

Rastreamento e Gerenciamento de Requisitos em Busca da Conformidade Legal

Ilueny Santos^{1,2}, Erica Miranda¹ e Márcia Lucena¹

¹ Universidade Federal do Rio Grande do Norte, Natal, Brasil

² Tribunal de Contas do Estado do Rio Grande do Norte, Natal, Brasil
{lueny, eestcunha}@gmail.com, marciaj@dimap.ufrn.br

Resumo A crescente complexidade dos sistemas de software e a exigência de conformidade com a legislação vigente são condições *sine qua non* à era digital. Atender estas condições e gerenciar requisitos legais tornam-se atividades custosas, quando feitas de forma não planejada. Requisitos legais referem-se ao conjunto de leis e regulamentos aplicáveis ao domínio do negócio do *software*, que se pretende desenvolver e por isso são altamente sensíveis às alterações que ocorrem na legislação. Neste trabalho, é apresentada uma proposta de gestão baseada na rastreabilidade de requisitos. Com isto, é possível realizar o acompanhamento e a descrição da vida de um requisito, visando o processo de manutenção e verificação da conformidade legal dos sistemas. Para melhor compreensão do problema foram realizadas pesquisas bibliográficas e entrevistas com especialistas em requisitos. Uma aplicação da abordagem no contexto de comércio eletrônico é apresentada.

Palavras-chave: Requisito legal; Rastreabilidade de requisitos; Conformidade legal.

1 Introdução

Ao longo das últimas décadas, temos experimentado um crescimento em infraestrutura, produtos e serviços que utilizam sistemas de informação (SI) para atingir os objetivos das partes interessadas. Leis e regulamentação de governos impõem desafios de conformidade para os engenheiros de requisitos. Cada vez mais instituições são obrigadas a desenvolver SI que estejam em conformidade com a legislação vigente. Esta conformidade legal (CL) deve ser mantida e monitorada durante o ciclo de vida dos sistemas. Os prejuízos de uma instituição, que não se preocupa com a CL dos requisitos de seus *softwares*, podem ir de perdas financeiras, sociais ou até mesmo prejuízos incalculáveis à sua imagem.

Neste trabalho, o foco está nos requisitos legais (RL). Requisitos que são de alta volatilidade e criticidade independente do contexto ao qual o SI está inserido. O dinamismo das leis, por exemplo as leis brasileiras em seus diferentes níveis, exige uma adaptação contínua dos RL modelados. Os custos e a complexidade para manutenção da conformidade do SI com essas leis - particularmente em contextos multijurisdicional - são notórios e podem ter efeitos nas estratégias

de comercialização das empresas [1,2,3], na segurança dos dados, na privacidade dos usuários, e nos aspectos de, por exemplo, legalidade, isonomia, economia, universalidade dos serviços do governos à sociedade.

RL, geralmente, são incompletos, desatualizados, ambíguos ou contraditórios e contribuem para um SI com baixa testabilidade. Desta forma, identificar a CL do SI tem sido um grande desafio interdisciplinar. A relação entre alguns tipos de *softwares* e as leis relacionadas ao domínio do SI despertou o interesse em responder às questões: i) Como evidenciar a conformidade do sistema com a legislação vigente? ii) Existem artefatos, nas metodologias de desenvolvimento atualmente adotadas nos projetos de *software*, que poderiam auxiliar a verificação da conformidade legal? iii) A sistematização da utilização de artefatos, possivelmente existentes, poderia melhorar ou facilitar o cumprimento da legislação vigente?

Neste contexto, a proposta desse trabalho foi, a partir de uma pesquisa bibliográfica e de entrevistas com especialistas no domínio, definir uma abordagem que ampliasse a eficiência através de um processo de desenvolvimento e manutenção centrado no gerenciamento e rastreabilidade dos RL. Buscou-se favorecer a verificação da conformidade dos *softwares* com a legislação vigente durante o ciclo de vida dos SI. Para isso, na Seção 2 deste trabalho, é apresentada a metodologia da pesquisa, os principais trabalhos da literatura que fundamentaram a abordagem proposta, e as entrevistas. As etapas, atividades, artefatos e aplicação da abordagem são apresentados na Seção 3, para que, na Seção 4, sejam feitas as discussões e apresentação dos trabalhos relacionados com cada uma das etapas da abordagem. Na Seção 5, constam as considerações finais.

2 Metodologia

A metodologia definida e aplicada foi dividida em quatro etapas: i) fundamentação a partir de pesquisa bibliográfica para conhecer o estado da arte relacionado com o tema; ii) desenvolvimento de uma abordagem preliminar conceitual e abstrata seguindo os trabalhos estudados; iii) entrevistas com especialistas no domínio para um melhor entendimento do problema e aprimoramento da abordagem; iv) nova versão da abordagem a partir dos resultados das entrevistas.

A fundamentação foi construída a partir de pesquisas bibliográficas nos principais repositórios de dados da área de computação, e nas principais conferências de Engenharia de Requisitos (ER), como IEEE Explorer, ACM Digital Library e Springer. Foram considerados como critérios de busca: “lei”, “requisito legal”, “rastreabilidade”, “gerenciamento de requisitos”, “engenharia de requisitos”, “requisito testável”, além dos mesmos termos em inglês; foram considerados os eventos: IEEE International Requirements Engineering Conference (RE), Workshop Engenharia de Requisitos (WER) - que engloba o Workshop Requirements Engineering and Law (RELAW); periódico: Requirements Engineering Journal. Esta pesquisa teve um caráter exploratório, como definido em [4].

