

SÉRGIO PATRIQUE ZOTTO

**O DESENVOLVIMENTO DE UM SISTEMA GERENCIADOR DE ARTIGOS  
ELETRÔNICOS UTILIZANDO A TECNOLOGIA JAWEB E O RATIONAL  
UNIFIED PROCESS NO GERENCIAMENTO DO PROJETO**

CURITIBA  
2009

SÉRGIO PATRIQUE ZOTTO

**O DESENVOLVIMENTO DE UM SISTEMA GERENCIADOR DE ARTIGOS  
ELETRÔNICOS UTILIZANDO A TECNOLOGIA JAWEB E O RATIONAL  
UNIFIED PROCESS NO GERENCIAMENTO DO PROJETO**

Monografia apresentada ao Curso de Pós Graduação Lato Sensu em Engenharia de Software, setor Escola Técnica da Universidade Federal do Paraná, como requisito parcial para obtenção da titulação de Especialista em Engenharia de Software.

Orientador: Professor Jaime Wojciechowski

CURITIBA  
2009

## **TERMO DE APROVAÇÃO**

**SÉRGIO PATRIQUE ZOTTO**

**O DESENVOLVIMENTO DE UM SISTEMA GERENCIADOR DE ARTIGOS  
ELETRÔNICOS UTILIZANDO A TECNOLOGIA JAVAWEB E O RATIONAL  
UNIFIED PROCESS NO GERENCIAMENTO DO PROJETO**

Monografia apresentada como requisito parcial para a obtenção da titulação de especialista, pelo Curso de Pós-Graduação Lato Sensu em Engenharia de Software, da Universidade Federal do Paraná, pela seguinte banca examinadora:

Orientador:

Professor Jaime Wojciechowski

Curitiba, 09 de junho de 2009

À minha esposa Patrícia, pela magnífica  
pessoa que é, por fazer parte em minha vida e  
por me trazer muitas felicidades e alegrias.

## **AGRADECIMENTOS**

A Deus (Jeová) e a Jesus Cristo, pela força espiritual que me concedem sempre.

Aos familiares e amigos, pela paciência que tiveram esperando dias e noites enquanto trabalhava em prol do desenvolvimento desta monografia.

A memória de meu pai, Eloe Zotto, pelo grande homem que foi.

A minha querida esposa Patrícia, por todo incentivo e ajuda.

A minha mãe Anézia, por toda compreensão e auxílio nos momentos mais difíceis.

Ao professor Jaime, pela ótima orientação e por me ajudar a conseguir realizar mais este grande passo em minha vida.

“Quem salva uma vida salva o mundo inteiro”.

*A Lista de Schindler - Citação de Talmud*

## RESUMO

O desenvolvimento de um sistema virtual de aprendizagem que simule a gerência de artigos eletrônicos é um tema que surgiu do interesse em se estudar e explorar as mais diversas linguagens de programação existentes, e pela necessidade de se encontrar um ambiente educacional que tivesse uma interface de fácil utilização por parte dos usuários comuns. Para auxiliar em todas as etapas do projeto, foi usado na abordagem de engenharia de software o rational unified process, um método prioritário que mostrou ser uma grande ferramenta que auxilia o desenvolvedor em todas as fases do projeto, contemplando desde a sua iniciação, ajudando a elaborar termos de abertura e planos de regras e de negócio, até as etapas de testes e homologação.

Apesar de existirem diversos sistemas que controlem a editoração de artigos, o presente projeto tem por diferencial utilizar a linguagem de programação Java voltada ao ambiente Web, ser totalmente orientado a objetos, e desenvolvido com uma interface amigável e com bibliotecas e plugins que ajudam a aperfeiçoar a relação interface-homem-computador e também por seguir todas as definições e passos estipulados pelo RUP.

## **ABSTRACT**

The development of a virtual environment of learning that simulated the electronic article management, is a subject that appeared of the interest in studying and exploring the most varied existing programming languages, and for the necessity of finding a learning system virtual that possessed an easy interface use for the common users. To assist in all of project stages, the rational unified process was used in the boarding of software engineering, a priority process that it showed to be a great tool that assists the development in all the project phases, contemplating since its initiation, helping to elaborate terms of opening and plans of rules and business, until the stages of tests and homologation.

Although exist diverse systems to that control the articles edit process, this project have the differential of use the programming language Java, be completely Object-oriented, and developed with a friendly interface and with libraries and plugins that help to improve the relation Computer-Human Interface and also following all the definitions and steps stipulated for the RUP.

## LISTA DE ILUSTRAÇÕES

FIGURA 01 - TELA DE CADASTRO DE NOVO USUÁRIO .....	35
FIGURA 02 - TELA DE CADASTRO DE NOVO USUÁRIO VALIDADA PELA JQUERY .....	36
FIGURA 03 - TELA DE ESCOLHA DE UF E CIDADE .....	37
FIGURA 04 - TELA DE ESCOLHA DE UF E CIDADE – BUSCA POR CIDADES.....	37
FIGURA 05 - TELA DE ESCOLHA DE UF E CIDADE – APRESENTAÇÃO DOS MUNICÍPIOS .....	38
FIGURA 06 - TELA DE ESCRITA DE UM NOVO ARTIGO.....	39
FIGURA 07 - TELA DE AVALIAÇÃO DE ARTIGOS.....	42
FIGURA 08 - FUNÇÃO DO CÁLCULO VETORIAL.....	43
FIGURA 09 - FUNÇÃO DO CÁLCULO DO TERM-FREQUENCY .....	44
FIGURA 10 - FUNÇÃO DO CÁLCULO DO INVERSE DOCUMENT FREQUENCY.....	44
FIGURA 11 - FUNÇÃO DO CÁLCULO DO TFXIDF.....	45
FIGURA 12 - FUNÇÃO DO CÁLCULO DO TFQ.....	45
FIGURA 13 - TELA INICIAL DE BUSCA DE ARTIGOS .....	46
FIGURA 14 - TELA DE EDIÇÃO DE PESOS .....	46
FIGURA 15 - TELA DE RESULTADO DA BUSCA POR RELEVÂNCIA .....	47

## LISTA DE TABELAS

TABELA 1 – ARTEFATOS DO SIGART PRODUZIDOS PELO RUP.....	20
---	----

## LISTA DE SIGLAS

AJAX	–	Asynchronous Javascript And XML
API	–	Application Programming Interface (ou Interface de Programação de Aplicativos)
AVA	–	Ambiente Virtual de Aprendizagem
CSS	–	Cascading Style Sheets
DAO	–	Model-view-controller
DOM	–	Document Object Model
DWR	–	Direct Web Remoting
GPL	–	General Public License
HTML	–	HyperText Markup Language (Linguagem de Marcação de Hipertexto)
IDE	–	Integrated Development Environment (Ambiente Integrado de Desenvolvimento)
IDF	–	Inverse Document Frequency
IHC	–	Interface Homem Computador
JS	–	Arquivo com código JavaScript
JSP	–	Java Server Pages
MVC	–	Model-view-controller
PDF	–	Portable Document Format
PGP	–	Plano Global do Projeto
POO	–	Programação Orientada a Objetos
RUP	–	Rational Unified Process
SGBD	–	Sistema Gerenciador de Banco de Dados
SIGART	–	Sistema Gerenciador de Artigos
SQL	–	Structured Query Language
TF	–	Term-Frequency
UML	–	Unified Modeling Language
XHTML	–	eXtensible Hypertext Markup Language
XML	–	EXtensible Markup Language (Linguagem extensível de formatação)
XSLT	–	eXtensible Stylesheet Language for Transformation

## SUMÁRIO

<b>1 INTRODUÇÃO .....</b>	<b>13</b>
<b>2 AMBIENTES VIRTUAIS DE APRENDIZAGEM .....</b>	<b>16</b>
<b>3 A ENGENHARIA DE SOFTWARE .....</b>	<b>18</b>
<b>4 O RATIONAL UNIFIED PROCESS .....</b>	<b>19</b>
<b>5 FERRAMENTAS E TECNOLOGIAS APLICADAS .....</b>	<b>22</b>
5.1 JAVA.....	22
5.1.1 Frameworks .....	23
5.1.2 Bibliotecas .....	24
5.2 ECLIPSE .....	25
5.3 JQUERY .....	25
5.4 AJAX.....	27
5.5 TOMCAT .....	28
5.6 MYSQL.....	29
<b>6 SIGART (SISTEMA GERENCIADOR DE ARTIGOS).....</b>	<b>31</b>
6.1 OBJETIVOS E JUSTIFICATIVA DO PROJETO .....	31
6.2 DEFINIÇÃO DO ESCOPO .....	31
6.3 ESTRUTURAÇÃO DO SISTEMA E PRINCIPAIS FUNÇÕES.....	34
6.3.1 Sistema de cadastro com Jquery e Ajax.....	34
6.3.2 Escrevendo um novo artigo.....	38
6.3.3 Definindo o avaliador.....	39
6.3.4 Avaliando um artigo.....	41
6.3.5 Sistema de busca inteligente .....	42
6.3.5.1 Método vetorial.....	42
6.3.5.2 Método TFXIDF .....	43
6.3.5.3 Keyword e TFXIDF na prática .....	45
6.3.6 Sistema de segurança.....	50
6.3.6.1 Controle de acessos.....	50
6.3.6.2 Restrições de acessos .....	51

6.3.7 Feed RSS .....	52
<b>7 DIAGRAMAS .....</b>	<b>53</b>
7.1 DIAGRAMA DE CASOS DE USO.....	53
7.2 DIAGRAMA DE CLASSES.....	54
7.3 DIAGRAMA DE BANCO DE DADOS .....	55
7.4 DIAGRAMA DE ESTADO .....	56
7.5 DIAGRAMA DE ATIVIDADES.....	57
7.6 DIAGRAMA DE COMPONENTES.....	58
7.7 DIAGRAMAS DE SEQUÊNCIA.....	59
7.7.1 DS01.01 - Efetuar Login no Sistema - FP.....	59
7.7.2 DS01.01.01 - Visualizar Página Principal - FP.....	60
7.7.3 DS01.02 - Realizar Cadastro no Sistema - FP .....	61
7.7.4 DS01.02.01 - Efetuar Confirmação de Cadastro - FP .....	62
7.7.5 DS01.03 - Reencaminhar Senha - FP .....	63
7.7.6 DS01.04 - Visualizar Categoria - FP .....	64
7.7.7 DS01.04.01 - Ver Detalhes da Categoria - FP.....	65
7.7.8 DS01.04.02 - Ver Feed da Categoria - FP.....	66
7.7.9 DS01.05 - Visualizar Artigo - FP .....	67
7.7.10 DS01.05.01 - Dar Nota ao Artigo - FP .....	68
7.7.11 DS01.05.02 - Visualizar Mensagens do Fórum do Artigo - FP.....	69
7.7.12 DS01.05.03 - Inserir Mensagem no Fórum do Artigo - FP.....	70
7.7.13 DS01.06 - Visualizar Buscas - FP.....	71
7.7.14 DS01.06.01 - Realizar Busca de Artigos - FP1.....	72
7.7.15 DS01.06.01 - Realizar Busca de Artigos - FP2.....	73
7.7.16 DS01.06.01 - Realizar Busca de Artigos - FP3.....	74
7.7.17 DS01.06.02 - Realizar Busca de Usuários - FP.....	75
7.7.18 DS01.06.03 - Realizar Buscas de Mensagens dos Fóruns - FP .....	76
7.7.19 DS01.07 - Ver Meus Artigos Favoritos - FP.....	77
7.7.20 DS01.08 - Visualizar Dados Pessoais - FP.....	78
7.7.21 DS01.08.01 - Alterar Imagem Pessoal - FP.....	79

7.7.22 DS01.08.02 - Alterar Senha - FP .....	80
7.7.23 DS01.08.03 - Alterar Dados Pessoais - FP.....	81
7.7.24 DS01.09 - Visualizar Meus Artigos - FP.....	82
7.7.25 DS01.09.01 - Escrever Artigo - FP.....	83
7.7.26 DS01.09.02 - Ver Artigos em Avaliação - FP.....	84
7.7.27 DS01.09.03 - Ver Artigos com Restrições - FP.....	85
7.7.28 DS01.09.04 - Visualizar Artigo Meus Artigos - FP.....	86
7.7.29 DS01.09.05 - Ver Avaliação do Artigo - FP.....	87
7.7.30 DS01.09.06 - Editar Artigo - FP .....	88
7.7.31 DS01.09.07 - Ver Artigos Rejeitados - FP .....	89
7.7.32 DS01.09.08 - Ver Artigos Publicados - FP.....	90
7.7.33 DS01.10 - Ver Estatísticas do Site - FP.....	91
7.7.34 DS01.11 - Entrar no Chat do Site - FP.....	92
7.7.35 DS01.12 - Visualizar Usuário do Sistema - FP .....	93
7.7.36 DS01.12.01 - Encaminhar email a um usuario - FP .....	94
7.7.37 DS02.01- Visualizar Página de Avaliação - FP.....	95
7.7.38 DS02.01.01 - Ver Artigos em Avaliação - FP.....	96
7.7.39 DS02.01.02 - Ver Avaliações Anteriores - FP.....	97
7.7.40 DS02.01.03 - Avaliar Artigo - FP.....	98
7.7.41 DS02.01.04 - Visualizar Artigo Avaliação - FP .....	99
7.7.42 DS02.01.05 - Ver Artigos Rejeitados - FP .....	100
7.7.43 DS02.01.06 - Ver Artigos Publicados - FP.....	101
7.7.44 DS02.01.07 - Banir Artigo - FP .....	102
7.7.45 DS03.01 - Visualizar Página de Administração - FP .....	103
7.7.46 DS03.01.01 - Ativar Usuário - FP.....	104
7.7.47 DS03.01.02 - Banir Usuário - FP .....	105
7.7.48 DS03.01.03 - Cadastrar Avaliador - FP .....	106
7.7.49 DS03.01.04 - Cadastrar Administrador - FP.....	107
7.7.50 DS03.01.05 - Cadastrar Categoria - FP.....	108
<b>8 CONCLUSÃO .....</b>	<b>109</b>

<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>110</b>
<b>ANEXO – PLANO GLOBAL DO PROJETO.....</b>	<b>112</b>

## 1 INTRODUÇÃO

Com o passar do tempo, as tecnologias ganharam espaço no cenário mundial e praticamente todas as áreas crescem paralelamente com o surgimento de inovações tecnológicas. A saúde, agricultura, transportes, educação, entre outras áreas, aprimoraram e continuam a aprimorar seus estudos.

Atualmente o sistema educativo vem ampliando seu espaço de aprendizado com o rápido crescimento dos meios de comunicação, como, por exemplo, jornais, revistas, programas de tv, e a internet, uma ferramenta que disponibiliza a seus usuários uma vasta gama de informações e que em apenas frações de segundo pode se trazer tudo o que for requerido. A internet não é uma tecnologia tão antiga, mas em pouco tempo ganhou espaço no mundo e tende a cada dia crescer mais e mais. Dentro do contexto web, se é possível fazer praticamente tudo, desde ler notícias em tempo real, ver filmes, ouvir músicas, consultar enciclopédias e até cursar uma universidade. A educação é um campo que acompanha paralelamente a internet em seu crescimento, pois a cada nova tecnologia desenvolvida, a base educacional aprimora os seus métodos de ensino.

Existem hoje diversos modelos de ensinios criados para a tecnologia web, tais como enciclopédias on-line, fóruns, livros digitais, aulas telepresenciais e os Ambientes Virtuais de Aprendizado (AVA) que fazem parte dos grandes sites educacionais e são uma excelente ferramenta de estudo e aprendizado. Um AVA é nada mais nada menos que uma interação homem computador (IHC) que potencializa e estimula a construção e elaboração de conhecimentos, tendo como resultado final à aprendizagem.

Todavia, com todos os avanços tecnológicos, tanto da educação, quanto de qualquer outra área, são necessários acompanhamentos no desenvolvimento de sistemas informatizados, e é nesse ponto que entra a Engenharia de Software, que visa proporcionar um melhor estudo de caso para elaboração, implantação e manutenção de sistemas.

A presente monografia intitulada “O desenvolvimento de um sistema gerenciador de artigos utilizando a tecnologia JAVAWEB e o rational unified process no gerenciamento do projeto” foi desenvolvida com base na elaboração de um sistema virtual de aprendizagem que simule um gerenciamento de artigos com o objetivo de facilitar a integração entre os usuários do sistema e facilitar a capacidade

de aprendizagem. No software, leitores, avaliadores e administradores podem interagir dinamicamente utilizando as mais novas ferramentas Web.

Apesar de existirem diversos sistemas que controlem a editoração de artigos, o presente projeto tem por diferencial utilizar e explorar a linguagem de programação Java, totalmente orientada a objetos, sendo desenvolvido com uma interface amigável e com bibliotecas e plugins que ajudam a desenvolver a relação IHC. Em paralelo ao desenvolvimento do projeto foram usadas as melhores técnicas e ferramentas de engenharia de software com base no RUP (Rational Unified Process), visando um ótimo acompanhamento e auxílio na estrutura arquitetural do projeto.

Assim, o objetivo geral da presente pesquisa é verificar a adequação dos temas organizacionais apresentados (tecnologia, gerência de projetos e desenvolvimento), no âmbito dos ambientes virtuais de aprendizados e concretizar o desenvolvimento de um sistema ágil com objetivos específicos da área educacional.

Do segundo ao quarto capítulo, serão apresentados, respectivamente, os conceitos básicos de ambientes virtuais de aprendizado, engenharia de software e o rational unified process, bem como suas aplicações dentro do projeto desenvolvido.

O quinto capítulo vem mostrar detalhadamente, todas as características das principais tecnologias utilizadas no desenvolvimento do sistema, quais as suas principais funções e aplicações realizadas.

No sexto capítulo, serão exibidas todas as informações necessárias sobre o sistema gerenciador de artigos, mostrando os seus principais objetivos, seu escopo, e toda estruturação do sistema e suas principais funções, tais como: sistema de cadastro com JQuery e Ajax, escrita e avaliação de artigos, definição de avaliadores, sistema de busca inteligente, segurança em acessos, entre outros.

No sétimo capítulo, são expostos os diagramas UML e de banco de dados elaborados durante as fases do RUP em sua última interação, contendo todas as associações, atores, relacionamentos, métodos e atributos. Dentre os diagramas estão: diagrama de casos de uso, diagrama de classes, diagrama de banco de dados, diagrama de estado, diagrama de atividades, diagrama de componentes e diagramas de seqüência dos fluxos principais dos casos de uso (os outros diagramas de seqüência, e também os arquivos originais de todos os diagramas, encontram-se na documentação do software dentro do CD que acompanha a monografia).

Para finalizar, é apresentado no Anexo deste trabalho o plano global do projeto, que contém todas as informações referentes à estipulação das metas, análise e regras definidas para o desenvolvimento do software, bem como uma visão geral dos objetivos e escopo do projeto de uma maneira macro para atender aos altos pontos do sistema.

## 2 AMBIENTES VIRTUAIS DE APRENDIZAGEM

Com o avanço tecnológico das últimas décadas, pudemos perceber o quanto à educação ultrapassou fronteiras antes impostas apenas a salas de aulas e livros. Com o surgimento da internet e o rápido crescimento dos meios de comunicação, como, por exemplo, jornais, revistas e programas de tv, acompanhamos que o campo educacional teve um crescimento relevante e bastante notado em nossos dias, pois a cada nova tecnologia desenvolvida, a base educacional aprimora os seus métodos de ensino.

Existem hoje diversos modelos de ensinamentos criados para a internet, tais como enciclopédias on-line, fóruns, livros digitais, aulas tele-presenciais e os Ambientes Virtuais de Aprendizado (AVA) que fazem parte dos grandes sites educacionais e são uma excelente ferramenta de estudo e aprendizagem. Um AVA é nada mais nada menos que uma interação homem computador (IHC) que potencializa e estimula a construção e elaboração de conhecimentos, tendo como resultado final à aprendizagem.

Para Pereira (2007) um AVA consiste em uma opção de mídia que está sendo utilizada para mediar o processo ensino-aprendizagem à distância tornando-o mais ativo, dinâmico e personalizado, não esquecendo que a tecnologia é apenas um meio de acesso, pois a ênfase sempre está na proposta, no conteúdo pedagógico e no desenvolvimento do processo educativo.

O SIGART foi desenvolvido para ser uma abordagem prática de um ambiente virtual de aprendizagem, com o intuito de estimular a capacidade de ensino e aprendizagem entre os usuários do sistema.

Um AVA deve ser dinâmico, eficaz e capaz de despertar o interesse do usuário nos tópicos apresentados, por isso, um sistema educacional deve possuir uma interface gráfica amigável e com recursos interativos entre seus usuários. Com essa idéia em mente, durante o desenvolvimento do sistema gerenciador de artigos, foram utilizadas tecnologias que auxiliaram a ambientação gráfica do software, para que tivesse uma interface rica, de fácil utilização e com recursos que estimulam o desejo de se utilizar o sistema.

