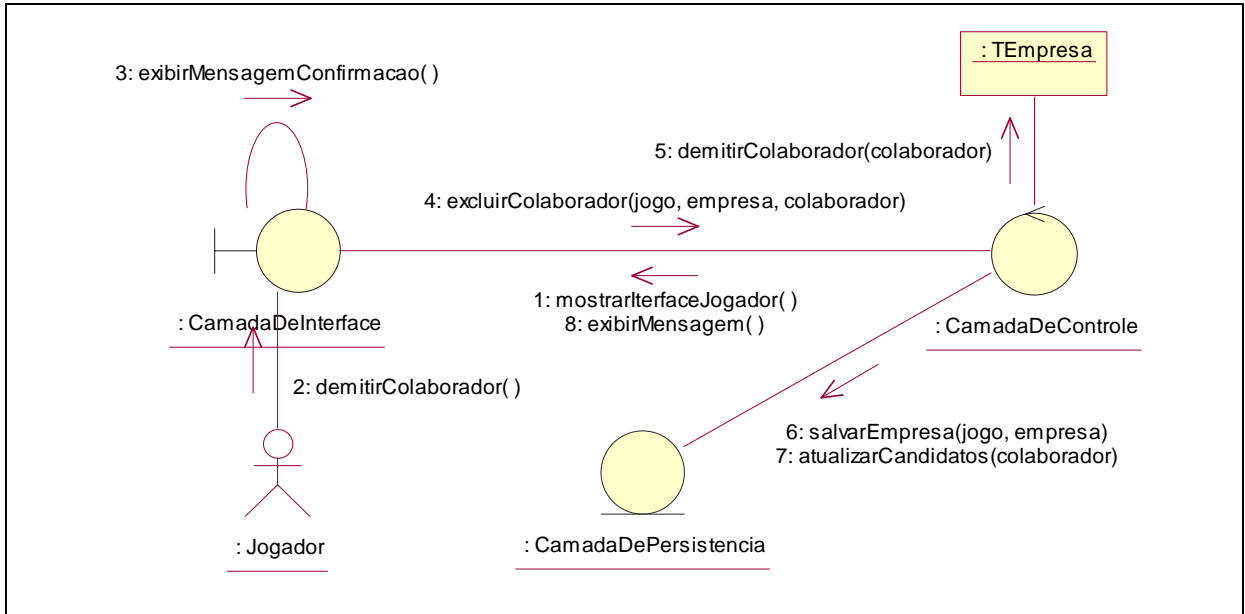


Figura 3.19: Diagrama de Colaboração – Demitir colaborador.

A figura 3.20 apresenta o diagrama de seqüência para este caso de uso.

