



UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ
CAMPUS LUIZ MENEGHEL

ALEXANDRE DE CAMARGO

**UMA FERRAMENTA PARA AUXÍLIO DA
VERIFICAÇÃO FORMAL DE CONTRATOS**

Bandeirantes

2014

ALEXANDRE DE CAMARGO

**UMA FERRAMENTA PARA AUXÍLIO DA
VERIFICAÇÃO FORMAL DE CONTRATOS**

Trabalho de Conclusão de Curso
submetido à Universidade Estadual do
Norte do Paraná - *Campus* Luiz
Meneghel, como requisito parcial para
obtenção do grau de Bacharel em
Sistemas de Informação.

Orientador: Prof. Wellington Aparecido
Della Mura

Bandeirantes

2014

ALEXANDRE DE CAMARGO

**UMA FERRAMENTA PARA AUXÍLIO DA
VERIFICAÇÃO FORMAL DE CONTRATOS**

Trabalho de Conclusão de Curso submetido à Universidade Estadual do Norte do Paraná - *Campus* Luiz Meneghel, como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

COMISSÃO EXAMINADORA

Prof. Wellington Aparecido Della Mura
UENP – *Campus* Luiz Meneghel

Prof. Glauco Carlos Silva
UENP – *Campus* Luiz Meneghel

Prof^a. Rafaella Aline Lopes da Silva
Neitzel
UENP – *Campus* Luiz Meneghel

Bandeirantes, 03 de Dezembro de 2014.

Dedico a DEUS, porque sem ele seria muito difícil de chegar até aqui. E em especial aos meus pais Alviriano e Cida.

AGRADECIMENTOS

Primeiramente gostaria de agradecer a Deus, Nossa Senhora Aparecida e São Miguel Arcanjo, por iluminar e abençoar a conquista de mais um sonho.

A minha mãe Cida e meu pai Alviriano, que sempre me apoiaram durante os anos de faculdade, me compreendendo e incentivando nos momentos difíceis e souberam entender às vezes minha ausência devido aos estudos. Obrigado por acreditar em mim.

A minha irmã Lucimara e ao meu cunhado Maurílio e principalmente ao meu sobrinho João Vitor que eu amo muito, que me incentivaram a lutar pelos meus estudos. Muito obrigado.

A todos os meus familiares que sempre presentes em minha vida e acompanhando minha trajetória, muito obrigado primos (as), tios (as), avó Tereza, avó Laura (In Memoriam).

Ao meu orientador, professor Wellington Della Mura, o qual dedicou parte do seu tempo para ajudar no desenvolvimento este trabalho. Muito obrigado pela atenção, paciência, orientação e amizade.

A amizade e companheirismo de todos da XIX Turma de Sistemas de Informação, em especial meus grandes amigos durante o curso: Alex Sandro, Andressa, Eduardo Quina, Guilherme Bassetto, Fernando Mossato, Victor, Elielson, José Mario, Paulo, Kelvin, Júlio, Eduardo, Dionatan, Isabella, Juliene, Dayane, Solange Fávero, Elicrécia.

Aos meus amigos inseparáveis: Everson Orlandini, Luana Fuzzo, Gisele Martins e Andressa Bonfim, obrigado pelo carinho, atenção, alegrias, emoções, festas e churrascos a mim proporcionados. Sem esquecer que sempre que tiver "BAD" seu dia, pode ser onde eu estiver estarei de coração aberto para te alegrar, podem contar comigo. Amo vocês.

A todos os funcionários da UENP, que sempre me ajudaram e apoiaram minha vitória, que vão morar no meu Coração Uenpiano. Em especial aos grandes amigos e pais ao mesmo tempo: Janete Kanneblay, Roberto Biggi, Rita Sampaio, muito obrigado a todos.

E por fim agradeço a todas as pessoas que direta ou indiretamente contribuíram para realização deste trabalho.

“Eu tentei 99 vezes e falhei, mas na centésima tentativa eu consegui, nunca desista de seus objetivos mesmo que esses pareçam impossíveis, a próxima tentativa pode ser a vitoriosa.” (***Albert Einstein***).

RESUMO

Os avanços tecnológicos tornaram-se constantes e imprescindíveis à interação entre organizações com empresas, empresas com seus clientes, pessoas com pessoas, principalmente por meio de sistemas computacionais, no qual auxiliam o processo de negócio das mesmas. Todo processo de negócio, tem como principais características a oferta e aceitação de um acordo para que possa ser adquirido um produto, bens ou serviço, etc. Diante dessa circunstância, visa que a garantia dos deveres e dos direitos, daqueles que fazem parte do negócio, sejam exercidas, por este motivo há necessidade de se criar contratos entre os envolvidos, para que eles possam ter suas garantias e deveres assegurados e preservados. Os contratos contém cláusulas, normas, regras, direitos e deveres que devem ser respeitos pelos envolvidos do contrato. Porém, no contrato podem ocorrer conflitos, devido à sua escrita subjetiva. Um conflito é uma situação em que as cláusulas remetem a uma situação impossível de ser resolvida. Portanto, o presente trabalho aborda o desenvolvimento de uma ferramenta computacional que auxilie as pessoas, empresas ou organizações na criação de contratos eletrônicos. Essa ferramenta converte estas cláusulas do contrato em linguagem de contrato CL por meio de expressões matemáticas, que apresentam características de lógica deôntica e lógica dinâmica. Após gerar as expressões, assim podem-se utilizar essas expressões para se verificar se existem conflitos normativos no contrato eletrônico, através da ferramenta CLAN. A ferramenta CLAN foi desenvolvida para verificação de conflitos em contratos eletrônicos.

Palavras-chave: CLAN; Contratos Eletrônicos; CL (Linguagem de Contrato); Lógica Deôntica, Conflitos Normativos.

ABSTRACT

Technological advances became constant and essential for the interaction between organizations with companies, companies with their customers, people to people, mainly through computer systems, which help the business process of them. Every business process has the following features to offer and acceptance of an agreement for it to be purchased a product, goods or service, etc. Given this circumstance, which aims to guarantee the duties and rights of those who are part of the business, are carried out, because of this it is necessary to create contracts between those involved, in order to preserve their warranties and obligations secured. The contracts contains clauses, norms, rules, rights and obligations that must be respect for those involved in it. However, in the contract could happen conflicts due to its subjective writing. One conflict is a situation where the terms refers to an impossible situation to be solved. Therefore, this paper discusses the development of a computational tool that helps people, businesses or organizations in the creation of electronic contracts. This tool converts these contracts clauses in CL contracts language through means of mathematical expressions, which features deontic logic characteristic and dynamic logic. After generating the expressions, they can be used to see if there are normative conflicts in electronic contracts through CLAN tool. The CLAN tool was developed to check conflicts in electronic contracts.

Keywords: CLAN; E-contracts; CL Contract Language; Deontic logic; Normative Conflicts.

LISTA DE ILUSTRAÇÕES

Figura 1- Exemplo de modelo de Kripke para lógica modal.	23
Figura 2 – Ferramenta CLAN analisando conflitos.	33
Figura 3 – Autômato $[a]O(b)^{[b]}F(b)$ com conflito.	33
Figura 4 – Adicionando conflito na Ferramenta CLAN.	34
Figura 5 – Autômato livre de conflito.	35
Figura 6 – Autômato gerado do contrato $[a]O(b)^{[b]}F(b)$	35
Figura 7 – Interface do Sistema desenvolvido.	47
Figura 8 – Passos para fazer as atividades do sistema.	48
Figura 9 – Tela de Cadastro das Cláusulas do Contrato.	49
Figura 10 - Diagrama de Caso de Uso.	50
Figura 11 – Diagrama de Classes.	51
Figura 12 – Diagrama de Atividades.	52
Figura 13 - Interface Gráfica da Tela Principal do Contrato.	53
Figura 14 – Interface Gráfica Cadastrar Ação.	54
Figura 15 – Tela Adicionar Cláusulas demonstrando as lógicas.	55
Figura 16 – Tela de Cadastrar Composição de Ações.	55
Figura 17 - Interface Gráfica de Contrato em CL.	56
Figura 18 – Criando um Novo Contrato.	57
Figura 19 – Adicionando Ações.	58
Figura 20 – Adicionando Agentes.	58
Figura 21 - Tela de Adicionar Cláusula no contrato.	59
Figura 22 – Tela Contrato com o contrato descrito.	60
Figura 23 – Tela Gerar Contrato Serviço Internet.	60
Figura 24 – Ferramenta CLAN aplicando Contrato Serviço Internet.	61
Figura 25 – Autômato gerado contrato serviço Internet.	62

LISTA DE TABELAS

Tabela 1 - Exemplo utilizando o operador modal de necessidade.	22
Tabela 2 - Exemplo utilizando o operador modal de possibilidade.....	23
Tabela 3 - Exemplo utilizando o operador dinâmico de necessidade em lógica dinâmica.....	26
Tabela 4 - Sintaxe da Linguagem de Contratos \mathcal{CL} (PRISACARIU; SCHNEIDER, 2012).....	29

LISTA DE ABREVIATURAS E SIGLAS

CL	Contract Language
CTD	Contrary-to-duty
CTP	Contrary-to-prohibition
DL	Dinamic Logic
GUI	Graphics User Interface
IDE	Integrated Development Environment
PDL	Propositional Dinamic Logic
SDL	Standard Deontic Logic

SUMÁRIO

1. INTRODUÇÃO	12
1.1 FORMULAÇÃO E ESCOPO DO PROBLEMA	13
1.2 JUSTIFICATIVA.....	13
1.3 OBJETIVOS	14
1.3.1 Objetivo Geral	14
1.3.2 Objetivos Específicos	14
1.4 ORGANIZAÇÃO DO TRABALHO	15
2. FUNDAMENTAÇÃO TEÓRICA	16
2.1 CONTRATOS.....	16
2.1.1 Importância do Contrato.....	17
2.1.2 Dificuldades de realizar um contrato convencional e eletrônico	17
2.2 CONTRATOS ELETRÔNICOS.....	19
2.2.1 Tipos de contratos eletrônicos	19
2.2.2 Motivos de utilizar contratos eletrônicos	20
2.3 REPRESENTAÇÃO DE CONTRATOS ELETRÔNICOS.....	21
2.3.1 Lógica Modal	21
2.3.2 Lógica Dinâmica.....	24
2.3.3 Lógica Deontica.....	26
2.3.4 A linguagem de contratos <i>CL</i>	28
2.3.5 Conflitos.....	31
2.3.6 Ferramenta CLAN	32
3. TESTE DE APLICAÇÃO DOS FORMALISMOS	37
3.1 CONTRATO SERVIÇO DE INTERNET.....	37
3.2 REPRESENTAÇÃO UTILIZANDO LÓGICA MODAL	38
3.3 REPRESENTAÇÃO UTILIZANDO LÓGICA DINÂMICA.....	40
3.4 REPRESENTAÇÃO UTILIZANDO LÓGICA DEONTICA	41
3.5 REPRESENTAÇÃO LINGUAGEM DE CONTRATO CL	42
3.6 ANÁLISES DOS TESTES APLICADOS	44
4. METODOLOGIA.....	45
4.1 MATERIAIS.....	45
4.2 PASSOS METODOLÓGICOS	46
5. DESENVOLVIMENTO DO SISTEMA	47
5.1 APRESENTAÇÃO DO SISTEMA.....	47

5.2 MODELAGEM	49
5.2.1 Diagrama de Casos de Uso	50
5.2.2 Diagrama de Classes	50
5.2.3 Diagrama de Atividades	52
5.3 IMPLEMENTAÇÃO DO SISTEMA	53
5.3.1 Interface Gráfica.....	53
6. RESULTADOS E DISCUSSÃO	57
7. CONSIDERAÇÕES FINAIS	64
REFERÊNCIAS	65
ANEXO 1 - PARTE DO CONTRATO ENTRE PROVEDOR DE INTERNET E UM CLIENTE (PACE; PRISACARIU, SCHNEIDER, 2007).	69

1. INTRODUÇÃO

A extraordinária transformação tecnológica que passou a existir com avanço da tecnologia da informação e das telecomunicações permitiu que o acesso à rede mundial de computadores se tornasse realidade em todo o mundo. A Internet proporciona um conjunto de tecnologias e representa uma nova mentalidade e uma nova cultura no mundo dos sistemas de informação, bem como uma nova função para a tecnologia da informação (LAUDON e LAUDON, 1999).

Diante desta atual evolução, as grandes organizações com empresas, empresas com empresas, empresas com seus clientes, pessoas com pessoas, vêm utilizando a Internet e os computadores praticamente em todas as situações de negociação. Essa interação entre as empresas tem colaborado para estabelecer novos desafios nas áreas de negociação, integração e interoperabilidade entre as organizações, o que reforça a necessidade de se obter um meio de garantir as regras de uma transação e da execução de tarefas de forma prática e confiável. A criação de contratos torna-se essencial em todos os cenários de negócio, quando uma transação é realizada entre as partes envolvidas em um contrato (XU, 2004a).

Os contratos englobam todos os participantes do processo em uma relação de direitos e deveres, ou seja, as cláusulas, as normas, as regras e o comprometimento de exercê-las. Eles ainda podem possuir regras definindo penalidades ou compensações no caso de alguma violação, é o que chamamos de cláusulas contrárias ao dever ou ainda cláusulas contrárias à proibição (PRISACARIU; SCHNEIDER, 2007).

A utilização de contratos é parte importante do processo de negócios, devido o mesmo possuir um conjunto de atividades ou tarefas estruturadas relacionadas que produzem um serviço ou produto específico para seus clientes. Isso se deve à necessidade de expressar o comprometimento dos participantes do negócio no que diz respeito aos seus direitos e deveres. Conforme Miranda (2008), "Contrato é o acordo de vontade entre duas ou mais pessoas com a finalidade de adquirir, resguardar, modificar, transferir ou extinguir direitos".

Os contratos são aplicados nas mais diversas áreas para garantir os direitos e deveres das partes envolvidas. No entanto, um contrato pode apresentar inconsistências devido a sua descrição em linguagem natural tornando-o ambíguo ou conflitante e, possivelmente, levando a desentendimentos e quebras contratuais.

Essas particularidades dos contratos motivam a sua formalização em meios matemáticos e eletrônicos criando assim o conceito de contratos eletrônicos (FENECH, 2008).

Uma descrição formal de um contrato permite a verificação automática de forma mais confiável, seja na análise, na negociação ou no monitoramento de um contrato. Dentre as representações para contratos, que são utilizadas para representar sistemas e propriedades, estão às lógicas modais, as lógicas temporais, a lógica deôntica e a lógica dinâmica (LEE, 1988; HERRESTAD e KUOGH, 1995; PRISACARIU e SCHNEIDER, 2012). Outros trabalhos utilizam XML (KOETSIER, GREFEN e VONK, 2000) e Redes de Petri (XU, 2004a; AALST, 2010).

As descrições formais dos contratos os tornam mais íntegros e detalhados na interpretação dos mesmos. No entanto, a interpretação é como qualquer outra informação representada em linguagem natural, subjetivo e ambígua, assim, possivelmente levando a discórdia contratual. A técnica de detecção de conflitos procura por situações conflitantes entre as cláusulas do contrato, verificando a ocorrência de inconsistências. Se uma cláusula obriga um indivíduo a executar uma ação e existe outra cláusula no contrato que proíbe essa mesma ação, se torna impossível cumprir o contrato (FENECH, 2008).

1.1 FORMULAÇÃO E ESCOPO DO PROBLEMA

A utilização de contratos é parte importante do processo de negócios. Isso se deve à necessidade de expressar o comprometimento dos participantes do negócio no que diz respeito a seus direitos e deveres. Ao realizar ou criar um contrato convencional é possível que este contrato possa haver conflitos ou cláusulas contraditórias devido à sua escrita. Se formularmos um contrato convencional por meio computacional, podemos verificar se o contrato está livre de conflitos, utilizando uma ferramenta para a verificação.

1.2 JUSTIFICATIVA

De acordo com Fenech (2008), os contratos são aplicados nas mais diversas áreas, devida sua importância. As particularidades dos contratos e possíveis falhas

em sua escrita, quando feitas somente por meio humano, motivam a sua formulação em meios matemáticos e eletrônicos criando o conceito de contratos eletrônicos.

O conflito normativo é uma situação onde as cláusulas de um contrato entram em contradição e podem gerar regras conflitantes. A proibição e obrigação (ou permissão) de uma mesma ação, por exemplo, pode levar o contrato a uma situação de conflito. Neste contexto o contrato não pode ser satisfeito, pois todas as ações realizadas levam a uma violação das normas. No meio jurídico o conflito normativo é muito comum, e para contornar essa situação alguns mecanismos são aplicados para eliminar essas contradições, como os critérios hierárquicos, cronológicos e de especialidade (AUSÍN, 2005). No entanto, se um conflito é detectado no processo de negociação de um contrato o impasse pode ser resolvido antes que este seja executado. Este processo de detecção de conflitos pode ser realizado utilizando técnicas de verificação formal, sobre algum formalismo adequado para descrever contratos.

Os conceitos presentes na linguagem de contrato CL são utilizados para definir os tipos de conflitos normativos segundo no trabalho de Fenech (2008), que aborda a análise de contratos eletrônicos expressos na linguagem de contratos CL.

1.3 OBJETIVOS

A seguir são apresentados os objetivos que se pretendia alcançar com este trabalho.

1.3.1 Objetivo Geral

O objetivo geral deste projeto foi construir uma ferramenta computacional que auxilie a criação de contratos eletrônicos e seu uso em uma ferramenta para a detecção de conflitos.

1.3.2 Objetivos Específicos

- Desenvolver uma interface gráfica para a construção de contratos utilizando formalismos, a qual possa ser inseridas as cláusulas de forma dinâmica poupando tempo sua na criação.

- Converter o contrato criado na ferramenta desenvolvida para a linguagem formal CL.
- Utilizar a ferramenta CLAN para identificar conflitos do contrato.

1.4 ORGANIZAÇÃO DO TRABALHO

O trabalho está organizado da seguinte forma. O Capítulo 2 apresenta uma revisão teórica. O Capítulo 3 apresenta o Estudo de Caso. O Capítulo 4 apresenta a Metodologia a serem utilizadas. O Capítulo 5 apresenta o Desenvolvimento do trabalho. O Capítulo 6 apresenta o Resultado e Discussão. O Capítulo 7 apresenta as Considerações Finais.

2. FUNDAMENTAÇÃO TEÓRICA

Na fundamentação teórica são apresentados os conceitos relacionados aos contratos e contratos eletrônicos, a importância que eles exprimem, as dificuldades em sua elaboração, os modelos existentes e algumas formas de representação de contratos eletrônicos.