Alem de uma interface bem apresentável, foram empregadas técnicas e opções que aumentassem o interesse dos usuários, como a utilização de fóruns de discussões, bate-papo dinâmico entre usuários online no site, encaminhamento de

emails e mensagens, utilização de RSS para acompanhamento de novos artigos escritos, entre outras.

Assim, o SIGART foi proposto com o intuito de ser o mais atraente possível, não só em matéria de apresentação, mas também de conteúdo e tecnologias empregadas, para poder ter uma boa aceitação por parte de seu público alvo.

### 3 A ENGENHARIA DE SOFTWARE

A Engenharia de Software tem por objetivo proporcionar um melhor estudo de caso para elaboração, implantação e manutenção de sistemas com abordagens sistemáticas, disciplinadas e quantificáveis. Desta forma, ela se preocupa nos métodos de como se proceder nas diversas atividades envolvidas no processo de desenvolvimento de software de forma que como resultado final se tenha um produto elaborado com maior qualidade, menor custo e dentro do prazo estimado.

A engenharia de software é uma área de conhecimento bastante abrangente, envolvendo diversos tipos de atividades, desde a programação até áreas mais gerenciais como controle de qualidade.

Segundo Pressman (2006), um princípio-chave de todos os modelos de engenharia de software é entender melhor o problema antes de tentar projetar uma solução e entender se os requisitos de um sistema estão entre as tarefas mais difíceis e mais críticas enfrentadas por um engenheiro de software. A dificuldade está relacionada, principalmente, ao processo de comunicação entre clientes e analistas.

Pensando em todo este contexto, foi abordado no acompanhamento do projeto de software, o RUP, que contempla diversas técnicas da engenharia de software, como planos de casos pré-definidos, artefatos e uma grande documentação UML envolvendo os mais diversos tipos de diagramas.

Além do RUP, foi desenvolvido o Plano Global do Projeto, que estabelece as diretrizes básicas de gerência de projetos, contendo todas as definições e análises do software desenvolvido, bem como cronograma com prazos e metas. O PGP do SIGART encontra-se no Anexo da presente monografia.

## 4 O RATIONAL UNIFIED PROCESS

Um grande problema enfrentado atualmente é a complexidade dos negócios dos projetos atuais, pois cada vez mais os sistemas se tornam complexos e precisam estar prontos em menor tempo. Com o intuito de auxiliar nesse ponto crítico, a Rational desenvolveu o RUP (Rational Unified Process), uma metodologia com um produto comercializado que é uma grande documentação baseada em hipertexto (HTML).

Para Kruchten (2003), o Rational Unified Process unifica toda a equipe de desenvolvimento de software e otimiza a produtividade de cada membro da equipe, colocando em suas mãos a experiência coletiva e derivada de milhares de projetos e de diversos líderes da indústria, além de oferecer:

- Maior orientação para desenvolvimento eletrônico
- Guias que fornecem avaliações de como aplicar o processo a uma ampla variedade de projetos e tecnologias
- Análise expandida de teste, abrangendo todo ciclo de vida do produto.
- Cobertura aperfeiçoada do design de interface da aplicação – especialmente como é aplicado ao desenvolvimento efetivo de aplicações Web
- Detalhes aperfeiçoados para desenvolvimento em tempo real e sistemas reativos
- Entendimentos dos sistemas de design usando padrões e estruturas.

O RUP tem sua estrutura básica formada por quatro fases por quais todo projeto deve passar, sendo que no final de cada fase temos um marco principal atingido:

- Concepção: ênfase no escopo do sistema contendo os workflows necessários para que as partes interessadas (stakeholders) concordem com os objetivos, arquitetura e o planejamento do projeto;
- Elaboração: ênfase na arquitetura, buscando complementar o levantamento / documentação dos casos de uso, voltado para a arquitetura do sistema.
- Construção: ênfase no desenvolvimento físico do software, produção de códigos, testes alfa e beta.

- Transição: ênfase na implantação, contemplando a entrega do software, o plano de implantação e acompanhamento e qualidade do software.

Além das quatro fases, o RUP engloba nove disciplinas que fazem parte do ciclo e vida do projeto de software:

- Modelagem do negócio
- Requisitos
- Análise & Projeto
- Implementação
- Testes
- Implantação
- Gerenciamento e planejamento
- Gerencia de configuração e mudanças
- Ambiente

O projeto SIGART seguiu rigorosamente os artefatos estipulados pelo RUP na sua versão 2002.05.00 em Português, aplicando desde sua análise e levantamento de requisitos, até sua implementação e implantação.

Abaixo segue uma tabela contendo todos os artefatos produzidos em cada fase e disciplina do RUP. Lembrando ainda, que todos os artefatos e documentações encontram-se dentro do CD que acompanha a monografia.

Tabela 01: Artefatos do SIGART produzidos pelo RUP

<b>Fase</b>	<b>Disciplina</b>	<b>Nome do Artefato</b>
<b>Iniciação</b>	Modelagem de Negócios	1. Avaliação da Organização-alvo 2. Documento de Arquitetura de Negócios 3. Especificação Suplementar de Negócios 4. Guia de Modelagem de Negócios 5. Regras de Negocio 6. Casos de Uso Negociais (Diagrama e especificação macro)
<b>Iniciação</b>	Requisitos (Iteração 1)	1. Visão 2. Solicitações dos Principais Envolvidos 3. Modelo de Casos de Uso (Atores, Casos de Uso) 4. Glossário 5. Especificações Suplementares

		do Projeto 6. Protótipo de Interface de Usuário
<b>Iniciação</b>	Requisitos (Iteração 2)	1. Especificações de Casos de Uso Negociais com Interfaces 2. Diagrama de Classes de Negócio (sem atributos) 3. Casos de Uso Negociais Detalhado
<b>Iniciação</b>	Gerenciamento de Projeto	1. Plano Global do Projeto ( Caso de Negócio, Plano de Desenvolvimento de Software, Lista de Riscos, Plano de Iteração, Avaliação da Iteração, Avaliação do Status)
<b>Iniciação</b>	Ambiente	1. Caso de Desenvolvimento 2. Guia de Programação
<b>Elaboração</b>	Análise e Design (Iteração 1)	1. Caso de Uso (Completo) 2. Diagramas de Seqüência 3. Diagramas de Classes com atributos e métodos 4. Modelo Físico de Dados 5. Diagrama de Atividades 6. Diagrama de Transição de Estados
<b>Elaboração</b>	Análise e Design (Iteração 2)	1. Modelo de Dados 2. Documento de Arquitetura de Software
<b>Construção</b>	Implementação	1. Plano de Integração do Building 2. Diagrama de Componentes 3. Documento de Arquitetura de Software - Revisado
<b>Construção</b>	Teste	1. Plano de Teste 2. Casos de Teste, Dados de Teste, Conjunto de Testes, Log de Testes 3. Sumário de Avaliação de Testes
<b>Transição</b>	Implantação	1. Plano de Implantação 2. Lista de Materiais 3. Produto 4. Material de Suporte para o Usuário - Notas de Release
<b>Transição</b>	Gerenciamento de Configuração de Mudança	1. Documento de Solicitação de Mudança

## 5 FERRAMENTAS E TECNOLOGIAS APLICADAS

### 5.1 JAVA

A linguagem Java<sup>1</sup> teve sua origem na década de 90, com a empresa Sun Microsystems, e uma de suas principais características, é ser uma linguagem de programação totalmente orientada a objetos (POO).

O diferencial da linguagem é a sua compilação que se denomina bytecode<sup>2</sup>, sendo posteriormente executada em uma máquina virtual, em contrapartida de outras linguagens que são compiladas a um código nativo, se tornando assim dependentes de determinada plataforma.

Pode-se dizer que linguagem Java foi projetada para possuir as seguintes características:

- Ter orientação a objeto, sendo baseada no modelo Smaltalk e Simula67.
- Ter portabilidade, ou seja, independência de plataforma, podendo rodar em qualquer sistema operacional.
- Possuir recursos de rede, com extensa biblioteca de rotina que pode facilitar a cooperação com protocolos TCP/IP, como http e ftp.
- Dar atenção à segurança, executando programas via rede com restrições.
- Ter sintaxe parecida com a linguagem C/C++.
- Facilidade de internacionalização, pois suporta caracteres unicode<sup>3</sup>.
- Simplicidade na especificação, linguagem e ambiente de execução da JVM.
- Distribuição de um grande conjunto de bibliotecas, chamadas API.
- Criação de programas distribuídos e multitarefas, ou seja, múltiplas linhas de execução em um mesmo programa.

---

<sup>1</sup> **Java:** Disponível em: <<http://java.sun.com>>. Acesso em: 30 out. 2008.

<sup>2</sup> **Bytecode:** Disponível em: <[http://developers.sun.com/appserver/reference/techart/tip6\\_august\\_2003.html](http://developers.sun.com/appserver/reference/techart/tip6_august_2003.html)>. Acesso em: 30 out. 2008.

<sup>3</sup> **Unicode:** Padrão em que cada código representa um símbolo, independente da localidade. Padronizada pelo Unicode-Consortium. Disponível em: <<http://www.unicode.org>>. Acesso em: 30 out. 2008.

- Deslocamento de memória por um processo coletor de lixo (Garbage Collector).
- Carga dinâmica de código, pois os programas são formados por uma coleção de classes armazenadas independentes e que são carregadas apenas quando utilizadas.

Para elaboração do sistema foram usados três padrões de projetos para auxiliar no desenvolvimento: MVC, Singleton e o padrão DAO.

O MVC (Model-View-Controller) é um padrão que tem por objetivo a separação das tarefas de acesso aos dados, lógica de negócio e lógica de apresentação. Dentro do SIGART podemos perceber isso através dos pacotes que recebem o mesmo nome e com funções específicas diferenciadas.

O DAO (Data Access Object) é um padrão para persistência de dados que permite separar regras de negócio das regras de acesso a banco de dados. No sistema gerenciador de artigos, existe um pacote chamado DAO que contém todas as classes de acesso ao banco de dados gerenciando e instanciando as classes modelos.

O padrão Singleton é um padrão de projeto de criação de objetos. O objetivo é garantir uma instância única e acessível de forma global para toda classe que implementar este padrão. Neste contexto, este padrão foi usado para a elaboração de uma classe que controle o sistema de segurança, onde um usuário escolhe os níveis de acesso para cada página ou arquivo, e o padrão se encarrega de instanciar uma única vez os dados, para futuras verificações de segurança.

### 5.1.1 FrameWorks

O DWR<sup>4</sup> foi o único framework utilizado no desenvolvimento. Ele permite construir aplicações web AJAX , ou seja, construir aplicações mais dinâmicas e interativas com conexões assíncronas.

---

<sup>4</sup> **DWR:** Framework Ajax exclusivo para linguagem Java. Disponível em: <<http://directwebremoting.org/>>. Acesso em: 02 dez. 2008.

### 5.1.2 Bibliotecas

Foram utilizadas diversas APIs de auxílio na programação do SIGART, todas gratuitas e de acesso livre. Abaixo segue a relação de bibliotecas e onde cada uma pode ser encontrada:

- activation.jar - Disponível em:  
<http://java.sun.com/javase/technologies/desktop/javabeans/jaf/downloads/index.html>
- commons-email-1.1.jar - Disponível em:  
[http://commons.apache.org/downloads/download\\_email.cgi](http://commons.apache.org/downloads/download_email.cgi)
- commons-fileupload-1.2.1.jar - Disponível em:  
[http://commons.apache.org/downloads/download\\_fileupload.cgi](http://commons.apache.org/downloads/download_fileupload.cgi)
- commons-io-1.4.jar - Disponível em:  
[http://commons.apache.org/downloads/download\\_io.cgi](http://commons.apache.org/downloads/download_io.cgi)
- commons-logging-1.1.1.jar - Disponível em:  
[http://commons.apache.org/downloads/download\\_logging.cgi](http://commons.apache.org/downloads/download_logging.cgi)
- core-renderer.jar - Disponível em:  
<http://source.concord.org/cgi-bin/viewcvs.cgi/Projects/Portfolio/Attic/core-renderer.jar?cvsroot=CodeBank>
- itext-155.jar - Disponível em:  
<http://itextpdf.sourceforge.net/>
- jstl.jar - Disponível em:  
[http://jakarta.apache.org/site/downloads/downloads\\_taglibs-standard.cgi](http://jakarta.apache.org/site/downloads/downloads_taglibs-standard.cgi)
- junit-4.5.jar - Disponível em:  
<http://www.junit.org/node/401>
- mail.jar - Disponível em:  
<http://java.sun.com/products/javamail/>
- mysql-connector-java-5.1.7-bin.jar - Disponível em:  
<http://dev.mysql.com/downloads/connector/j/5.1.html>
- rssutils.jar - Disponível em:

[http://java.sun.com/developer/technicalArticles/javaserverpages/rss\\_utilities/](http://java.sun.com/developer/technicalArticles/javaserverpages/rss_utilities/)

- standard.jar - Disponível em:  
[http://jakarta.apache.org/site/downloads/downloads\\_taglibs-standard-1.0.cgi](http://jakarta.apache.org/site/downloads/downloads_taglibs-standard-1.0.cgi)
- Tidy.jar - Disponível em:  
<http://sourceforge.net/projects/jtidy>

## 5.2 ECLIPSE

O Eclipse<sup>5</sup> é o resultado do trabalho realizado por uma comunidade de software livre que tem seu foco na construção de uma plataforma de desenvolvimento com frameworks, ferramentas e runtimes que possibilitam a construção, desenvolvimento e gerenciamento do programa durante o seu ciclo de vida.

A plataforma Eclipse está sendo desenvolvida, complementada e tem suporte por parte de um grande grupo de comerciantes tecnológicos, inovadores, acadêmicos, instituições de pesquisa e usuários comuns.

A IDE eclipse é considerada no meio acadêmico como uma das melhores ferramentas de desenvolvimento de código aberto, com possibilidade de adaptação de diversos plugins e tem seu foco principal no apoio à codificação (automação, refatoração, extreme programming) em vez da construção visual.

## 5.3 JQUERY

jQuery é uma leve biblioteca para desenvolvimento rápido de javascripts que interage com páginas web, e com ela podemos atribuir eventos, definir efeitos, alterar ou criar elementos na página, dentre diversas outras ações. A sua principal função é facilitar a manipulação de DOM (Document Object Model) via javascript, sendo que a partir dessas definições foram desenvolvidos vários plugins com o

---

<sup>5</sup> **Eclipse**: Disponível em: <<http://www.eclipse.org.>>. Acesso em: 30 out. 2008

intuito de auxiliar as mais diversas interações, tais como validações de formulários, apresentações gráficas, conexões AJAX entre outras.

A JQuery faz conexão também com o CSS, podendo assim estruturar toda interface gráfica da aplicação, produzindo efeitos avançados, com uma biblioteca pequena e leve de se carregar, e com efeitos ricos e parecidos com o FLASH, mas sem a necessidade de o navegador possuir plugins.

A JQuery é uma extensão do javascript, e sendo assim, todos seus processos e de seus plugins são realizados diretamente no lado do cliente. Os mais modernos navegadores já possuem compatibilidades com a JQuery, ressaltando que o SIGART foi testado usando os navegadores Mozilla Firefox 3.0 e o Internet Explorer 7.0, e em ambos não apresentou nenhuma incompatibilidade com a biblioteca e seus plugins.

Abaixo são apresentados todos os plugins utilizados para o desenvolvimento do sistema gerenciador de artigos e onde cada um pode ser encontrado:

Biblioteca:

- Biblioteca JQuery - Disponível em:  
[http://docs.jquery.com/Downloading\\_jQuery](http://docs.jquery.com/Downloading_jQuery)

Plugins:

- Animatedcollapse - Disponível em:  
<http://www.dynamicdrive.com/dynamicindex17/animatedcollapse.htm>
- Autotab - Disponível em:  
<http://dev.lousyllama.com/autotab/>
- Bounce - Disponível em:  
<http://plugins.jquery.com/project/Bounce>
- Datepicker - Disponível em:  
<http://www.kelvinluck.com/assets/jquery/datePicker/>
- FCKEditor - Disponível em:  
<http://www.fyneworks.com/jquery/FCKEditor/>
- Interface - Disponível em:  
<http://interface.eyecon.ro/>
- Jcarousel - Disponível em:  
<http://sorgalla.com/projects/jcarousel/>

- JqModal - Disponível em:  
<http://www.webresourcesdepot.com/lightweight-window-alert-plugin-for-jquery-jqmodal/>
- Jqtooltip - Disponível em:  
<http://bassistance.de/jquery-plugins/jquery-plugin-tooltip/>
- Jquery-validate - Disponível em:  
<http://bassistance.de/jquery-plugins/jquery-plugin-validation/>
- Jquery.rater - Disponível em:  
<http://dev.jquery.com/~john/plugins/rating/>
- Maskedinput - Disponível em:  
<http://digitalbush.com/projects/masked-input-plugin/>
- Nicejforms - Disponível em:  
<http://www.jqueryplugins.com/plugins/view/7/>
- Quicksearch - Disponível em:  
<http://rikrikrik.com/jquery/quicksearch/>
- Simplemodal - Disponível em:  
<http://www.ericmmartin.com/projects/simplemodal/>
- Tablesorter - Disponível em:  
<http://tablesorter.com/docs/#Download>
- Tabs - Disponível em:  
<http://stilbuero.de/jquery/tabs/>
- Thickbox - Disponível em:  
<http://jquery.com/demo/thickbox/>

## 5.4 AJAX

O termo Ajax apareceu em 2005 e significa Asynchronous JavaScript and XML, e não é uma tecnologia e sim o uso de tecnologias incorporadas que tem as principais o JavaScript e o XML, onde juntos são capazes de tornar o navegador mais interativo, utilizando-se de solicitações assíncronas de informações. O Ajax compõe de:

- HTML/XHTML E CSS: juntos compõe a apresentação visual da página Web;
- DOM: exibe e interage com o usuário;
- XML e XSLT: fazem o intercâmbio e manipulam dados;
- XMLHttpRequest: recupera dados de forma assíncrona;
- JavaScript: a linguagem de scripts do lado do cliente que une essas tecnologias;

No modelo clássico de desenvolvimento para Web, as informações são repassadas ao servidor através de links ou formulários e o servidor se encarrega de devolver o conteúdo solicitado. Na maioria dos sites atuais, o conteúdo retornado está armazenado em bancos de dados, e o servidor tem o papel de trazer para o layout o conteúdo requisitado.

A idéia do Ajax é tornar esse processo mais simples, ou seja, se o layout já está carregado, porque não carregar apenas o conteúdo que é o objeto da solicitação, sem a necessidade de realizar o carregamento do layout inteiro novamente (Refresh).

O Ajax contém um mecanismo que é um conjunto de funções escritas em JavaScript que são chamadas sempre que uma informação se acha necessária ou enviada ao servidor. A chamada ao servidor fica agora ligada a esse mecanismo conhecido como Ajax, onde ele se encarrega de transmitir e receber informações ao servidor.

Quando o mecanismo do Ajax recebe a resposta do servidor, entra em ação a análise dos dados, que sofrem várias mudanças para trazer a informação de forma agradável ao usuário.

Em uma aplicação Ajax, parte de lógica da aplicação é movida para o navegador, através dos scripts em JavaScript e enquanto o usuário não fechar o seu navegador, ele permanecerá com essa parte lógica por toda a sessão.

No Sistema Gerenciador de Artigos, como mencionado anteriormente, foi usado o framework DWR que auxiliou todo o processo de conexão Ajax na aplicação desenvolvida.

## 5.5 TOMCAT

Para que o Java funcione em aplicações Web, é necessário um Container Servlet. Um Container Servlet pode ser um servidor, ou a integração de um, trabalhando exclusivamente para processar páginas escritas em Java.

O Tomcat<sup>6</sup> iniciou juntamente com a origem da tecnologia servlet, e tecnicamente é um Container Web com a capacidade de atuar também como servidor Web/http assim como pode também funcionar integrado a um servidor web dedicado como o Apache, mas não implementa um Container EJB (Enterprise Java Beans).

Desenvolvido pela Apache Software Foundation, o Tomcat é distribuído como software livre dentro do conceituado projeto Apache Jakarta, sendo oficialmente adotado pela Sun Microsystems como a implementação de referência para as tecnologias Java Servlet e JavaServer Pages (JSP).

## 5.6 MYSQL

O MySQL<sup>7</sup> é um sistema de gerenciamento de banco de dados (SGBD) com licença base na GPL, que utiliza a linguagem SQL e é um dos bancos de dados mais populares.

O Mysql tem como características principais: ser portátil a praticamente qualquer plataforma atual, compatibilidade com diversas linguagens de programação através de drivers e módulos de interface, bom desempenho e estabilidade, pouco exigente quanto a recursos de hardware, facilidade de uso, entre outras.

Uma das principais vantagens em se utilizar o SGBD é seu código aberto e também sua capacidade de rodar nos mais diversos sistemas operacionais disponíveis hoje no mercado.

No SIGART, foi usada a versão 5.0 do MySQL, juntamente com a ferramenta MySQL GUI Tools<sup>8</sup> que é um pacote de ferramentas para administração e manutenção do banco de dados. Na aplicação o banco se mostrou totalmente

---

<sup>6</sup> **Tomcat**: Disponível em: <<http://tomcat.apache.org/>>. Acesso em: 05 nov. 2008

<sup>7</sup> **MySQL**: Disponível em: <<http://www.mysql.com>> . Acesso em: 01 nov. 2008.