As entrevistas com os especialistas foram semi-estruturadas para ampliar o entendimento de dificuldades, preocupações, mecanismos utilizados, fluxos de trabalho na busca pela manutenção da CL, dentre outros assuntos. Esta es-

tratégia foi utilizada por permitir a coleta de informações com maior controle do informante com relação uniformidade dos estímulos apresentados, como tópicos abordados, questões propriamente ditas e entonação de voz. Para planejamento das entrevistas foram consideradas as recomendações de [5], e [6]. O roteiro foi planejado de modo que cada entrevista tivesse a duração de 30 a 50 minutos. Desta forma, foi feita uma pequena introdução de cinco minutos para, na sequência, serem feitas perguntas sobre informações demográficas, formação e experiência do participante, ambiente profissional, gerenciamento de requisitos (GR), rastreabilidade e testes de *software*, RL e seu monitoramento; para então haver o encerramento e o agradecimento pela participação. A seguir são apresentados os resultados da pesquisa exploratória e das entrevistas realizadas.

2.1 Fundamentação Teórica

Identificou-se que muitos pesquisadores têm concentrado esforços para definir uma forma eficiente de extração dos requisitos a partir de leis e regulamentos [7,8,9,2]. Segundo [10], uma norma, de modo geral, é uma determinação que alguém, com autoridade ou poder, dirige a quem deva submissão ou obediência. As leis incluem normas que foram criadas com a intenção de determinar a operação de sistemas de computação [11]. As leis, assim como a sociedade, as organizações e os *softwares*, também evoluem [12]. Saber quais normas estão vigentes em determinadas datas, exige do cidadão familiaridade com o tema. Ademais, não é todo texto em linguagem natural que pode ser traduzido para linguagem formal sem que diversas suposições sejam feitas. Estas suposições representam risco para qualidade da especificação dos requisitos do *software* [11].

[13] reforçam a necessidade de desenvolvimento de SI, onde a legislação tem um impacto direto no que deve ou não ser feito. Segundo os autores, os RL são a forma encontrada pela ER para tornar possível que as normas relevantes para o domínio da aplicação sejam codificadas corretamente nos SI. [14] *apud* [15] chama de RL, os requisitos que são passíveis de sofrer alterações devido a prioridades decorrentes de legislação. Diante da existência de conflito de interesse entre os requisitos do usuário e a legislação vigente, [16] argumentam que para derivar os RL, deve-se estabelecer um processo sistemático para compatibilização dos conceitos jurídicos com os objetivos dos *stakeholders*, garantindo o cumprimento da lei. Para [17], leis e regulamentos tornam o processo de priorização mais difícil devido ao elevado custo de não cumprimento e do montante substancial de conhecimento do domínio do negócio necessário para tomada das decisões de priorização. Estes autores propõem um método específico de priorização de RL, que considera entre outros parâmetros, o número de referências cruzadas e a quantidade de exceções possíveis para o trecho da lei relacionado ao RL.

O GR ocorre durante o processo de ER, e envolve as atividades que ajudam a equipe de desenvolvimento a identificar, controlar e rastrear requisitos, gerenciando suas mudanças em qualquer momento ao longo do ciclo de vida do *software* [18,19]. As principais preocupações do GR são: i) gerenciar mudanças; ii) gerenciar o relacionamento entre os requisitos; iii) gerenciar as dependências entre os documentos de requisitos e os outros artefatos do sistema; iv) analisar

o impacto e o custo das alterações relacionadas aos requisitos alterados. [20] esclarecem que a gerência por requisitos tem por objetivo estabelecer uma visão comum entre o cliente e a equipe de projeto em relação aos requisitos que serão atendidos pelo *software*. [21] argumentam que este é um aspecto fundamental e crítico em qualquer processo de *software*. Assim, pode-se dizer que a gerência por requisitos pode ser recomendada nos processos ligados a melhoria da qualidade dos *softwares*. O uso desta prática podem também auxiliar o desenvolvimento orientado à CL. Isso porque o processo de monitoramento e avaliação dos RL do sistema exercem um papel de destaque durante o ciclo de vida do sistema.

Diversos trabalhos de pesquisas têm destacado a rastreabilidade como um dos mais importantes pré-requisitos para o desenvolvimento de *software* de qualidade [18,21,22,23,24]. A rastreabilidade de requisitos pode ser vista como a capacidade de acompanhar e descrever a vida de um requisito, em ambas as direções. A pré-rastreabilidade documenta o contexto de onde derivam os seus requisitos. A pós-rastreabilidade relaciona os requisitos ao projeto do sistema e sua implementação [21] *apud* [25]. Em [26], os autores apresentam um modelo de referência para rastreabilidade de requisitos legais no processo de contratação de soluções de TI na administração pública e argumentam que abordagens de rastreamento de requisitos são essenciais ao contexto de processos de negócio, principalmente nos casos onde os RL estão presentes. Para [27], a decomposição para quando o requisito está precisamente definido, isto geralmente acontece após três ou quatro níveis de refinamento, quando não há mais ambiguidade. Este mesmo autor diz que as condições de precisamente definido e não ambíguo somente são satisfeitas quando o requisito torna-se testável. Nesta seção, motiva-se o uso de rastreabilidade e do termo requisito testável, que será usado na abordagem.