2.1 CONTRATOS

Segundo Salomo (2005), os contratos foram criados devido ao progresso da civilização e a necessidade de se garantir os interesses das pessoas. Os contratos são uma forma de manifestar e garantir as vontades dos indivíduos sobre um determinado assunto, ou seja, é um meio de proteger os signatários.

O contrato é uma forma de manifestar uma vontade de adquirir um determinado produto ou serviço e ao mesmo tempo a ter condições de exercer e fornecer seu produto para que assim satisfaça a necessidade do outro, isso de forma pura e que não possua restrições, que entre em conflitos que podem ser considerados ilegais em seus atos jurídicos. Contrato é um acordo de vontades, na conformidade da lei, e com a finalidade de adquirir, resguardar, transferir, conservar, modificar ou extinguir direitos (PEREIRA, 2001).

A lei (artigo 104 do Código Civil Brasileiro) apenas estabelece que ao contratar (trata-se de um ato jurídico) é necessário que as partes tenham capacidade de exercício, que o objeto seja lícito e que tenha forma prescrita ou não proibida pela lei (BACCIOTTI, 2013).

Os contratos podem ser classificados de acordo com a redução de obrigações. Os contratos unilaterais ocorrem quando apenas um indivíduo assume responsabilidades, no caso de doações e. Nos contratos bilaterais (entre duas partes) ou multilaterais (mais de duas partes) todos os indivíduos assumem obrigações (MIRANDA, 2008). Um contrato pode, então, ser definido como sendo um conjunto de normas, ou cláusulas, que rege a negociação entre os indivíduos de uma sociedade.

2.1.1 Importância do Contrato

A utilização de contratos como parte importante do processo de negócios, se deve a necessidade de expressar o comprometimento de indivíduos no que diz respeito aos seus direitos e deveres. O contrato na área jurídica é uma declaração de vontade dentro da lei, capaz de produzir o efeito jurídico desejado pelo indivíduo que declarou tal vontade (SOUZA, 1997).

São nos contratos escritos que estão estabelecidos os prazos de entrega, prazos de pagamento, forma de pagamento, multas, forma de cumprimento de deveres, obrigações, todas as cláusulas importantes para um bom contrato. Uma das finalidades de um contrato é ter a garantia que os participantes ao estabelecerem um contrato sejam beneficiados pelas condições acordadas entre as partes, de forma que sejam cumpridas adequadamente, dando segurança aos mesmos perante a lei.

2.1.2 Dificuldades de realizar um contrato convencional e eletrônico

Os problemas com contratos vão além da dificuldade de sua representação em relação aos mesmos. As principais dificuldades que podem ocorrer ao se realizar um contrato convencional e eletrônico: Negociação, Verificação, Monitoramento e Assinatura.

- **Negociação**

Baseado no processo de negociação de contratos legais, é um cenário em que pelo menos dois indivíduos visam alcançar um acordo que é aceitável por ambas as partes envolvidas. Normalmente, cada parte inicia a negociação oferecendo a solução preferida de seu interesse. A outra parte, caso não aceite, deve fazer contrapropostas para que cheguem a um acordo. Este processo ocorre em contratos bilaterais ou multilaterais através de mediadores ou não (BOSSE, 2008). O sistema desenvolvido não trata essa questão, devido à negociação ser um processo inicial entre as organizações com as empresas, na busca de entrarem em acordo para que assim possam fazer um contrato.

- **Verificação**

A detecção de conflitos Fenech (2008) objetiva a eliminação de conflitos normativos de um contrato. Um conflito normativo é uma situação em que as normas se contradizem proibindo e obrigando ou proibindo e permitindo a mesma ação, realizando a execução de duas ações conflitantes (OCHOA, 2003). Esta situação invalida um contrato, levando a uma situação de violação (BOBBIO, 2006). A verificação de conflitos deve ocorrer antes da execução do contrato e, se for o caso, após sua negociação. O sistema desenvolvido auxilia nesta verificação, onde se pode estar averiguando se existem cláusulas contraditórias, que inviabilize a celebração do contrato.

- **Monitoramento**

No processo de execução de um contrato, cada participante tem suas características particulares. Um participante de um contrato pode, então, executar as ações descritas numa cláusula ou então ignorá-las, levando a violação do contrato. Com o monitoramento da execução é possível tornar as transações e relacionamentos entre os indivíduos mais confiáveis, flexíveis, eficientes e aceitáveis. Assim, tenta-se manter os benefícios entre todos os indivíduos mesmo na presença de violações (XU e JEUSFELD, 2003). O sistema desenvolvido não possui recurso para identificar se um envolvido no contrato violou o contrato, pois os indivíduos são incorporados apenas nas ações e não podem ser representados pela linguagem de contrato CL.

- **Assinatura**

O problema da assinatura digital dos contratos é posterior à etapa de negociação. Em geral, a assinatura de um contrato digital é mais complicada do que no modo convencional. O problema surge pelo fato de que nenhum indivíduo envolvido quer ser o primeiro a assinar o contrato. Pode ocorrer de que o outro indivíduo se recuse a assinar depois de ter obtido a assinatura e o contrato do primeiro indivíduo (CHADHA; KRAMER; SCEDROV, 2004). Esse problema pode causar uma vantagem indesejável para um dos indivíduos envolvidos.

2.2 CONTRATOS ELETRÔNICOS

Atualmente, com o advento da tecnologia da informação integrada em praticamente todos os âmbitos da sociedade, as corporações utilizam-na para auxiliar em seus processos de negócio e interações das mais variadas formas: troca de informações ou produtos, realização de processos em nome de outra parte, relacionamentos em longo prazo, prestação de serviços, entre outros (ANGELOV; GREFFEN, 2005). Se em um relacionamento de negócios tradicional, os contratos tem um papel essencial na garantia dos interesses dos participantes, em relacionamentos pelo meio eletrônico isso não é diferente (XU; JEUSFELD, 2003).

Um contrato eletrônico ou “*e-Contract*” é uma conversão de contratos convencionais, utilizados para firmar acordos entre pessoas ou entidades, para o meio digital com o propósito de utilizar a tecnologia como aliada na solução dos problemas encontrados neste contexto. Um contrato eletrônico possui as obrigações de cada participante, atividades exercidas e responsabilidades definidas para cada um dos indivíduos para satisfazer os termos e condições do contrato firmado (XU, 2004b).

O contrato eletrônico somente pode ser concluído se houver uma vontade de manifestação entre seus contratantes (proposta e aprovação), do qual não pode ser afirmado nas formas verbais e nem documento escrito, deve ser registrado através meio eletrônico.

2.2.1 Tipos de contratos eletrônicos

Segundo Prisacariu e Scheneider (2007), contratos eletrônicos (*e-contract*) são entendíveis por máquinas e dessa forma podem ser verificados, controlados ou monitorados por sistemas específicos. Dentro do conceito de *e-contract*, ainda é possível distinguir dois tipos importantes:

- **Legal**

As áreas de aplicação dos contratos eletrônicos são as mais variadas como meios legais, em que se converte ou substitui um contrato físico para o meio eletrônico, ou utilizando as técnicas de contratação eletrônica em sistemas, lógicas de negócio ou integração de serviços.

A contratação eletrônica legal obtida diretamente de um contrato convencional obtido pela tradução da sua versão física para o meio eletrônico, que consiste em transcrever um contrato legal, o mesmo que é utilizada em papel para reger a prestação de um serviço, a compra de um imóvel ou as leis de uma instituição (XU, 2004b).

O sistema desenvolvido trata o uso do contrato eletrônico legal, devido ao mesmo descrever as cláusulas, ações, pré-condições que estão contidas no contrato eletrônico assim criado pelo sistema.

- **Tecnológico**

Contrato eletrônico tecnológico advindo de certa aplicação, como um Webservice ou um processo de negócio que, rege o conjunto de regras de um sistema de computação, como um serviço web, um fluxo de trabalho ou processo, entre outros (PRISACARIU; SCHNEIDER, 2007).

A tarefa de se evitar conflitos normativos é extremamente importante e necessária na área de contratação eletrônica tanto legal quanto tecnológica. Por isso, se torna imprescindível obter um contrato livre de conflitos para proteger os indivíduos envolvidos, seja em contratos legais ou em contratos tecnológicos.

2.2.2 Motivos de utilizar contratos eletrônicos

Ao utilizar um contrato eletrônico levamos sempre em conta a sua formação, o encontro da vontade proveniente das partes contratantes, sendo assim apto a produzir um efeito jurídico.

Segundo Diniz (2005) os contratos devem possuir duas partes importantes, a iniciativa que demonstra a vontade de celebrar o contrato e a outra parte se faz através da aceitação para a formação do contrato. Portanto, a oferta e a concordância são elementos indispensáveis à formação de qualquer contrato, visto que deverá haver o consentimento entre ambos, um com a proposta e outro com aceitação para o ponto final.

Por sua vez, entende que os contratos eletrônicos podem ser definidos como instrumentos que viabiliza a transmissão de formal digital principalmente pela

Internet permitindo a determinação de deveres e obrigações jurídicos. (WIELEWICKI, 2001, p. 198).

No entanto, o motivo primordial pelo qual os contratos eletrônicos são utilizados está relacionado à questão de possuir cláusulas ou termos em suas descrições, feito pelo intermédio dos contratantes afirmados entre si, de forma que possuam deveres e obrigações a serem cumpridos ou exercidos, os quais estejam previstos dentro da lei, para que no futuro não ocorram problemas.

Grandes organizações estão migrando para o uso do contrato eletrônico já que o mesmo se torna interessante devido a fácil aquisição e atinge um grande número de clientes através do uso do mesmo, como por exemplo, a internet. Viabilizando as organizações, possíveis em tomadas de decisões.

2.3 REPRESENTAÇÃO DE CONTRATOS ELETRÔNICOS

Em geral, um contrato possui ambiguidades devido à utilização da linguagem natural para descrevê-lo. A utilização de formalismos para expressar contratos evita as inconsistências e possibilita a verificação sistemática com apoio computacional. Por isso, uma das tarefas para se verificar contratos eletrônicos de forma automática é definir um formalismo adequado para expressá-los (FENECH, 2008).

A seguir são descritas as lógicas modais e dinâmicas, utilizadas para representar sistemas e propriedades, bem como a lógica deôntica convencional, indicada para representar sistemas normativos. Por fim, a CL (PRISACARIU; SCHNEIDER, 2012), uma linguagem baseada na lógica deôntica, na lógica dinâmica e nas lógicas temporais, é apresentada para a representação de contratos eletrônicos. Essa linguagem permite a especificação de contratos de forma intuitiva, capturando várias características dos contratos legais e evitando diversos paradoxos deônticos.

2.3.1 Lógica Modal

A lógica modal, dizendo de forma simples, é o estudo do comportamento dedutivo das expressões “é necessário que” e “é possível que”, ou seja, sua principal característica é capacidade de expressar necessidades e possibilidades (CHELLAS, 1998). A lógica modal se refere a qualquer sistema de lógica formal que

procure lidar com modalidades (tratar de modos quanto o tempo, possibilidade, probabilidade, etc.). Tradicionalmente, as modalidades mais comuns são possibilidades e necessidades.

O termo lógica modal descreve um grande número de lógicas na literatura que possuem características modais, como as lógicas epistêmicas (MEYER, 2001), lógicas temporais (PNUELLI, 1977) e lógica deôntica (VON WRIGHT, 1951). De forma simplificada, a lógica modal estende a lógica clássica com operadores modais que expressam não somente que uma proposição é verdadeira, mas que ela pode ser verdadeira ou necessariamente verdadeira (BLACKBURN; BENTHEM; WOLTER, 2006).

A sintaxe da lógica modal é baseada nos conectivos da Lógica Proposicional acrescido dos operadores modais de necessidade, denotado por $\Box(p)$ e de possibilidade, denotado por $\Diamond(p)$ (BLACKBURN; BENTHEM; WOLTER, 2006), conforme é apresentada a seguir:

$$\varphi ::= p \mid \perp \mid \top \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \rightarrow \varphi \mid \varphi \leftrightarrow \varphi \mid \Box \varphi \mid \Diamond \varphi$$

Apesar de a lógica modal abrir um leque extenso de possibilidades para as linguagens formais, seu uso muitas vezes se restringe às linguagens proposicionais, devido grande parte às dificuldades que surgem quando se adicionam a uma mesma linguagem os operadores modais (RAMOS, 2003). Com isso houve a indispensabilidade da inserção dos operadores de $\Box(p)$ de necessidade e $\Diamond(p)$ de possibilidade para completar a linguagem da lógica modal em relação à lógica proposicional.

O operador de necessidade pode ser representado da seguinte forma em lógica modal, por exemplo:

Tabela 1 - Exemplo utilizando o operador modal de necessidade.

É obrigatório ter a carteirinha de estudante para emprestar livros.	$\Box c$
É obrigatório ao aluno devolver o livro na data correta.	$\Box d$

O operador de possibilidade pode ser representado da seguinte forma em lógica modal, por exemplo:

Tabela 2 - Exemplo utilizando o operador modal de possibilidade.

Emprestar livro.	$\diamond e$
Renovar o empréstimo.	$\diamond r$

A semântica da lógica modal é definida considerando o conceito de mundos possíveis, sendo modelada utilizando uma estrutura de Kripke (KRIPKE, 1959). Esta estrutura é denotada pela tupla $M = \langle W, R, \models \rangle$, onde W é o conjunto de mundos possíveis, $R \subseteq W \times W$ é a relação de acessibilidade entre os mundos, de modo que se w e w' são mundos pertencentes a W e $(w, w') \in R$, o mundo w' é acessível a partir de w . Por fim, a relação entre os mundos possíveis e as proposições válidas em um determinado mundo é denotado por \models , e com isso, considerando que $w \in W$, se $(w, p) \in \models$, então p é verdadeiro no mundo w .

Um exemplo de modelo de Kripke pode ser dado conforme o grafo direcionado na Figura 1, considerando $M = \langle W, R, \models \rangle$ com as definições a seguir:

- $W = \{w_0, w_1, w_2\}$
- $R = \{(w_0, w_1), (w_0, w_2)\}$
- $\models = \{(w_0, p), (w_1, p), (w_1, q), (w_2, q)\}$

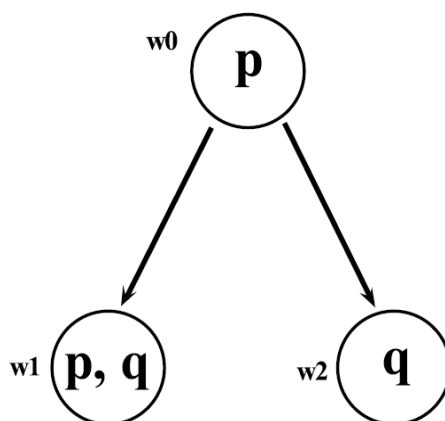


Figura 1- Exemplo de modelo de Kripke para lógica modal.

Neste exemplo, pode ser observado que $M, w_0 \not\models \Box(p)$, uma vez que o mundo alcançável $w_2 \not\models p$. Por outro lado, $M, w_0 \models \Diamond(p \wedge q)$, pois $w_1 \models p \wedge q$, ou seja, existe um mundo alcançável a partir de w_0 que satisfaz tanto p quanto q .

A lógica modal é uma extensão da lógica clássica que inclui operadores extras para lidar com inferências envolvendo necessidade e possibilidade. Por sua

vez, um cálculo simbólico, baseado em axiomas e regras de inferência, com a peculiaridade de tratar não apenas da verdade, mas também da verdade, que pode ser necessária, possível ou impossível.

2.3.2 Lógica Dinâmica

A lógica dinâmica (DL) é um sistema formal para raciocínio sobre programas (PRATT, 1976). Esta lógica é utilizada na formalização de especificações de correção e provas rigorosas para que essas especificações sejam atendidas por um programa em particular. Existem outras atividades que se enquadram nas aplicações da lógica dinâmica, como: determinar a equivalência dos programas, comparar o poder expressivo de varias construções de programas ou sistematizar programas a partir de especificações (HAREL; KOZEN; TIURYN, 1984).

A lógica dinâmica pode ser estendida como uma união da lógica de predicados, da lógica modal e da álgebra de eventos regulares (HAREL; KOZEN; TIURYN, 1984). A diferença principal da Lógica Dinâmica comparada com a Lógica Proposicional é que o valor verdade não é estático, assim como ocorre na Lógica Temporal. Já na Lógica Dinâmica existem construções sintáticas explícitas chamados programas, cujo papel principal é mudar os valores das variáveis, mudando assim, os valores verdade das fórmulas.

A lógica dinâmica proposicional (PDL), proposta por Fischer e Ladner (1979), é baseada na ideia original da lógica dinâmica (DL), associando um programa a com um operador modal de necessidade $[]$. A fórmula $[a]p$ significa que sempre o programa a terminar, deve fazê-lo num estado que satisfaça a proposição p . Um estado é a representação de um instante de tempo que descreve a realidade, associando um valor a cada variável do programa. Já, o operador modal da possibilidade $\langle a \rangle p$ representa que há uma computação a que termina num estado que satisfaça a proposição p .

A equivalência entre os operadores de necessidade e possibilidade pode ser denotada por:

$$\langle a \rangle p \equiv \neg[a]\neg p$$

Há dois conjuntos de símbolos presentes no sistema formal da PDL. O primeiro representa o conjunto de fórmulas atômicas, Φ_0 . O segundo conjunto é dos

programas atômicos, Σ_0 , considerados indivisíveis numa linguagem de programação. Os Programas atômicos utilizados na PDL são básicos e indivisíveis, ou seja, são executados em apenas um passo. A PDL se abstrai da natureza do domínio da computação e estuda a interação pura entre os programas e as proposições (HAREL, KOZEN e TIURYN; 1984).

As fórmulas são utilizadas para descrever propriedades que ocorrem depois da execução de certo programa. Por exemplo, a fórmula $\varphi = [a \cup b]p$ representa que sempre que o programa a ou o programa β são executados, é alcançado um estado onde a proposição a é verdadeira. Por outro lado, a fórmula $\varphi' = \langle (a; \beta) * \rangle p$ significa que há uma sequência de execuções alternadas de a e β que podem chegar até um estado em que a proposição p seja verdadeira.

Os conjuntos dos programas (Σ) e das fórmulas (Φ) são definidos indutivamente pelas seguintes regras (FISCHER; LADNER, 1979):

- Programas

1. Programas atômicos e θ são programas.
2. Se a e β são programas e φ é uma fórmula, então $a; \beta, a \cup \beta, a^*$ e $\varphi?$ são programas.