<sup>8</sup> **MySQL GUI Tools**: Disponível em: <<http://dev.mysql.com/downloads/gui-tools/5.0.html>>. Acesso em: 01 nov. 2008

integrável ao contexto exposto, rápido e com desempenho satisfatório, mesmo quando se exigiam diversas conexões e consultas, como nos casos de buscas de artigos por grau de relevância.

## **6. SIGART (SISTEMA GERENCIADOR DE ARTIGOS)**

### **6.1 OBJETIVOS E JUSTIFICATIVA DO PROJETO**

O sistema de gerenciamento de artigos visa proporcionar o desenvolvimento de um software eficiente para a melhor execução de controle e administração de artigos. Um artigo é o conjunto de um tema e o seu desenvolvimento, classificado em uma determinada categoria.

O SIGART tem por objetivo atender a todos interessados em participar e trocar idéias e informações em um Ambiente Virtual de Aprendizagem, auxiliando desde o cadastro do usuário, até a publicação e edição dos artigos.

Apesar de existirem diversos sistemas que controlem a editoração de artigos, o presente projeto tem por diferencial utilizar e explorar a linguagem de programação Java, totalmente orientada a objetos, sendo desenvolvido com uma interface amigável e com bibliotecas e plugins que ajudam a desenvolver a relação IHC. Em paralelo ao desenvolvimento do projeto foram usadas as melhores técnicas e ferramentas de engenharia de software com base no RUP (Rational Unified Process), visando um ótimo acompanhamento e auxílio na estrutura arquitetural do projeto.

Assim, o objetivo geral da presente pesquisa é verificar a adequação dos temas organizacionais apresentados (tecnologia, gerência de projetos e desenvolvimento), no âmbito dos ambientes virtuais de aprendizados e concretizar o desenvolvimento de um sistema ágil com objetivos específicos da área educacional.

### **6.2 DEFINIÇÃO DO ESCOPO**

O sistema possui várias regras funcionas e características que integram o escopo do projeto. Abaixo são relacionados os principais propósitos e regras do SIGART, exemplificados por cinco etapas elaboradas com a finalidade de se obter uma melhor compreensão do funcionamento do sistema:

Artigos:

- Os artigos a serem escritos devem seguir regras básicas como, por exemplo, ter um título bem fundamentado e um desenvolvimento bem elaborado, e o autor do artigo deve verificar e definir no momento do envio do artigo a qual categoria que o mesmo pertence.
- Todos podem escrever um artigo (Leitores, Avaliadores e Administradores) desde que não estejam inabilitados. Lembrando ainda que um artigo escrito por um avaliador não pode ser encaminhado para a categoria a qual ele mesmo avalia. Cada usuário pode escrever quantos artigos achar necessário
- Os avaliadores são responsáveis pela análise de artigos. Os avaliadores são escolhidos pelos administradores e podem avaliar apenas artigos de categorias das quais foram habilitados.

#### Categorias:

- Cada artigo deve pertencer a uma e apenas uma categoria. Por exemplo, um artigo que retrate sobre tópicos de programação com Java Server Faces, deve fazer parte da categoria JAVA. Um artigo não pode ser salvo e enviado para avaliação se não tiver sido escolhida uma categoria por parte do usuário.
- Cada artigo possui um fórum de discussões onde os usuários poderão trocar informações a respeito da matéria publicada e também sanar dúvidas a respeito do conteúdo do artigo.

#### Usuário leitor:

- Os interessados em participar no sistema de artigos, devem previamente realizar um cadastro para poderem navegar no site. Esse cadastro é aberto a todas as pessoas e é através dele que o usuário informa previamente seus dados pessoais, que serão usados durante todas as suas ações no sistema. Após informar seus dados pessoais, o sistema enviará para o e-mail do usuário uma confirmação indicando que ele realizou um cadastro no site. Apenas após o usuário realizar a efetuação de confirmação através do e-mail é que o sistema ficará disponível para utilização.
- Todos os usuários podem escrever artigos nos mais diversos temas abordados no site. Para cadastrar um artigo, o usuário deve

primeiramente indicar a qual categoria que o mesmo pertence, e após escolher o tema e o desenvolvimento.

- Após escrever um novo artigo, o usuário o encaminha para avaliação, e o mesmo será enviado para um avaliador realizar a análise e dar o seu parecer sobre o artigo.
- Um artigo avaliado e finalizado com a opção de “Retornar Para Revisão” retorna ao usuário para eventuais alterações. O autor pode fazer as mudanças necessárias no artigo e reencaminhá-lo para uma nova avaliação.
- Todos os usuários podem ao encontrar um artigo de seu interesse, incluí-los na opção “Meus Favoritos”, assim, toda vez que o usuário realizar o login no sistema, na página principal, os artigos escolhidos como meus favoritos serão apresentados ao usuário, sem a necessidade de o mesmo ter de procurá-los no sistema.
- Cada usuário pode dar uma nota variando de 1 - 10 por artigo.

Usuário avaliador:

Além das regras de leitor, o Avaliador possui as seguintes funcionalidades:

- Se uma categoria tiver mais de um avaliador, os artigos que forem encaminhados para avaliação deverão ser distribuídos igualmente a cada avaliador, evitando que um fique com muitos artigos para avaliação, enquanto outro sem nenhum.
- O avaliador visualiza o artigo para avaliação e após análise, escolhe uma nota para o artigo e depois seleciona uma das opções de avaliação “Publicar Artigo / Retornar Para Revisão / Cancelar Artigo”. Os artigos que forem escolhidos como “Publicar Artigo” vão para publicação imediata, os escolhidos como “Retornar para Revisão” devem ser acrescidos de um comentário do avaliador e após, voltam para o autor do artigo realizar as alterações. Os artigos selecionados como “Cancelar Artigo” são cancelados momentaneamente e não podem mais ser alterados.

Usuário administrador:

- O administrador do site é quem cria as categorias e escolhe quem serão os seus avaliadores. Cada categoria deve ter no mínimo um e no

máximo três avaliadores. Assim ao se criarem categorias, o administrador deve informar quem são os avaliadores respectivos a aquela categoria. Cada avaliador pode realizar a avaliação também de apenas uma categoria.

- O administrador também tem o direito de banir ou ativar usuários. Ele pode realizar o banimento de usuários que desrespeitarem as regras e normas do fórum ou ativar usuários que estejam com problemas para ativação de sua conta por e-mail.

## 6.3 ESTRUTURAÇÃO DO SISTEMA E PRINCIPAIS FUNÇÕES

### 6.3.1 Sistema de cadastro com JQuery e Ajax

Um ponto diferencial no desenvolvimento do sistema é a sua função de cadastro de usuários, que engloba simultaneamente o uso da tecnologia Ajax e JQuery.

Com o uso da JQuery e seus plugins, podemos estabelecer facilmente uma validação dos dados antes de serem encaminhados para registro, juntamente com uma interface rica e bem apresentável ao usuário com a utilização dos recursos CSS. Primeiramente criamos um arquivo JavaScript (JS) e o referenciamos em nossa página JSP junto com a biblioteca jquery.js e o plugin jquery-validate. Dentro no arquivo podemos indicar através das funções “\$(document).ready( function () {})” as validações e as mensagens de saída e erro que deverão ser apresentadas.

Por exemplo, caso desejarmos colocar o campo nome do usuário com preenchimento obrigatório e que o mesmo tenha no mínimo 8 caracteres de tamanho, podemos definir estas regras com o seguinte código:

```

rules : {
    nomeUsuario : {
        required :true,
        minlength :8
    },
}

messages : {

```

```

nomeUsuario : {
    required : "Entre com o nome do usuário",
    minlength : jQuery
        .format("Entre com um nome de no mínimo {0}
        caracteres")
    },
}

```

No código acima, definimos primeiramente as regras do campo a serem implementadas, como por exemplo, “required: true” indica que o campo é obrigatório, e a regra “minlength: 8” define que o nome de usuário não pode ter menos de 8 caracteres. Após as validações devemos estabelecer as mensagens de erro caso alguma regra não seja validada, como é demonstrado na segunda parte do código.

Figura 01: Tela de cadastro de novo usuário

The screenshot shows a web page titled "World Programming" with a header containing logos for Java, PHP, C/C++, and Delphi. The main content is a registration form titled "Formulário de Cadastro de Novo Usuário". The form contains the following fields and elements:

- Nome Completo:
- Data de Nascimento:
- Sexo:
- UF:
- Cidade:
- Email:
- Login:
- Senha:
- Confirme a Senha:
- [Clique aqui para ler os termos de uso do sistema.](#)
- Eu declaro que li e aceito os termos de uso:
- Navigation buttons: << Voltar, Resetar, Cadastrar >>

Na Figura01 é mostrada a tela principal de cadastro de um novo usuário. E quando se é pressionado o botão “Cadastrar >>”, o sistema chama as funções do plugin que validam o form, indicando se cada campo foi preenchido corretamente ou não. O resultado é apresentado na Figura02 abaixo:

Figura 02: Tela de cadastro de novo usuário validada pela JQuery

**Formulário de Cadastro de Novo Usuário**

---

Nome Completo:  ✖ **Entre com um nome de no mínimo 8 caracteres**

Data de Nascimento:   ✖ **Entre com uma data de Nascimento Válida**

Sexo:  ✖ **Escolha o Sexo**

UF:  ✔

Cidade:  ✔

Email:  ✖ **Por favor, entre com um endereço de email válido**

Login:  ✔

Senha:  ✔

Confirme a Senha:  ✖ **Entre com uma senha de no mínimo 5 caracteres**

[Clique aqui para ler os termos de uso do sistema.](#)

Eu declaro que li e aceito os termos de uso:  ✖ **Para se cadastrar, você deve declarar que leu e aceita os termos do site**

Após pressionar o botão cadastrar, verificamos toda a validação do formulário, com indicadores visuais ao lado de cada campo, com o ícone ✔ caso a validação tenha retornado sem erros, ou com o ícone ✖ e uma mensagem caso a validação retorne com inconsistências. Lembrando ainda que a questão de cores, imagens, ícones e fontes podem ser modificadas com o uso do CSS, assim cada um pode deixar sua formatação do modo que desejar.

Dentro deste mesmo formulário ainda existe outras aplicações de plugins da JQuery, como no campo data de nascimento e no item termos de uso do sistema, que abrem ambos uma tela com informações referentes a suas funções. Nesse contexto podemos perceber que podemos utilizar simultaneamente diversos plugins da JQuery sem se preocupar com incompatibilidades.

Outra tecnologia presente no cadastro é o AJAX, que dentro do formulário, busca todas as cidades quando se é selecionado uma UF. No software foi utilizado o framework DWR, que através de um código simples e eficaz faz todo o processo de comunicação assíncrona, retornando uma resposta rápida e precisa.

Abaixo temos a Figura03 que mostra os campos UF e Cidade sem preenchimento. Podemos notar que enquanto não é selecionada nenhuma UF, o campo cidade fica vazio e esperando uma ação do usuário.

Figura 03: Tela de escolha de UF e cidade

UF: --SELECIONE A UF --

Cidade: --SELECIONE PRIMEIRO A UF --

Email: patriquesz

Login:

Senha:

Confirme a Senha:

No momento em que o usuário escolhe um Estado, o DWR através de seu código JavaScript, faz uma conexão assíncrona e passa os dados da UF escolhida disparando uma solicitação HTTP para o servidor web. O servidor processa as informações recuperando todas as cidades que pertencem a UF informada, e para finalizar, o servidor retorna os dados dos municípios preenchendo a combobox cidade na página JSP.

Figura 04: Tela de escolha de UF e cidade – Busca por Cidades

UF: ACRE

Cidade:

Email: patriquesz

Carregando Cidades ... Aguarde por favor

No término da busca, teremos como resposta todos os municípios pertencentes à UF escolhida através de uma conexão assíncrona, sem necessidade de ser realizado um refresh na página, agilizando assim o tempo de procura e resposta da página.

Figura 05: Tela de escolha de UF e Cidade – Apresentação dos Municípios

UF: ACRE

Cidade: ACRELANDIA

Email: ASSIS BRASIL

Login: BRASILEIA

Senha: BUJARI

Senha: CAPIXABA

Senha: CRUZEIRO DO SUL

Senha: EPITACIOLANDIA

Senha: FEIJO

Senha: JORDAO

Senha: MANCIO LIMA

Senha: MANOEL URBANO

Senha: MARECHAL THAUMATURGO

Senha: PLACIDO DE CASTRO

Senha: PORTO ACRE

Senha: PORTO WALTER

Senha: RIO BRANCO

Senha: RODRIGUES ALVES

Senha: SANTA ROSA

Senha: SENA MADUREIRA

Senha: SENADOR GUIOMARD

Dentro do SIGART temos ainda outras funções AJAX e JQuery implementadas que podem ser melhor analisadas através do código fonte do programa, presente no CD que acompanha a monografia.

### 6.3.2 Escrevendo um novo artigo

Todos os usuários podem realizar a principal função do sistema, que é a escrita de um artigo. Algumas regras já definidas previamente mostram que cada pessoa pode escrever quantos artigos quiser e encaminhá-los para avaliação, mas é somente um avaliador cadastrado e autorizado quem definirá se o artigo está apto ou não para ser publicado no site.

O princípio básico de escrita de um artigo consiste em o usuário escolher um título, uma categoria e fundamentar o corpo do artigo. Neste ponto foi usado um editor textual em HTML chamado FCKeditor<sup>9</sup> que ajuda de uma maneira fácil e ágil, a formatação do corpo do artigo a ser desenvolvido.

---

<sup>9</sup> **FCKeditor**: Editor de texto HTML Web com grande eficiência de formatação. Disponível em: <<http://www.fckeditor.net/>>. Acesso em: 25 nov. 2008.

Figura 06: Tela de escrita de um novo artigo

**Escrever Artigo**

Preencha todos os campos corretamente e verifique todas as informações de seu artigo, pois após o envio, até a correção do Avaliador, não será possível realizar alterações.

Título do Artigo:

Categoria:

Descrição do Artigo:

Com o editor, o usuário tem a simulação de escrita como se estivesse usando um editor de texto desktop comum, como o NotePad ou até o Microsoft Word. Nele se é possível formatar o artigo através dos botões disponíveis na parte superior da página, colocando, por exemplo, dados em negrito, inserindo imagens e links, alterando o tamanho e formato da fonte, entre outras opções.

O editor transforma toda a formatação manual do usuário em um código HTML que é enviado para o banco de dados no momento do cadastro do artigo. Quando o artigo for requisitado para visualização, basta retirar do banco o código e apresentá-lo novamente em uma página HTML que não se perderá a formatação salva.

### 6.3.3 Definindo o avaliador

Após um artigo ter sido escrito e salvo, entra em ação uma função especial do sistema de gerenciamento de artigos. Cada categoria pode ter um, dois ou três avaliadores cadastrados, e com essa definição, era preciso encontrar uma solução para que no momento em que um artigo fosse salvo, se definisse quem seria o seu avaliador, de modo que a distribuição deveria ser imparcial a todos os avaliadores da categoria. No momento em que o artigo é salvo, o sistema verifica qual o avaliador que se encontra com menor numero de artigos a serem avaliados. A função consiste em uma consulta a Fila de Artigos em Avaliação, que retorna qual o avaliador que possui a menor quantidade de artigos na fila de espera.

Abaixo segue o código em Java do método que define a função de busca:

```
// Metodo que retorna o codigo do avaliador com menor numero de
avaliacoes por categoria, para insercao na fila de avaliacao do cadastro de
novo artigo
public int
getCodigoAvaliadorComMenorQtidadeAvaliacoesByCategoria(int codigoCategoria)
throws DaoException,
        SQLException {
    Connection conn = null;
    PreparedStatement pst = null;
    ResultSet rs = null;
    int codigoAvaliador = 0;
    try {
        conn = this.conn;

        //primeiro verifica se há algum avaliador sem
artigo em avaliação
        pst = (PreparedStatement)
conn.prepareStatement("SELECT avaliadorcategoria.codigoAvaliador From
avaliadorcategoria Where avaliadorcategoria.codigoCategoriaAvaliador = ?
and avaliadorcategoria.codigoAvaliador NOT IN (SELECT codigoAvaliador from
filaavaliacao Where filaavaliacao.statusAvaliacao = 'EM AVALIACAO') ");
        pst.setInt(1, codigoCategoria);
        rs = pst.executeQuery();
        if (rs.next()) {
            codigoAvaliador =
rs.getInt("codigoAvaliador");
        }

        //se nao houver, procura o avaliador com menor
numero de artigos para avaliacao
        if(codigoAvaliador == 0){
            pst = null;
            rs = null;
            pst = (PreparedStatement)
conn.prepareStatement("SELECT filaavaliacao.codigoAvaliador,
Count(filaavaliacao.codigoArtigo) AS ContarDecodigoArtigo,
artigo.codigoCategoriaArtigo " +
                        "FROM artigo INNER JOIN
filaavaliacao ON artigo.codigoArtigo = filaavaliacao.codigoArtigo "+
```

```

                                "WHERE
(((filaavaliacao.statusAvaliacao)='EM AVALIACAO')) and
artigo.codigoCategoriaArtigo = ? " +
                                "GROUP BY
filaavaliacao.codigoAvaliador "+
                                "ORDER BY
Count (filaavaliacao.codigoArtigo)");
                                pst.setInt(1, codigoCategoria);
                                rs = pst.executeQuery();
                                if (rs.next()) {
                                    codigoAvaliador =
rs.getInt("codigoAvaliador");
                                }
                                }

                                //se nao houver avaliador com artigo para ser
avaliado na categoria, retorna o primeiro avaliador que encontrar da
categoria
                                if(codigoAvaliador == 0){
                                    pst = null;
                                    rs = null;
                                    pst = (PreparedStatement)
conn.prepareStatement("SELECT avaliadorcategoria.codigoAvaliador From
avaliadorcategoria Where avaliadorcategoria.codigoCategoriaAvaliador = ?
");
                                    pst.setInt(1, codigoCategoria);
                                    rs = pst.executeQuery();
                                    if (rs.next()) {
                                        codigoAvaliador =
rs.getInt("codigoAvaliador");
                                    }
                                }
                                return codigoAvaliador;
                            }
    }
}

```

Assim, teremos um balanceamento entre o numero de artigos por avaliador, não deixando um avaliador sobrecarregado enquanto outro fica sem nenhum artigo a analisar.

### 6.3.4 Avaliando um artigo

O processo de avaliação consiste em uma análise minuciosa de um artigo e, após análise, é definida uma nota, uma vista ao artigo e um status que definirá se o mesmo irá para correção, será cancelado ou publicado.

Figura 07: Tela de avaliação de artigos



O ponto chave na avaliação é que quando um artigo vai para publicação, o sistema através da biblioteca IText<sup>10</sup> converte todo o código HTML em PDF, deixando assim duas possibilidades de visualização ao usuário. A primeira é no próprio formato HTML padrão, e a segunda no formato PDF, sendo que através deste, o usuário poderá salvar o artigo, imprimi-lo ou até enviá-lo por email com os recursos disponibilizados, sem perder a formatação e dados originais.

### 6.3.5 Sistema de busca inteligente

Dentro do SIGART, foi inserida uma função especial na parte de localização de artigos. Foram abordadas técnicas de recuperação inteligente de informações, para que as buscas não se tornassem padronizadas e específicas apenas ao uso da linguagem SQL. Para isso foram utilizadas duas técnicas de recuperação: O método vetorial e o método TFxIDF.

#### 6.3.5.1 Método vetorial

<sup>10</sup> IText: Biblioteca livre Java-PDF. Disponível em: <<http://www.lowagie.com/iText/>>. Acesso em: 29 abr. 2009

Para a representação de textos, utiliza-se freqüentemente a representação vetorial (Keyword), proposta inicialmente por Salton (1988), no qual os documentos são considerados como vetores multi-dimensionais, onde cada dimensão do vetor representa um radical, e seu valor é a freqüência de ocorrência de um termo no documento.

O método vetorial é baseado no fato de que se um texto contém uma maior quantidade dos termos que estão sendo pesquisados, a tendência é que seja mais relevante ao usuário e para que isto ocorra, os documentos são armazenados com a escolha e determinação da quantidade das Keywords.

O grau de similaridade entre a consulta e um documento é calculado pelo produto vetorial entre os vetores da consulta e o vetor de quantidade de Keywords de cada documento.

Figura 08: Função do cálculo vetorial

$$Sim(Q, D_i) = \frac{\vec{Q} \cdot \vec{D}_i}{|\vec{Q}| |\vec{D}_i|}$$

Sendo:

- $Q = (q_1, q_2, \dots, q_n)$  o vetor formado pelos pesos da consulta
- $D_i = (d_1, d_2, \dots, d_n)$  os vetores dos documentos onde  $d_n$  são as quantidades de keywords do documento  $D_i$
- $Sim(Q, D_i)$  o grau de relevância da recuperação da informação

Se prestarmos bem atenção à fórmula, podemos perceber que se caso apenas uma palavra for encontrada em um artigo, o vetor virará unidimensional e teremos sempre o mesmo resultado caso outro artigo contenha também apenas uma palavra encontrada. Para conseguir passar por esta falha, foi estabelecida uma regra básica dentro do sistema, que quando a recuperação de informações retornar que apenas uma única palavra foi encontrada, o cálculo da relevância do artigo será a quantidade de vezes que a palavra apareceu dentro do artigo.