2.2 Entrevistas

Foram feitas entrevistas individuais com cinco profissionais voluntários e participantes projetos com alguma interseção com a universidade, que são de três diferentes instituições, com reconhecidas expertises nas áreas de ER, desenvolvimento e gerenciamento de projetos de sistemas no mercado profissional com o propósito de identificar: 1) O entendimento dos conceitos relacionados com rastreabilidade, RL e CL; 2) As maiores preocupações relacionadas com aspecto de CL nos sistemas; 3) As maiores dificuldades de acompanhamento, evolução e gerenciamento dos requisitos (de sistemas) atualizados e integrados com os dados, como código-fonte, banco de dados e testes de *software*; 4) Os mecanismos utilizados para apoiar estas atividades (acompanhamento, evolução e gerenciamento dos requisitos); 5) Os fluxos de trabalho, definição de responsáveis e técnicas de priorização de requisitos; 6) Os fluxos do processo de testes, e se há planejamento para sua realização; 7) Se existe, e como é a participação de profissionais da área de Direito nas atividades relacionadas com a ER.

Com base nas entrevistas, os problemas encontrados foram sumarizados em: 1) embora haja sólidos conhecimentos da necessidade de garantir a rastreabilidade entre os artefatos relacionados com o sistema, é notório que, para estes profissionais, a realização de atividades de rastreabilidade não era a prática for-

mal (institucionalizada) e, sim, uma consequência do processo da atribuição de tarefas ou obtida parcialmente através de iniciativas individuais; 2) a adoção inapropriada de metodologias ágeis de desenvolvimento e deficiência na documentação os projetos também traziam alguma dificuldade à verificação da CL dos sistemas, quando não eram observadas, por exemplo, questões relacionadas à comunicação dos *stakeholders*, a criação de elos entre os artefatos disponibilizados e produzidos, a definição de estratégias, boas práticas, planos e políticas para o ciclo de vida dos sistemas pelos participantes, além de sua real perfilhação; 3) a ausência de acompanhamento da evolução da legislação relacionada ao *software* poderia fazer com que, em algum caso, acontecesse a inconformidade legal. O ônus disso poderia resultar em horas de retrabalho, multas e, até mesmo, prejuízos à credibilidade dos *stakeholders* envolvidos; 4) existia uma dificuldade em quantificar e controlar os RL dos projetos. Em parte, isto era o resultado do uso de ferramentas não adequadas, como às boas práticas, rastreabilidade, controle de versão; da não execução da política de rastreabilidade ou da adoção de puramente técnicas manuais. No *link*, <https://goo.gl/zWS0nh>, podem ser encontrados os documentos utilizados durante as entrevistas, testes e avaliações.

3 Abordagem

A razão da preocupação com os RL, diferenciando-os dos demais tipos de requisitos, recai nas distinções já citadas destes requisitos e nas sanções aplicáveis pelo não atendimento da legislação. Assim, qualquer outro requisito que tenha as mesmas característica pode se beneficiar da abordagem. Na Figura 1, apresenta-se uma perspectiva geral da abordagem proposta onde é possível identificar quatro etapas fundamentais, dividida nas raias de "equipe de desenvolvimento e testes", "gerência de projetos" e "equipe de requisitos", e alguns dos artefatos: 1. planejar rastreabilidade de requisitos; 2. selecionar requisitos legais; 3. realizar rastreabilidade de requisitos; 4. verificar e validar a conformidade legal.

A seguir, as etapas são desmembradas em atividades e artefatos/resultados esperados. Considerando um ambiente de desenvolvimento para linguagem de programação Java (opção feita para preparação das futuras experimentações), são apresentadas algumas sugestões de ferramentas digitais de código aberto (estrategicamente, pensando nas possíveis adaptações pelas equipes que resolverem adotá-las) e gratuitas para apoiar as atividades propostas.

ETAPA 1: Planejar rastreabilidade de requisitos - para estabelecer relações entre os requisitos e outros artefatos do SI, e possibilitar a rastreabilidade entre os artefatos, devem ser consideradas as próximas atividades. Os artefatos resultantes, como a política de rastreabilidade e as ferramentas/ambiente configurados, normalmente, podem ser reaproveitados em novos projetos. Desta forma, deve ser avaliada a necessidade de atualizar os artefatos ou gerar novos.

Atividade 1.1. Desenvolver/atualizar o modelo de rastreabilidade: devem ser definidos ou atualizados quais os itens serão controlados - artefatos/resultados esperados: versão preliminar da política de rastreabilidade, e (novo) modelo de rastreabilidade. A política de rastreabilidade influencia dire-

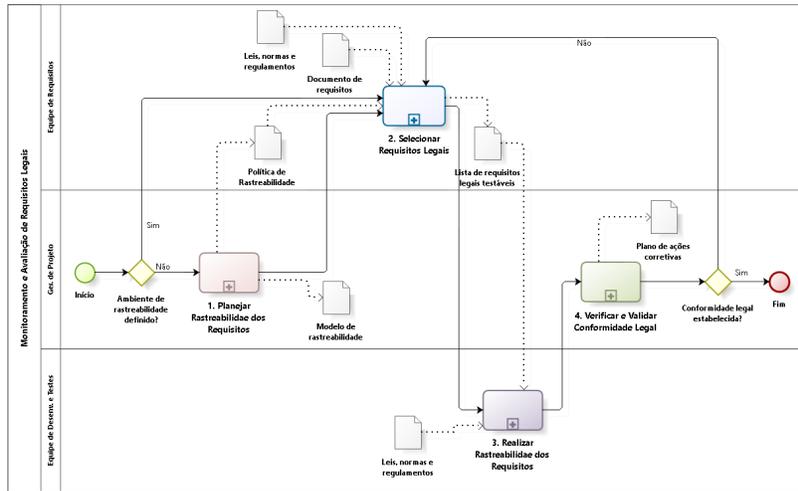


Figura 1. Visão Geral da Abordagem Proposta

tamente a construção do modelo de rastreabilidade. A política deve apresentar as técnicas e ferramentas para manutenção dos elos de rastreabilidade entre os artefatos; descrição dos pontos em que a informação de rastreabilidade deverá ser coletada; definição dos papéis das pessoas responsáveis pela manutenção das informações de rastreabilidade. Para a criação desta política devem ser considerados, segundo [28], fatores como número de requisitos, tempo de vida útil do sistema, nível de maturidade da organização, dimensão da equipe, tipo de sistemas desenvolvidos, requisitos específicos do cliente.