- Fórmulas

1. Fórmulas atômicas, \top e \perp são fórmulas.
2. Se p e q são fórmulas e a é um programa, então $p \wedge q, \neg p, \langle a \rangle p$ e $[a]p$ são fórmulas.

A sintaxe da PDL é formada por operadores da lógica modal Chellas (1980), associadas a estruturas algébricas. Já os operadores proposicionais $\leftrightarrow, \wedge, \vee, \rightarrow$ e \neg são definidos da sua maneira usual.

A semântica dos operadores da PDL, proposta por Fischer e Ladner (1979), é representada por uma estrutura $\mathcal{A} = \langle W, \pi, \rho \rangle$, onde:

- W é o conjunto de estados do programa.
- $\pi : \Phi_0 \rightarrow 2^W$ provê uma interpretação das fórmulas atômicas, e $w \in \pi(p)$ significa que a fórmula p é verdadeira no estado w .

- $\rho : \Sigma_0 \rightarrow 2^{W \times W}$ retorna uma interpretação sobre os programas atômicos, onde $(u, v) \in p(a)$ significa que existe uma execução do programa a que se inicia no estado u e termina no estado v .

O operador de necessidade pode ser representado da seguinte forma em lógica dinâmica, por exemplo:

Tabela 3 - Exemplo utilizando o operador dinâmico de necessidade em lógica dinâmica.

Mostrar a carteirinha de estudante.	$[c]e$
Pagar multa	$[\neg d \wedge \neg r]p$
Renovar o empréstimo	$[e]r$

A lógica Dinâmica é um instrumento criado para verificação do programa e para o raciocínio sobre a semântica dos programas e linguagens de programação. As fórmulas são utilizadas para descrever propriedades que ocorrem depois da execução de um certo programa.

2.3.3 Lógica Deôntica

A lógica deôntica pode expressar obrigações, permissões e proibições. A representação pode ser declarada pelas letras maiúsculas O (Obrigado), P (Permitido) e F (Proibido), no aspecto conceitual normativa expressa deveres, possibilidades e impossibilidades de certa ação.

O operador da lógica deôntica Pa representa que a ação a é permitida. Se uma ação a não é permitida, esta é denominada proibida e representada por Fa . Por último, se a negação de uma ação a é proibida, a ação é considerada obrigatória, denotada por Oa (VON WRIGHT, 1951).

Segundo Hilpinem (2001), muitos estudos apontam a lógica deôntica como uma ramificação da lógica modal e seus conceitos de obrigação, permissão e obrigação podem ser entendidos como as modalidades de necessidade (\square), possibilidade (\diamond) e impossibilidade ($\neg \diamond$).

Segundo Wieringa e Meyer (1993) a lógica deôntica foi, originalmente, proposta para definir estruturas normativas da lei e possibilitar a sua verificação. No entanto, essa lógica pode ser aplicada nas mais diversas áreas onde deseja-se

verificar o comportamento de sistemas. Alguns exemplos de aplicações da lógica deôntica são:

- **Tolerância a falhas em sistemas:** nenhum sistema de computador é livre de falhas e muitas vezes é recomendado que seja definido o que acontece quando ocorre uma falha no sistema. Esta técnica é conhecida como tratamento de exceções, em que o sistema tolerante a falhas pode entrar em um comportamento que garanta seu funcionamento mesmo depois de um problema ocorrido.
- **Normativas de comportamento de usuários:** é claro que nenhum sistema está pronto para tratar todos os tipos de comportamentos dos usuários, pois suas ações são muitas vezes imprevisíveis. O tratamento de erros nos sistemas utilizando a lógica deôntica pode ajudar a controlar os erros criados pelos usuários já que é possível determinar o comportamento aceito e com isso indicar as ações do sistema.
- **Especificação de políticas:** com a lógica deôntica é possível descrever o comportamento dos componentes dentro das organizações. Assim sendo, a definição das regras e políticas das organizações precisam definir o que pode ou não ser realizado e o que ocorre quando uma norma não é respeitada. Neste caso, a lógica deôntica pode ser utilizada para definir políticas não ambíguas e indicar as consequências de uma violação.
- **Especificação de normas legais:** a representação da lei é uma área vasta e muito explorada da lógica deôntica mesmo antes de sua automação pelos computadores. Desta maneira, atos e leis são formalizados e manipulados de acordo com as regras de alguma lógica deôntica. A formalização das normas utilizando lógica permite que sistemas sejam capazes auxiliar advogados e juízes nas mais diversas tarefas que envolvem leis e normas.

Por exemplo, na lógica deôntica é possível expressar situações reais do nosso dia-a-dia como essa:

- O código de trânsito brasileiro não permite você dirigir enquanto bebe ou fuma então:

$$P(\text{dirigir}) \wedge F(\text{dirigir} \wedge \text{beber}) \wedge F(\text{dirigir} \wedge \text{fumar})$$

- Você é obrigado a pagar seus impostos do carro e é proibido dirigir se você não pagar o IPVA:

$$O(\text{pagarIPVA}) \wedge F(\text{dirigir} \wedge \neg \text{pagarIPVA})$$

Na lógica deôntica esta correlata aos conceitos de obrigação e permissão (bem como o de proibição), respectivamente, visto que será o que os envolvidos no contrato deve fazer ou satisfazer para que seja executada uma ação.

2.3.4 A linguagem de contratos \mathcal{CL}

A linguagem para contratos, denominada \mathcal{CL} é proposta por Prisacariu e Schneider (2012). A \mathcal{CL} é uma linguagem desenvolvida para a representação formal de contratos legais, serviços web, interfaces e protocolos de comunicação. Essa linguagem é expressiva o suficiente para capturar comportamentos dos contratos e permitir sua verificação formal, sendo aplicada em estudos sobre model checking (PACE; PRISACARIU; SCHNEIDER, 2007), verificação de conflitos (FENECH; PACE; SCHNEIDER, 2009) e no monitoramento de contratos (KYAS; PRISACARIU, SCHNEIDER, 2008). A \mathcal{CL} é baseada em ações síncronas e nos princípios da lógica deôntica, para definir normas (VON WRIGHT, 1951), e da lógica dinâmica proposicional, para representar as ações num sistema (HAREL; KOZEN; TIURYN, 1984). A parte deôntica da \mathcal{CL} representa os operadores de permissão, obrigação e proibição, bem como a compensação sobre a violação de obrigações ou proibições. Já a parte dinâmica expressa quais cláusulas serão válidas após alguma ação ser executada.

A \mathcal{CL} linguagem do contrato é uma lógica (PRISACARIU; SCHNEIDER, 2009), que pode ajudar o processo de análise e automatização de contratos: detecção de contradições e inconsistências, a identificação de cláusulas supérfluas e verificando algumas propriedades desejadas em um contrato. No entanto segundo (PRISACARIU; SCHNEIDER, 2007). Uma característica interessante da \mathcal{CL} é a

captura de conceitos e propriedades naturais encontrados em contratos legais e eletrônicos buscando evitar os principais paradoxos deônticos.

A sintaxe da \mathcal{CL} é apresentada na Tabela 4. O símbolo C representa uma cláusula do contrato. As cláusulas podem conter operadores lógicos \rightarrow | \wedge | \vee | \neg | \leftrightarrow | \top que têm o mesmo significado da lógica proposicional e podem ser representados de forma simplificada com os operadores \rightarrow e \perp . O conjunto Φ_B contém as constantes proposicionais ϕ . Uma constante proposicional descreve um caso ou um determinado acontecimento, por exemplo, "ocorreu um sinistro" ou "o valor do reparo é menor que R\$500".

Tabela 4 - Sintaxe da Linguagem de Contratos \mathcal{CL} (PRISACARIU; SCHNEIDER, 2012).

C	$:=$	$\phi Oc(a) P(a) Fc(a) C \rightarrow C [\delta]C \perp$	(Expressões em \mathcal{CL})
a	$:=$	$a 0 1 a \times a a . a a + a$	(ações deônticas)
δ	$:=$	$a 0 1 \delta \times \delta \delta . \delta \delta + \delta \delta^* \varphi?$	(ações dinâmicas)
φ	$:=$	$\phi 0 1 \varphi \vee \varphi \varphi \wedge \varphi \neg \varphi$	(testes lógicos)

As ações deônticas consistem num conjunto de ações básicas (ou atômicas). Além disso, existem os tipos especiais de ações que não pertencem ao conjunto, representadas por 0 e 1. A ação 0 representa uma violação do contrato e a ação 1, chamada a ação de salto (skip), que representa a execução de qualquer ação do conjunto e indica que naquele momento do contrato é indiferente qual ação é executada. As ações podem ser combinadas pelos operadores: +, que indicam uma escolha entre as ações; ., que expressam uma sequência de ações; e x, que representam ações síncronas ocorrendo ao mesmo tempo. Nesta linguagem, os indivíduos são incorporados nas ações, seguindo o conceito da lógica deôntica padrão (SDL) (PACE; PRISACARIU; SCHNEIDER, 2007), o que impede a identificação e análise dos envolvidos no contrato.

As modalidades deônticas $Oc(a)$, $Fc(a)$, $P(a)$ representam respectivamente os operadores de obrigação, proibição e permissão da lógica deôntica padrão, onde a é uma ação e C representa a penalidade ou reparação no caso da cláusula ser violada. O operador $Oc(a)$ indica que a ação a deve ser executada e, em caso de violação uma penalidade deve ser imposta, especificada na cláusula C . O mesmo acontece com o operador $Fc(a)$ onde a execução de a exige a execução de C .

Estas estruturas de reparação representam as cláusulas contrárias ao dever (CTD) e contrárias à proibição (CTP). As cláusulas CTD representam situações em que há uma violação de obrigação primária que implica em executar uma obrigação secundária. Essa obrigação secundária é uma reparação ou penalidade da violação gerada pela obrigação primária. O mesmo ocorre para o CTP que trata da violação de uma proibição (PRAKKEN; SERGOT, 1996).

Obrigações ou proibições sem reparação, $O_{\perp}(a)$ e $F_{\perp}(a)$, também podem ser representadas, onde o símbolo \perp tem valor falso. Obrigações (ou proibições) sem reparações são denominadas de categóricas, pois não podem ser violadas de forma alguma. Para facilitar a sintaxe, as obrigações categóricas podem ser representadas por $O(a)$ e, do mesmo modo, as proibições como $F(a)$. As permissões, por sua vez, não possuem nenhum tipo de reparação já que são facultativas e não implicam numa violação.

Segundo Prisacariu e Schneider (2012):

[...] o projeto da CL e, especialmente, sua semântica é impulsionada pela necessidade de expressar e raciocinar sobre contrato. Mais particularmente, a CL precisa raciocinar obrigações, permissões e proibições, mas específico, sobre as ações e o fato de que tais ações são obrigatórias, permitidas e proibidas.

Devido à CL ser baseada na lógica dinâmica proposicional e na lógica deôntica padrão, algumas propriedades destas duas lógicas também são observadas nesta linguagem de contratos, conforme são descritas a seguir:

- A obrigação de uma ação implica na sua permissão: $O_c(a) \leftrightarrow O_c(a)$.
Exemplo: “Se o cliente é obrigado a pagar então o cliente tem a permissão de pagar”.
- A permissão de uma ação implica que esta ação não é proibida: $P(a) \rightarrow \neg F_c(a)$.
Exemplo: “Se o provedor tem o direito de alterar dados pessoais do cliente então o provedor não é proibido de alterar os dados pessoais do cliente”.
- Se duas expressões representam a mesma ação, então a obrigação de uma ação implica na obrigação da outra ação: se $a = \beta$ então $O_c(a) \leftrightarrow O_c(\beta)$.
- A obrigação de uma violação jamais deve ocorrer no contrato: $\neg O_c(0)$.

- A obrigação de executar qualquer ação, em que a ação realizada é indiferente no contrato, pode ser inserida em qualquer contrato: $O_c(1)$.

2.3.5 Conflitos

Um conflito é uma situação em que as cláusulas remetem a uma situação impossível de ser resolvida. Enquanto uma norma proíbe uma certa ação e uma outra permite a execução dessa ação, ou quando uma norma proíbe a ação uma outra a obriga. Essa situação pode ser demonstrada pela incompatibilidade entre as normas que obrigam e proíbem ou as que permitem e proíbem. Para evitar estes conflitos, é necessário garantir que o contrato não possua situações em que qualquer ação tomada leva a uma violação (FENECH, 2008).

Os conflitos podem ser verificados tanto com operadores deônticos conflitantes com a mesma ação quanto com ações distintas, situação na qual estas ações seguem o princípio da exclusão mútua e, sendo assim, não podem ser executadas ao mesmo tempo. Essa relação de conflito é representada em CL pelo símbolo #. Desta maneira, a exclusão entre as ações a e b pode ser representada por $a \# b$.

Um exemplo de um contrato em conflito seria uma obrigação para abrir a porta e, ao mesmo tempo proibição de fazê-lo.

De acordo com Fenech (2009), conflitos podem acontecer por quatro motivos:

1. Obrigações e proibições aplicados na mesma ação;
Exemplo: $O(\text{abrirJanela}) \wedge F(\text{abrirJanela})$
2. Permissão e proibição aplicados na mesma ação;
Exemplo: $P(\text{abrirJanela}) \wedge F(\text{abrirJanela})$
3. Obrigação de executar ações mutuamente exclusivas;
Exemplo: $O(\text{abrirJanela}) \wedge O(\text{fecharJanela})$
4. Permissão e obrigação de executar ações mutuamente exclusivas.
Exemplo: $O(\text{abrirJanela}) \wedge P(\text{fecharJanela})$

Entretanto, um outro tipo de conflito não pode ser caracterizado pelas definições dadas. Por exemplo, na fórmula $O(a + b) \wedge F(a)$ existe uma escolha na obrigação de executar a ação a ou a ação b . Caso seja escolhida a ação a , um conflito vai ocorrer com a proibição de a , mas se a ação b é executada, o conflito não ocorre. Este tipo de situação é chamado de conflito fraco e não caracteriza um problema, já que na ocorrência de conflitos não há escolhas para se evitar uma violação.

Deve-se notar que uma vez que o contrato está solicitando essas condições conflitantes não há nenhuma maneira em que o contrato pode ser satisfeita. É o próprio contrato que está nos colocando em uma posição em que não pode satisfazê-lo e, assim, tal contrato não é desejado.

Dentre os conflitos que acima descritos podemos deparar com as seguintes circunstâncias: em primeiro lugar, pode-se obrigar e proibir de fazer a mesma ação, e, segundo, que se pode permitir e proibir de realizar a mesma ação. Os demais casos correspondem a obrigações e obrigações; permissões e obrigações de ações mutuamente exclusivas.

A abordagem de detecção de conflitos em CL baseia-se em técnicas de verificação de modelos, onde propriedades desejadas sejam satisfeitas no modelo que especifica o contrato. A propriedade que define um contrato livre de conflitos pode ser descrita como: "dada qualquer sequência de ações que não viola o contrato, a execução do contrato não termina num estado que ocorre um conflito". Para todos os caminhos de computação válidos no modelo, nenhum destes caminhos podem chegar a um estado que possua conflitos.

2.3.6 Ferramenta CLAN

A ferramenta CLAN foi desenvolvida para fazer a identificação de conflitos nas cláusulas do contrato para verificar se ele está livre de conflitos e não. A interface da ferramenta, conforme a figura 2.

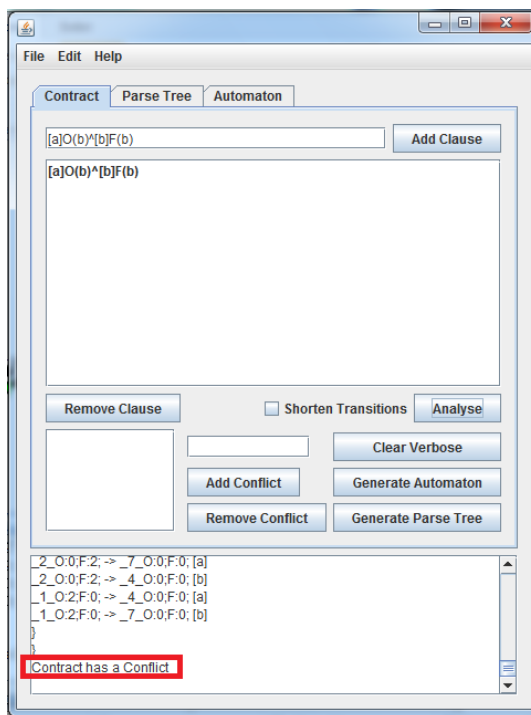


Figura 2 – Ferramenta CLAN analisando conflitos.

A ferramenta, ao verificar se a fórmula $[a]O(b)^{[b]}F(b)$ como exemplo, identifica que contém conflito na formulação. Também há a possibilidade de gerar um autômato dessa fórmula para verificar em qual momento se ocorre o conflito, como podemos ver na figura 3.

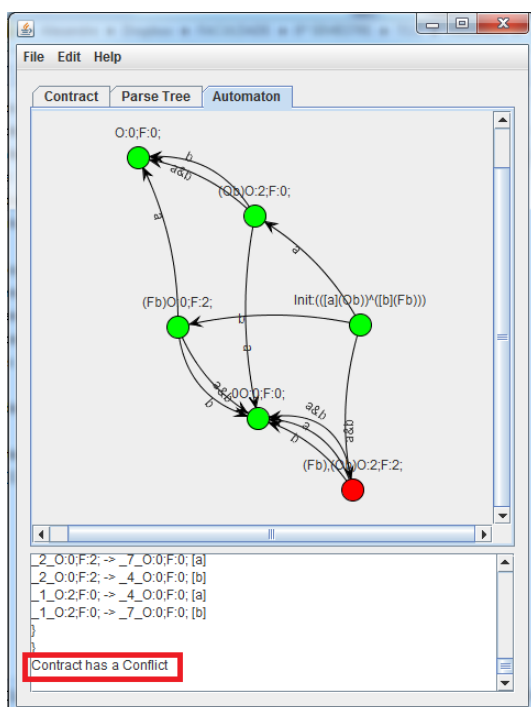


Figura 3 – Autômato $[a]O(b)^{[b]}F(b)$ com conflito.

Pode-se verificar que no autômato anterior, encontra-se a localização que ocorreu o conflito, o que facilita a identificação do estado que se gera a situação de conflito.

A Ferramenta CLAN nos permite colocar em situação de conflito, representada na figura 4, na área identificada. No exemplo tem-se a situação de conflito $a\#b$, que condiz a ação “a” deve conflitar com “b”.