#### 6.3.5.2 Método TFXIDF

O modelo de recuperação de informações TFxIDF é um método cuja maneira de consulta não difere do método anterior, somente os pesos dos vetores são substituídos pela medida do TFxIDF da keyword. Para saber quais termos do documento são mais relevantes ou menos relevantes, Salton (1988) propôs várias técnicas para calcular seus pesos. No modelo vetorial os pesos mais utilizados para relacionar um termo  $i$  em um documento  $d$  são o TF ( $i,d$ ) e o TF-IDF ( $i,d$ ).

O TF (Term-Frequency) mede a proporção da quantidade de um determinado termo em relação ao termo de maior quantidade em um documento. É uma métrica dada pela seguinte fórmula:

Figura 09: Função do cálculo do Term-Frequency

$$tf = \frac{QtdeKeyword}{QtdeMaxKeyword}$$

Onde:

- QtdeKeyword é a quantidade de vezes que uma determinada Keyword aparece no Documento.
- QtdeMaxKeyword é a quantidade da Keyword que mais aparece no Documento.

O IDF (Inverse Document Frequency) mede a proporção de termo em relação ao aparecimento deste termo no total de documentos da coleção. É uma métrica dada pela seguinte fórmula:

Figura 10: Função do cálculo do Inverse Document Frequency

$$idf = \log \frac{N}{ni}$$

Onde:

- $N$  é a quantidade total de Documentos cadastrados
- $ni$  é a quantidade de Documentos em que aparece a Keyword.

Segundo Larocca (2002), o IDF de uma palavra é baixo se esta ocorre em muitos documentos e alto se a palavra ocorre somente em um documento.

Neste método tanto os vetores com a consulta, como documentos, terão seus pesos calculados pela fórmula do TFxIDF:

Figura 11: Função do cálculo do TFxIDF

$$tfidf = tfxidf$$

Assim, segundo Ferreira (2004), um termo que ocorre freqüentemente em um documento é considerado importante (TF alto), e um termo que é muito freqüente na coleção de documentos é considerada pouco importante (IDF baixo).

Ainda temos de calcular o TF do vetor da consulta, que é calculada pela fórmula:

Figura 12: Função do Cálculo do TFq

$$TFq = (0,5 + (0,5 \times \frac{Peso}{PesoMax}))$$

Onde:

- Peso é o peso atribuído para a Keyword.
- PesoMax é o valor do maior peso atribuído a todas às keyword

Com todos os dados em mãos, podemos agora calcular o grau de relevância utilizando o método vetorial KeyWord, passando o TFq como dados dos pesos das consultas, e o TFxIDF como vetores da quantidade de vezes que cada palavra foi encontrada. Assim teremos como resultado um número entre 0 e 1 que será o grau de relevância de nossa recuperação de informações.

### 6.3.5.3 KeyWord e TFXIDF na prática

Como já mencionado anteriormente, os dois métodos de recuperação de informações, foram implementados no SIGART. Abaixo será apresentada a parte gráfica da busca e recuperação de informações por parte dos usuários comuns do sistema, e após a implementação das funções e classes de cálculos vetoriais.

Na prática temos o seguinte caso de busca:

Quando um usuário quer encontrar um artigo dentro do sistema, ele informa quais as palavras que deseja que o(s) artigo(s) contenha(m) e seleciona o botão Buscar Artigo.

Figura 13: Tela inicial de busca de artigos

**Buscas**

Busca de Artigos		
Buscar Artigo por:	CONTEXTO DO ARTIGO TITULO CONTEXTO DO ARTIGO AUTOR DATA DE PUBLICACAO	Descrição: PHP Web Inicial - Data Final:
Data de Publicação:		
<input type="button" value="Buscar Artigo &gt;&gt;"/>		

Nesse ponto o sistema identifica as palavras e pede para que o usuário escolha um peso entre 0 e 10 para cada palavra informada.

Figura 14: Tela de edição de pesos

#### Edição de Pesos

Para realizar a busca por grau de relevância, por favor indique um peso entre 0 e 10 para a(s) palavra(s) apresentada(s) abaixo :

Palavra:	Peso
PHP	10
Web	6

Com os pesos informados o sistema realiza todos os cálculos vetoriais, e retorna os artigos apresentando-os por ordem decrescente do grau de relevância calculado.

Figura 15: Tela de resultado da busca por relevância

## Artigos por Grau de Relevância

Data Publicacao	Título Artigo	Autor	Categori	Relevância	Visualizaç
24/02/2009	CAMINHOS PARA CERTIFICAÇÃO PHP / ZEND (ZCE)	PATRICIA CRISTIANE DE PAULA ZOTTO	PHP	91,74 %	
24/02/2009	INSTALAÇÃO DO PHP-FUSION + TRADUÇÃO	PATRICIA CRISTIANE DE PAULA ZOTTO	PHP	91,74 %	
11/02/2009	ENVIANDO EMAILS COM JAVA	ELOE ZOTTO	JAVA	58,36 %	
11/02/2009	HELLO, WORLD - SEU PRIMEIRO PROGRAMA EM JAVA!	SERGIO PATRIQUE ZOTTO	JAVA	41,05 %	
11/02/2009	PRAZER, JAVA!	BENEDITO RIBAS DE PAULA	JAVA	40,21 %	
11/02/2009	INTRODUÇÃO AO JDBC	SERGIO PATRIQUE ZOTTO	JAVA	40,19 %	
11/02/2009	INTRODUÇÃO AO JAVA.NIO	ANEZIA DO CARMO NUNES	JAVA	40,13 %	
11/02/2009	OLÁ JAVA MICRO EDITION!	SERGIO PATRIQUE ZOTTO	JAVA	39,99 %	

1/1 10

Foi desenvolvida uma classe Java, chamada TFxIDF que realiza os dois tipos de cálculos vetoriais, tanto o próprio TFxIDF quanto o Keyword. Quando as palavras e pesos são informados, chama-se a classe Java ArtigoDao que faz primeiramente uma consulta rápida SQL em todos os artigos do banco de dados, retornando apenas os artigos que contenham pelo menos uma das palavras pesquisadas.

Quando os artigos encontrados são retornados, a classe ArtigoDao faz um loop passando por cada artigo encontrado, e dentro do loop referencia a classe TFxIDF enviando as palavras a serem consultadas, os pesos e o artigo que está em questão na passagem do loop.

Dentro da classe TFxIDF, temos as seguintes funções de cálculos:

- Método de cálculo do TF:

```
public void setCalcularDnComTF() {
    int maiorOcorrenciaEmDN = 0;
    for(int i = 0; i < getQtdePalavrasBuscadas() ; i++){
        if(Dn[i] > maiorOcorrenciaEmDN){
            maiorOcorrenciaEmDN = Dn[i];
        }
    }

    if(maiorOcorrenciaEmDN == 0){
        maiorOcorrenciaEmDN++;
    }
}
```

```

    }

    for(int i = 0; i < getQtdePalavrasBuscadas() ;
i++){
        this.DnComTF[i] = ((double)
Dn[i]) / ((double)maiorOcorrenciaEmDN;
    }
}

```

- Método de cálculo do IDF:

```

public void setCalcularIDF() throws DaoException{
    double qtdadeAparicoesPalavraEmArtigos = 0;
    double totalArtigosEscritos = 0;
    for(int i = 0; i < getQtdePalavrasBuscadas() ; i++){

        qtdadeAparicoesPalavraEmArtigos =
this.qtdadeAparicoesEmArtigos[i];
        totalArtigosEscritos = getTotalDocumentosGeral();
        if(qtdadeAparicoesPalavraEmArtigos == 0){
            qtdadeAparicoesPalavraEmArtigos++;
            totalArtigosEscritos++;
        }
        this.idfn[i] = Math.log10(((double)totalArtigosEscritos)
/ ( (double) qtdadeAparicoesPalavraEmArtigos));
    }
}

```

- Função de cálculo do TFXIDF:

```

public void setCalcularTFIDF() throws DaoException{
    for(int i = 0; i < getQtdePalavrasBuscadas() ; i++){
        this.tfidfn[i] = (double)this.DnComTF[i] *
(double) this.idfn[i];
    }
}

```

- Função de cálculo do TFXIDFq:

```

public void setCalcularTFIDFConsulta() throws DaoException{
    double maiorPeso = 0;
    for(int i = 0; i < getQtdePalavrasBuscadas() ; i++){
        if(this.QPesosPalavras[i] > maiorPeso){
            maiorPeso = this.QPesosPalavras[i];
        }
    }
    if(maiorPeso == 0){
        maiorPeso++;
    }
    for(int i = 0; i < getQtdePalavrasBuscadas() ; i++){
        this.tfq[i] = (0.5 +
(((double)0.5*this.QPesosPalavras[i] ) / (double)maiorPeso ));
    }
    for(int i = 0; i < getQtdePalavrasBuscadas() ; i++){
        this.qTFIDF[i] = this.tfq[i] * this.idfn[i];
    }
}

```

- Função de cálculo do SIM do artigo (Keyword):

```

    public void setCalculoSim(){
        double vetorBDn = 0, vetorBQ = 0;
        double resultadoCima = 0, resultadoBaixo = 0;
        String calculoVetorialCima = "";
        for(int i = 0; i < getQtdePalavrasBuscadas() ; i++){
            resultadoCima = resultadoCima + (this.tfidfn[i] *
this.qTFIDF[i]);
            vetorBDn = vetorBDn + (double) this.tfidfn[i] *
(double) this.tfidfn[i];
            vetorBQ = vetorBQ + (double) this.qTFIDF[i] *
(double) this.qTFIDF[i];
        }
        resultadoBaixo = Math.sqrt((double)vetorBDn) *
Math.sqrt((double)vetorBQ);
        if(resultadoBaixo == 0){
            resultadoBaixo++;
        }
        String resultadoFinal =
df.format(resultadoCima/resultadoBaixo);
        resultadoFinal = resultadoFinal.replaceAll(",", ".");
        if(getQtdePalavrasBuscadas() == 1){
            setSimArt( (double) tfidfn[0] + ((double) Dn[0] /
10000));
        }else{
            if(verificarQtidadePalavrasVetor()){
                int indice = 0;
                for(int i = 0; i < getQtdePalavrasBuscadas() ;
i++){
                    if(this.Dn[i] != 0){
                        indice = i;
                    }
                }
                setSimArt(Double.parseDouble(resultadoFinal)+
((double) Dn[indice] / 10000));
            }else{
                setSimArt(Double.parseDouble(resultadoFinal));
            }
        }
    }
}

```

No final de todo processo, na classe ArtigoDao, é criado após o loop um ArrayList contendo as informações pertinentes ao artigo pesquisado, como código, título, autor, e também o Sim do artigo que será o indicador do grau de relevância deste artigo para a busca realizada.

Assim, para testar a pesquisa, basta invocar o método buscarArtigosByDescricao KeywordReturnArrayList (String[] palavrasChave, double[] pesosPalavrasChave) da classe ArtigoDao, passando as palavras a serem consultadas e seus pesos, que o sistema irá retornar um ArrayList com os artigos encontrados e seus graus de relevância para serem exibidos.

### 6.3.6 Sistema de Segurança

Na maioria dos sistemas web, é necessário implantar algum tipo de controle de acesso às áreas da aplicação para se aumentar o grau de segurança do sistema desenvolvido. Dentro do SIGART foram usadas as técnicas dos filtros da API Servlet, com o objetivo de mostrar de uma forma simples e rápida como realizar esse controle.

Os filtros foram aplicados em duas etapas distintas:

#### 6.3.6.1 Controle de acessos

No primeiro caso e mais simples, um filtro foi criado e configurado no arquivo web.xml com a finalidade de verificar se todos os usuários que estão usando o sistema, estão devidamente logados. Os filtros da API Servlet são responsáveis por interceptar a chamada a certo recurso, fazer algum processamento, e liberá-la ou não.

Para isso, basta dentro do filtro criado, inserir a seguinte função:

```

public void doFilter(ServletRequest req, ServletResponse res,
                    FilterChain chain) throws IOException,
ServletException {
    HttpSession session =
((HttpServletRequest) req).getSession();
    Usuario usuario =
(Usuario) session.getAttribute("usuario");
    if(usuario==null){
        session.setAttribute("msg", "Você não está
logado no sistema!");
((HttpServletResponse) res).sendRedirect("../index.jsp");
    }else{
        chain.doFilter(req, res);
    }
}

```

Ao fazer uma requisição à /admin/index.jsp, por exemplo, o sistema compara o pedido com o arquivo configurado no filter-mapping em web.xml. Assim é chamado primeiramente o filtro, no método doFilter(), que verifica a existência do atributo “usuário” na sessão. Caso exista um usuário é porque o mesmo encontra-se logado, assim o método doChain() libera a execução do Servlet. Caso o atributo seja

nulo, o usuário é redirecionado para a página de login e o Servlet (JSP) sequer é executado.

#### 6.3.6.2 Restrições de acessos

Dentro do sistema podem existir áreas que apenas certos usuários devam ter permissões, e os filtros também ajudam a facilitar a implementação desse controle.

Exemplificando, dentro do SIGART temos três tipos de usuários: Leitores, Avaliadores e Administradores. Os leitores não podem ter acesso a nenhuma função dos avaliadores, e estes dois não podem ter nenhum acesso ao conteúdo dos administradores.

Para conseguir implementar este problema, foi primeiramente criado um filtro chamado `FiltroSegurancaAcesso` e um arquivo na aplicação chamado `access.properties`.

O arquivo `access.properties` tem a finalidade de que o desenvolvedor do sistema, indique nele linha a linha cada página do site e quem deve ter acesso. Por exemplo, se contarmos que cada leitor tem permissão 1, avaliadores permissão 2, e os administradores permissão 3, podemos colocar no arquivo `access.properties` o seguinte conteúdo para cada página do site:

- `ecreverArtigo.jsp` 1
- `administração.jsp` 3
- `avaliarArtigo.jsp` 2

Desta maneira, todas páginas que tiverem índice 1 podem ser visualizadas por qualquer usuário do sistema. Páginas que contenham índice 2 só podem ser visualizadas por Avaliadores e páginas com índice 3 apenas por administradores.

Assim o `FiltroSegurancaAcesso` implementa uma função de verificação básica parecida com a do controle do acesso, mas desta vez ele verifica no arquivo `access.properties` qual a referência para a página visualizada, e qual o grau de visualização do usuário que está tentando abrir a página. Se a verificação passar o usuário terá acesso aos dados requisitados, senão será informada uma mensagem dizendo que o mesmo não tem permissão para acessar o local indicado.

Por exemplo, se um usuário leitor que tem permissão 1, tentar acessar diretamente de seu navegador a página `administração.jsp`, o filtro imediatamente é

chamado e verifica primeiramente que para esta página é necessária uma permissão de tipo 3 e verifica que o usuário que está tentando acessar a página tem permissão tipo 1. Sendo assim o sistema irá informar ao usuário que ele está tentando acessar uma área do sistema restrita.

### **6.3.7 Feed RSS**

RSS é um subconjunto de dialetos XML que servem para agregar conteúdos, podendo ser acessados mediante programas ou sites que possuem a tecnologia.

A abreviatura do RSS é usada para se referir aos seguintes padrões:

- Rich Site Summary (RSS 0.91)
- RDF Site Summary (RSS 0.9 e 1.0)
- Really Simple Syndication (RSS 2.0)

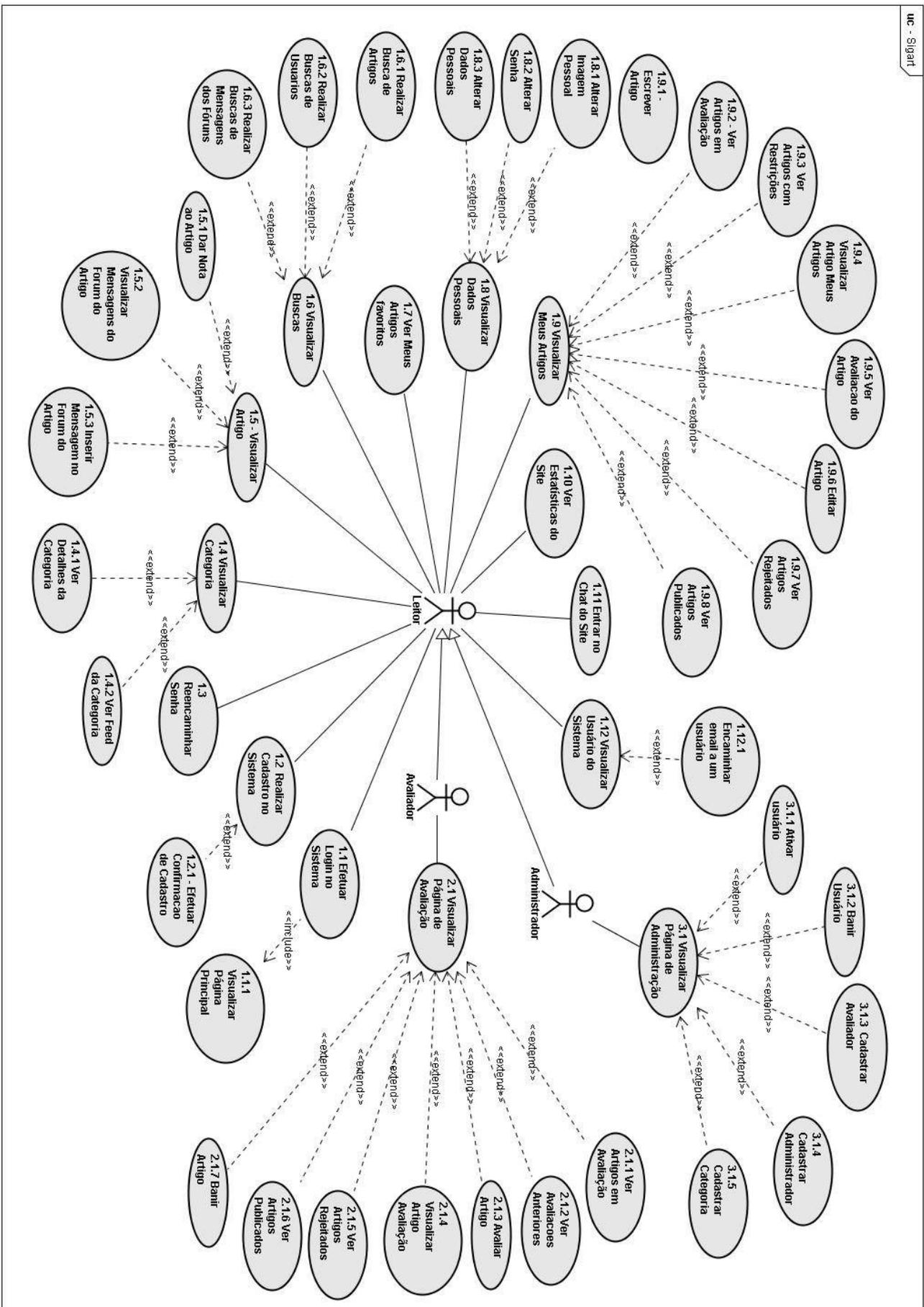
A tecnologia do RSS permite aos usuários da internet se inscreverem em sites que fornecem "Feeds" (alimentadores ou fontes) RSS. Estes são tipicamente sites que mudam ou atualizam o seu conteúdo regularmente. Para isso, são utilizados Feeds RSS que recebem estas atualizações, desta maneira o utilizador pode permanecer informado de diversas atualizações em diversos sites sem precisar visitá-los um a um.

Os Feeds RSS oferecem conteúdo Web ou resumos de conteúdo juntamente com os links para as versões completas deste conteúdo e outros metadados. Esta informação é entregue como um arquivo XML chamado "RSS Feed", "webfeed", "Atom" ou ainda canal RSS.

No SIGART cada categoria de artigos possui um Feed RSS próprio, sendo assim, um usuário pode se inscrever no RSS da categoria a qual desejar e no seu navegador ou programa Feed poderá visualizar instantaneamente um artigo quando o mesmo for publicado, sem a necessidade de ter de ficar entrando no site para verificar se existe alguma novidade.