A Figura 2 apresenta um modelo de rastreabilidade elaborado com base na proposta apresentada em [29]. As classes destacadas são as modificações feitas. O modelo permite o gerenciamento de diversos tipos de rastreabilidade. Sendo também capaz de armazenar diferentes versões do projeto permitindo a criação de *baselines*. Este metamodelo é genérico, assim os atributos devem contextualizados e preenchidos. Ressalta-se a importância de se manter atualizadas as informações no modelo de rastreabilidade definido pela equipe e na política de rastreabilidade, e as informações operacionalizadas pelas ferramentas de GR, de gerenciamento de atividades e de codificação do *software*. Sugestão de ferramenta: Astah Community (ferramenta de modelagem UML).

Atividade 1.2. Validar o modelo de rastreabilidade: a validação do modelo de rastreabilidade a ser adotado deverá ser realizada pelos principais envolvidos no processo de desenvolvimento do *software*. O nível de granularidade da rastreabilidade deve ser adequado às necessidades do projeto para ser obtida a melhor relação entre a profundidade dos detalhes a serem registrados para tornar a tarefa de rastreabilidade mais gerenciável e o custo imputado. Esta validação pode ocorrer nas reuniões dos envolvidos para análise de relatórios das ferramentas adotadas para GR, por exemplo - artefato esperado: modelo

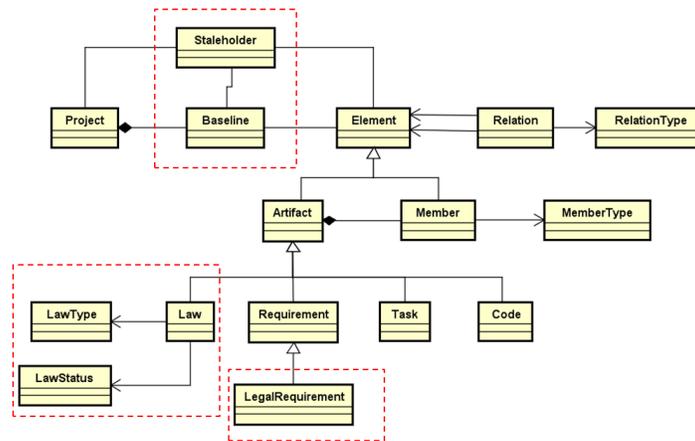


Figura 2. Modelo de rastreabilidade para Requisitos Legais, adaptado de [29]

de rastreabilidade validado. O aprimoramento e a consolidação do modelo de rastreabilidade poderão ocorrer ao longo das fases do desenvolvimento e manutenção do sistema. Sugestão de ferramenta: Filtros personalizados do Redmine (relatórios da ferramenta de GR).

Atividade 1.3. Preparar o ambiente de desenvolvimento para rastreabilidade: como tornar operacional o ambiente facilitando o registro dos elos de rastreabilidade. Nesta atividade são definidas boas práticas de codificação, regras de nomeação dos itens de configuração, de versionamento do código fonte, definição de papéis, fluxo de trabalho. Recomenda-se uso do plano elaborado pela Gerência de Configuração de Software (GCS) e, quando necessário, atualizá-lo com novas definições ou correções - artefato esperado: política de rastreabilidade consolidada e implementada na ferramenta de GR e atividades. Sugestões de ferramentas: Git (sistema de controle de versão); Redmine (sistema de Gerenciamento de requisitos e controle de tarefas (GRCT)); Redmine-RE (*plugin* para personalização do Redmine); IDE Eclipse (ferramenta de desenvolvimento); Mylyn (*plugin* para integração da IDE Eclipse com Redmine).

ETAPA 2: Selecionar requisitos legais - esta Etapa inclui as atividades para reunião das informações relevantes, elicitação, seleção, priorização e refinamento dos RL para aprimoramento da testabilidade do sistema/requisito.

Atividade 2.1. Reunir informações relevantes: identificar e analisar as informações relacionadas às partes da legislação, que têm relação com o domínio do negócio. Em seguida, devem ser feitos os procedimentos relacionados com a elicitação, a análise e a documentação dos requisitos do SI, considerando a compatibilização das leis com as necessidades dos *stakeholders*, ou os requisitos que foram implementados com novas determinações encontradas na legislação - artefatos esperados: artefato contendo o conjunto das informações relevantes sobre a lei aplicável, e lista de requisitos legais identificados. Sugestões de ferramentas:

LexML Brasil (banco de dados legislativo disponível via WEB); Redmine (utilização do sistema de GRCT para armazenamento das informações relevantes).

Atividade 2.2. Priorizar requisitos legais: a técnica a ser utilizada pela equipe para priorização dos RL, dentre os próprios RL, deve estar implementada no sistema de GR, ou equivalente, de modo que o foco esteja nos requisitos mais sensíveis à CL - artefato esperado: lista de requisitos legais priorizados de acordo com o plano de GR. Sugestão de ferramenta: Redmine (utilização de campo específico do sistema de GRCT para priorização dos requisitos).