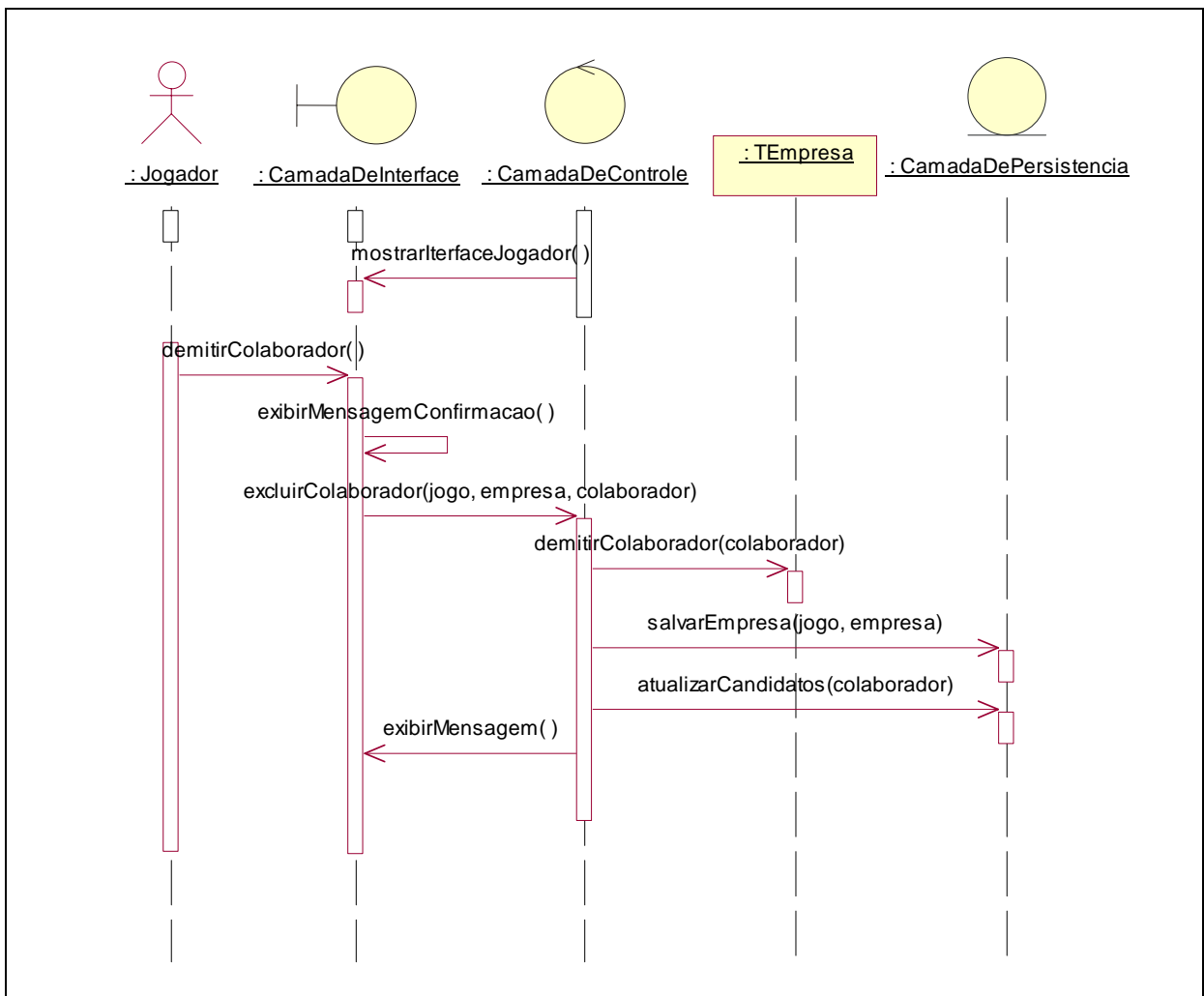


Figura 3.20: Diagrama de Seqüência – Demitir colaborador.

- f) **Consultar o ranking do jogo:** funcionalidade que lista as empresas do jogo ordenadas por lucro acumulado. A idéia é verificar qual empresa é mais lucrativa no jogo corrente.

A figura 3.21 apresenta o diagrama de colaboração para este caso de uso.