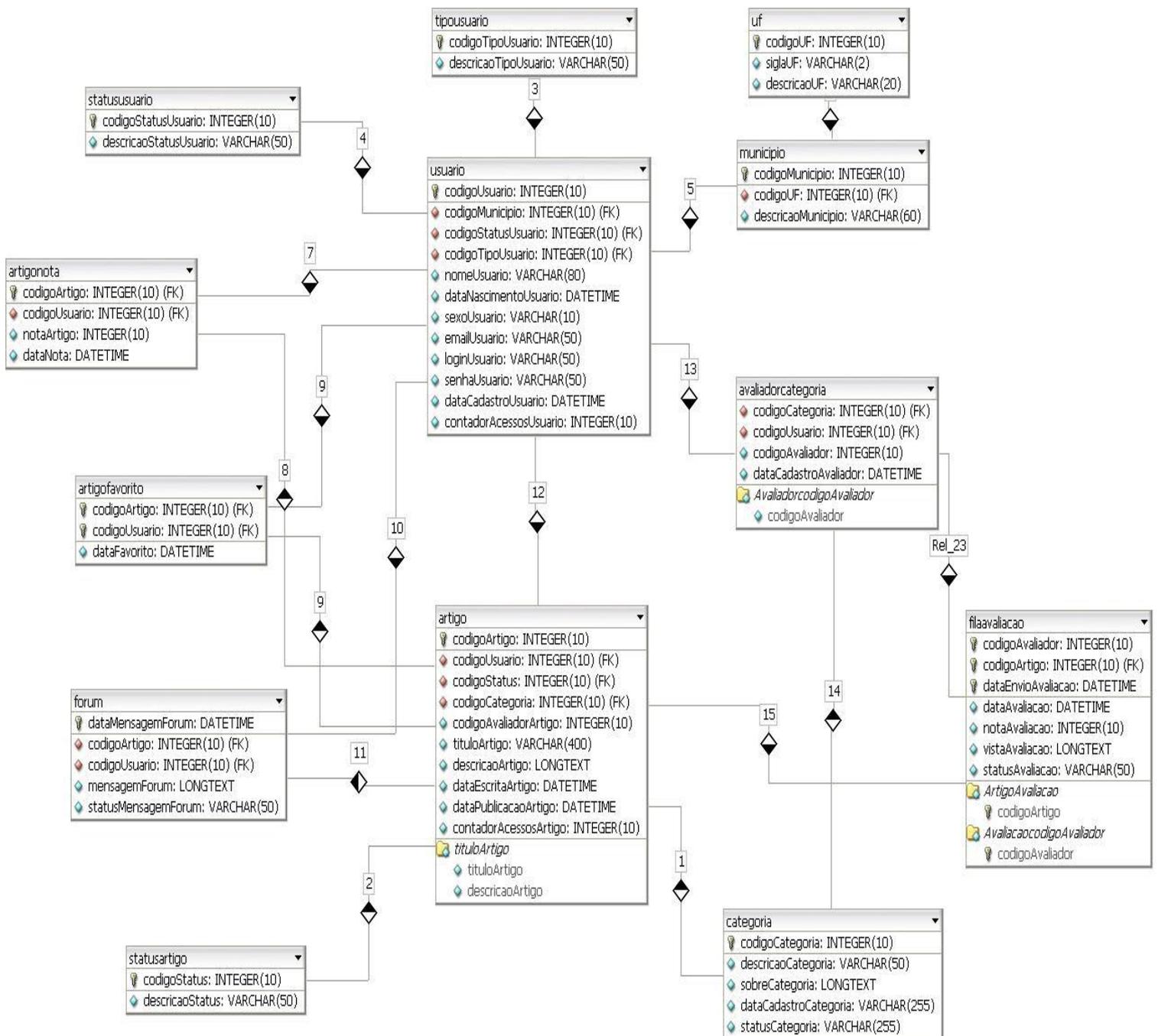
# 7 DIAGRAMAS

## 7.1 DIAGRAMA DE CASOS DE USOS

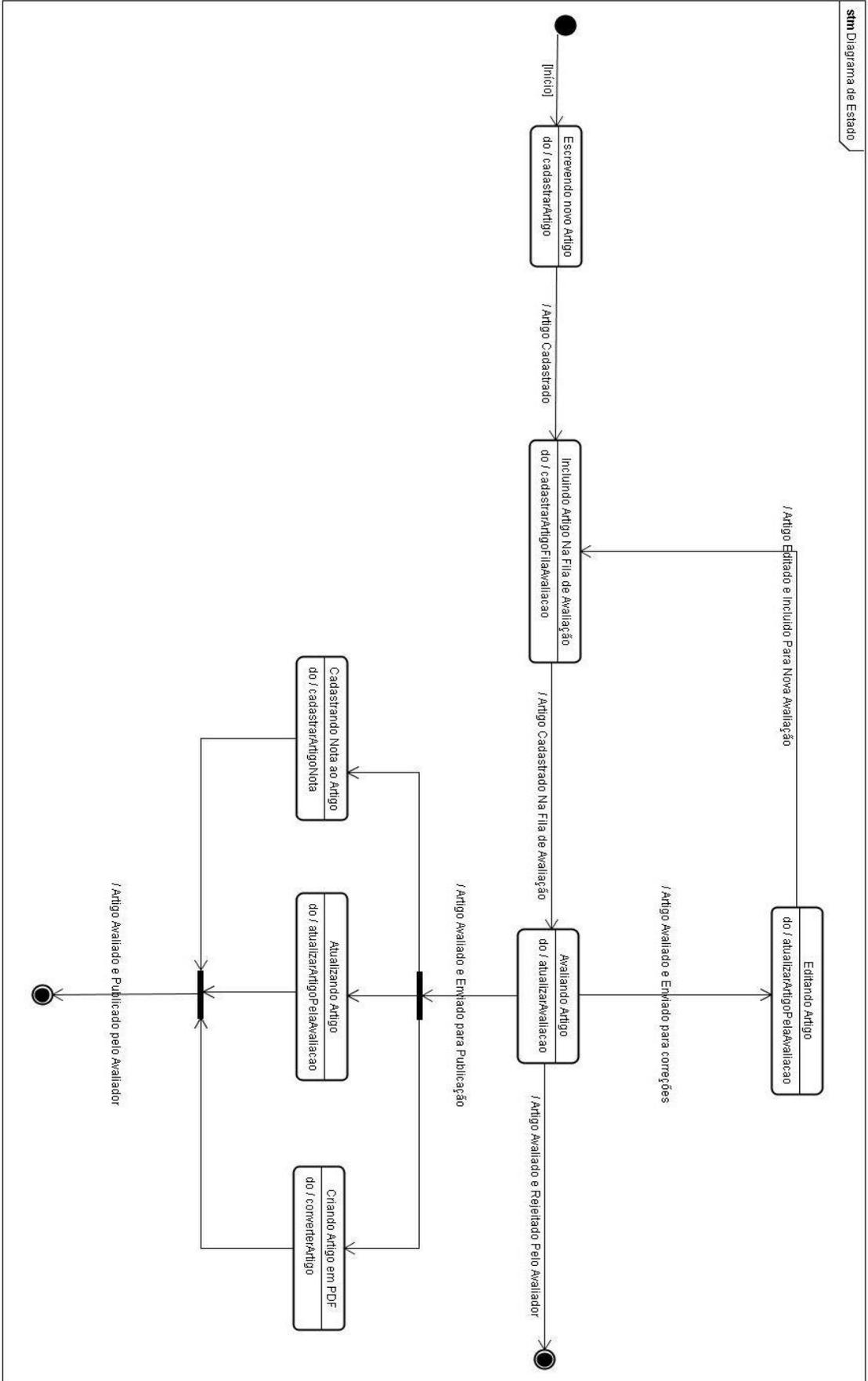




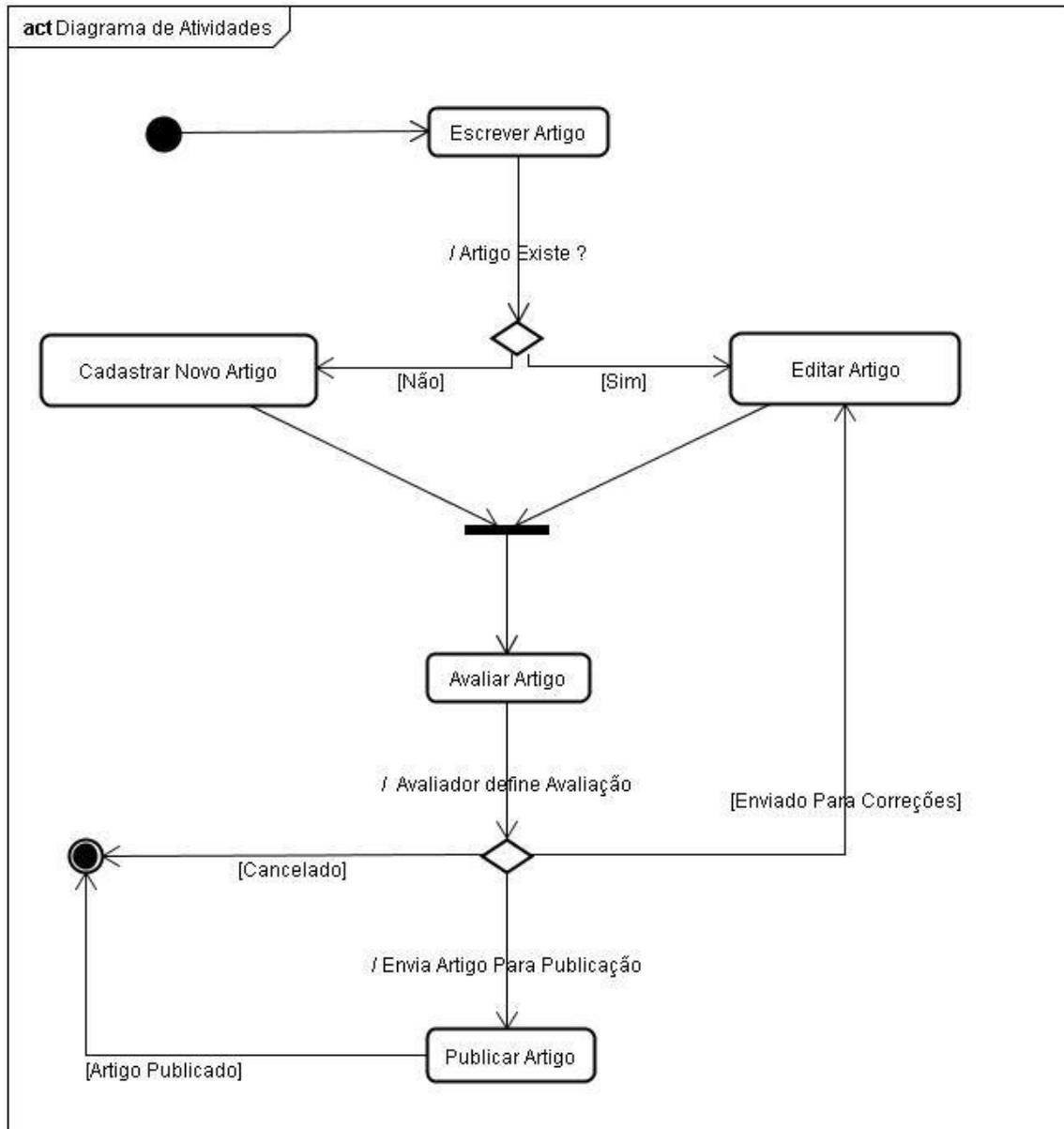
## 7.3 DIAGRAMA DE BANCO DE DADOS



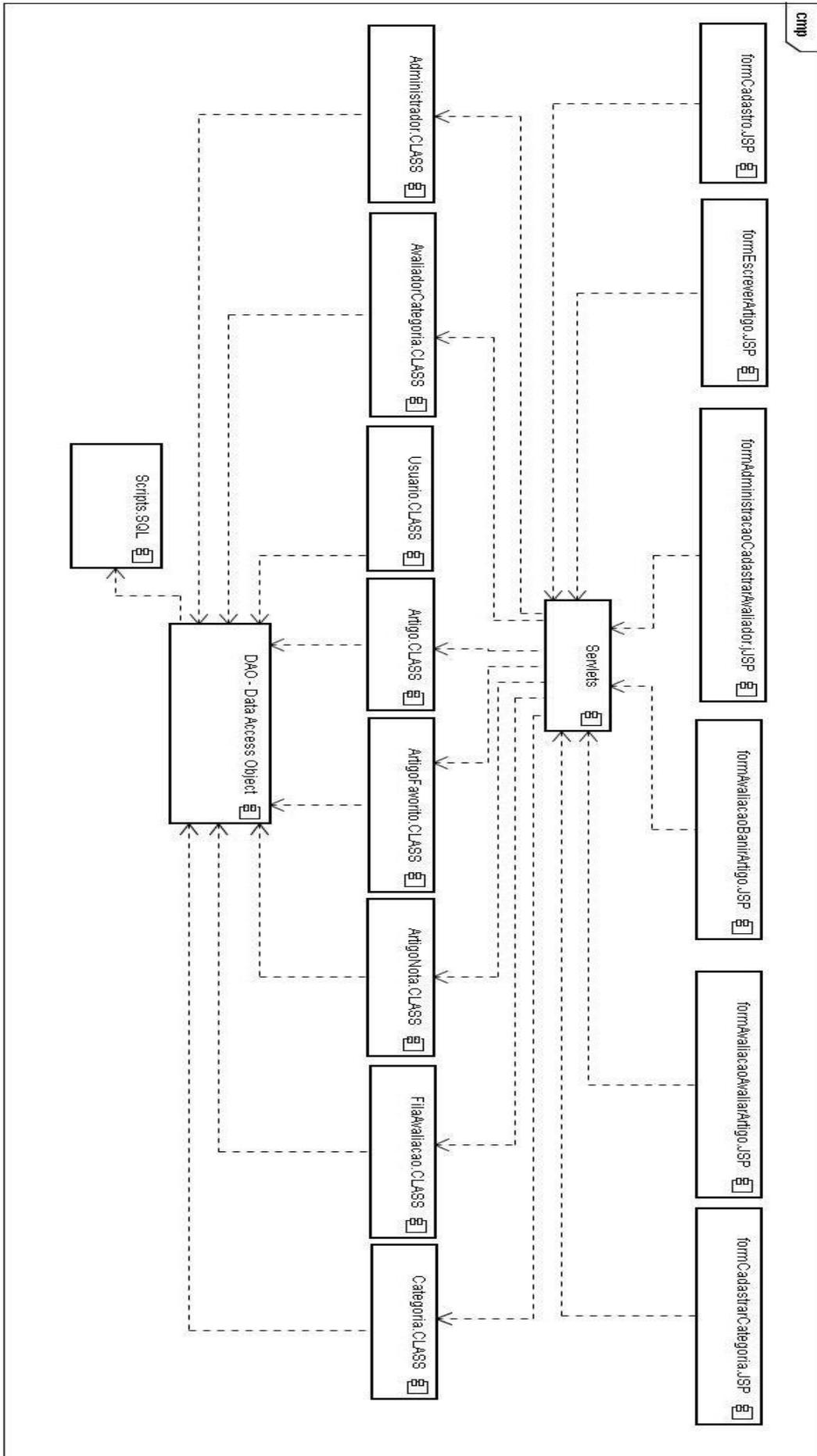
### 7.4 DIAGRAMA DE ESTADO



## 7.5 DIAGRAMA DE ATIVIDADES

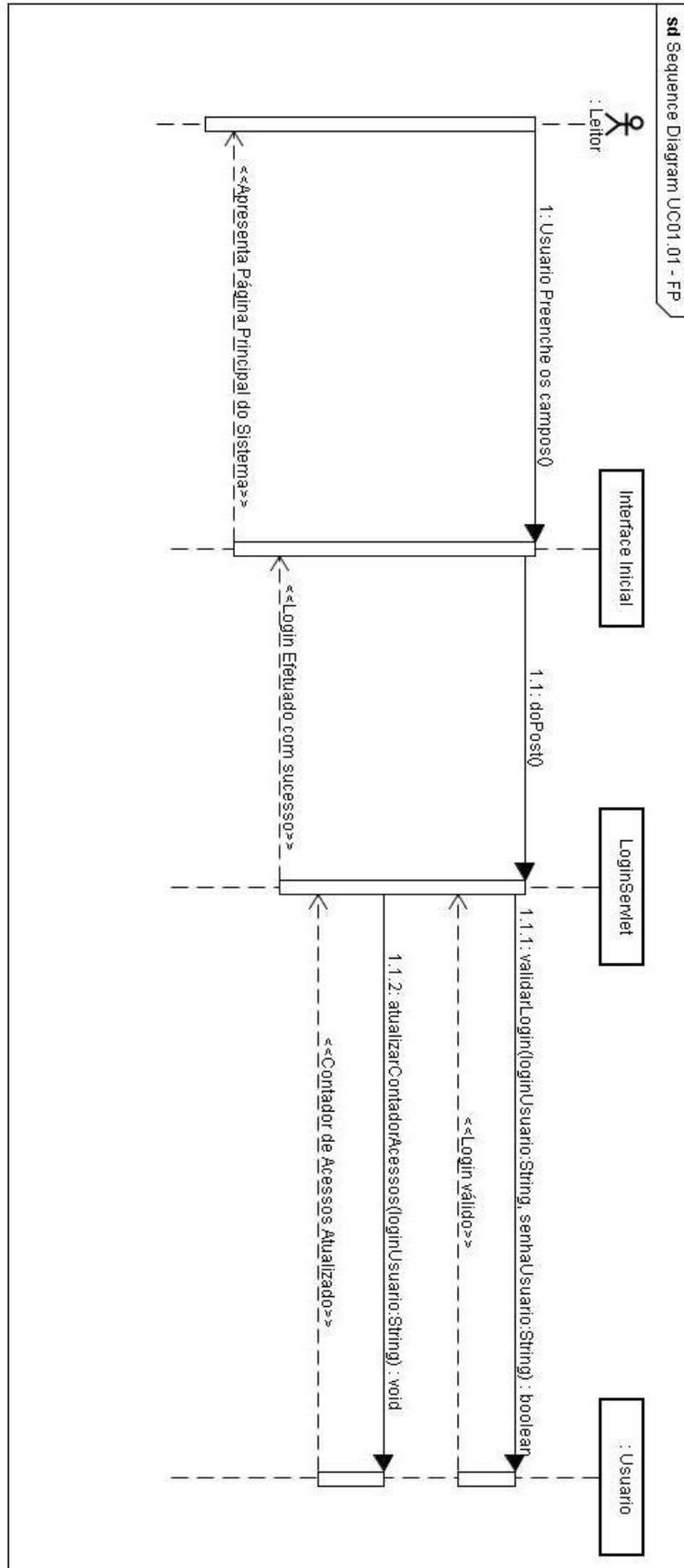


### 7.6 DIAGRAMA DE COMPONENTES

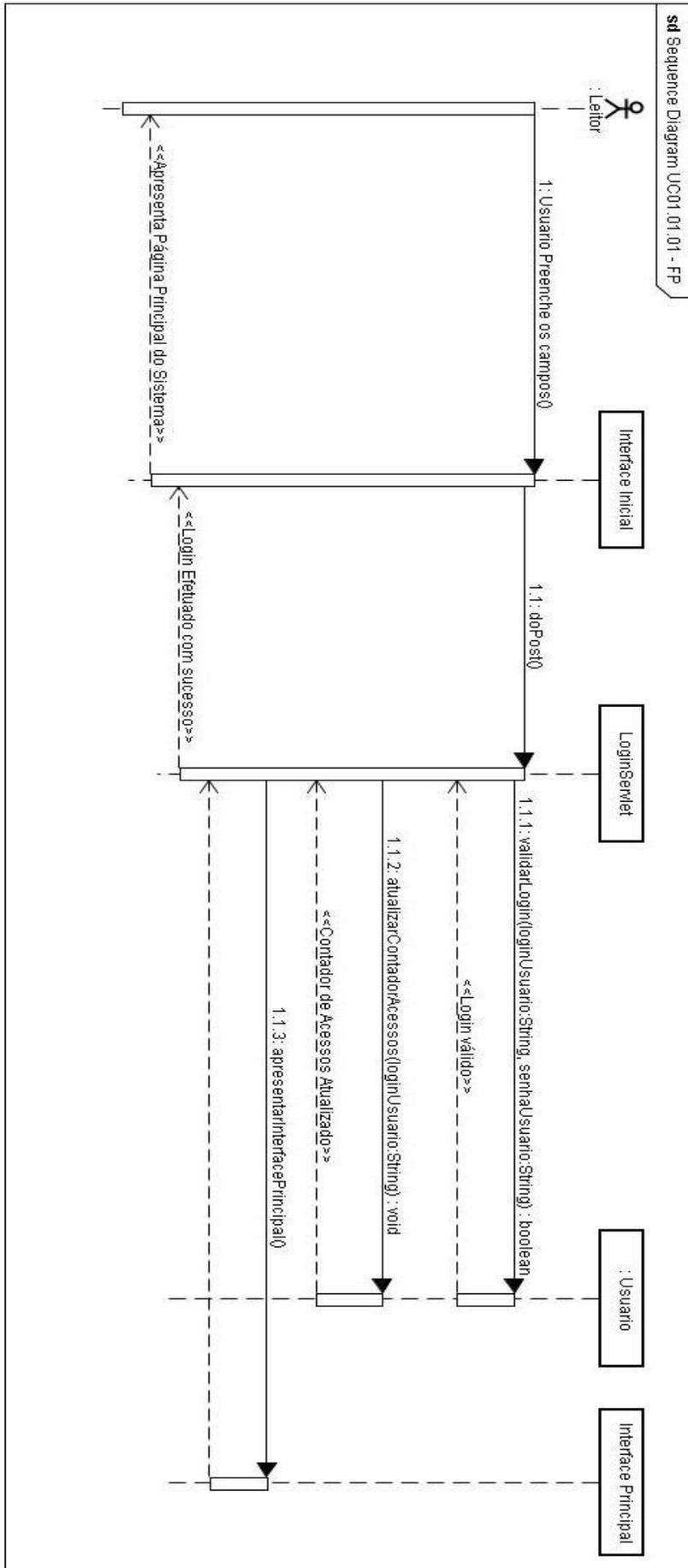


## 7.7 DIAGRAMAS DE SEQUÊNCIA

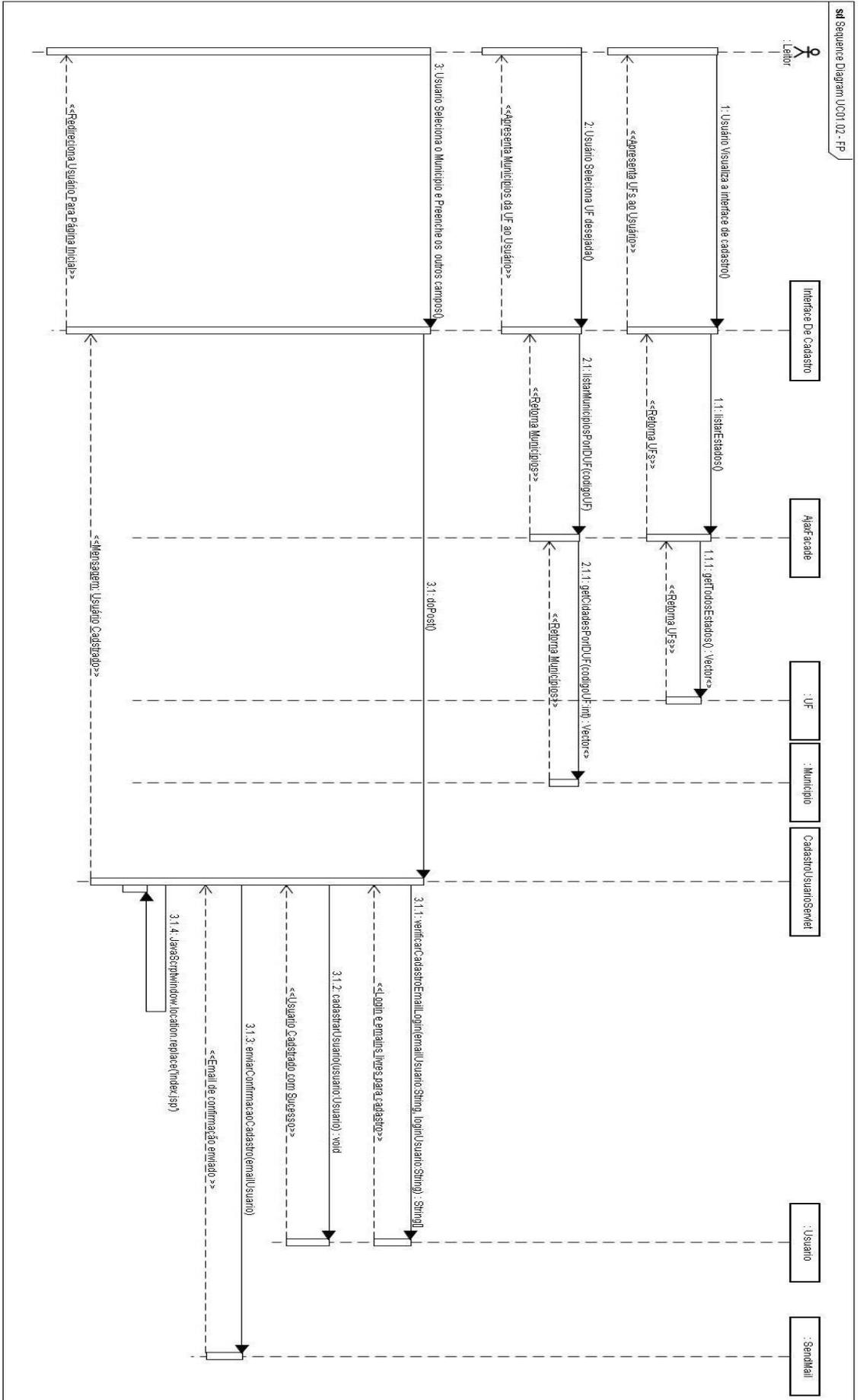