Atividade 2.3. Refinar requisitos legais para aprimorar a testabilidade do sistema/requisito: o requisito deve ter um alto grau de refinamento, como defendido em [27], a ponto de ser implementado e testado, dando origem às funcionalidades do SI. Cada requisito derivado deve ser cadastrado no sistema de GR, ou equivalente, estabelecendo *links* diretos entre os requisitos mais abstratos até os requisitos testáveis - artefatos esperados: lista de RL testáveis, e lista de RL testáveis validados pelos *stakeholders*. Sugestão de ferramenta: Redmine (utilização do sistema de GRCT para cadastro de requisitos e tarefas associadas de forma hierárquica - pai/filho).

ETAPA 3: Realizar rastreabilidade de requisitos - o principal objetivo é possibilitar que a implementação das funcionalidades e o desenvolvimento de testes do *software* estejam orientados à verificação da CL do sistema. Assim, devem ser consideradas as seguintes atividades:

Atividade 3.1. Implementar e rastrear os requisitos legais: o foco da equipe está na implementação das funcionalidades dos sistema, entretanto a preocupação deve ser garantir o estabelecimento dos *links* entre o código fonte, tarefas (issues) e os diferentes artefatos envolvidos. Estratégias como o uso de *Annotations* (em Java) ou *Decorators* (em Python) no código fonte, e *Tagging* nas operações relacionadas ao sistema de controle de versões (SCV) são boas práticas de codificação - artefatos esperados: lista de tarefas cadastradas, política de rastreabilidade executada, artefatos com elos de rastreabilidade definidos, e lista de artefatos de prova (que são os artefatos que comprovam que um RL foi implementado no sistema. Podem ser diagramas, desenho, requisitos, dentre outros). Sugestões de ferramentas: IDE Eclipse (ambiente de implementação do código da aplicação); TagSEA *plugin* para criação de *tags* no código fonte); Git (uso de *tags* e mensagens de *logs* no SCV).

Atividade 3.2. Projetar testes de *software* dos requisitos legais: o planejamento para os testes deve começar com os primeiros estágios de desenvolvimento do *software*, e serem refinados contínua e sistematicamente. Idealmente, deve ser utilizado o modelo do plano de testes estabelecido - artefatos esperados: plano de testes de *software*, e roteiro de testes de *software*. Sugestões de ferramentas: Redmine (utilização do sistema de GRCT para atribuição de responsável pelo teste); JUnit (*framework* para execução de testes unitários); Selenium (ferramenta para execução de testes de aceitação no navegador).

Atividade 3.3. Realizar testes de *software* automáticos: diversos tipos de testes podem ser realizados. Durante a pesquisa bibliográfica, não foram encontradas referências que justifiquem a utilização de um tipo específico de teste

para verificação de requisitos legais. Este processo depende de diversos fatores, como o grau de maturidade da equipe e o tamanho do sistema que está sendo produzido. A prática de utilização de uma ferramenta de gerenciamento das atividades de testes do sistema deve ser observada - artefatos esperados: caso de testes de *software* dos requisitos legais, arquivos de *log* de execução dos testes de *software*, e relatório de execução dos testes de *software*. Sugestão de ferramenta: Maven (ferramenta de gerenciamento de dependência de bibliotecas - permite a execução automática dos testes no momento do empacotamento do *software*).

ETAPA 4: Verificar e validar conformidade legal - os processos de verificação e de validação asseguram que o *software* atende às especificações e às necessidades para as quais ele foi desenvolvido [30]. Logo, no contexto deste trabalho, a verificação e a validação da implementação dos requisitos legais têm como objetivo principal não somente o atendimento das especificações e das necessidades dos *stakeholders*, mas também o atendimento das normas vigentes e aplicáveis ao domínio do sistema. As atividades associadas a essa etapa podem permitir, conseqüentemente, ao gestor do projeto efetuar um conjunto de ações corretivas para garantir a qualidade do *software*. Essas atividades são:

Atividade 4.1. Analisar requisitos legais com artefatos de prova: esta atividade consiste em verificar se os RL foram devidamente implementados no sistema. Os elos de rastreabilidade estabelecidos nas etapas anteriores facilitarão a análise dos RL e a identificação dos artefatos produzidos. Ao inspecionar os artefatos de prova, RL, os RL testáveis, tarefas e o código fonte na busca pela CL - artefatos esperados: lista de tarefas cadastradas, política de rastreabilidade executada, artefatos com elos de rastreabilidade definidos, e lista de artefatos de prova (modelos, desenho, requisitos, *logs*, dentre outros). Sugestões de ferramentas: Redmine (relatórios disponíveis do sistema de GRCT); Surefire Maven plugin (*plugin* do Maven para geração de relatórios de testes realizados); SonarQube (ferramenta para auxiliar a análise da qualidade do código fonte).

Atividade 4.2. Validar os resultados da análise da conformidade legal junto aos *stakeholders*: os resultados da análise de CL do sistema devem ser comunicados aos *stakeholders*. Os riscos de não conformidade do sistema devem estar claros nessa fase do projeto para, então, serem realizadas as negociações finais - artefatos esperados: lista de requisitos legais validados, lista de requisitos legais não validados, plano de ações corretivas, e política de rastreabilidade validada. Sugestões de ferramentas: Redmine (sistema de GRCT) ou sistema de gerenciamento de projetos, para registrar as decisões, os conflitos e os encaminhamentos dados pelos *stakeholders* durante as reuniões realizadas.