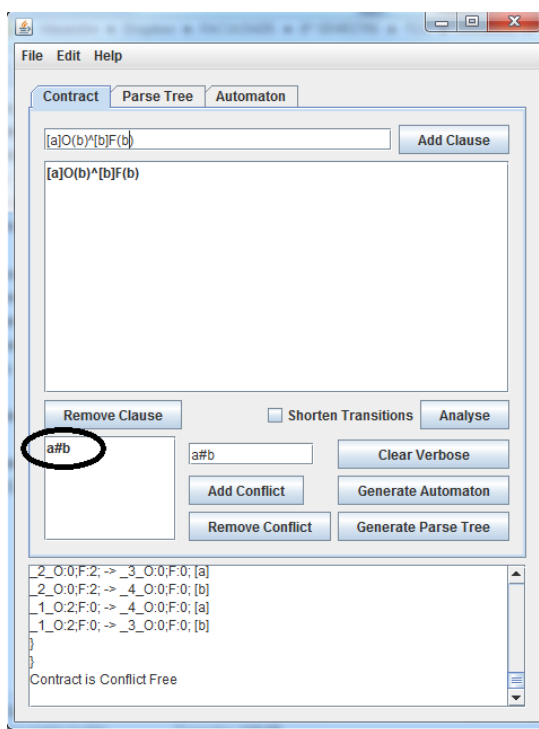


Figura 4 – Adicionando conflito na Ferramenta CLAN.

Ao adicionar uma situação de conflito a CLAN, aonde havia um conflito, automaticamente fazendo a verificação novamente, identifica-se que o conflito foi removido, pois se tornou uma situação oposta de conflito, conforme podemos observar no autômato da figura 5.

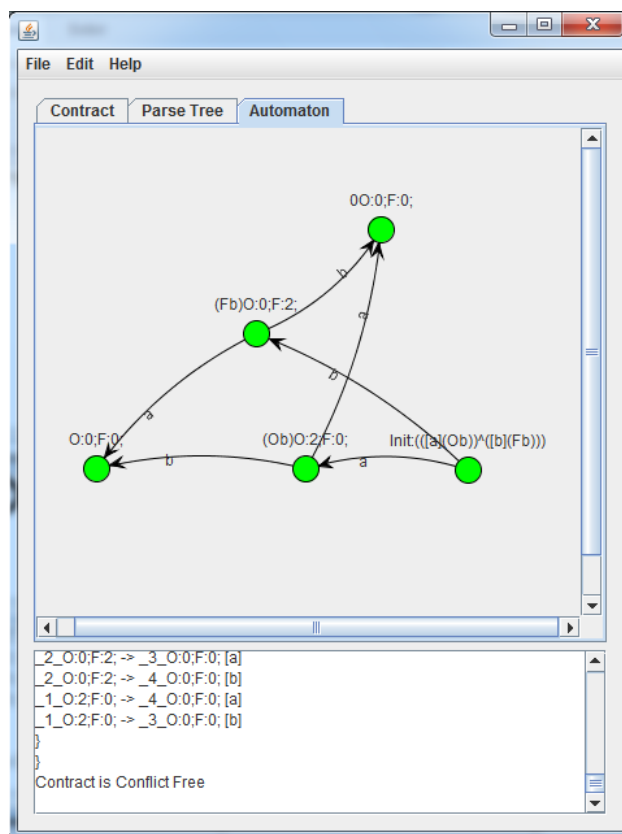


Figura 5 – Autômato livre de conflito.

Como pode ver a ferramenta CLAN auxilia no processo de verificação e identificação de conflito, além de fazer com que sejam inseridos os conflitos para seu teste.

Depois de verificar na ferramenta CLAN, ela nos cria um autômato e o mesmo verifica se existe conflito ou não. Para melhor identificação e entendimento, a figura 6, que demonstra o autômato gerado.

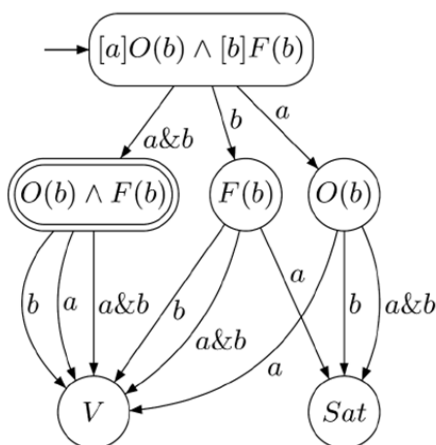


Figura 6 – Autômato gerado do contrato $[a]O(b) \wedge [b]F(b)$.

Para ilustrar a aplicação do algoritmo, assumamos o contrato $C = [a]O(b) \wedge [b]F(b)$, cujo autômato obtido é mostrado na Figura 6. O autômato é construído com base na função f_c a partir do estado inicial S_0 . A rotulação atribuída é $l(S_0) = [a]O(b) \wedge [b]F(b) \wedge (s_0)$. A partir desta chamada todas as transições possíveis são criadas, utilizando o alfabeto de ações concorrentes $\{a, b, a \& b\}$, partindo deste estado S_0 para um novo estado rotulado com o resultado da função f . Neste caso, f é executado com a fórmula original e o rótulo da transição dando origem ao rótulo obtido com a ação a , denotado por $f([a]O(b) \wedge [b]F(b), a) = O(b)$. Se houver outro estado s' com o mesmo rótulo, uma transição é criada para este estado s' . Caso contrário, é criado um novo estado e a função f_c é chamada novamente para este estado criado. O algoritmo termina quando é encontrado um estado que satisfaça a fórmula, quando o rótulo obtido é vazio, ou quando ocorre uma violação num estado já rotulado com a fórmula.

3. TESTE DE APLICAÇÃO DOS FORMALISMOS

Este capítulo consiste em teste de aplicação dos formalismos, demonstra-se a representação do contrato nas lógicas modal, lógica dinâmica e lógica deôntica, e na linguagem de Contrato CL.

Para iniciar se deve ter um contrato. Neste caso utiliza-se o Contrato Serviço de Internet, para que se faça uso no teste de aplicação de formalismo. O contrato será descrito no Anexo 1.

3.1 CONTRATO SERVIÇO DE INTERNET

Apresenta-se uma parte de um contrato convencional entre um fornecedor de serviços e um cliente, em que o prestador dá acesso à Internet para o cliente (PACE; PRISACARIU; SCHNEIDER, 2007).

No que se segue considera-se a parte 7 (PARTE DECISÓRIA) do contrato dado do Anexo 1 entre um prestador de serviços e um cliente, em que o prestador dá acesso à Internet para o cliente. Existem dois parâmetros do serviço: altos e normais, que denota o tráfego da Internet do cliente. Sendo assim, utiliza-se apenas as seguintes cláusulas do o contrato.

7.1. O cliente não deve:

a) fornecer informações falsas ao Departamento de Relações com o Cliente do Provedor.

7.2. Sempre que o tráfego Internet é alto, em seguida, o cliente deve pagar [preço] imediatamente, ou o Cliente deverá notificar o provedor, enviando um e-mail especificando que ele vai pagar mais tarde.

*7.3. Se o cliente atrasa o pagamento, conforme estipulado no item 7.2, após a notificação, ele deve imediatamente diminuir o tráfego da Internet para o nível normal, e pagar depois duas vezes (2 * [preço]).*

*7.4. Se o cliente não diminuir o tráfego da Internet imediatamente, em seguida, o cliente terá de pagar 3 * [preço].*

7.5. O Cliente deverá, assim que o serviço de Internet torna-se operacional, apresentar, no prazo de 7 (sete) dias, o formulário de dados pessoais de sua conta

na página web do Provedor para o Departamento de Relações com o Cliente do Provedor.

Também é adicionada a cláusula 11.2, uma vez que está fortemente relacionada com a cláusula 7.1 e os dois devem ser tomados em conjunto:

11.2. Provedor poderá, a seu exclusivo critério, sem aviso prévio ou indicar qualquer motivo ou incorrer em qualquer responsabilidade para fazê-lo:

b) suspender os serviços de Internet imediatamente se o cliente está em descumprimento da Cláusula 7.1

Assim nas seções a seguir foram apresentados os testes de aplicação dos formalismos, a representação do contrato nas lógicas modal, lógica dinâmica e lógica deôntica, e na linguagem de Contrato CL.

3.2 REPRESENTAÇÃO UTILIZANDO LÓGICA MODAL

Este contrato pode ser representado em Lógica Dinâmica, como parte da formalização de um contrato primeiramente tem que definir as afirmações e ações:

a = o tráfego da Internet é alto;

b = cliente fornece informações falsas ao Departamento de Relações com Cliente;

c = cliente aumenta o tráfego de Internet para o nível alto;

d = cliente paga [preço];

e = cliente atrasa pagamento;

f = cliente notifica por e-mail;

g = cliente diminui o tráfego de Internet;

h = cliente envia o Formulário de Dados Pessoais a área de Relações com Cliente;

i = ativa o provedor de serviços de Internet (que torna-se operacional);

j = provedor suspende serviço.

A fórmula é detalhada da seguinte forma:

7.1. O cliente não deve:

a) fornecer informações falsas ao Departamento de Relações com o Cliente do Provedor.

$\diamond \neg b$

7.2. Sempre que o tráfego Internet é alto, em seguida, o cliente deve pagar [preço] imediatamente, ou o Cliente deverá notificar o provedor, enviando um e-mail especificando que ele vai pagar mais tarde.

$\square a \rightarrow (\square d \vee \square f)$

7.3. Se o cliente atrasa o pagamento, conforme estipulado no item 7.2, após a notificação, ele deve imediatamente diminuir o tráfego da Internet para o nível normal, e pagar depois duas vezes (2 * [preço]).

$\diamond e \rightarrow \square g \wedge \square d$

Não é possível representar a ação de pagar 2 vezes

7.4. Se o cliente não diminuir o tráfego da Internet imediatamente, em seguida, o cliente terá de pagar 3 * [preço].

$\diamond \neg g \rightarrow \square d$

Não é possível representar a ação de pagar 3 vezes

7.5. O Cliente deverá, assim que o serviço de Internet torna-se operacional, apresentar, no prazo de 7 (sete) dias, o formulário de dados pessoais de sua conta na página web do Provedor para o Departamento de Relações com o Cliente do Provedor.

$\square i \rightarrow \square h$

11.2. Provedor poderá, a seu exclusivo critério, sem aviso prévio ou indicar qualquer motivo ou incorrer em qualquer responsabilidade para fazê-lo:

b) suspender os serviços de Internet imediatamente se o cliente está em descumprimento da Cláusula 7.1

$\diamond j$

3.3 REPRESENTAÇÃO UTILIZANDO LÓGICA DINÂMICA

Este contrato pode ser representado em Lógica Dinâmica, como parte da formalização de um contrato primeiramente tem que definir as afirmações e ações:

a = o tráfego da Internet é alto;

b = cliente fornece informações falsas ao Departamento de Relações com Cliente;

c = cliente aumenta o tráfego de Internet para o nível alto;

d = cliente paga [preço];

e = cliente atrasa pagamento;

f = cliente notifica por e-mail;

g = cliente diminui o tráfego de Internet;

h = cliente envia o Formulário de Dados Pessoais a área de Relações com Cliente;

i = ativa o provedor de serviços de Internet (que torna-se operacional);

j = provedor suspende serviço.

A fórmula é detalhada da seguinte forma:

7.1. O cliente não deve:

a) fornecer informações falsas ao Departamento de Relações com o Cliente do Provedor.

Não é possível representar

7.2. Sempre que o tráfego Internet é alto, em seguida, o cliente deve pagar [preço] imediatamente, ou o Cliente deverá notificar o provedor, enviando um e-mail especificando que ele vai pagar mais tarde.

$[a](d \vee f)$

7.3. Se o cliente atrasa o pagamento, conforme estipulado no item 7.2, após a notificação, ele deve imediatamente diminuir o tráfego da Internet para o nível normal, e pagar depois duas vezes ($2 * [\text{preço}]$).

$[e]g \rightarrow [g]d$

Não é possível representar a ação pagar 2 vezes

7.4. Se o cliente não diminuir o tráfego da Internet imediatamente, em seguida, o cliente terá de pagar 3 * [preço].

(g)d

Não é possível representar a ação pagar 3 vezes

7.5. O Cliente deverá, assim que o serviço de Internet torna-se operacional, apresentar, no prazo de 7 (sete) dias, o formulário de dados pessoais de sua conta na página web do Provedor para o Departamento de Relações com o Cliente do Provedor.

[i]j

11.2. Provedor poderá, a seu exclusivo critério, sem aviso prévio ou indicar qualquer motivo ou incorrer em qualquer responsabilidade para fazê-lo:

b) suspender os serviços de Internet imediatamente se o cliente está em descumprimento da Cláusula 7.1

(j)b

3.4 REPRESENTAÇÃO UTILIZANDO LÓGICA DEÔNICA

Este contrato pode ser representado em Lógica Dinâmica, como parte da formalização de um contrato primeiramente tem que definir as afirmações e ações:

a = o tráfego da Internet é alto;

b = cliente fornece informações falsas ao Departamento de Relações com Cliente;

c = cliente aumenta o tráfego de Internet para o nível alto;

d = cliente paga [preço];

e = cliente atrasa pagamento;

f = cliente notifica por e-mail;

g = cliente diminui o tráfego de Internet;

h = cliente envia o Formulário de Dados Pessoais a área de Relações com Cliente;

i = ativa o provedor de serviços de Internet (que torna-se operacional);

j = provedor suspende serviço.

A fórmula é detalhada da seguinte forma:

7.1. O cliente não deve:

a) fornecer informações falsas ao Departamento de Relações com o Cliente do Provedor.

F(b)

7.2. Sempre que o tráfego Internet é alto, em seguida, o cliente deve pagar [preço] imediatamente, ou o cliente deverá notificar o provedor, enviando um e-mail especificando que ele vai pagar mais tarde.

O(a) → O(dvf)

7.3. Se o cliente atrasa o pagamento, conforme estipulado no item 7.2, após a notificação, ele deve imediatamente diminuir o tráfego da Internet para o nível normal, e pagar depois duas vezes (2 * [preço]).

P(e) → O(g^d) = Não consegui representar pagar 2 vezes

7.4. Se o cliente não diminuir o tráfego da Internet imediatamente, em seguida, o cliente terá de pagar 3 * [preço].

F(g) → O(d)

Não consigo representar pagar 3 vezes

7.5. O Cliente deverá, assim que o serviço de Internet torna-se operacional, apresentar, no prazo de 7 (sete) dias, o formulário de dados pessoais de sua conta na página web do Provedor para o Departamento de Relações com o Cliente do Provedor.

P(i) → O(h)

11.2. Provedor poderá, a seu exclusivo critério, sem aviso prévio ou indicar qualquer motivo ou incorrer em qualquer responsabilidade para fazê-lo:

b) suspender os serviços de Internet imediatamente se o cliente está em descumprimento da Cláusula 7.1

P(j)

3.5 REPRESENTAÇÃO LINGUAGEM DE CONTRATO CL

Este contrato pode ser representado em Lógica Dinâmica, como parte da formalização de um contrato primeiramente tem que definir as afirmações e ações:

a = o tráfego da Internet é alto;

b = cliente fornece informações falsas ao Departamento de Relações com Cliente;

c = cliente aumenta o tráfego de Internet para o nível alto;

d = cliente paga [preço];

e = cliente atrasa pagamento;

f = cliente notifica por e-mail;

g = cliente diminui o tráfego de Internet;

h = cliente envia o Formulário de Dados Pessoais a área de Relações com Cliente;

i = ativa o provedor de serviços de Internet (que torna-se operacional);

j = provedor suspende serviço.

A fórmula é detalhada da seguinte forma:

7.1. O cliente não deve:

a) fornecer informações falsas ao Departamento de Relações com o Cliente do Provedor.

F(b)

7.2. Sempre que o tráfego Internet é alto, em seguida, o cliente deve pagar [preço] imediatamente, ou o Cliente deverá notificar o provedor, enviando um e-mail especificando que ele vai pagar mais tarde.

[a]O(d + f)_O(d. d)

7.3. Se o cliente atrasa o pagamento, conforme estipulado no item 7.2, após a notificação, ele deve imediatamente diminuir o tráfego da Internet para o nível normal, e pagar depois duas vezes (2 * [preço]).

[e]O(d. d)

7.4. Se o cliente não diminuir o tráfego da Internet imediatamente, em seguida, o cliente terá de pagar 3 * [preço].

[g]O(d. d. d)

7.5. O cliente deverá, assim que o serviço de Internet torna-se operacional, apresentar, no prazo de 7 (sete) dias, o formulário de dados pessoais de sua conta

na página web do Provedor para o Departamento de Relações com o Cliente do Provedor.

O(h)

11.2. Provedor poderá, a seu exclusivo critério, sem aviso prévio ou indicar qualquer motivo ou incorrer em qualquer responsabilidade para fazê-lo:

b) suspender os serviços de Internet imediatamente se o cliente está em descumprimento da Cláusula 7.1

[b]P(j)

3.6 ANÁLISES DOS TESTES APLICADOS

Ao analisar os contratos anteriormente descritos, a aplicação dos testes de formalismos assim representados pelo contrato de serviço de Internet, através da lógica modal, lógica dinâmica, lógica deôntica e linguagem de contrato CL, dentro do ponto de vista, as lógicas respectivamente acima utilizadas, encontra-se uma dificuldade de se representar por meio do formalismo, em determinados casos impossível de descrever suas expressões para representação.

Já se utilizando a linguagem de contrato CL para aplicar o formalismo, nas expressões, nota-se, mais clareza na interpretação e além de poder utilizar a reparação em determinados casos, pois as reparações são chamadas de cláusulas para que se complete uma cláusula se após a execução ela for falsa, então se faz a chamada de outra cláusula para se satisfaça a execução.

A CL é uma linguagem desenvolvida para a representação formal de contratos legais, serviços web, interfaces e protocolos de comunicação. Essa linguagem é expressiva o suficiente para capturar comportamentos dos contratos e permitir sua verificação formal, sendo aplicada em estudos sobre model checking (PACE; PRISACARIU; SCHNEIDER, 2007), verificação de conflitos (FENECH; PACE; SCHNEIDER, 2009) e no monitoramento de contratos (KYAS; PRISACARIU, SCHNEIDER, 2008).

4. METODOLOGIA

Os materiais se referem às tecnológicas e ferramentas utilizadas para modelar e implementação da proposta deste trabalho. O método está relacionado às principais atividades desenvolvidas para a realização deste trabalho.

4.1 MATERIAIS

As ferramentas e as tecnologias utilizadas no desenvolvimento deste trabalho foram o Astah Professional para a construção de diagramas, a Linguagem Java como linguagem de programação, IDE NetBeans 8.0.1 para construção das interfaces do projeto, a ferramenta CLAN para verificação de conflitos em contratos.