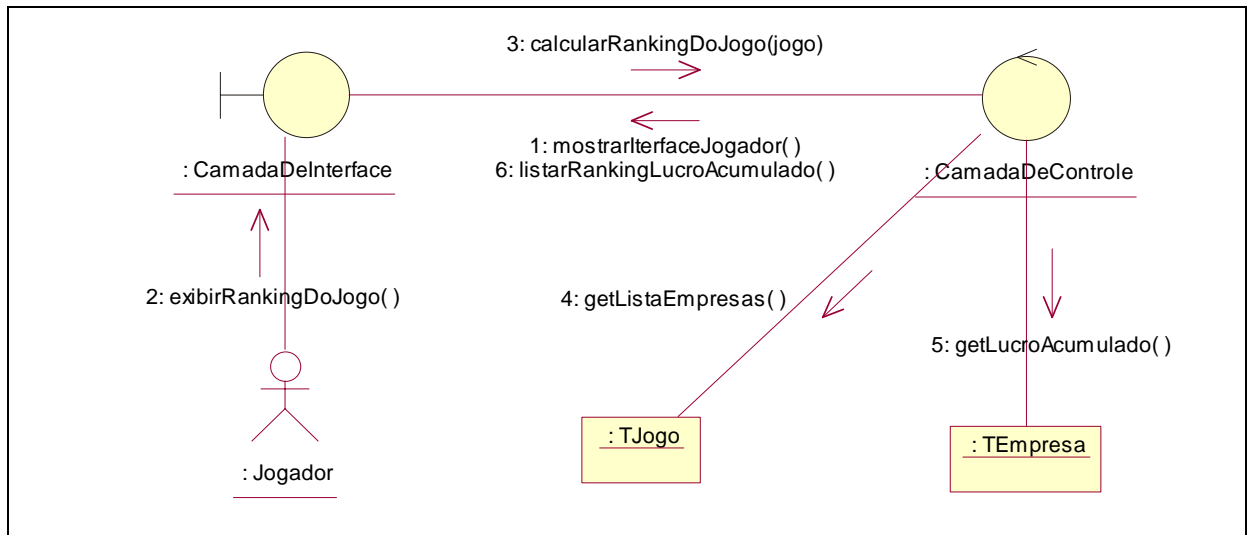


Figura 3.21: Diagrama de Colaboração - Consultar ranking do jogo.

A figura 3.22 apresenta o diagrama de seqüência para este caso de uso.