3.1 Aplicação da Abordagem

Para ilustrar a aplicação da abordagem, buscou-se utilizar como exemplo um sistema cujos RL tenham sido previamente elicitados. O cenário apresentado em [14], que utilizou os modelos de *e-commerce* disponíveis nos trabalhos de [31] e [32], foi utilizado para validação do *framework* GenNormas. Foi adaptada deste trabalho uma história de usuário “ID: RL09 - Como cliente, eu quero ter acesso

seguro ao site, para gerenciar minhas transações”, como fonte de um requisito para facilitar o entendimento de cada atividade da abordagem desenvolvida.

Para realização da Etapa 1, a atividade de desenvolver/atualizar o modelo de rastreabilidade foi realizada partindo da identificação de quais itens de configuração precisavam ser controlados. Os membros da equipe identificaram os elementos que precisavam ser rastreados, e elaborar um diagrama de rastreabilidade. Operacionalização foi feita com a construção de um diagrama de classes auxiliado por uma ferramenta de modelagem UML. A validação do modelo de rastreabilidade ocorreu durante a preparação do ambiente de desenvolvimento para apoiar a construção dos elos de rastreabilidade entre os primeiros artefatos do projeto. A ferramenta de controle de versão integrada com a ferramenta de GR foi útil para satisfazer o modelo de rastreabilidade.

Para realização da Etapa 2, a atividade de reunir informações relevantes considerou a Lei do Comércio Eletrônico - Lei 7.962/2013 (artefato de lei - L001), Código de defesa do consumidor - Lei 8.078/1990 (artefato de lei - L002), o Marco Civil da Internet - 12.965/2014 (artefato de lei - L003) e outros documentos importantes que tratavam da política da organização. Assim, o artefato de lei L001 foi extraído como informação relevante o Art. 43, §1 (L001.1); de L002, o Art. 10 (L002.1); e de L003, o Art. 4º, inciso VII (L003.1). Seguindo o plano de rastreabilidade e as informações coletadas foram cadastradas na ferramenta de GR preparada na etapa anterior. A priorização de requisitos foi feita com a presença de um profissional experiente nas questões legais que envolviam cada RL. O RL09 foi priorizado como essencial na ferramenta, segundo os critérios de priorização definidos e acordados com os *stakeholders*.

Seguindo critérios de testabilidade, identificou-se que havia necessidade de refinar o requisito, e a descoberta de alguns requisitos funcionais (RF) derivados, com testabilidade satisfatória. O requisito RL09 transformou-se em “RL09.1 - Para realizar *login* no *site*, o cliente deverá realizar cadastro onde deverá fornecer uma senha classificada como, no mínimo, nível 8 de segurança segundo a política de definição de senhas da empresa, para que a segurança de suas transações seja garantida.” e “RL09.2 - Para garantir o tratamento seguro das informações do usuário, o fornecedor, através do site de comércio eletrônico, deverá realizar a autenticação do usuário utilizando um método combinado de criptografia de chave simétrica e assimétrica como SSL em todas as transações de compra”.

Na Etapa 3, as tarefas para implementação das funcionalidades foram criadas e distribuídas para os seus responsáveis fictícios, neste contexto, considerando o cronograma e a prioridade de cada uma; a rastreabilidade foi mantida entre as tarefas e os RL criados. As práticas estabelecidas na política de rastreabilidade, como a padronização dos nomes dos artefatos gerados estava institucionalizada, e o participante executou-a sistematicamente. Durante a implementação, os trechos de código que diziam respeito aos RL foram devidamente relacionados com as leis através das tarefas e do SCV. Diagramas, requisitos, relatórios e *logs*, foram “marcados” como artefatos de prova da implementação do RL. Os casos de testes também foram implementados tendo como base no plano de rastreabilidade. Os elementos do projeto foram devidamente rastreados.

Na Etapa 4, a análise dos RL e dos seus artefatos de prova foi realizada. A lista dos RL serviu de base para a verificação da conformidade do sistema pelo gerente do projeto. A partir da ferramenta de GR, foi verificado se todos os RL elicitados estavam sendo controlados e se possuíam tarefas cadastradas e devidamente realizadas. Os *links* e as marcações permitiram que os artefatos de prova fossem identificados. O gestor analisou os pontos de não conformidade, identificados a partir do relatório de testes dos RL, e realizou a inspeção do código fonte do sistema. Ao final, as funcionalidades implementadas que satisfizeram os RL foram apresentadas aos *stakeholders* para validação. Nos casos de não validação de algum RL, foram propostas ações corretivas.

4 Discussão

As soluções para os desafios ligados a derivação dos requisitos do sistema a partir das leis avançaram de forma a existirem algumas propostas satisfatórias como as apresentadas por [13,33,34]. Dentre as metodologias estudadas, as que têm relação mais próxima com a abordagem apresentada neste trabalho são as de [33] e [35]. A primeira, por mostrar-se mais sistemática nos processos iniciais de elicitação de requisitos; a segunda, apesar de tratar de RL para processos de negócio, pela promessa de melhoria da rastreabilidade do gerenciamento de requisitos legais do sistema. A série de tarefas propostas em [9], principalmente a “vi - manter a atualização e a gestão dos itens listados anteriormente em face à constante evolução das regulamentações e da organização”, também estão diretamente relacionadas a abordagem apresentada neste artigo.

Os trabalhos relevantes para proposição das atividades das Etapas 1 e 3 da abordagem foram os trabalhos de: i) [36], que introduziram questionamentos importantes sobre rastreabilidade e propuseram um dos primeiros e mais influentes modelos; ii) [26], que apresentaram um modelo de referência para rastreabilidade de requisitos legais no processo de contratação de soluções de TI na administração pública e argumentam que abordagens de rastreamento de requisitos são essenciais ao contexto de processos de negócio; iii) [29], que propôs um modelo para diferentes tipos de rastreabilidade, gestão de mudança, gestão de impactos e versionamento dos artefatos, além de permitir a criação de *baselines*.