## 7.7.1 DS01.01 – Efetuar Login no Sistema - FP



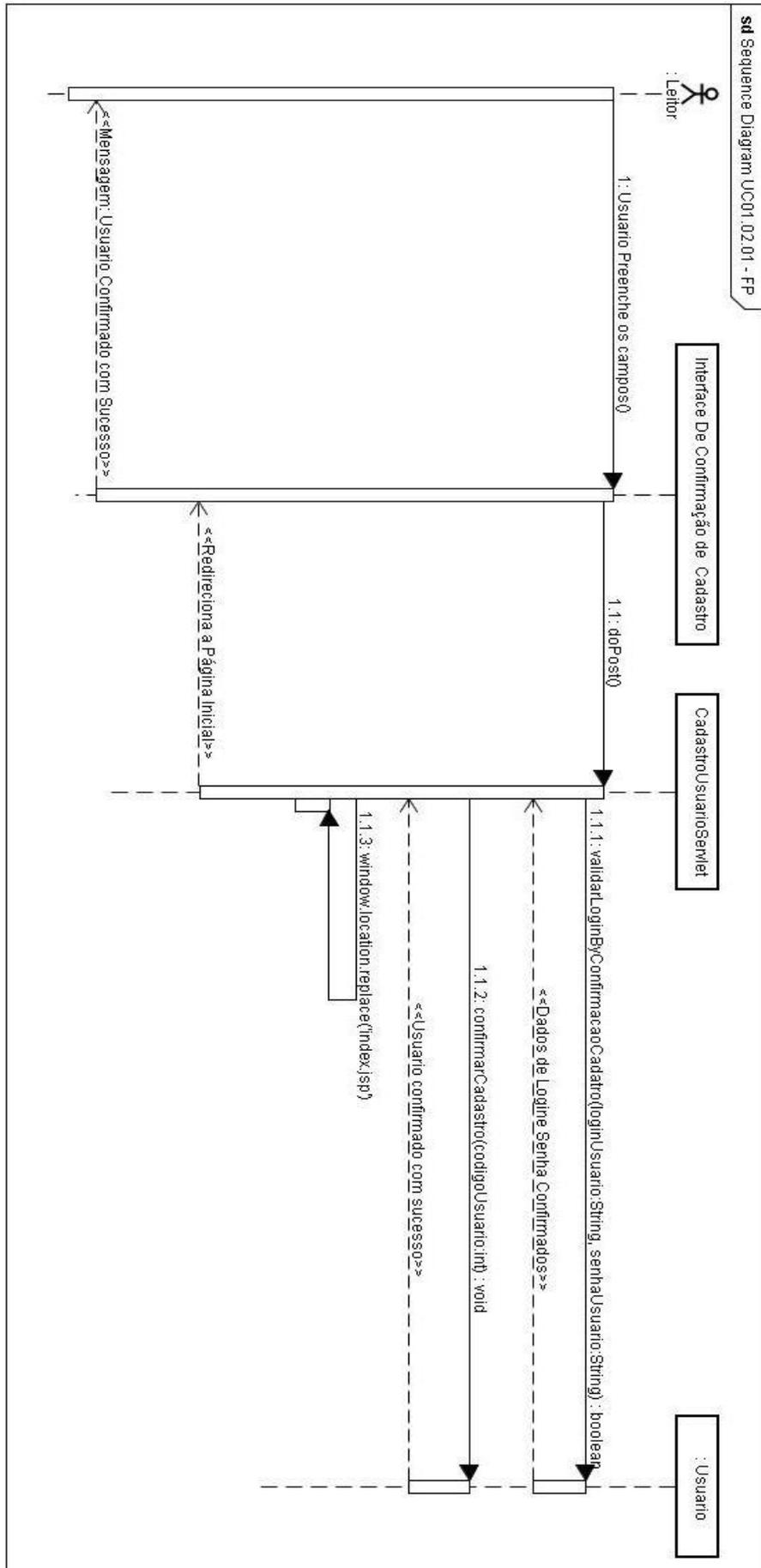
## 7.7.2 DS01.01.01 – Visualizar Página Principal - FP



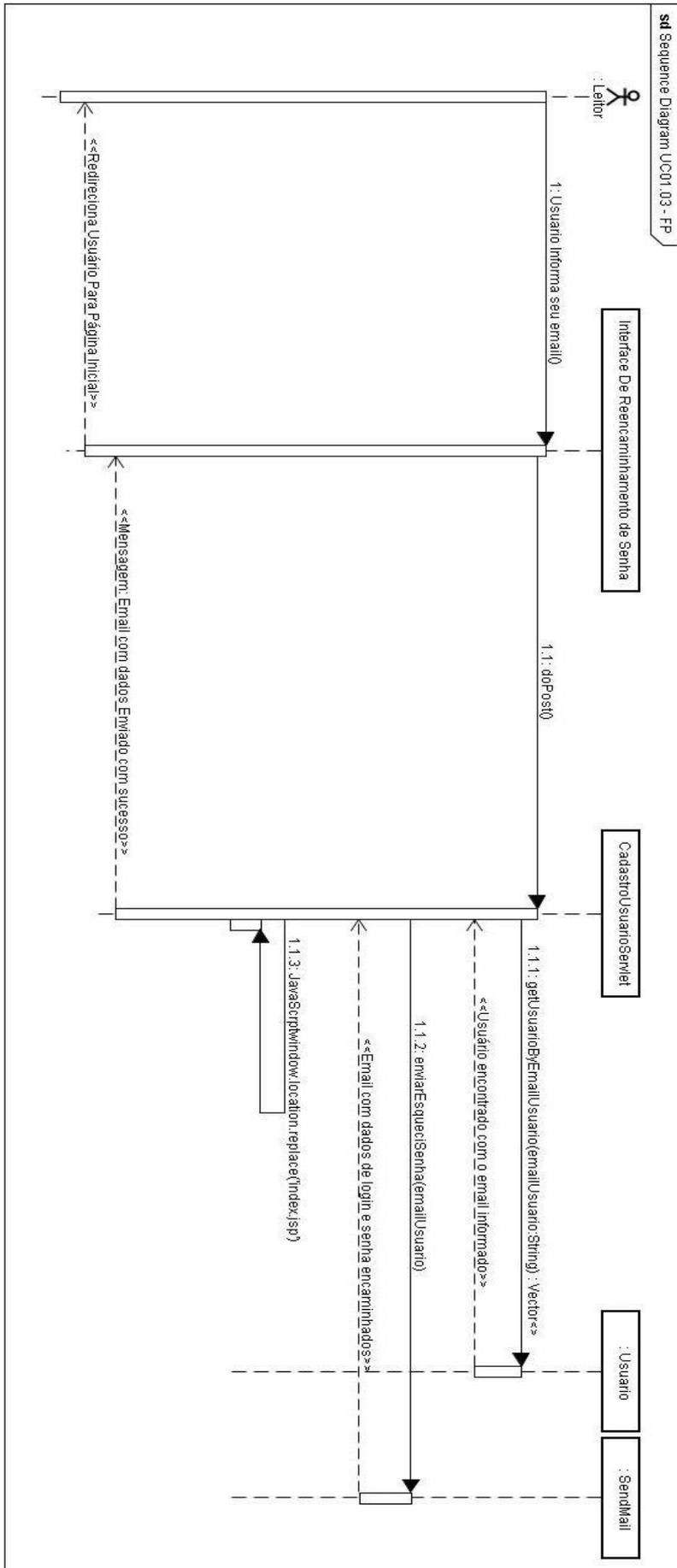
### 7.7.3 DS01.02 – Realizar Cadastro no Sistema - FP



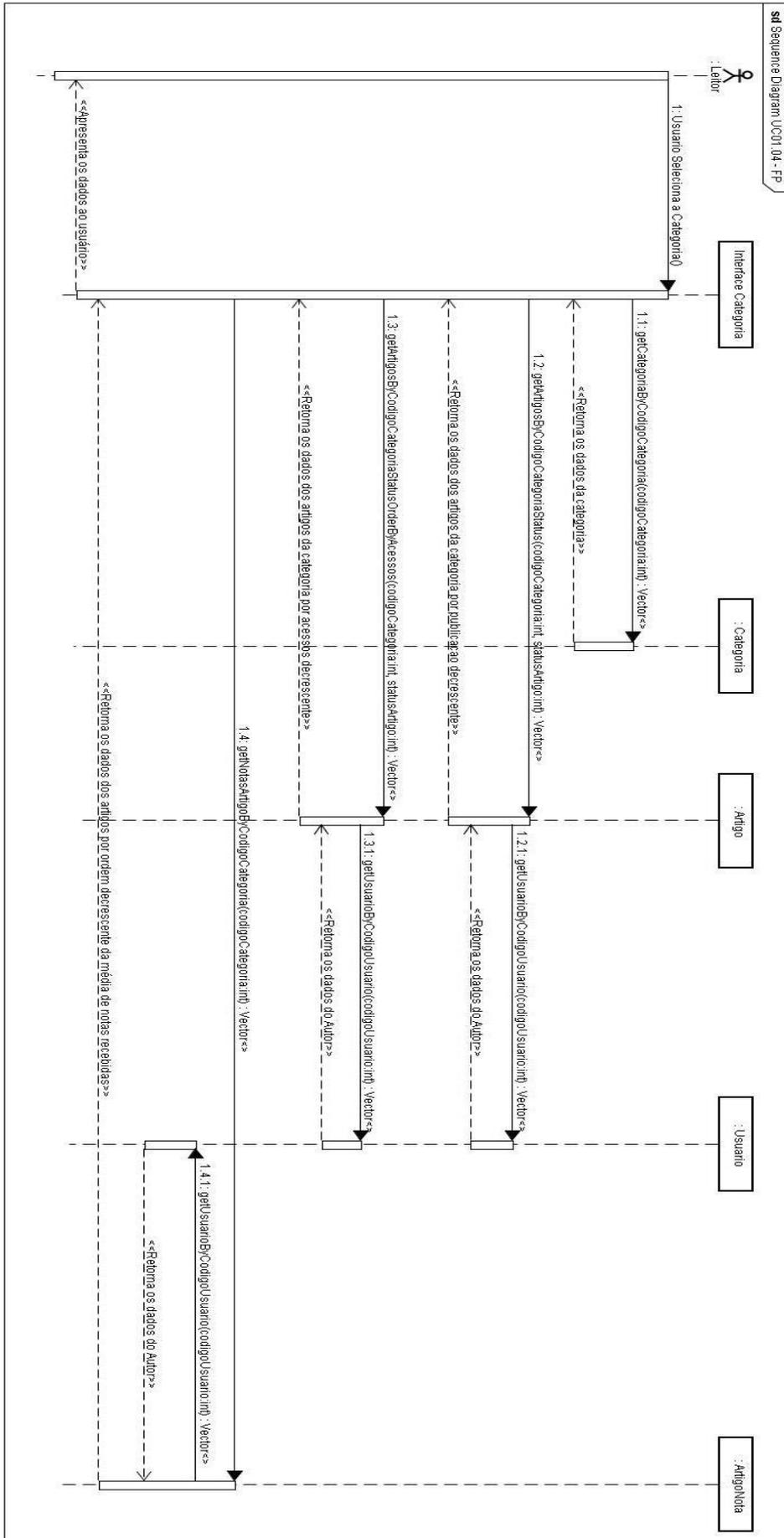
## 7.7.4 DS01.02.01 – Efetuar Confirmação de Cadastro - FP



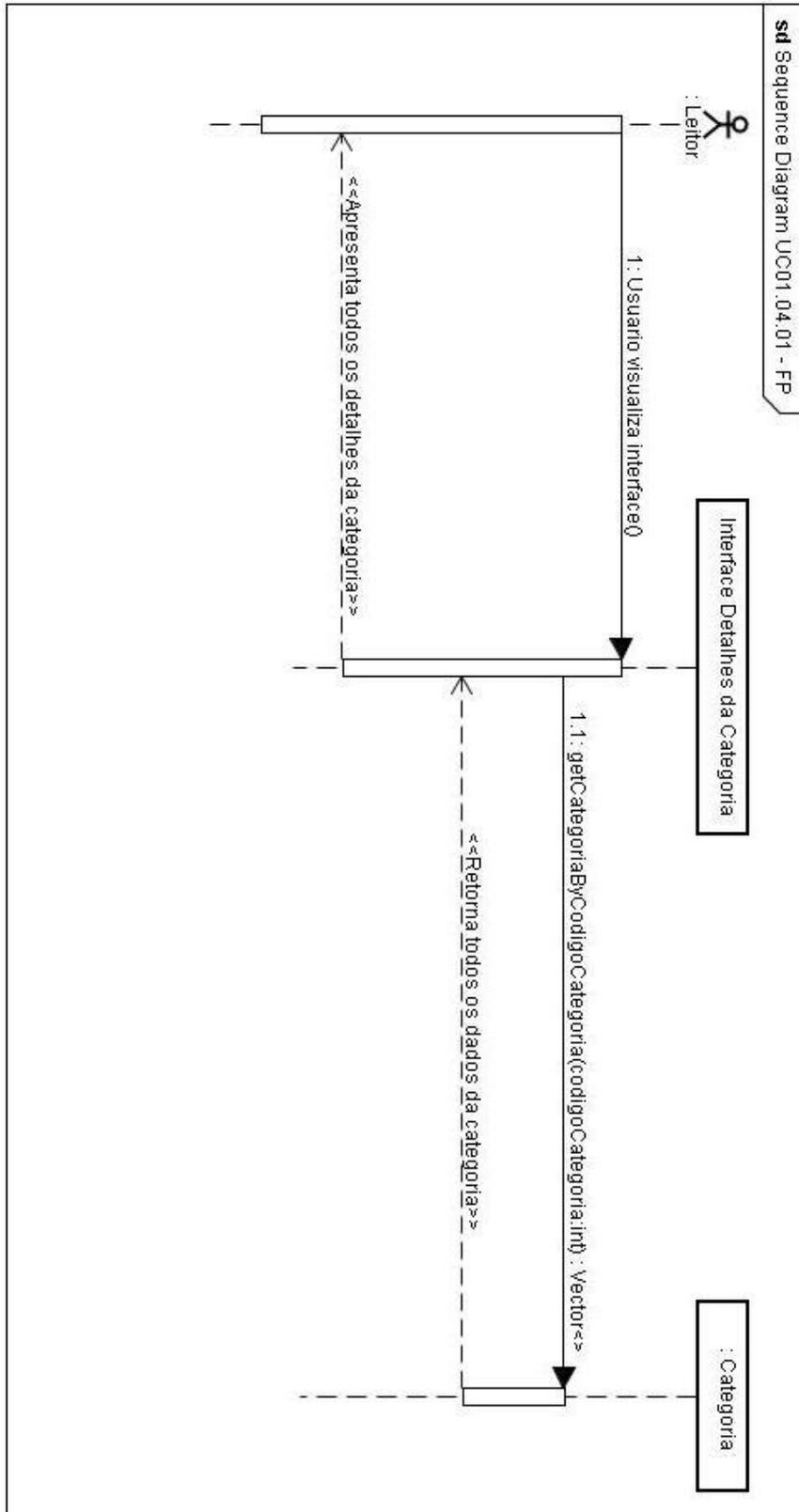
## 7.7.5 DS01.03 – Reencaminhar Senha - FP