A escolha da ferramenta Astah Professional foi devido à sua facilidade de interação com o usuário e por ser gratuita, no qual o proprietário disponibilizou a licença para os acadêmicos da UENP – Universidade Estadual do Norte do Paraná – Campus Luiz Meneghel. Com essa ferramenta foi possível desenvolver todos os diagramas necessários para a modelagem do sistema que são resultados deste trabalho.

A ferramenta IDE NetBeans 8.0.1 e linguagem de programação Java, devido às compatibilidades entre ambos. O Netbeans é um ambiente integrado de desenvolvimento (IDE) que permite ao programador criar programas utilizando recursos gráficos. É um programa de computador da Sun sob licença free (gratuita) e possui uma Interface Gráfica com o Usuário (GUI) para desenvolvimento de softwares. As suas ferramentas de desenvolvimento são integradas com as mais recentes tecnologias profissionais para a criação e manutenção de grandes projetos na linguagem Java.

A CLAN é uma ferramenta de análise automática de conflito de cláusulas de contratos escritos na linguagem de contrato *CL*, permitindo: (i) A análise automática do contrato de conflitos normativos; (ii) a geração automática de um monitor para o contrato. As análises são particularmente úteis quando o contrato está sendo escrito (que permite a descoberta de conflitos indesejáveis), antes de aderir a um determinado contrato (para assegurar a aplicação inequívoca do contrato), e durante a execução de contratos (monitoramento) (FENECH; PACE; SCHNEIDER, 2009).

4.2 PASSOS METODOLÓGICOS

Fundamentação Teórica: Os conceitos abordados neste trabalho são: contratos, contratos eletrônicos, conflitos, dificuldades para realizar um contrato e as lógicas: modal, dinâmica e deôntica, a linguagem de contrato CL. Todas as referências pesquisadas são científicas, que foram buscadas em revistas, livros, artigos, home page, sítios científicos, dentre outros meios de buscas.

Aplicação de teste dos formalismos: caracterizou-se pela aplicação da parte de um contrato de uma empresa de prestação de serviço de Internet, assim aplicando esse contrato nas lógicas modais, dinâmica, deôntica e linguagem de contrato CL. Após a aplicação foram analisados os contratos assim representados pelo formalismo e aderindo a linguagem de contrato CL para representar os contratos na ferramenta desenvolvida, pois a mesma é uma linguagem desenvolvida para a representação de contratos.

5. DESENVOLVIMENTO DO SISTEMA

Neste capítulo apresenta-se o sistema proposto, a modelagem do sistema, o desenvolvimento do sistema proposto e a demonstração do resultado.

5.1 APRESENTAÇÃO DO SISTEMA

O sistema analisado, modelado e desenvolvido e tem como finalidade auxiliar na criação de contratos, visando que os contratos são muito importantes para a área de negociação.

O sistema tem uma tela que facilita a elaboração de contrato, conforme figura 7, visando que o mesmo gerará fórmulas para que possa ser utilizado na ferramenta CLAN, para se averiguar que o contato esteja livre de conflitos ou divergências.

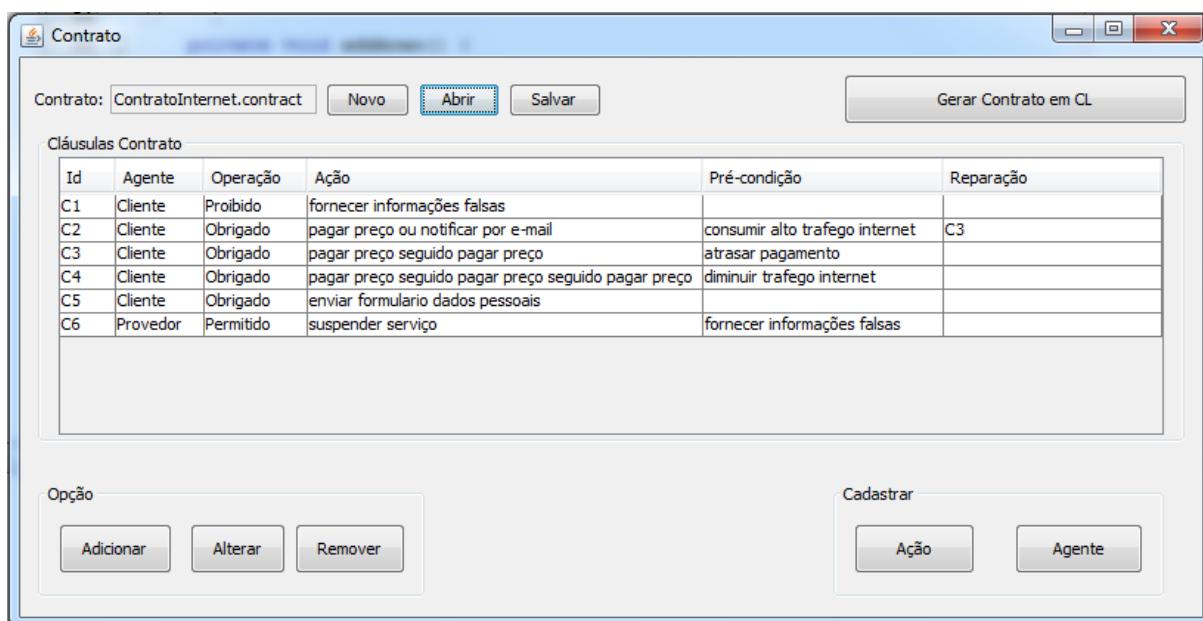


Figura 7 – Interface do Sistema desenvolvido

O sistema possibilita ao usuário acesso a todas as suas atividades, as quais podem ser exercidas no sistema. Consistem em que o usuário pode cadastrar agente, ações, cláusulas, remover, alterar e gerar o contrato em CL. Conforme os passos respectivamente na Figura 8.

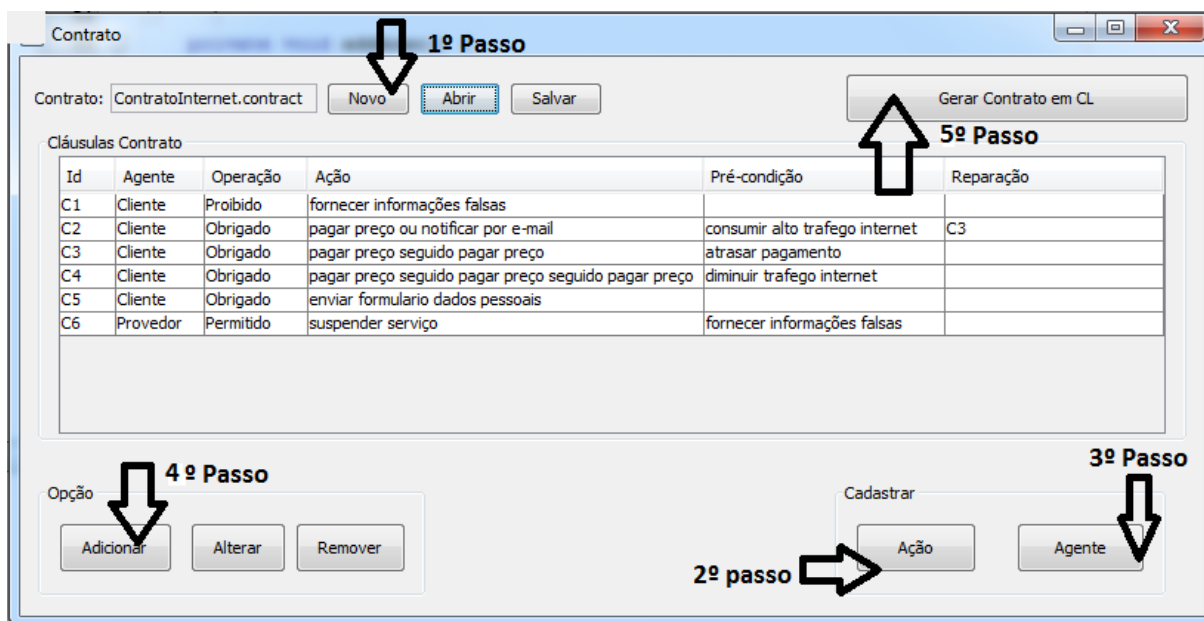


Figura 8 – Passos para fazer as atividades do sistema.

O contrato pode ser criado da seguinte maneira: 1º Passo: O usuário deve criar um “novo contrato”. 2º Passo: “cadastrar” as ações do contrato, logo após o 3º Passo: deve “cadastrar” os agentes que fazem parte do contrato. Feito todos os cadastros de ações e agentes, da início a criação do contrato. 4º Passo: O usuário aciona o botão “adicionar” selecionando as opções que contém nas cláusulas do contrato, na tela cláusulas do contrato, conforme a figura 9, como as “pré-condições”, se é necessário que aconteça alguma ação ou nenhuma, caso não possua a ação ele pode cadastrá-la. Seguindo o processo, se faz a seleção de “quem” esta fazendo a ação e o que o mesmo estará “autorizado ou proibido” de fazer determinada “ação”, por exemplo: “O cliente está proibido de fornecer informações falsas”. Depois de concluído ele acionar o botão “Ok” e será adicionando a cláusula do contrato na página de contrato.

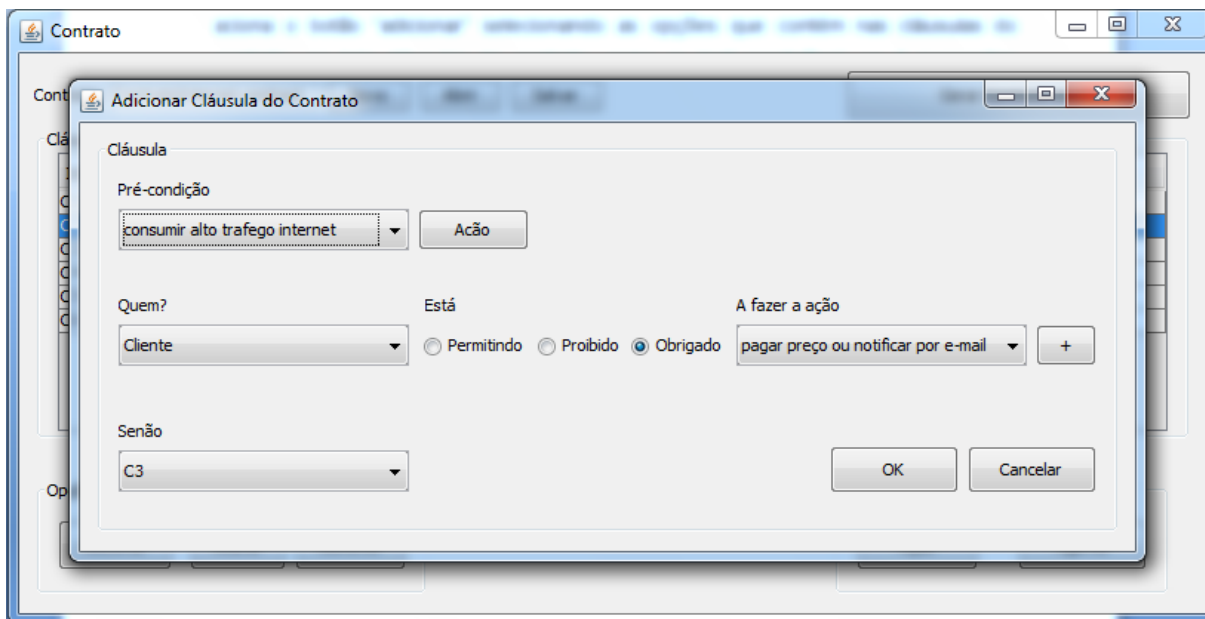


Figura 9 – Tela de Cadastro das Cláusulas do Contrato.

Depois criado o contrato através do sistema proposto, vamos aos 5º Passo: acionar o botão “gerar o contrato em CL” o usuário deverá selecionar e copiar a fórmula que gerar e assim fazer a verificação de conflito na ferramenta CLAN, no qual demonstra um autômato de verificação de conflito. Porém, o processo de gerar a CL se define da seguinte maneira: primeiramente de buscar na cláusula se existe reparação, caso exista, o processa pausa e se chama a cláusula que condiz a reparação assim executando-a e retornando para a cláusula que está em pausa para a sua execução. Após identificado quem está permitido, obrigado e proibido de fazer determinada ação. Assim então, analisa se existe alguma pré-condição para ser executada antes da execução da cláusula, montado assim a fórmula a ser gerada para o contrato em CL.

5.2 MODELAGEM

Modelagem de software é a atividade que consiste em construir modelos que expliquem as características ou o comportamento de um software ou de um sistema de software. Na construção do software os modelos podem ser usados na identificação das características e funcionalidades que o software deverá prover (análise de requisitos), e no planejamento de sua construção. Assim, modelamos os diagramas de casos de uso, diagrama de classes e o diagrama de sequência.

5.2.1 Diagrama de Casos de Uso

Um diagrama de caso de uso ilustra as funcionalidades do sistema descrevendo os cenários aos qual um ator está envolvido. Um ator é um agente que interage com o sistema, podendo ser um ser humano ou outro sistema.

Para esta ferramenta um ator foi identificado como usuário, pode-se cadastrar, alterar, remover ações e agente; poderá cadastrar, alterar e remover as cláusulas, dentre outras especificadas no diagrama caso de uso, além de gerar contrato em CL. Assim pode-se demonstrar na Figura 10.

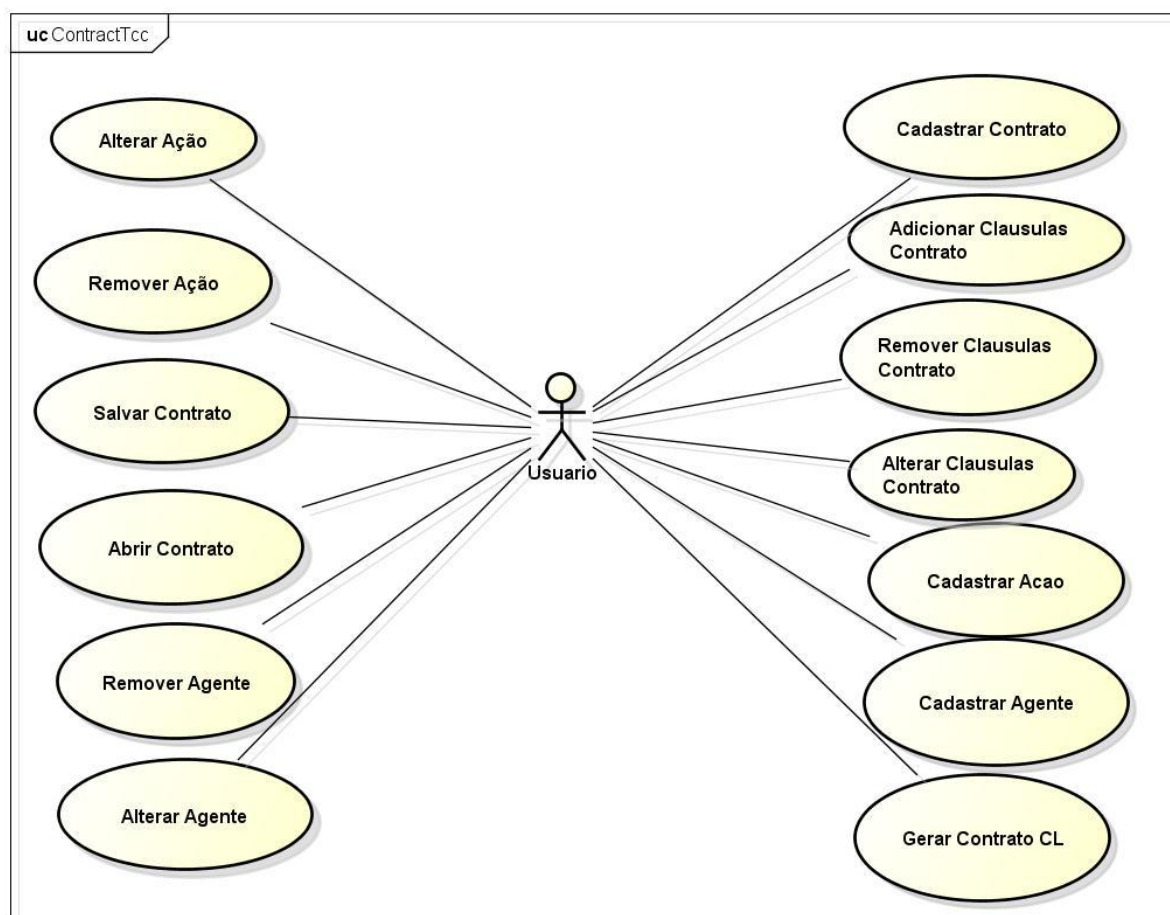


Figura 10 - Diagrama de Caso de Uso

5.2.2 Diagrama de Classes

Diagrama de classe descreve o objeto e as estruturas usadas pelo sistema internamente e a comunicação. Ele descreve as informações sem referência a qualquer implementação específica. Assim pode-se demonstrar na Figura 11.

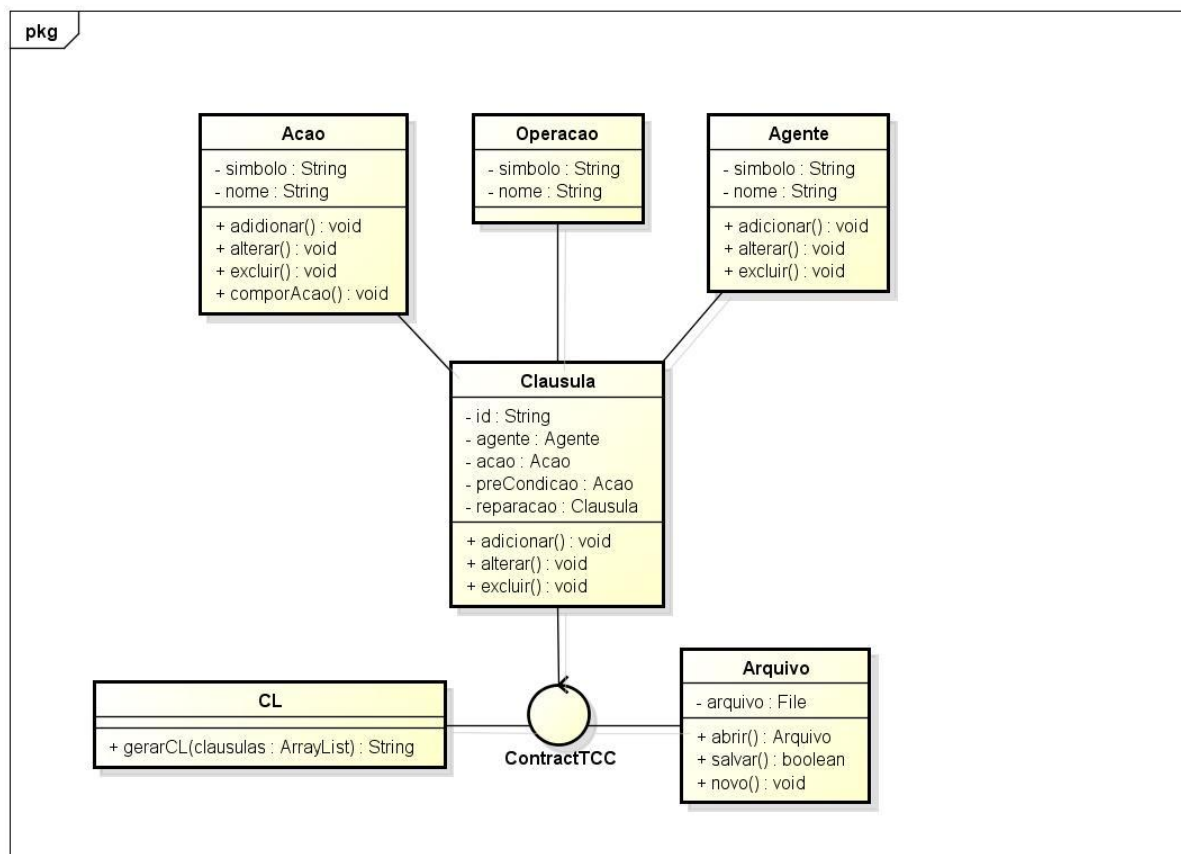


Figura 11 – Diagrama de Classes.