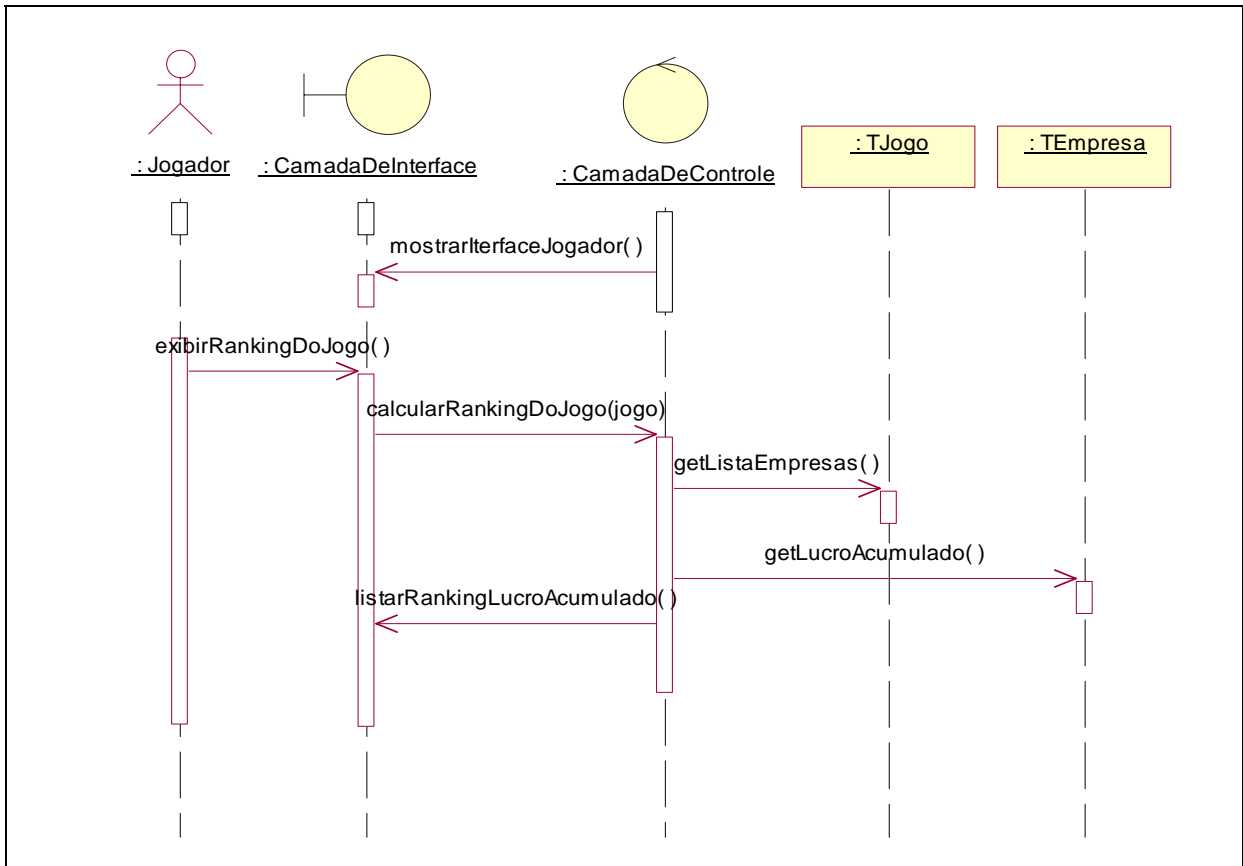


Figura 3.22: Diagrama de Seqüência - Consultar ranking do jogo.

3.5.1 Diagrama de Classes

No diagrama apresentado na figura 3.23, tem-se a visualização das classes que compõem o domínio do sistema e seus respectivos relacionamentos. Uma descrição completa de cada classe (atributos e serviços) é disponibilizada no diagrama de classes (**Apêndice G**).

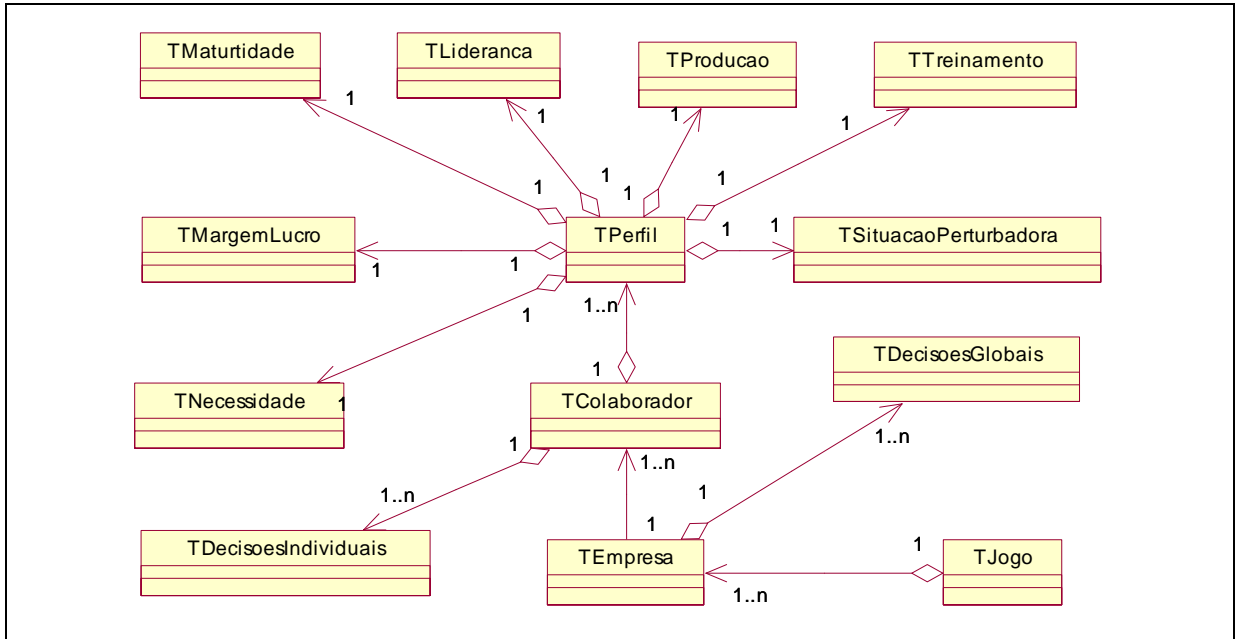


Figura 3.23: Diagrama de domínio.