Atender a legislação relacionada com o domínio é um pressuposto para boa qualidade dos sistemas, os RL precisam estar em constante adaptação ao dinamismo da lei, compatibilizando os objetivos das partes interessadas e cumprimento da lei. A Etapa 2, relacionada com elicitação e refinamento dos RL foi alicerçada principalmente nos trabalhos de [14,16,33]. A CL do sistema, como descrita por [37], é evidenciada pelos elos de rastreabilidade dos artefatos do sistema e da legislação, definidos a partir dos relacionamentos entre dois conjuntos particulares de especificações: as do sistema que se quer desenvolver e as especificações legais, oriundas do entendimento de documentos legais. O caráter da CL é alcançado quando há um perfeito alinhamento entre as especificações dos requisitos do usuário e as exigências expressas em lei [14]. A manutenção destes *links* está para além da criação dos elos, está no monitoramento contínuo como defendido por [38]. Estes trabalhos foram pilares para Etapa 4.

Em respostas às questões elaboradas, a solução oferecida pela abordagem baseia-se no gerenciamento e na rastreabilidade dos requisitos legais e seus relacionamentos com os artefatos do sistema. A abordagem oferece uma sistematização de práticas e artefatos, que evidenciam a CL do sistema, dispersos em diferentes metodologias de desenvolvimento, propondo atividades e artefatos esperados para cada atividade. Aparentemente, a abordagem pode ser aplicada de forma mais ampla, no controle de requisitos voláteis e críticos de sistemas de informação, já que tem um foco no monitoramento dos requisitos implementados com o apoio da rastreabilidade. A flexibilidade atribuída à abordagem pode facilitar a adaptação às metodologias e atividades utilizadas por qualquer equipe de desenvolvimento. A proposta de uma abordagem contendo elementos práticos com vistas à melhoria do processo de implementação e monitoramento dos requisitos legais pode facilitar a verificação e o cumprimento da legislação vigente, como indicaram estudos preliminares. Uma das limitações encontradas durante a realização das entrevistas diz respeito ao processo de execução da avaliação da abordagem em um ambiente real de produção de *softwares* que envolvam requisitos legais. Para criação de um ambiente para estudos e testes foram utilizados artefatos definidos e ferramental com características adequadas à abordagem. A abordagem sugere práticas que podem não estar presentes no ambiente, será necessário conhecer os processos das equipes para melhor adaptação.

5 Considerações Finais

A conformidade legal do sistema é imperativa para qualidade de *software*. Neste sentido, muito esforços são empenhados para alcançá-la e garanti-la. Elicitar bem os requisitos legais é um primeiro passo. Uma solução global deve incluir a implementação de mecanismo de execução eficiente de rastreabilidade, não só dos requisitos legais do sistema, mas dos artefatos envolvidos com as leis vigentes.

Essa pesquisa culminou na proposta de uma abordagem que foi dividida em etapas, atividades e artefatos necessários à promoção da conformidade de sistema com a legislação vigente. Para maior entendimento das atividades, foi utilizado uma aplicação de comércio eletrônico e sugeridas ferramentas que podem apoiar a instanciação da abordagem. Essa abordagem foi fundamentada em pesquisas bibliográficas e entrevistas com especialista no domínio.

Os trabalhos futuros identificados foram: i) estudos visando a aplicação em ambiente real de desenvolvimento; e ii) a instanciação da abordagem em uma ferramenta para facilitar sua implantação.

Referências

1. D.G. Gordon and T.D. Breaux. Assessing regulatory change through legal requirements coverage modeling. In *Requirements Engineering Conference (RE), 2013 21st IEEE International*, pages 145–154, July 2013.
2. D.G. Gordon and T.D. Breaux. The role of legal expertise in interpretation of legal requirements and definitions. In *Requirements Engineering Conference (RE), 2014 IEEE 22nd International*, pages 273–282, Aug 2014.