O diagrama possui uma controladora “ContractTCC” que gerencia todo o sistema através das demais classes. A classe “Arquivo” contém métodos que permitem criar um novo contrato, abrir um contrato já existente e salvar um arquivo criado. A classe “Cláusula” é responsável por adicionar, alterar e excluir uma cláusula, além de delegar a classes “Ação” (adicionar, remover, alterar e comporAção), a classe “Agente”(adicionar, remover, alterar) e a classe “Operação”, tendo suas operações já definidas, como permitido, proibido e obrigado. A classes “CL” que possuem o método gerar CL, que processam as cláusulas e convertem em linguagem de contrato CL.

5.2.3 Diagrama de Atividades

O diagrama de atividades demonstra as atividades em fluxo, tendo como sequência as atividades que serão exercidas consecutivamente. A figura 12 demonstra o fluxo dessas atividades.

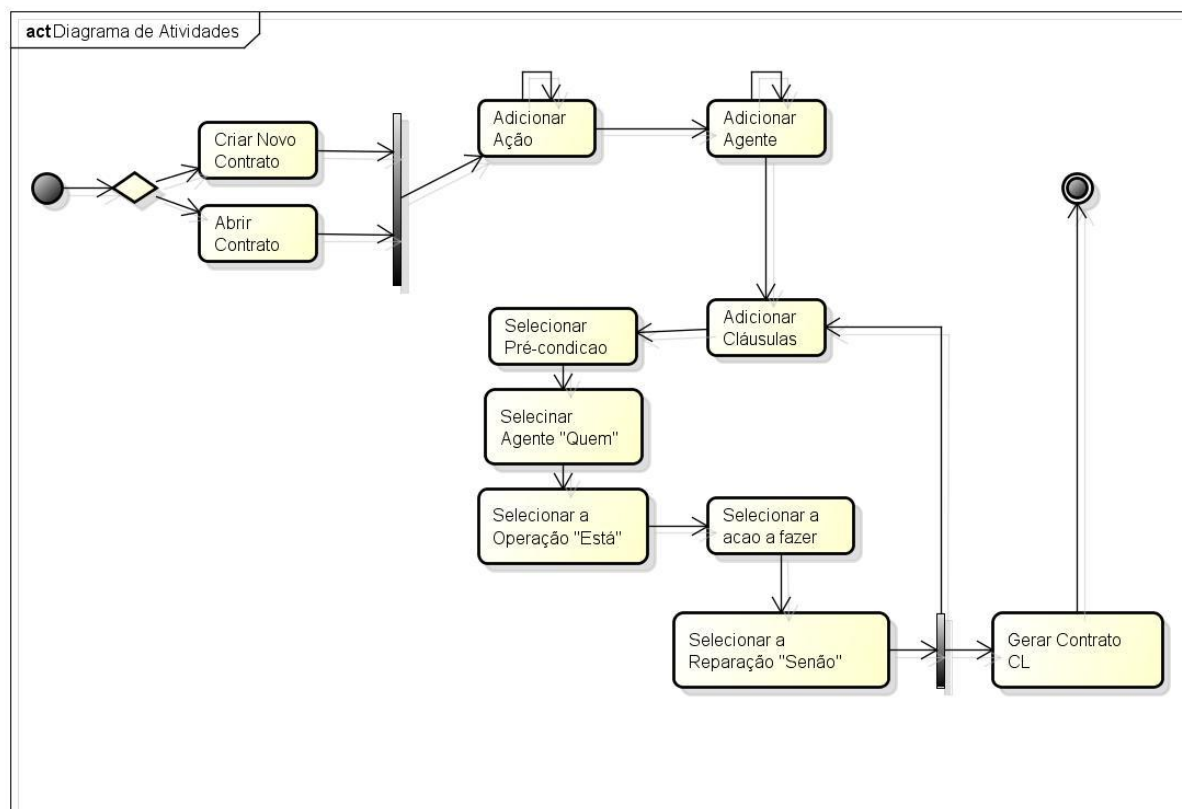


Figura 12 – Diagrama de Atividades

O diagrama de atividades nos diz que ao usuário iniciar o sistema ele deve criar um novo contrato ou abrir um contrato já existente. Em seguida deve cadastrar as ações e agentes, contudo, podem adicionar quantas vezes forem necessárias. Logo após, pode se dar a criação das cláusulas do contrato, através do adicionar cláusulas, que possui selecionar pré-condição, selecionar o agente, selecionar o tipo de operação que está fazendo, a ação a fazer e se for necessário a cláusulas de reparação. Depois de adicionar todas as cláusulas o usuário pode gerar o contrato em CL e em seguida encerrar o sistema.

5.3 IMPLEMENTAÇÃO DO SISTEMA

Esta seção objetiva justificar o estudo desenvolvido durante a fase de projeto, aplicando as metodologias descritas ao longo deste, de forma a deixar claro o processo de desenvolvimento e as funcionalidades da ferramenta proposta, de forma a atingir o objetivo definido para este trabalho.

As ferramentas utilizadas neste trabalho para o desenvolvimento são de caráter livre de forma viabilizar o desenvolvimento de baixo custo.

5.3.1 Interface Gráfica

Para o desenvolvimento desta interface gráfica foi utilizada a ferramenta IDE Netbeans 8.0.1 e a linguagem de programação Java.

Para proporcionar uma melhor utilização da ferramenta, optou-se por fazer a interação desta e o usuário em uma interface gráfica principal, que contenha todos os dados de um contrato em uma única tela, de forma a possibilitar um fácil acesso das funcionalidades do sistema e a fácil interação com o usuário. Conforme a figura 13.

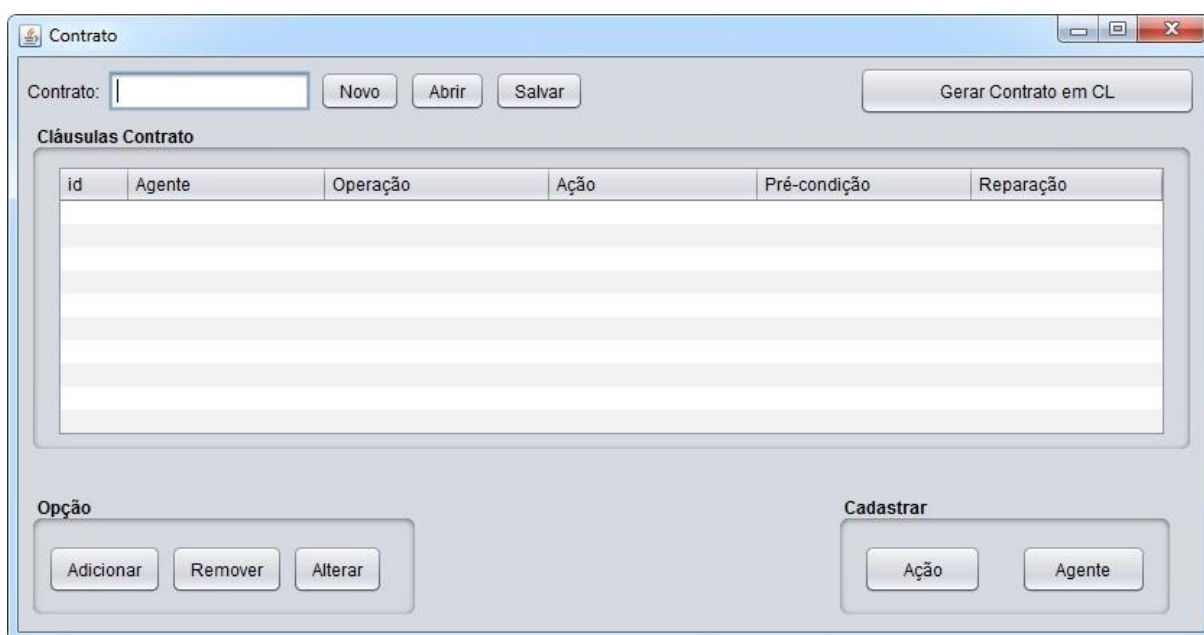


Figura 13 - Interface Gráfica da Tela Principal do Contrato

A figura 14, demonstra a Tela de Cadastro de Ação, que o usuário pode descrever a ação a ser executar e adicionar, criando o Id Ação e a Descrição da

Ação. Logo após, criar o usuário pode remover ou alterar a ação cadastrada. Caso não deseja mais adicionar ações pode se fechar a tela. Dentro dos mesmos conceitos da Tela “Cadastrar Ação”, assim deve acontecer na Tela “Cadastrar Agentes”.

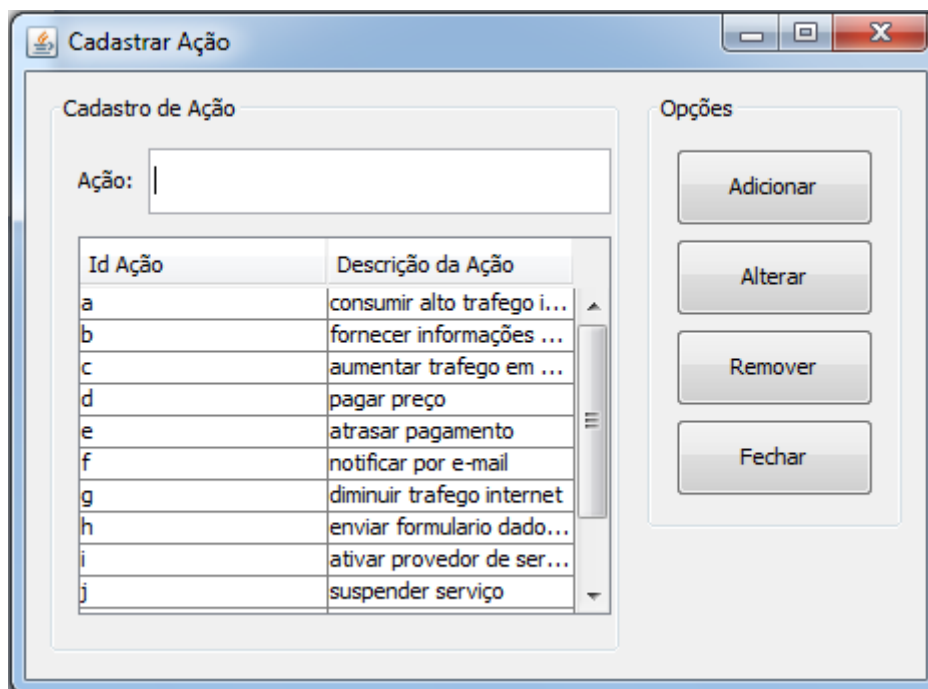


Figura 14 – Interface Gráfica Cadastrar Ação

Encontra-se os operadores dinâmicos, que representam uma pré-condição para a existência de uma cláusula, e os operadores deônticos, que representam as obrigações, permissões e proibições do contrato.

Conforme a sintaxe da linguagem de contrato acima, visualiza-se na figura 15, as lógicas e linguagem sendo aplicadas no sistema nos seguintes casos: (1) lógica dinâmica (refere-se a ação a ser executar para satisfazer outra ação), (2) lógica deôntica (tipo de operação) e (3) linguagem de contrato CL (que condiz a reparação ou compensação da cláusula).

Figura 15 – Tela Adicionar Clausulas demonstrando as lógicas

Sendo assim, é preciso construir uma interface que permita a coleta de informações necessária para a criação destas cláusulas do contrato.

Porém, se uma ação é composta por mais de uma ação ao mesmo tempo, assim foi desenvolvido uma interface que possa criar uma ação composta, conforme a figura 16 abaixo.

Id	Ações
d	pagar preço
e	atrasar pagamento
f	notificar por e-mail
g	diminuir trafego internet
h	enviar formulario dados pessoais
i	ativar provedor de serviços
j	suspender serviço
d+f	pagar preço ou notificar por e-mail
d.d	pagar preço seguido pagar preço
d.d.d	pagar preço seguido pagar preço seguido pagar preço

Figura 16 – Tela de Cadastrar Composição de Ações

Na tela principal é possível adicionar cláusulas do contrato, as ações e os atores, que serão usados nas cláusulas.

A interface gráfica da ferramenta possibilita a execução do contrato durante todo o processo de execução, desde a inserção de dados da cláusula, demonstração das cláusulas até a criação do contrato em si.

A interação entre o usuário e a ferramenta se inicia com o nome do Contrato Novo e clicando no botão novo. Conforme na figura 6. O mesmo possui a opção Abrir contrato existente e Salvar o contrato aberto.

Ao clicar no botão Gerar Contrato em CL na Tela do Contrato, será direcionada a outra interface que assim apresentada na figura 17.

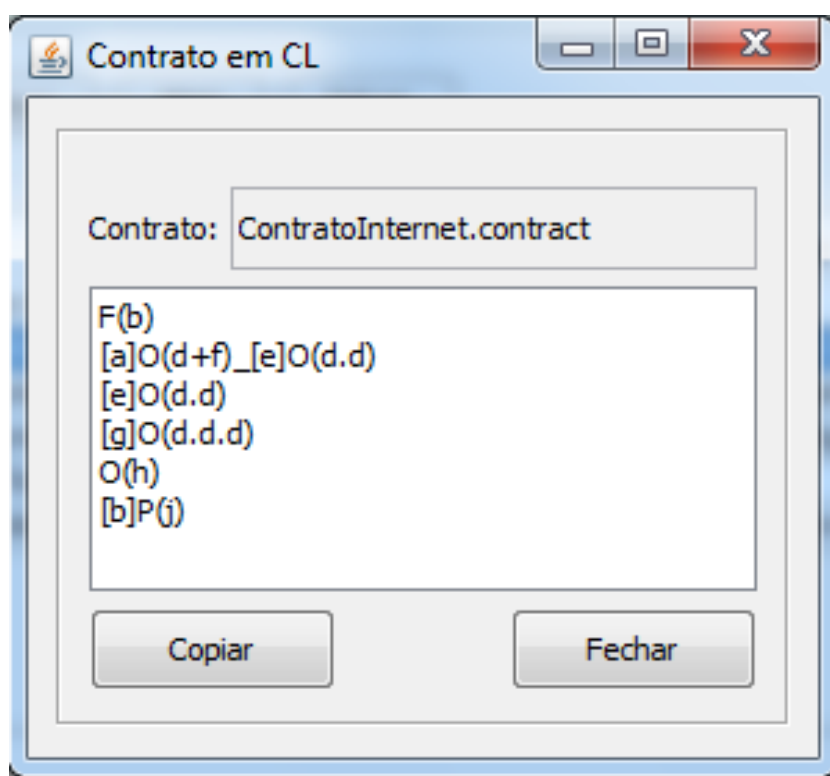


Figura 17 - Interface Grafica de Contrato em CL

A tela do contrato em CL demonstra o nome do contrato que foi gerado e as suas fórmulas, que o usuário deve selecionar e copiar para que possa ser utilizado na ferramenta CLAN para fazer a verificação de conflito na cláusula do contrato. Para encerrar clicar em salvar no qual irá armazenar o contrato em arquivo com extensão “.contract”, podendo ser aberto somente na ferramenta desenvolvida.

6. RESULTADOS E DISCUSSÃO

Neste capítulo se aplica o contrato no sistema desenvolvido, gerando a fórmulas em CL necessárias para aplicar a ferramenta CLAN, assim verificando a existência de conflito no contrato. Diante disso, demonstrando do resultado utilizando a ferramenta e a ferramenta CLAN.

Depois de representado o Contrato Serviço de Internet inicia-se a utilização e aplicação do contrato no sistema criado, criando um novo Contrato, conforme a figura 18.

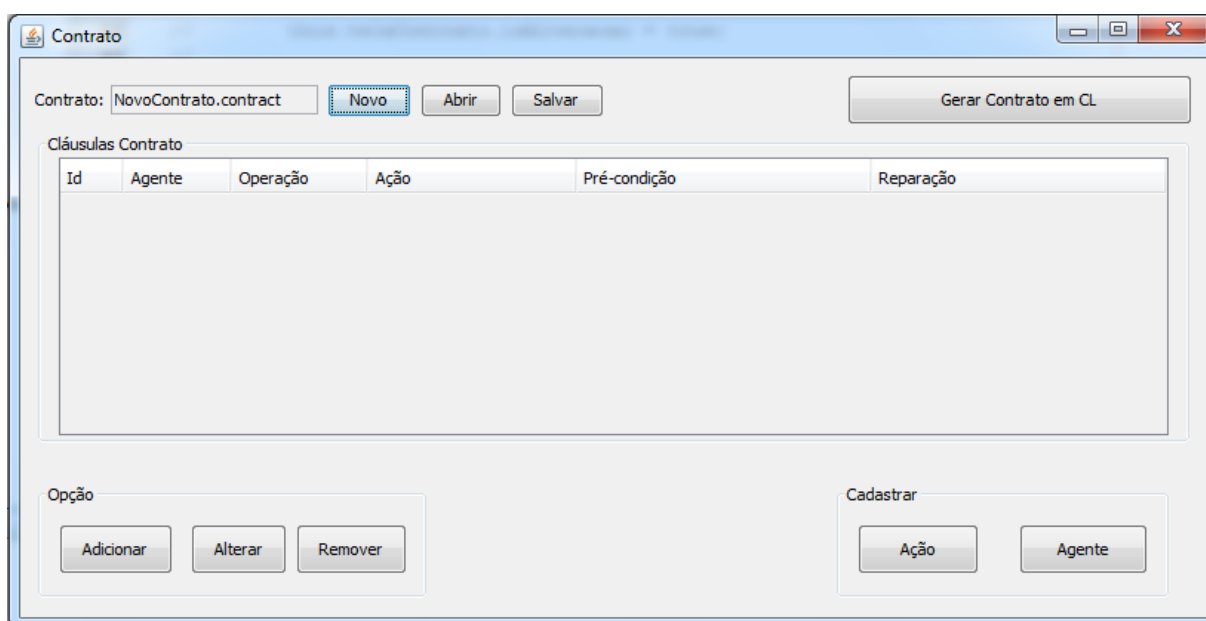


Figura 18 – Criando um Novo Contrato.