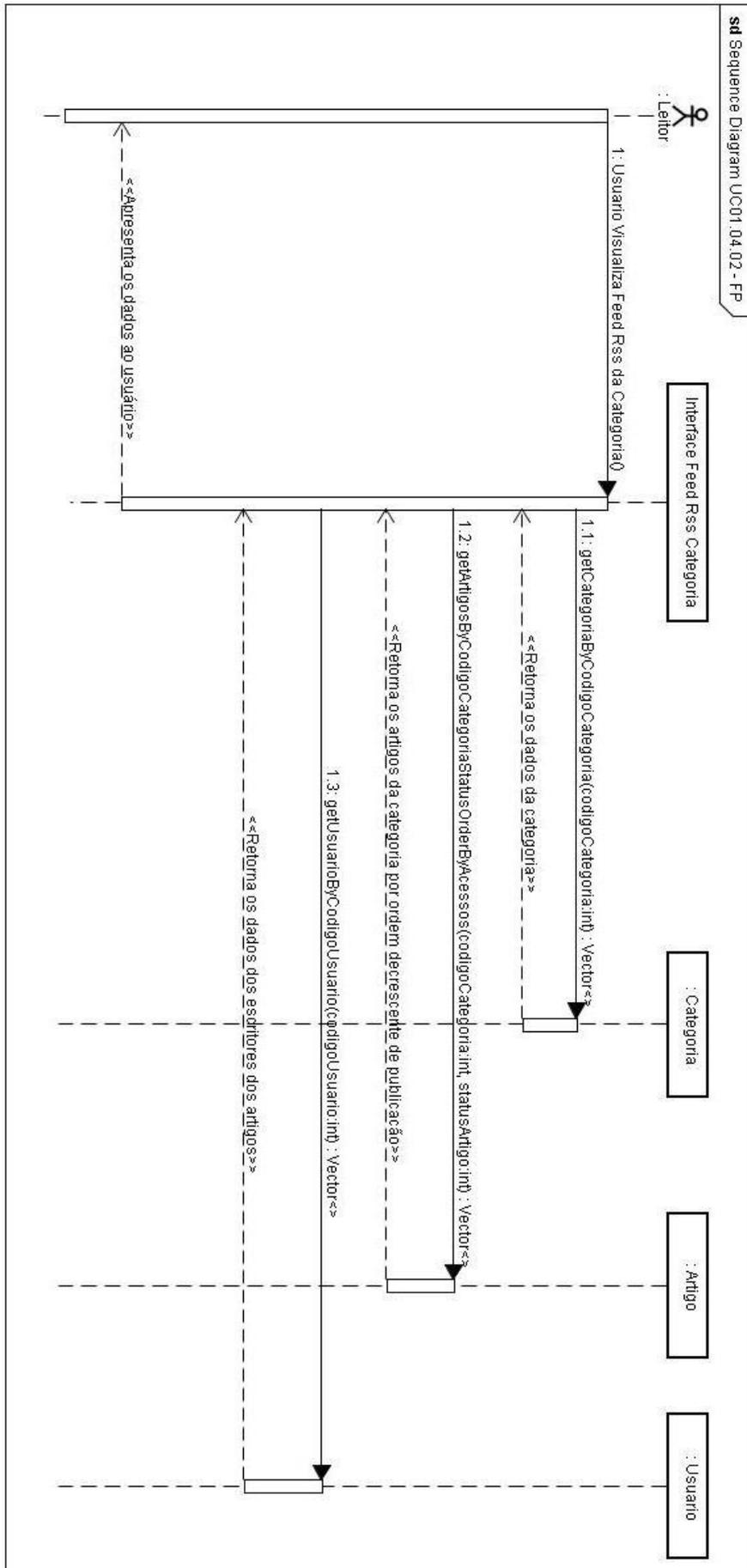
### 7.7.6 DS01.04 – Visualizar Categoria - FP



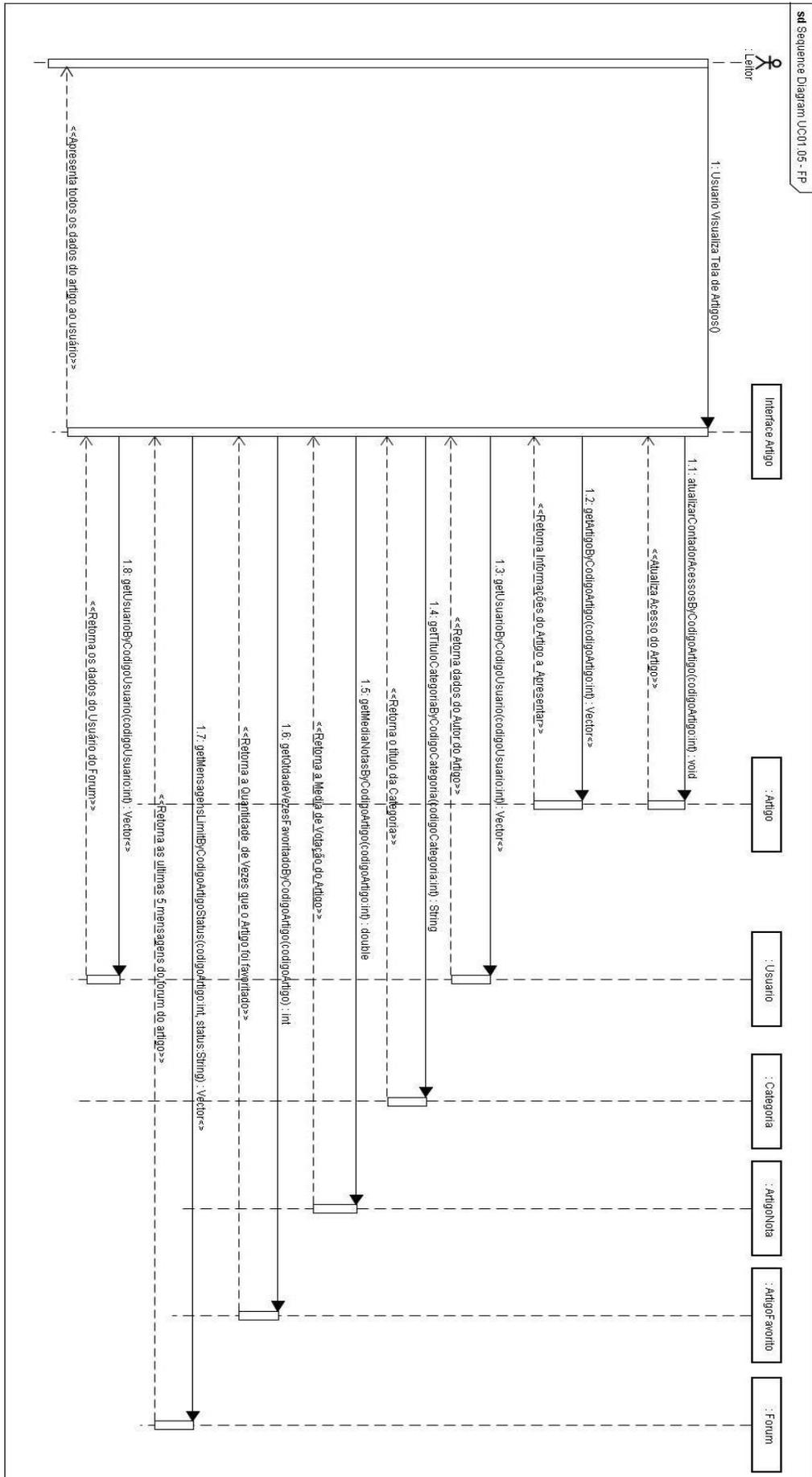
### 7.7.7 DS01.04.01 – Ver Detalhes da Categoria - FP



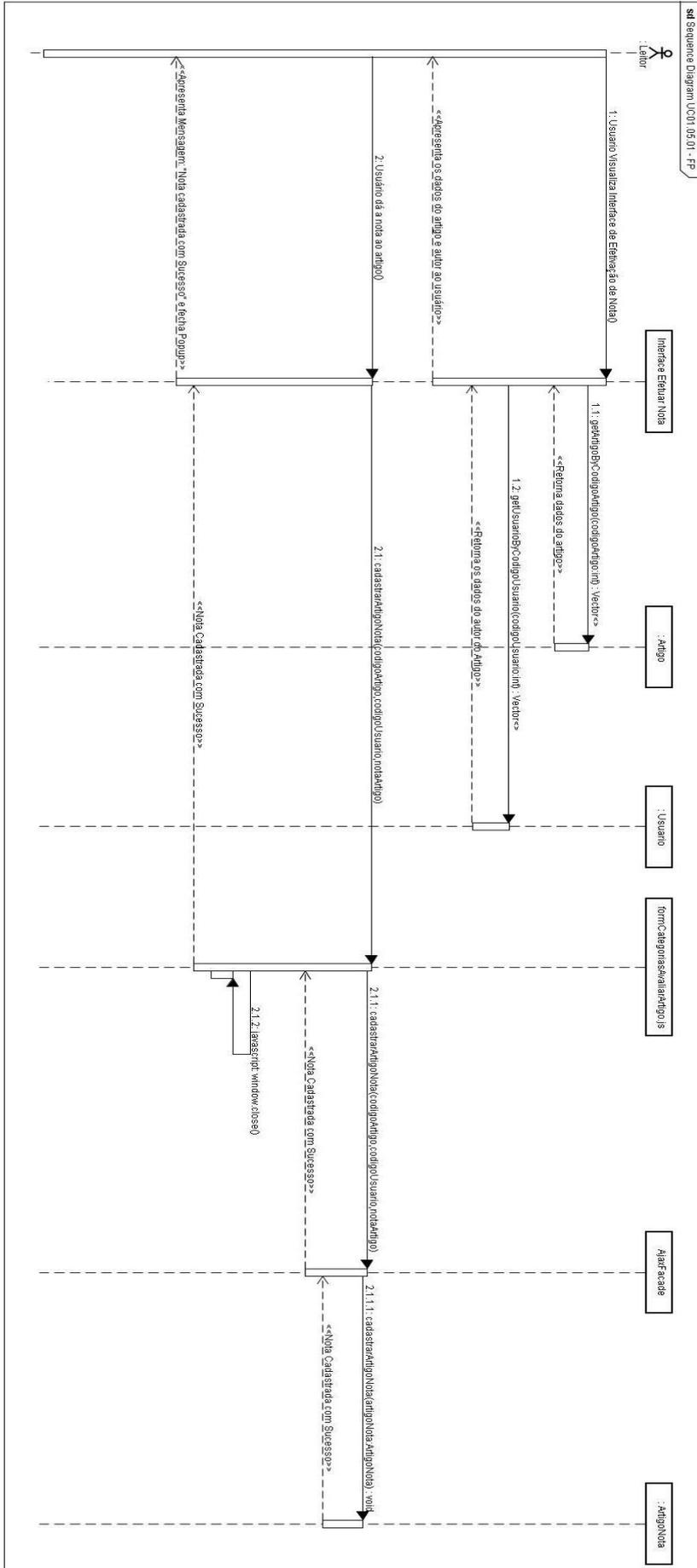
## 7.7.8 DS01.04.02 – Ver Feed da Categoria - FP



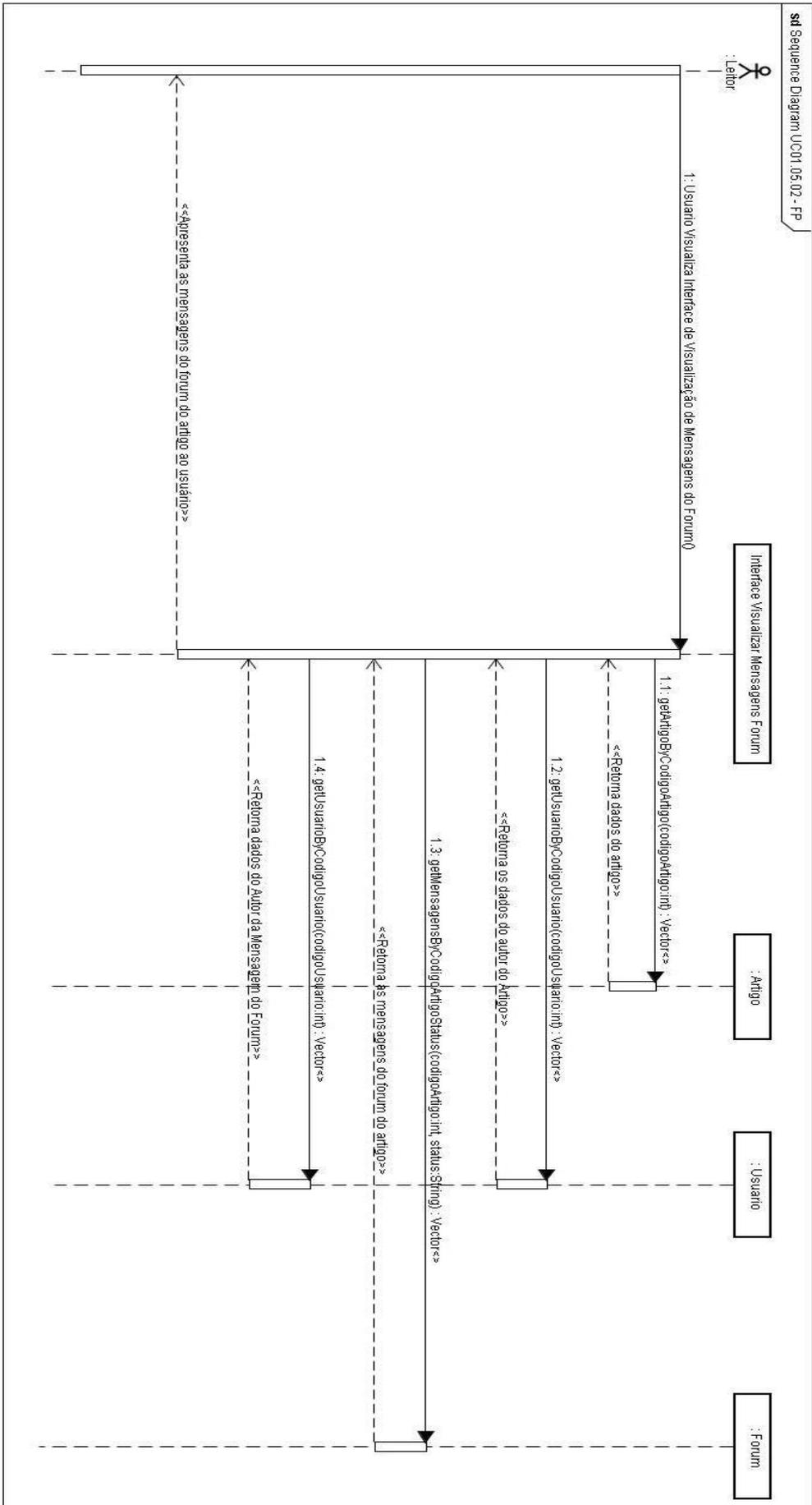
## 7.7.9 DS01.05 – Visualizar Artigo - FP



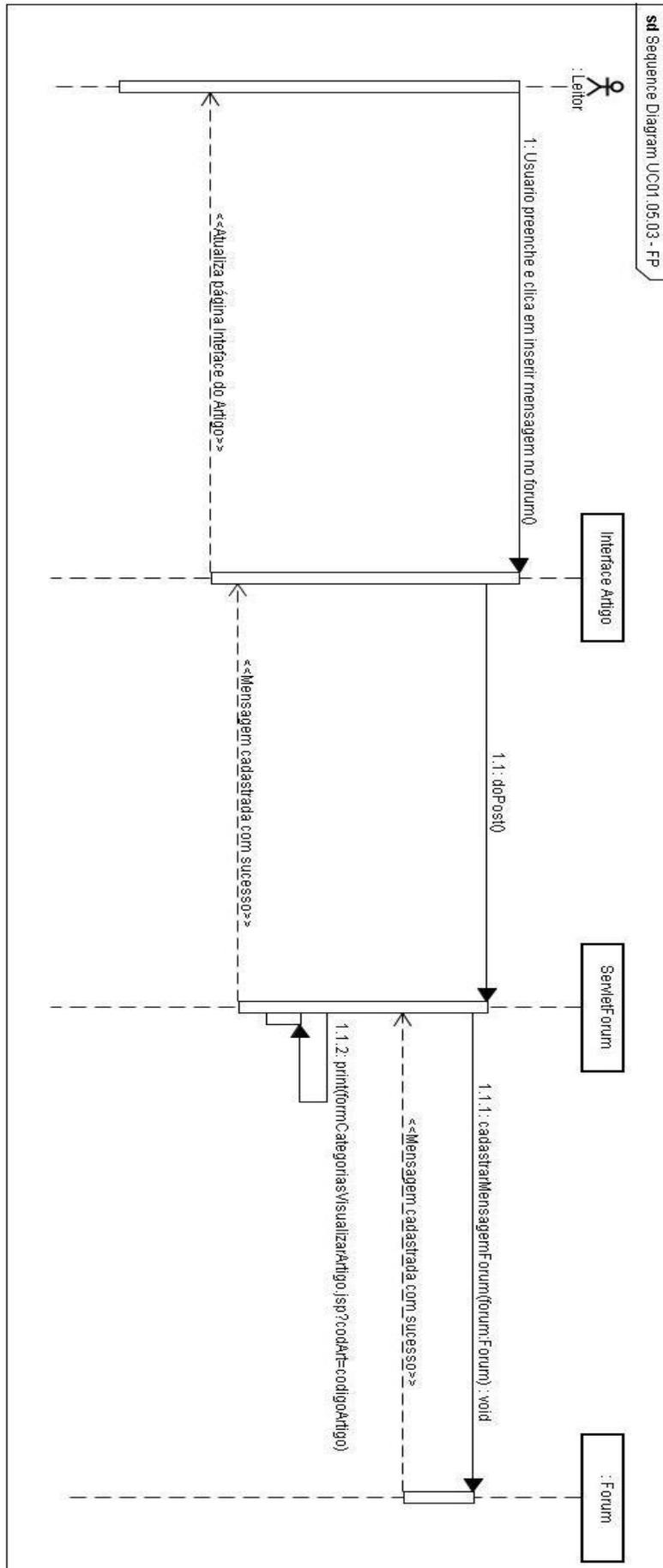
### 7.7.10 DS01.05.01 – Dar Nota ao Artigo - FP



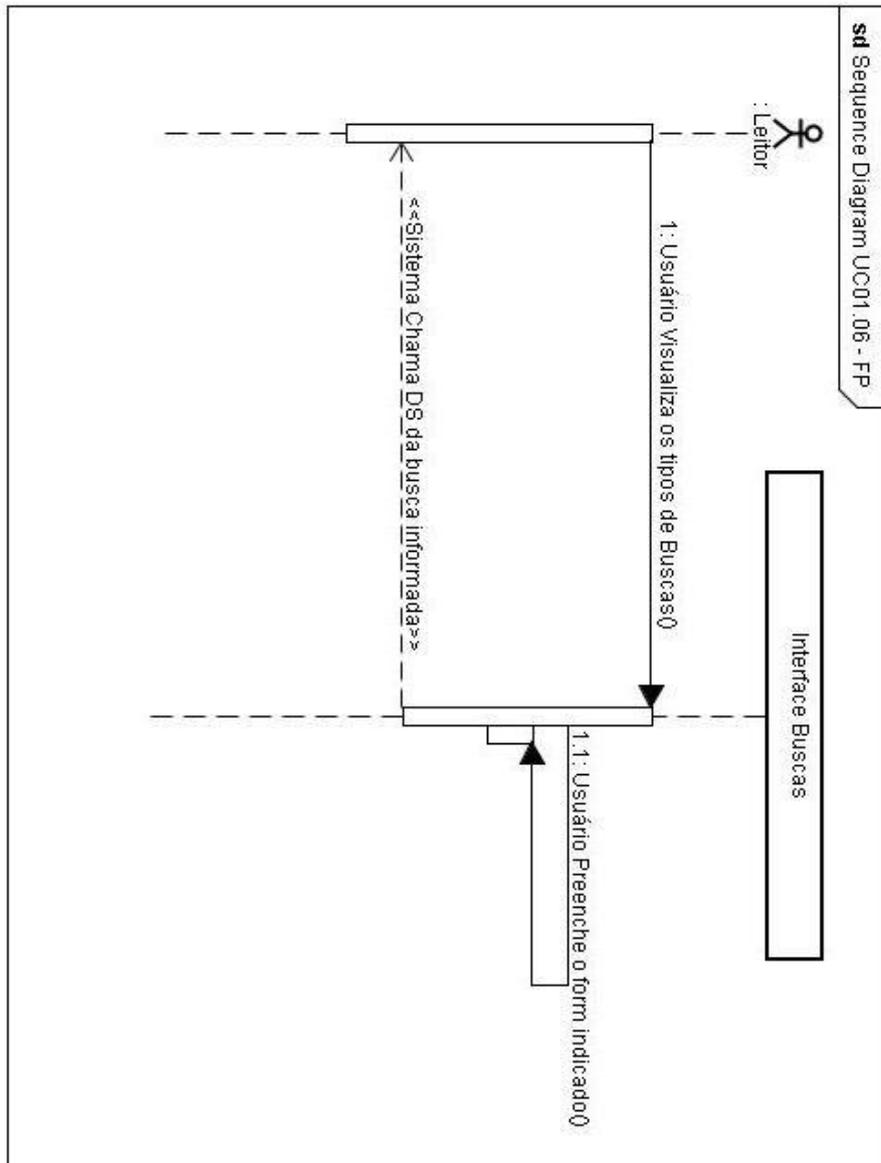
## 7.7.11 DS01.05.02 – Visualizar Mensagens do Fórum do Artigo - FP