Na seqüência tem-se uma breve descrição do papel de cada classe do diagrama no contexto global do sistema:

- a) **Classe TJogo:** representa um jogo existente no sistema, é a classe mais externa do modelo;
- b) **Classe TEmpresa:** representação da empresa, onde estão os colaboradores e as decisões globais;
- c) **Classe TDecisoesGlobais:** parte da classe TEmpresa, contendo todos os atributos e métodos para o acesso as decisões globais;
- d) **Classe TColaborador:** representa cada colaborador pertencente à empresa, contendo todo histórico dos perfis ;
- e) **Classe TPerfil:** parte da classe TColaborador, que agrupa as outras classes necessárias para o processamento do modelo matemático;
- f) **Classe TDecisoesIndividuais:** parte da classe TColaborador, contendo todas as decisões individuais aplicadas a cada colaborador em um determinado jogo;
- g) **Classe TLideranca:** parte da classe TPerfil, responsável pelos cálculos relativos à eficácia na liderança do colaborador;

- h) **Classe TNeccidade:** parte da classe TPerfil, responsável pelos cálculos relativos às necessidades dos colaboradores;
- i) **Classe TSituacaoPerturbadora:** parte da classe TPerfil, responsável pelos cálculos relativos às situações perturbadoras do colaborador;
- j) **Classe TTreinamento:** parte da classe TPerfil, responsável pelos cálculos relativos aos efeitos dos treinamentos ou a ausência deles nos colaboradores;
- k) **Classe TProducao:** parte da classe TPerfil, responsável pelos cálculos relativos à produtividade do colaborador e conseqüentemente da empresa;
- l) **Classe TMaturidade:** parte da classe TPerfil, responsável pelos cálculos relativos à maturidade dos colaboradores;
- m) **Classe TMargem:** parte da classe TPerfil, responsável pelos cálculos das receitas e despesas geradas pelo colaborador, e conseqüentemente a margem de lucro da empresa.

O detalhamento das classes é disponibilizado no Diagrama de Classes (**Apêndice G**).

3.6 Considerações

Neste capítulo foi apresentado o sistema Líder X por intermédio de sua modelagem na linguagem UML, complementada por diagramas, ilustrações e textos descritivos.

A implementação dos métodos das classes não faz parte do escopo da pesquisa, uma vez que o objetivo desta é o desenvolvimento de uma modelagem, que por definição é independente de plataforma e linguagem computacional. As rotinas algorítmicas para os métodos das classes relacionadas com o perfil dos colaboradores, devem ser desenvolvidas baseadas nos cálculos necessários para o processamento do modelo matemático do jogo Líder, e métodos das demais classes devem ser implementados de forma a contemplar as funcionalidades do sistema proposto, apresentadas neste capítulo.

Capítulo 4 – Conclusões e Recomendações

Este capítulo da dissertação traz as conclusões obtidas do desenvolvimento da pesquisa, as limitações identificadas e as recomendações para futuros trabalhos que envolvam o jogo de empresas Líder.

4.1 Avaliação dos objetivos

O principal objetivo desta dissertação foi o estudo de um novo sistema para aplicação do jogo líder, que retratasse o ambiente simulado proposto pelo jogo de uma forma mais real, e o desenvolvimento de sua modelagem utilizando a linguagem UML. A motivação para o estudo deste sistema era de modelar um software que pudesse intensificar a vivência dos jogadores com o modelo durante as aplicações do jogo Líder.

Para atingir este objetivo foi incorporada a tecnologia de agentes inteligentes ao sistema, não com o intuito de agregar um valor tecnológico ao software propriamente dito, e sim como meio para viabilizar a implementação de um modelo comportamental mais real aos colaboradores que compõem a estrutura do jogo, e conseqüentemente atingir os objetivos desejados.