Após criar um novo contrato vamos adicionar as ações que serão executadas pelos agentes. Conforme as figura 19.

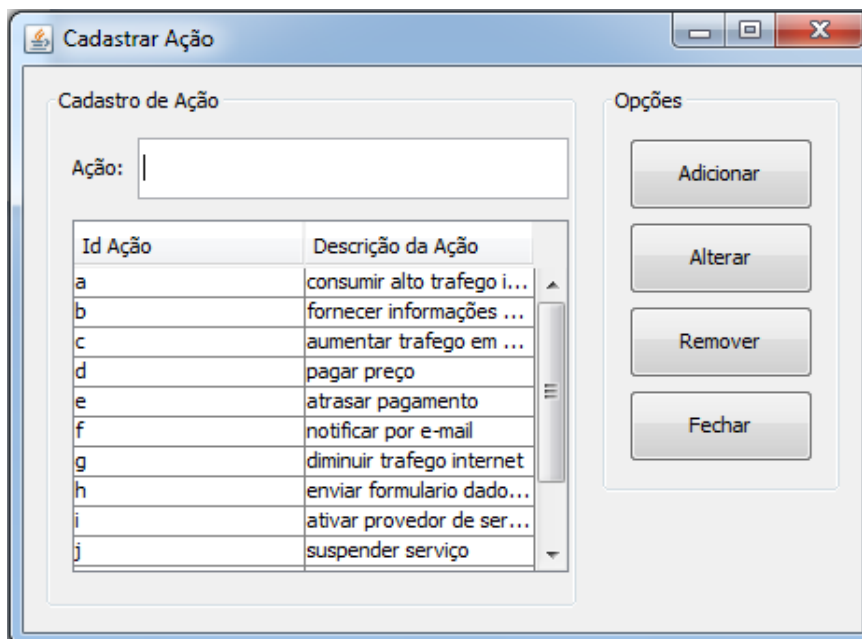


Figura 19 – Adicionando Ações

Todas as ações foram cadastradas pelo autor, por exemplo, as seguintes descrições: consumir tráfego Internet, fornecer informações falsas, pagar preço, atrasar pagamento, ativar servidor de serviço, dentre outras.

Da mesma forma foi feita com a tela de cadastrado de agente, conforme a figura 20.

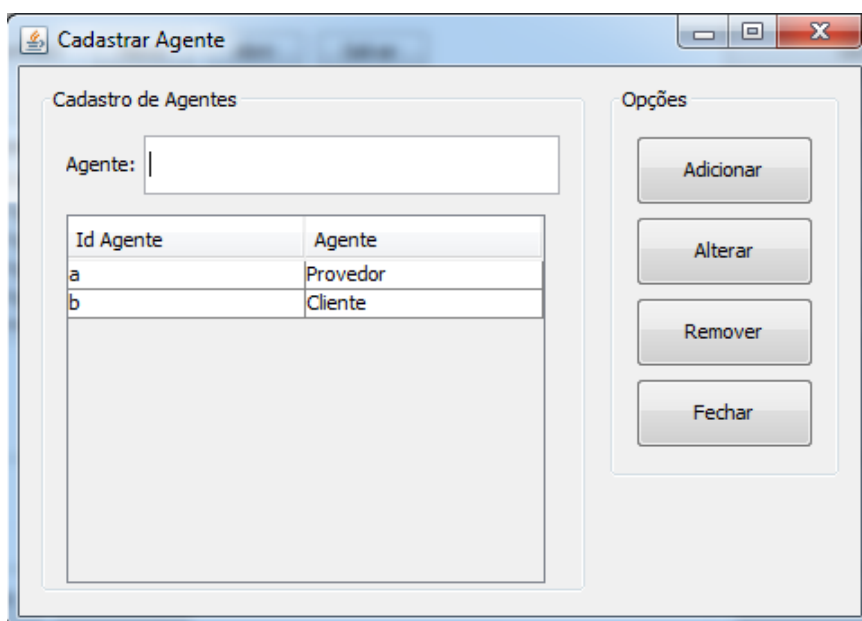


Figura 20 – Adicionando Agentes

As ações foram cadastradas pelo autor, tendo com os agentes envolvidos no contrato o Provedor e o Cliente.

Depois de Cadastrados os agentes e ações passaram a adição de cláusulas no contrato. O processo de adicionar consiste em clicar no botão de opção Adicionar na tela principal do contrato, após a pressionar o botão será encaminhado para a tela de Adicionar cláusula do contrato. O preenchimento se dá através da seleção do campo Pré-Condição, Quem, Está, A Fazer a Ação, Senão, respectivamente, conforme a figura 21 abaixo.

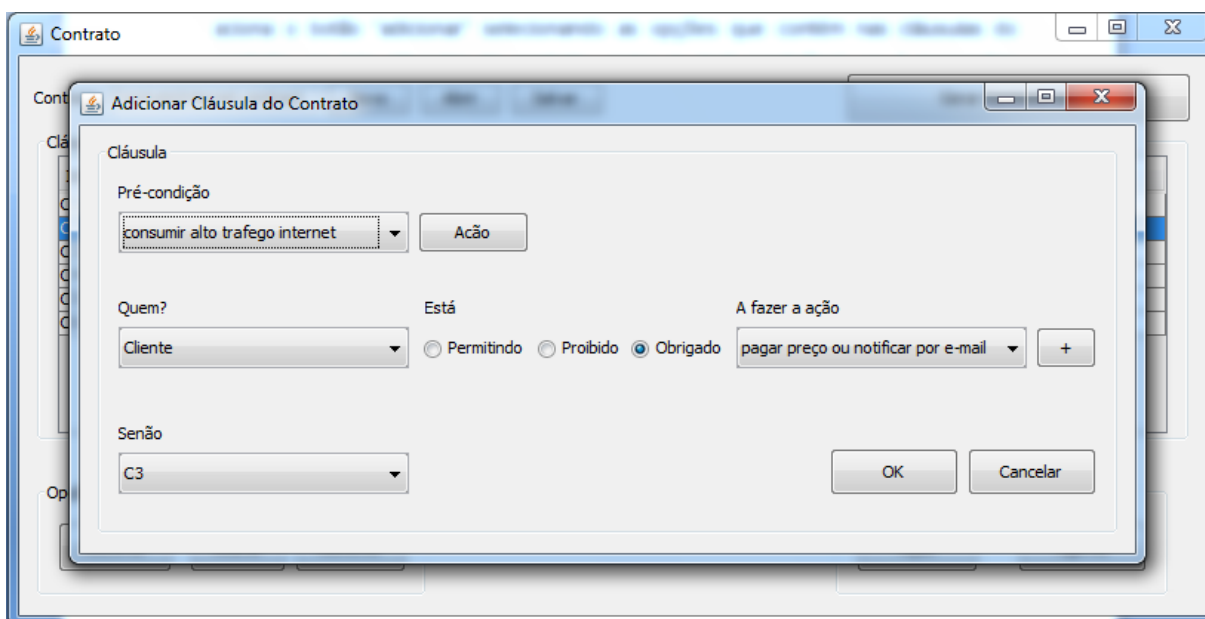


Figura 21 - Tela de Adicionar Cláusula no contrato.

Depois de todos os campos selecionados da tela Adicionar cláusulas do contrato, clicar em Ok para se adicionar a cláusula do contrato na tela principal e Cancelar para o processo de adição.

Após o processo de adicionar as cláusulas do contrato e assim sucessivamente o quanto de cláusulas necessárias na sua inserção, após as inserções podemos ver o contrato assim descrito na figura 22.

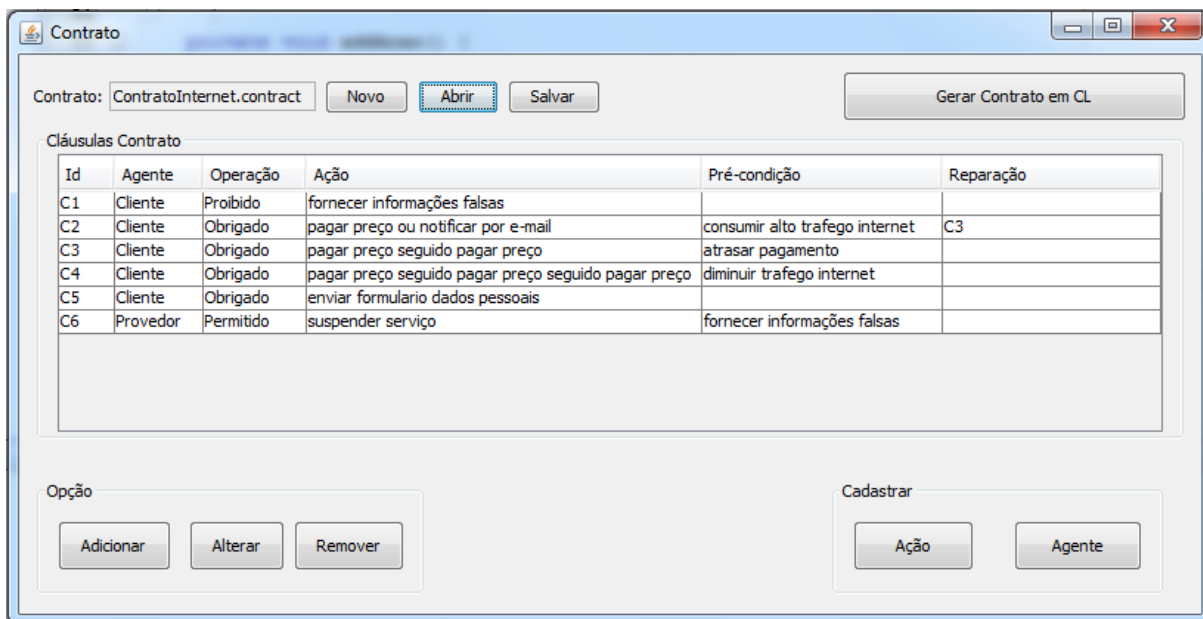


Figura 22 – Tela Contrato com o contrato descrito.

Após a inserção de todas as cláusulas do contrato, conforme a figura acima, agora se aplica o processo de criação das fórmulas do contrato no qual será copiado e executado na ferramenta para verificação de conflitos assim denominada CLAN. A tela, conforme a figura 23 demonstra as fórmulas que foram geradas em CL.

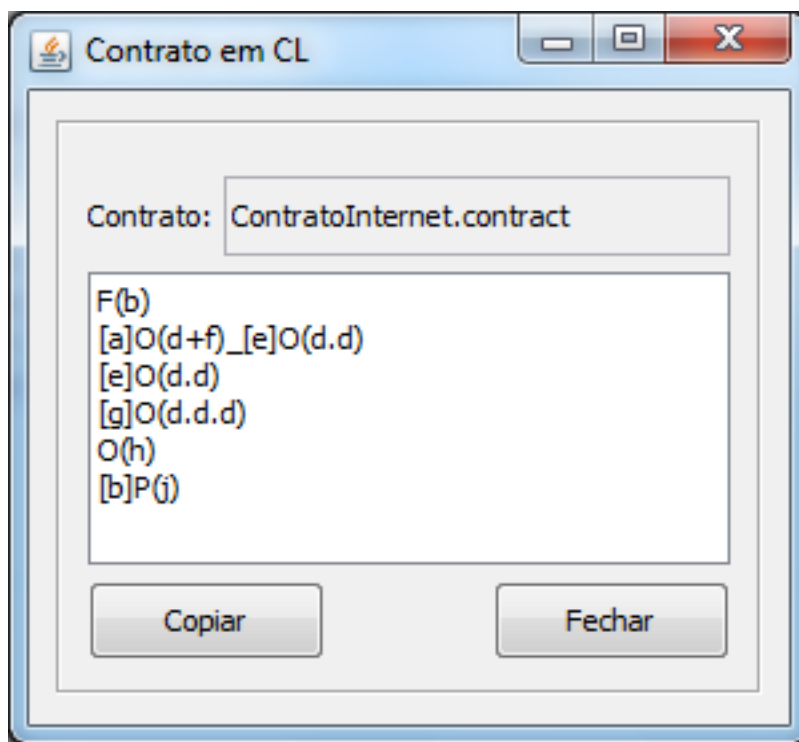


Figura 23 – Tela Gerar Contrato Serviço Internet.

Após as fórmulas terem sido geradas pela ferramenta desenvolvida, assim vamos aplicar as fórmulas na ferramenta CLAN e assim demonstrar os resultados obtidos com a análise da ferramenta. A figura 24 demonstra a tela da CLAN com o contrato Serviço Internet.

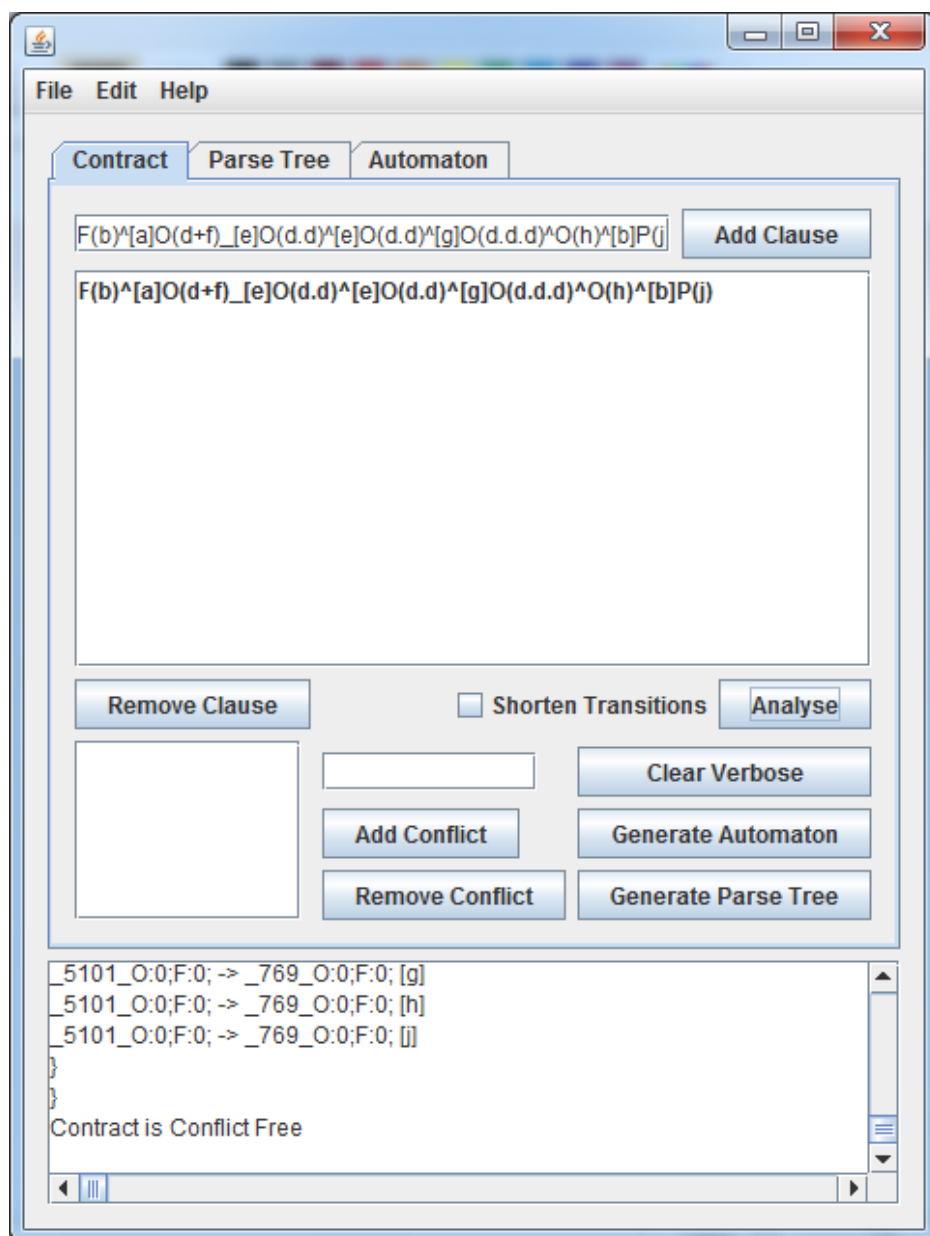


Figura 24 – Ferramenta CLAN aplicando Contrato Serviço Internet

Ao verificar o contrato de serviço de Internet podemos assim criar o autômato que corresponde ao contrato. Podemos ver na figura 25 o autômato.