3. S. Ghanavati, A. Rifaut, E. Dubois, and D. Amyot. Goal-oriented compliance with multiple regulations. In *Requirements Engineering Conference (RE), 2014 IEEE 22nd International*, pages 73–82, Aug 2014.
4. Armando Piovesan and Edméa Rita Temporini. Pesquisa exploratória: procedimento metodológico para o estudo de fatores humanos no campo da saúde pública. *Rev Saúde Pública*, 29(4):318–25, 1995.
5. Robert S Weiss. *Learning from strangers: The art and method of qualitative interview studies*. Simon and Schuster, 1995.
6. Yvonne Rogers, Helen Sharp, and Jenny Preece. *Interaction design: beyond human-computer interaction*. Wiley, 2011.
7. Travis D Breaux, Matthew W Vail, and Annie I Anton. Towards regulatory compliance: Extracting rights and obligations to align requirements with regulations. In *Requirements Engineering, 14th IEEE International Conference*, pages 49–58. IEEE, 2006.
8. J.C. Maxwell and A.I. Anton. Checking existing requirements for compliance with law using a production rule model. In *Requirements Engineering and Law (RELAW), 2009 Second International Workshop on*, pages 1–6, Sept 2009.
9. Sepideh Ghanavati, Daniel Amyot, and Liam Peyton. A Systematic Review of Goal-oriented Requirements Management Frameworks for Business Process Compliance 2011 Fourth International Workshop on. *2011 Fourth International Workshop on Requirements Engineering and Law, (Relaw)*:25–34, 2011.
10. Carlos Schmieguel. Conceito de lei em sentido jurídico. *Ágora: revista de divulgação científica*, 17(1):p–128, 2010.
11. W. Hassan and Luigi Logrippo. Requirements and compliance in legal systems: a logic approach. In *Requirements Engineering and Law, 2008. RELAW '08.*, pages 40–44, Sept 2008.
12. N. Kiyavitskaya, A. Krausova, and N. Zannone. Why eliciting and managing legal requirements is hard. In *Requirements Engineering and Law, 2008. RELAW '08.*, pages 26–30, Sept 2008.
13. G. Boella, L. Humphreys, R. Muthuri, P. Rossi, and L. van der Torre. A critical analysis of legal requirements engineering from the perspective of legal practice. In *Requirements Engineering and Law (RELAW), 2014 IEEE 7th International Workshop on*, pages 14–21, Aug 2014.
14. Hidelberg Oliveira Albuquerque. Gennormas: um processo genérico para conformidade legal na engenharia de requisitos, Jul 2014.
15. A.K. Massey, P.N. Otto, and A.I. Anton. Prioritizing legal requirements. In *Requirements Engineering and Law (RELAW), 2009 Second International Workshop on*, pages 27–32, Sept 2009.
16. Alberto Siena, John Mylopoulos, Anna Perini, and Angelo Susi. From laws to requirements. In *Requirements Engineering and Law, 2008. RELAW'08.*, pages 6–10. IEEE, 2008.
17. A.K. Massey, P.N. Otto, and A.I. Anton. Prioritizing legal requirements. In *Requirements Engineering and Law (RELAW), 2009 Second International Workshop on*, pages 27–32, Sept 2009.
18. Ian Sommerville and Gerald Kotonya. *Requirements engineering: processes and techniques*. John Wiley & Sons, Inc., 1998.
19. Roger S Pressman. *Engenharia de software*. AMGH Editora, 2009.
20. Jose Marcos Goncalves and Andre Villas Boas. Modelo de maturidade de capacidade de software (cmm), Oct 2001.
21. Miriam Sayão and Julio Cesar Sampaio do Prado Leite. Rastreabilidade de requisitos. *RITA*, 13(1):57–86, 2006.

22. Matthias Jarke. Requirements tracing. *Communications of the ACM*, 41(12):32–36, 1998.
23. Balasubramaniam Ramesh and Matthias Jarke. Toward reference models for requirements traceability. *Software Engineering, IEEE Transactions on*, 27(1):58–93, 2001.
24. Richard Torkar, Tony Gorschek, Robert Feldt, Mikael Svahnberg, Uzair Akbar Raja, and Kashif Kamran. Requirements traceability: a systematic review and industry case study. *International Journal of Software Engineering and Knowledge Engineering*, 22(03):385–433, 2012.
25. Alan M Davis. *Software requirements: objects, functions, and states*. Prentice-Hall, Inc., 1993.
26. Lamartine da Silva Barboza, Cysneiros de A Filho, A Gilberto, and Ricardo AC de Souza. Towards a legal compliance verification approach on the procurement process of it solutions for the brazilian federal public administration. In *Requirements Engineering and Law (RELAW), 2014 IEEE 7th International Workshop on*, pages 39–40. IEEE, 2014.
27. Peter B Wilson. Sizing software with testable requirements. *Systems Development Management*, pages 34–10, 2000.
28. Pedro Rodrigo Caetano Strecht Ribeiro. *Metodologia para Equipas de Desenvolvimento de Requisitos de Sistemas de Informação*. PhD thesis, UNIVERSIDADE DO PORTO, 2008.
29. Pablo Dall’Oglio, João Pablo Silva da Silva, and Sérgio Crespo Coelho da Silva Pinto. Um modelo de rastreabilidade com suporte ao gerenciamento de mudanças e análise de impacto. *Workshop de Engenharia de Requisitos (WER 2010)*, 2010.
30. Ian Sommerville and Pete Sawyer. *Requirements engineering: a good practice guide*. John Wiley & Sons, Inc., 1997.
31. Jaelson Castro, Manuel Kolp, and John Mylopoulos. Towards requirements-driven information systems engineering: the tropos project. *Information systems*, 27(6):365–389, 2002.
32. MJNR Lucena. Stream: a systematic process to derive architectural models from requirements models. *Doctoral theses. Universidade Federal de Pernambuco, Centro de Informática. Fev*, 2010.
33. A. Siena, A. Perini, A. Susi, and J. Mylopoulos. A meta-model for modelling law-compliant requirements. In *Requirements Engineering and Law (RELAW), 2009 Second International Workshop on*, pages 45–51, Sept 2009.
34. Silvia Ingolfo, Alberto Siena, and John Mylopoulos. Goals and compliance in nomos 3. In *iStar*, 2014.
35. Sepideh Ghanavati. *Legal-URN Framework for Legal Compliance of Business Processes*. PhD thesis, Université d’Ottawa/University of Ottawa, 2013.
36. Balasubramaniam Ramesh, Timothy Powers, Curtis Stubbs, and Michael Edwards. Implementing requirements traceability: a case study. In *Requirements Engineering, 1995., Proceedings of the Second IEEE International Symposium on*, pages 89–95. IEEE, 1995.
37. Guido Governatori and Sidney Shek. Rule based business process compliance. In *RuleML (2)*. Citeseer, 2012.
38. William N Robinson. Implementing rule-based monitors within a framework for continuous requirements monitoring. In *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, pages 188a–188a. IEEE, 2005.