A modelagem desenvolvida incorporou alguns comportamentos humanos aos colaboradores que compõem a estrutura do jogo Líder, comportamentos estes, presentes em nosso cotidiano no mundo real, que quando simulados no contexto do sistema, influenciam diretamente na percepção do jogador.

Alem de incrementar o comportamento dos colaboradores do jogo, o modelo apresentado também modificou alguns aspectos das empresas que compõem a estrutura do jogo Líder. Estas passaram a “funcionar” independente da presença ou não do gerente (papel exercido pelo jogador), ou seja, o modelo matemático permanece em execução mesmo quando o jogador não está “conectado” ao sistema do jogo, aspecto esse, que também contribui para aproximar o ambiente simulado do modelo real.

Com relação aos objetivos específicos:

- a) Os seguintes comportamentos específicos foram estudados e incorporados aos colaboradores:
- ✓ Ausência aleatória durante o expediente por motivos particulares;
 - ✓ Pedido de demissão por parte do colaborador para ir trabalhar em uma outra empresa de uma mesma aplicação do jogo;
 - ✓ Capacidade para perceber alterações no ambiente e autonomia para alterar seu estado (perfil), iniciando seu próprio processamento sem a intervenção dos jogadores, apenas com base em suas percepções.
- b) O comportamento incorporado aos colaboradores, por intermédio dos agentes, não interferiu na função matemática que estes exercem dentro da estrutura do jogo Líder, apenas alteram o momento em que são ativados os cálculos.
- c) A revisão da dinâmica de aplicação do jogo determinou desnecessária qualquer alteração significativa no fluxo global, uma vez que as etapas continuam as mesmas, apenas a tomada de decisões foi separada em tomada de decisões individuais e globais e ficaram independentes.

Com base nos pontos supra-enumerados, os objetivos específicos podem ser considerados como atingidos, assim como o objetivo geral.

4.2 Limitações

Como limitações:

- Sem a implementação do modelo e conseqüentemente uma simulação real, torna-se inviável determinar o comportamento do modelo matemático nas situações em que a empresa fica “abandonada”, situação onde os colaboradores

realizam seu processamento automático consecutivamente utilizando as decisões anteriores como novas.

- O critério utilizado para que os agentes decidam se devem ou não pedir demissão, ou iniciarem seu processamento por conta própria, pode ser reformulado, se os eventos acontecerem de forma excessiva, e prejudicarem o realismo do jogo.
- Uma vez que o modelo proposto define um módulo destinado exclusivamente aos jogadores, usuários estes, que não necessariamente conhecem o modelo de aplicação do jogo. Fatores como ergonomia, interatividade e facilidade de uso tornam-se pontos relevantes para sua implementação. Tais aspectos, relacionados à interface com os usuários, não foram contemplados nesta pesquisa.

4.3 Sugestões

Como sugestões destacam-se:

- Refinamento da modelagem, com objetivo de direcioná-la para uma efetiva implementação, definindo uma linguagem computacional e detalhando as camadas de interface, persistência e controle.
- Mapeamento do modelo matemático do Líder para diagramas de estados e atividades relacionados aos métodos das classes desta modelagem, o que a tornaria uma documentação completa do modelo do jogo Líder.
- Implementação da modelagem apresentada.

Referências Bibliográficas

BOOCH, Grady, RUMBAUGH, James, JACOBSON, Ivar. **The Unified Modeling Language – User Guide**. United States: Addison-Wesley Longman Inc, 1998.

DETTMER, Armando Luiz. **Concebendo um laboratório de engenharia de produção utilizando um jogo de empresas**. Tese de doutorado em Engenharia de Produção/UFSC. Florianópolis, 2001.

FRANKLIN, Stan, GRAESSER, Art (1996) "Is it an agent, or just a Program? A Taxonomy for Autonomous Agents", *Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages*, Springer-Verlag.
<http://www.msci.memphis.edu/~franklin/AgentProg.html>, em 29/10/97.

FURLAN, José Davi. **Modelagem de objetos por intermédio da UML – The unified modeling language**. São Paulo: Makron Books, 1998.