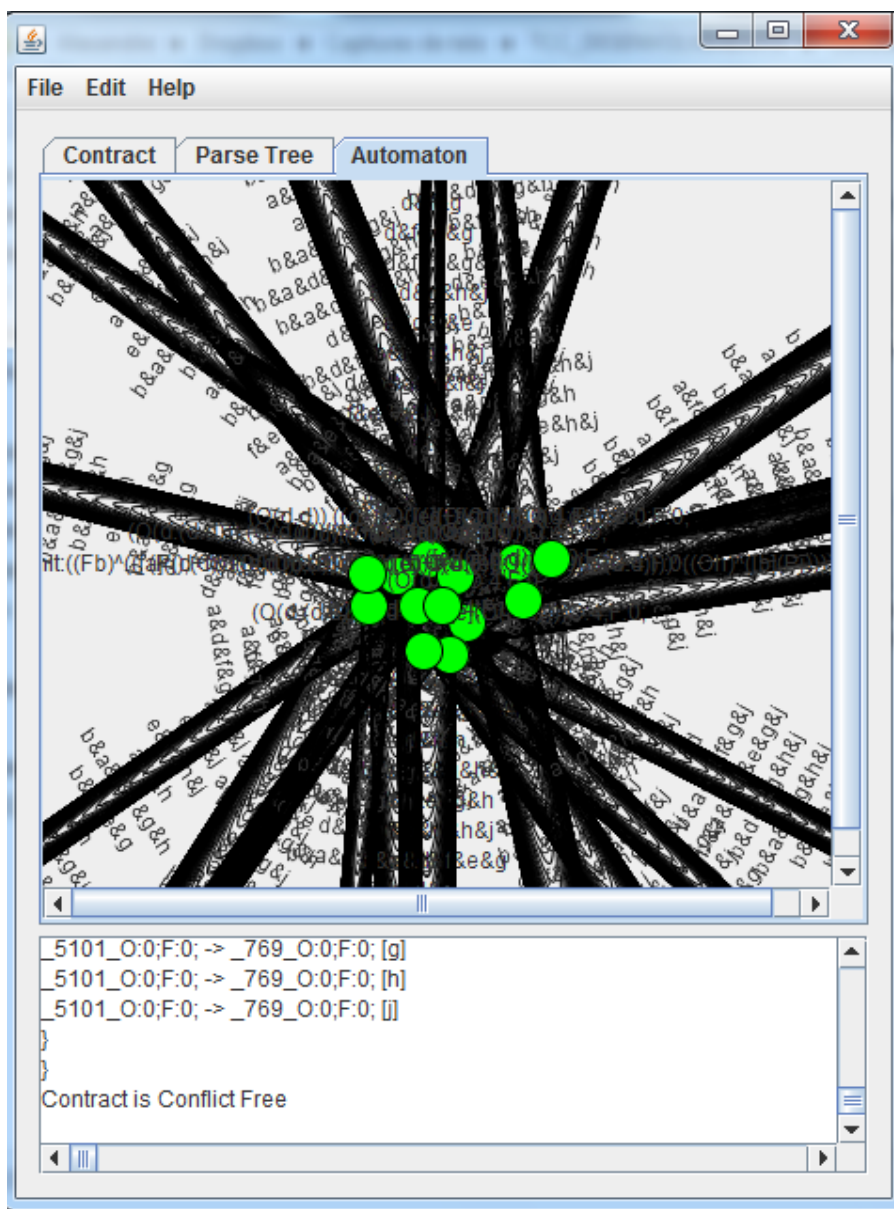


Figura 25 – Autômato gerado contrato serviço Internet.

Conclui-se que após todo o processo de criação do contrato, aplicando os conceitos de formalismo, linguagem de contrato CL, principalmente ter como aliadas aos conceitos a esse processo as lógicas modais, as lógicas deônticas, se tem um parecer positivo em relação ao contrato. A análise do contrato Serviço de Internet aplicado à ferramenta CLAN, após verificar nas cláusulas contidas no contrato por meio do formalismo, nos traz o seguinte resultado, o contrato está livre de conflitos, no qual se pode ser feito sem problemas e assim os seus envolvidos sejam satisfeitos.

Diante do problema que possuía como fator de principal de poder criar contratos eletrônicos utilizando formalismos para a sua representação e assim para que se pudesse verificar se os contratos possuíam cláusulas contraditórias ou se o mesmo estava livre de conflito; pois sistema que assim foi desenvolvido trouxe a confirmação que é possível verificar se existem os conflitos contidos no contrato.

As restrições que possuem no sistema são: o sistema é apenas para um autor, pois não possui restrições de acesso; possui extensão “.contract” que apenas pode ser aberto pela ferramenta e obrigatório ter ações cadastradas.

As dificuldades quanto ao desenvolvimento, se deu através da implementação de cláusulas compostas, que consistem e ter mas de uma ação ao mesmo tempo.

7. CONSIDERAÇÕES FINAIS

Em meio a uma sociedade cada vez mais preocupada com suas obrigações e seus deveres encontramos a necessidade de firmar por escrito as combinações formuladas para as relações comerciais, surgindo então à figura do contrato.

A finalidade de um contrato é garantir que as condições acordadas entre as partes sejam cumpridas à risca oferecendo segurança aos contratantes, de maneira especial em casos de descumprimentos que gerem uma rescisão. No momento em que uma das partes fica insatisfeito o que influencia aos desacordos e levando a quebras contratuais.

A tarefa de se evitar conflitos é muito importante e necessária na área de contratação eletrônica, diante disto o presente trabalho teve como objetivo o desenvolvimento de uma ferramenta que auxilie na verificação formal de contratos, no qual possa raciocinar sobre contrato convencional e assim transformando em contratos eletrônicos.

Esta ferramenta beneficia o autor do contrato na verificação de suas cláusulas de forma dinâmica, fácil e bem interativa. Dessa forma, há um grande ganho de tempo de análise e melhor raciocínio nos contratos desenvolvidos, além de ser uma ferramenta de baixo custo.

Como trabalhos futuros, é sugerido que seja integrada a ferramenta CLAN a ferramenta desenvolvida, visando que o autor ao criar um contrato possa automaticamente estar verificando se existem conflitos no contrato, o que ele terá ainda, mais ganho de tempo na elaboração dos contratos.

REFERÊNCIAS

AALST, W. M. P. van der et al. *Multiparty contracts: Agreeing and implementing interorganizational processes*. Comput. J., v. 53, n. 1, p. 90-106, 2010.

ALMEIDA, M. B. *Uma introdução ao XML, sua utilização na Internet e alguns conceitos complementares*. Universidade Federal de Minas Gerais. Minas Gerais, 2002. Disponível em <<http://www.scielo.br/pdf/ci/v31n2/12903.pdf>>, Acesso em 18 ago. 2013.

ANGELOV, S.; GREFEN, P. *B2B eContract Handling - A Survey of Projects, Papers and Standards*. [S.l.], 2001.

ANGELOV, S.; GREFEN, P. *Requirements on a b2b e-contract language*. In: *BETA publicatie*. [S.l.]: Eindhoven: Technische Universiteit Eindhoven, 2005, (BETA publicatie: working papers, 140). p. 46.

AUSÍN, T. *Entre la Lógica y el Derecho - Paradojas y conflictos normativos*. Barcelona, 2005. Disponível em: <<http://books.google.com.br/books?id=5WD2u1u9qjUC>>. Acesso em 22 ago. 2013.

BACCIOTTI, R. C. D. *CONTRATOS – Conceito e Espécies*. Disponível em: <<http://www.advogado.adv.br/artigos/2000/ruibacciotti/contratos1.htm>>. Acesso em: 08 abr. 2013.

BEDREGA, B. R. C.; ACÍOLY, B. M. *Introdução à Lógica Clássica para a Ciência da Computação*. Natal, 2007. 277 p. Disponível em: <http://www.dimap.ufrn.br/~jmarcos/books/BA_Jul07.pdf>. Acesso em: 01 set. 2014.

BLACKBURN, P.; BENTHEM, J. van; WOLTER, F. *Handbook of Modal Logic*. Elsevier Science, 2006. (Studies in Logic and Practical Reasoning). ISBN 9780080466668. Disponível em: <<http://books.google.com.br/books?id=urINMvvsT5MC>>. Acesso em: 02 fev. 2014.

BOBBIO, N. *Teoria do Ordenamento Jurídico*. 1ª edição. ed. Brasília: Editora UnB, 2006.

BOSSE, T. et al. Automated formal analysis of human multi-issue negotiation processes. *Multiagent Grid Syst.*, IOS Press, Amsterdam, The Netherlands, The Netherlands, v. 4, n. 2, p. 213-233, abr. 2008. ISSN 1574-1702. Disponível em: <<http://dl.acm.org/citation.cfm?id=1402618.1402622>>. Acesso em: 18 mar. 2013.

CHADHA, R.; KRAMER, S.; SCEDROV, A. *Formal analysis of multi-party contract signing*. In: Computer Security Foundations Workshop, 2004. Proceedings. 17th IEEE. [S.l.: s.n.], 2004. p. 266-279. ISSN 1063-6900.

CHELLAS, B. F. *Modal Logic: an introduction*. Cambridge University Press, 1980.

DINIZ, M. H. *Curso de Direito Civil Brasileiro: Teoria das obrigações contratuais e*

extracontratuais. 21 ed. rev. e atual. São Paulo: Saraiva, 2005. 867 p. 3 v.

FENECH, S. Mestrado, *Conflict Analysis of Deontic Contracts*. Department of Computer Science and Artificial Intelligence: [s.n.], setembro 2008.

FENECH, S.; PACE, G. J.; SCHNEIDER, G. *Clan: A tool for contract analysis and conflict discovery*. In: LIU, Z.; RAVN, A. P. (Ed.). 7th International Symposium on Automated Technology for Verification and Analysis (ATVA'09). Macao, China: Springer, 2009. (Lecture Notes in Computer Science, v. 5799), p. 90-96. ISBN 978-3-642-04760-2.

FENECH, S.; PACE, G.; SCHNEIDER, G. *Automatic conflict detection on contracts*. In: LEUCKER, M.; MORGAN, C. (Ed.). Theoretical Aspects of Computing - ICTAC 2009. Springer Berlin Heidelberg, 2009, (Lecture Notes in Computer Science, v. 5684). p. 200-214. ISBN 978-3-642-03465-7. Disponível em: <http://dx.doi.org/10.1007/978-3-642-03466-4_13>. Acesso em: 08 mar. 2013.

FISCHER, M. J.; LADNER, R. E. *Propositional dynamic logic of regular programs*. Journal of Computer and System Sciences, v.18, n.2, p.194 – 211, 1979. INSS 0022-0000. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0022000079900461>>. Acesso em: 08 mar. 2013.

GAGLIANO, P. S. PAMPLONA, R. F. *Novo curso de direito civil: contratos*, tomo 1: teoria geral, 2008, p. 11 (v. IV).

GENESERETH, M. R.; NILSON, N. J. *Logic Foundations of Artificial Intelligence*, Morgan Kaufmann Publishers, 1988.

GOMES, O. *Contratos*. 17 ed. Atual, Rio de Janeiro: Forense, 1997. 1997, p.4.

HAREL, D.; KOZEN, D.; TIURYN, J. *Dynamic logic*. In: Handbook of Philosophical Logic. [S.I.]: MIT Press, 1984. p. 497 - 604.

HERRESTAD, H.; KROGH, C. *Deontic logic relativised to bearers and counterparties*. p. 453-522, 1995.

HILPINEN, R. *Deontic logic*. Blackwell, 2001. cap. 8.

KOETSIER, M.; GREFEN, P.; VONK, J. *Contracts for cross-organizational workflow management*. In: BAUKNECHT, K.; MADRIA, S.; PERNUL, G. (Ed.). Electronic Commerce and Web Technologies. Springer Berlin Heidelberg, 2000, (Lecture Notes in Computer Science, v. 1875). p. 110-121. ISBN 978-3-540-67981-3. Disponível em: <http://dx.doi.org/10.1007/3-540-44463-7_10>. Acesso em: 08 mar. 2013.

LAUDON, K. C.; LAUDON, J. P. *Sistemas de informação*. Rio de Janeiro: LTC, 1999.

LEE, R. M. *A logic model for electronic contracting*. Decision Support Systems, n. 4,

p. 27-44, March 1988.

MARQUES, C. L. *Contratos no Código de Defesa do Consumidor: o novo regime das relações contratuais*. 4. ed. rev., São Paulo: Editora Revista dos Tribunais, 2002.

MEYER, J.-J. C. *Epistemic logic*. In: GOBLE, L. (Ed.). *The blackwell guide to philosophical logic*. [S.l.]: Blackwell, 2001. cap. 9.

MIRANDA, M. B. *Teoria geral dos contratos*. In: *Revista Virtual Direito Brasil*. 2. ed., 2008. volume 2, n. 2, p. 15.

OCHOA, C. H. *Conflictos normativos*. México: Universidad Nacional Autónoma de México, 2003.

PEREIRA, C. M. S. *Instituições de direito civil: fontes das obrigações*. 2001. p.2.

PNUELI, A. *The temporal logic of programs*. In: FOCS. [S.l.]: IEEE Computer Society, 1977. p. 46-57.

PRAKKEN, H.; SERGOT, M. *Contrary-to-duty obligations*. *Studia Logica*, v. 57, p. 91-115, 1996.

PRISACARIU, C.; SCHNEIDER, G. *A dynamic deontic logic for complex contracts*. *The Journal of Logic and Algebraic Programming*, v. 81, n. 4, p. 458-490, 2012. ISSN 1567-8326. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1567832612000197>>. Acesso em: 08 mar. 2013.

PRISACARIU, C.; SCHNEIDER, G. *A formal language for electronic contracts*. In: *In FMOODS'07*, volume 4468 of LNCS. [S.l.]: Springer, 2007. p. 174-189.

RAMOS, D. *Quine: quantificação, existência e identidade*. 2003. Disponível em: <<http://www.cfh.ufsc.br/~dkrause/LogicaIII/DiogoRamos.doc>>. Acesso em: 02 set. 2014.

SALOMO, J. *Contratos de prestação de serviços: manual teórico e pratico*. Juarez de Oliveira, 2005. ISBN 9788574535302. Disponível em: <<http://books.google.com.br/books?id=hvGdc5PXIUyC>>. Acesso em : 24 jul. 2014.

SOUZA, M. C. J. d. *Contratos Eletrônicos*. MAUAD, 1997. (Série Jurídica, 2). Disponível em: <<http://books.google.com.br/books?id=r7ptQFFK4V4C&printsec=frontcover&hl=pt-BR>>. Acesso em: 20 fev. 2013.

VON WRIGHT, G.H., *Deontic logic*, *Mind* 60 (1951) 1–15.

WIELEWICK, L. *Contratos e Internet - Contornos de uma breve análise*. In: *Comércio Eletrônico*. Organizadores: Ronaldo L. da Silva Jr., Ivo Waisberg. São Paulo: Revista dos Tribunais, 2001.

XU, L. *A multi-party contract model*. *ACM SIGecom Exchanges*, v. 5, 2004a.

XU, L. *Monitoring Multi-party contracts for e-business*. Tese (Tese) - Faculty of Economics and Business Administration of Tilburg University, 2004b.

XU, L.; JEUSFELD, M. A. *A concept for monitoring of electronic contracts*. In: In The 2003 IEEE Conference on E-Commerce (CEC'03). IEEE Computer. [S.l.]: Society Press, 2003. p. 584-600.

ANEXO 1 – Parte do contrato entre provedor de Internet e um cliente (PACE; PRISACARIU, SCHNEIDER, 2007).

Este documento de acordo é feito entre:

1. [nome], a partir de agora referido como Provedor e.
2. [nome], a partir de agora referido como o Cliente.

INTRODUÇÃO

3. O Provedor é obrigado a fornecer os serviços de Internet, conforme estipulado no presente Acordo.

5. DEFINIÇÕES

5.1. a) o tráfego de Internet pode ser medido tanto pelo Cliente e Fornecedor por meio de equipamento e podem levar os dois valores de alta e normal.

PARTE DECISÓRIA

7. RESPONSABILIDADES E DEVERES DO CLIENTE

7.1. O cliente não deve:

a) fornecer informações falsas ao Departamento de Relações com o Cliente do Provedor.

7.2. Sempre que o tráfego Internet é alto, em seguida, o cliente deve pagar [preço] imediatamente, ou o Cliente deverá notificar o provedor, enviando um e-mail especificando que ele vai pagar mais tarde.

7.3. Se o cliente atrasa o pagamento, conforme estipulado no item 7.2, após a notificação, ele deve reduzir imediatamente o tráfego da Internet para o nível normal, e pagar depois duas vezes (2 * [preço]).

7.4. Se o cliente não diminuir o tráfego da Internet imediatamente, em seguida, o cliente terá de pagar 3 * [preço].

7.5. O Cliente deverá, assim que o serviço de Internet torna-se operacional, apresentar, no prazo de 7 (sete) dias, o formulário de dados pessoais de sua conta na página web do Provedor para o Departamento de Relações com o Cliente do Provedor.

8. DIREITOS DO CLIENTE

8.1. O cliente pode optar por pagar ou:

a) a cada mês; b) a cada 3 (três) meses; c) a cada 6 (seis) meses;

9. SERVIÇO DA OPERADORA

9.1. Como parte do serviço oferecido pelo Provedor do Cliente tem o direito a um e-mail e uma conta de usuário.

9.2. Provedor é obrigado a oferecer, sem qualquer limitação e dentro de um período de sete (7) dias a senha e qualquer outro equipamento específico para cliente, necessário para o uso correto da conta de usuário, ao receber de todos os dados necessários sobre o cliente do Departamento de Relações com cliente do Provedor.

9.3. Cada mês, o cliente paga a conta do provedor é obrigado a enviar um Relatório de Uso da Internet para o cliente.

10. OBRIGAÇÕES DO PRESTADOR

10.1. O provedor tem a obrigação de devolver os dados pessoais do cliente ao estado original após a rescisão do presente acordo e, posteriormente, para apagar e não usar para qualquer finalidade na totalidade ou em parte dela.

10.2. As garantias do provedor que o Departamento de Relações com o Cliente, como parte de sua organização administrativa, será sensível aos pedidos do cliente ou qualquer outro departamento do Provedor, ou o próprio Provedor de dentro de um período de menos de duas (2) horas durante o horário de trabalho ou no dia seguinte.

11. DIREITOS DO PROVEDOR

11.1. O provedor tem o direito de alterar, apagar ou utilizar os dados pessoais do cliente apenas para as estatísticas, monitoramento e uso interno na confiança do Provedor.

11.2. Provedor poderá, a seu exclusivo critério, sem aviso prévio ou indicar qualquer motivo ou incorrer em qualquer responsabilidade para fazê-lo:

b) suspender os serviços de Internet imediatamente se o cliente está em descumprimento da Cláusula 7.1;

13. RESCISÃO

13.1. Sem limitar a generalidade de qualquer outra cláusula no presente acordo o cliente pode rescindir este Contrato imediatamente, sem qualquer aviso prévio e sendo vindicada de qualquer cláusula do presente Acordo, se:

a) o provedor não oferece o serviço de Internet por sete (7) dias consecutivos.

13.2. O Provedor é proibido de denunciar o presente Acordo, sem notificação prévia, por escrito, por correio normal e por e-mail.

13.3. O Provedor poderá rescindir o presente Contrato se:

a) qualquer pagamento devido do cliente ao fornecedor de conformidade com este Acordo não foi pago por um período de 14 (quatorze) dias;

16. LEI APLICÁVEL

16.1. O Provedor e do presente Acordo são regidos e interpretados de acordo com a lei que regulamenta os Serviços de Internet e para o Direito do Estado.

a) A lei do Estado estabelece que qualquer provedor ISP é obrigada, mediante pedido para aproveitar qualquer atividade até novo aviso dos representantes do Estado.