HAYES-ROTH, B. (1995). "An Architecture for Adaptive Intelligent Systems" *Artificial Intelligence: Special Issue on Agents and Interactivity*, 72, 329-365.

HÜBNER, Jomi Fred. **Migração em sistemas multiagentes abertos**. Porto Alegre: Universidade Federal do Rio Grande do Sul, 1995. Dissertação de Mestrado.

HÜBNER, Jomi Fred. **Desenvolvimento de um ambiente de bases de conhecimento distribuídas**. Blumenau, 1996. Relatório final projeto PIBIC/CNPq. Universidade Regional de Blumenau.

JENNINGS, Nicholas R. (1994) "**THE ARCHON SYSTEM AND ITS APPLICATIONS**", *Second International Working Conference on Cooperating Knowledge Based Systems (CKBS-94) (Invited Paper)*, Keele, UK, pp.241-257, <ftp://ftp.elec.qmw.ac.uk/pub/isag/distributed-ai/publications>, em 29/10/97.

JENNINGS, Nicholas R., WOOLDRIDGE, Michael (1994) "Intelligent Agents: Theory and Practice", <http://www.doc.mmu.ac.uk/STAFF/mike/papers.html>, em 30/09/97.

JENNINGS, Nicholas R., WOOLDRIDGE, Michael. **Agent technology foundations, applications and markets**. New York: Springer, 1998.

LOPES, Maurício Capobianco. **Jogo de empresas LÍDER: aperfeiçoamento do modelo e do sistema**. Florianópolis, 1994. Dissertação de Mestrado em Engenharia de Produção. Universidade Federal de Santa Catarina.

MAES, Pattie. "**Artificial Life Meets Entertainment: Life Like Autonomous Agents**", Communications of the ACM, 1995.

MALDONADO, Luis Alberto Taja. **Implantação em micro computador de um modelo comportamental para treinamento de liderança empresarial: um enfoque da liderança situacional**. Florianópolis, 1990. Dissertação de Mestrado em Engenharia de Produção. Universidade Federal de Santa Catarina.

MORIE, J.F. Inspiring the Future: Merging Mass Communication, Art, Entertainment and Virtual environments, Computer Graphics, 28(2):135-138, May 1994.

NIVEIROS, Sofia Inês. **Estudo e aperfeiçoamento do modelo das maturidades dos funcionários no jogo de empresas líder**. Florianópolis, 1998. Dissertação. Programa de Pós-Graduação em Engenharia de Produção. Universidade Federal de Santa Catarina.

RUSSELL, Stuart e NORVIG, Peter. **Artificial intelligence: a modern approach**. New Jersey : Prentice Hall Series in Artificial Intelligence, 1995.

Using Rose - Rational Rose ® VERSION: 2001A.04.00 - G126-5348-00. Disponível em: <www.rational.com>. Acessado em Novembro de 2004.

SALVATIERRA, Edwin Gery Maldonado. **Implantação em micro computador de um modelo comportamental para treinamento de liderança empresarial: um enfoque nas necessidades humanas**. Florianópolis, 1990. Dissertação de Mestrado em Engenharia de Produção. Universidade Federal de Santa Catarina.

SMITH, D. C. CIFRA e J. SPOHER. "**KidSim: Programming Agents Without a Programming Language**", Communications of the ACM, 1994.

Apêndices

Apêndice A - Diagrama de colaboração do caso de uso “Criar Jogo”.

Apêndice B - Diagrama de seqüência do caso de uso “Criar Jogo”.

Apêndice C – Diagrama de colaboração do caso de uso “Tomar decisões individuais”.

Apêndice D - Diagrama de seqüência do caso de uso “Tomar decisões individuais”.

Apêndice E – Diagrama de seqüência do caso de uso “Tomar decisões globais”.

Apêndice F - Diagrama de seqüência do caso de uso “Consultar Perfil dos Colaboradores”.

Apêndice G - Diagrama de Classes