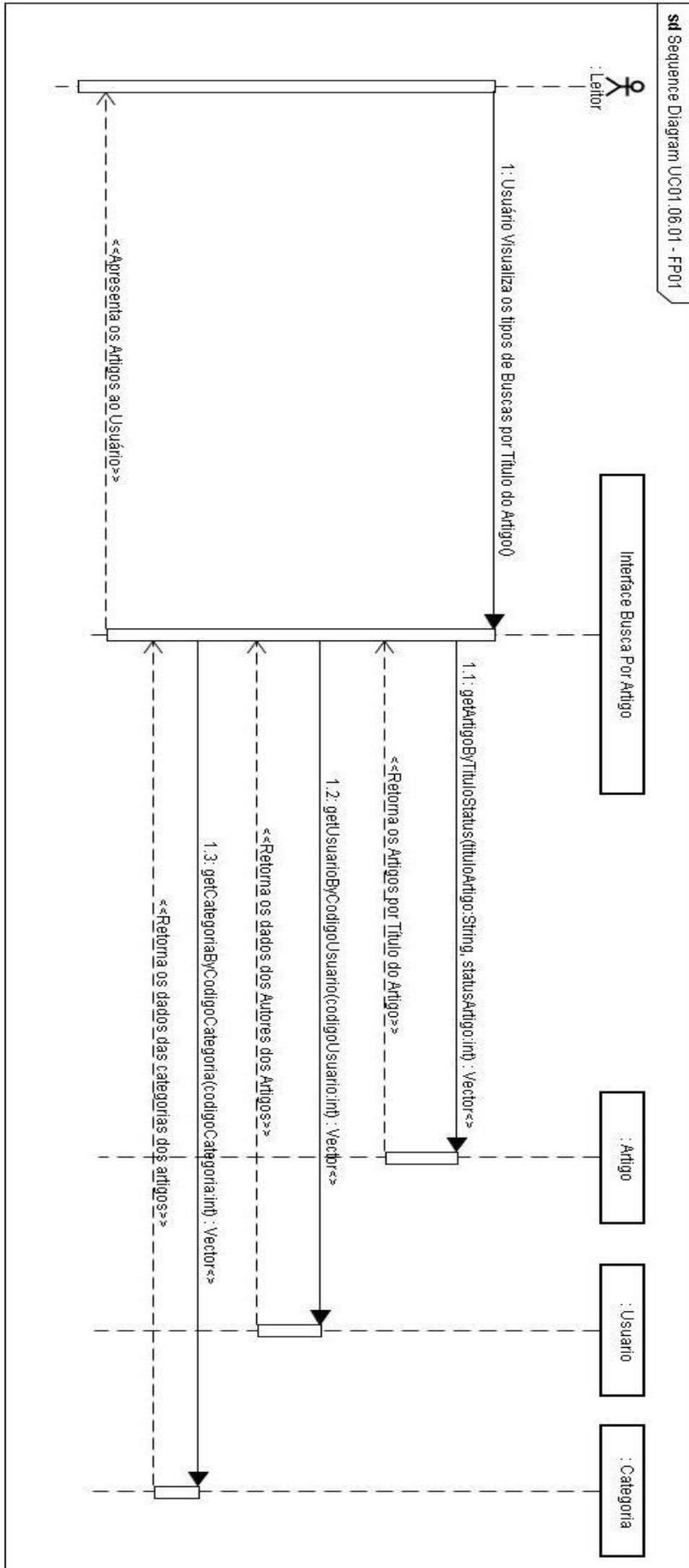
## 7.7.12 DS01.05.03 – Inserir Mensagem no Fórum do Artigo - FP



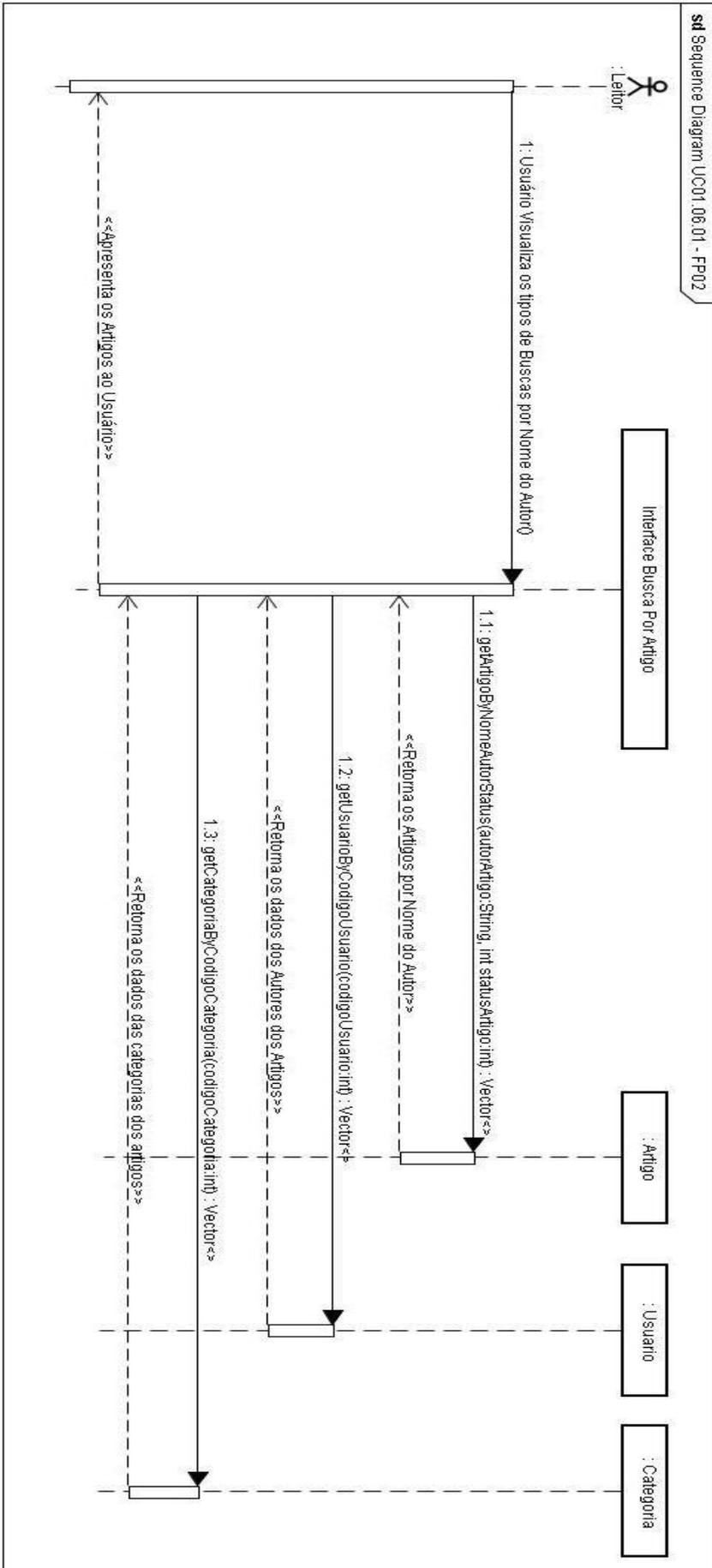
## 7.7.13 DS01.06 – Visualizar Buscas - FP



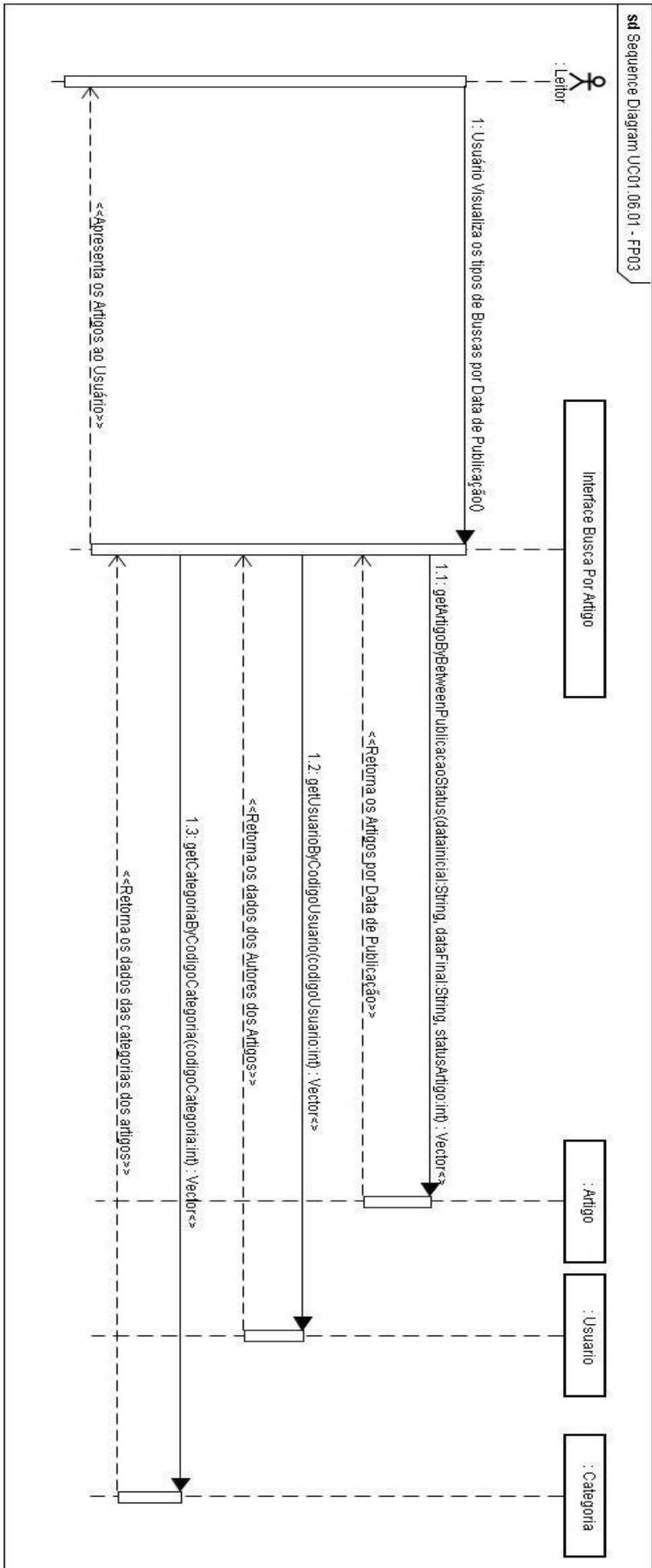
## 7.7.14 DS01.06.01 – Realizar Busca de Artigos - FP1



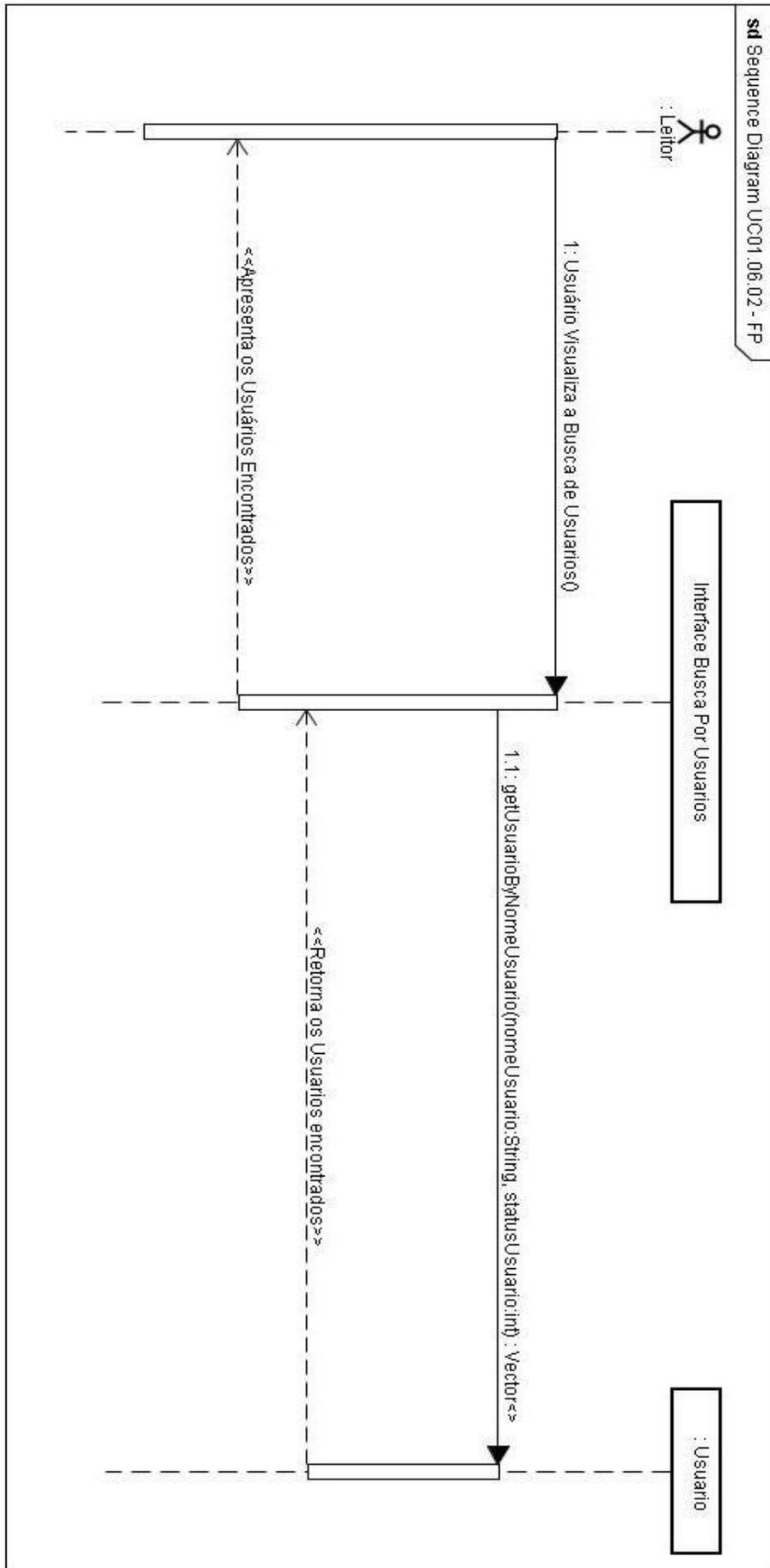
### 7.7.15 DS01.06.01 – Realizar Busca de Artigos – FP2



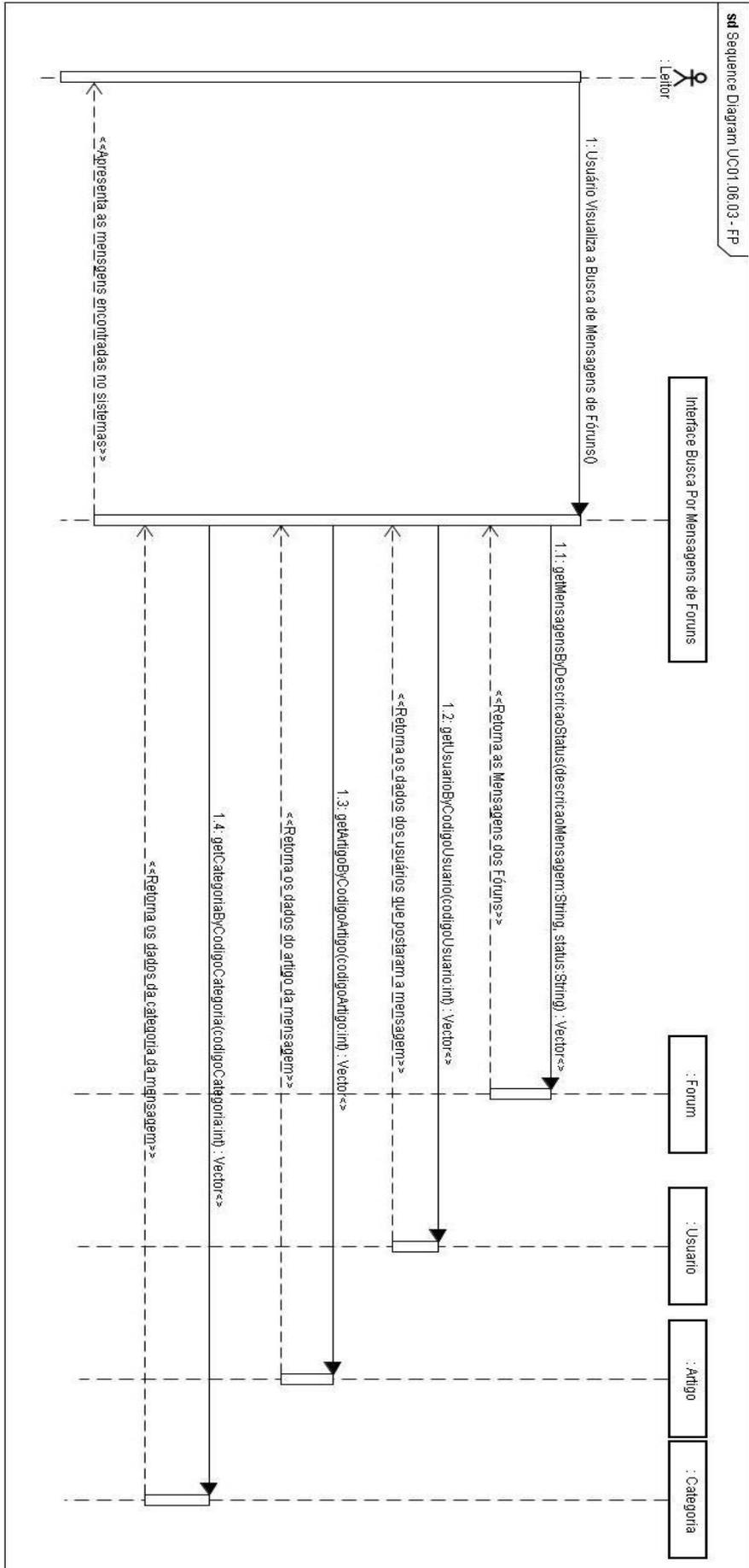
### 7.7.16 DS01.06.01 – Realizar Busca de Artigos – FP3



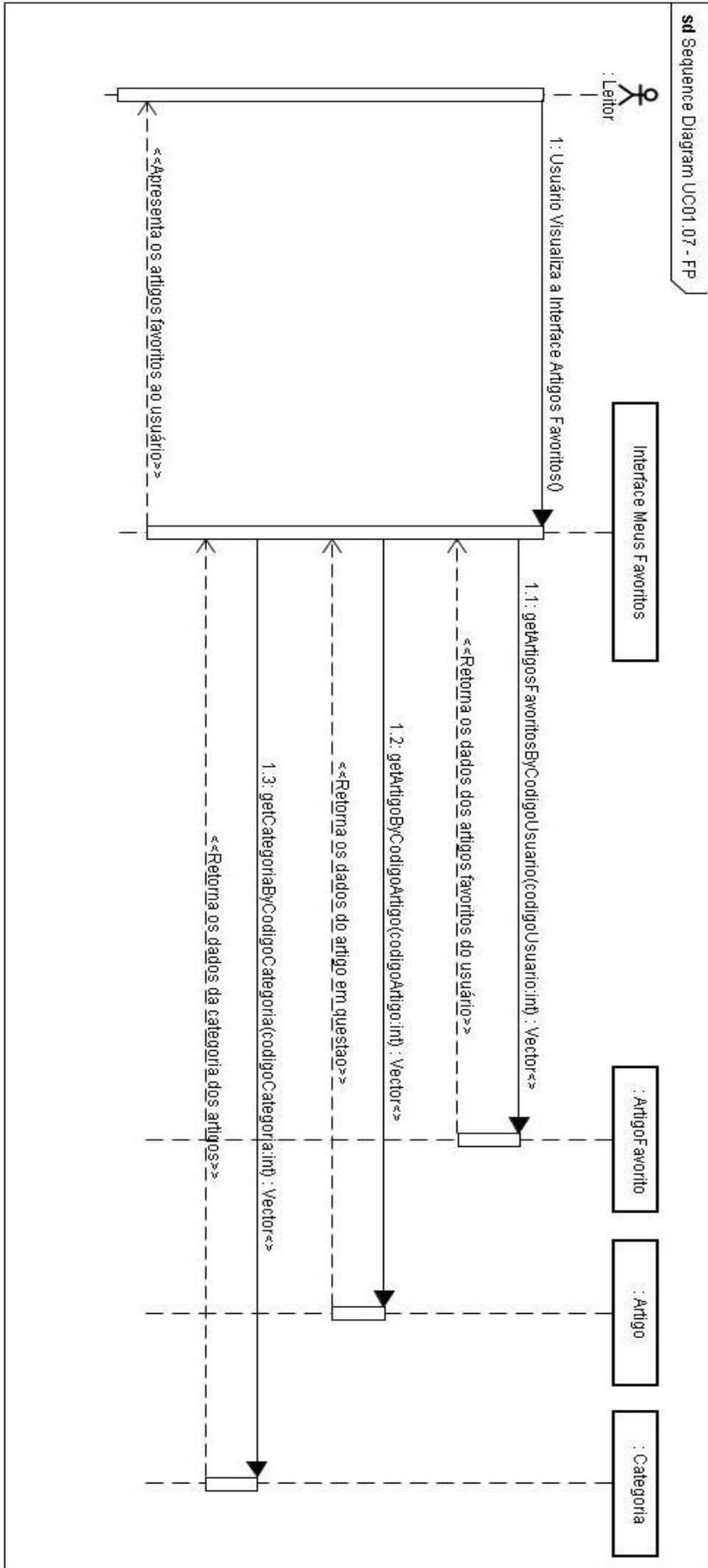
## 7.7.17 DS01.06.02 – Realizar Busca de Usuários - FP



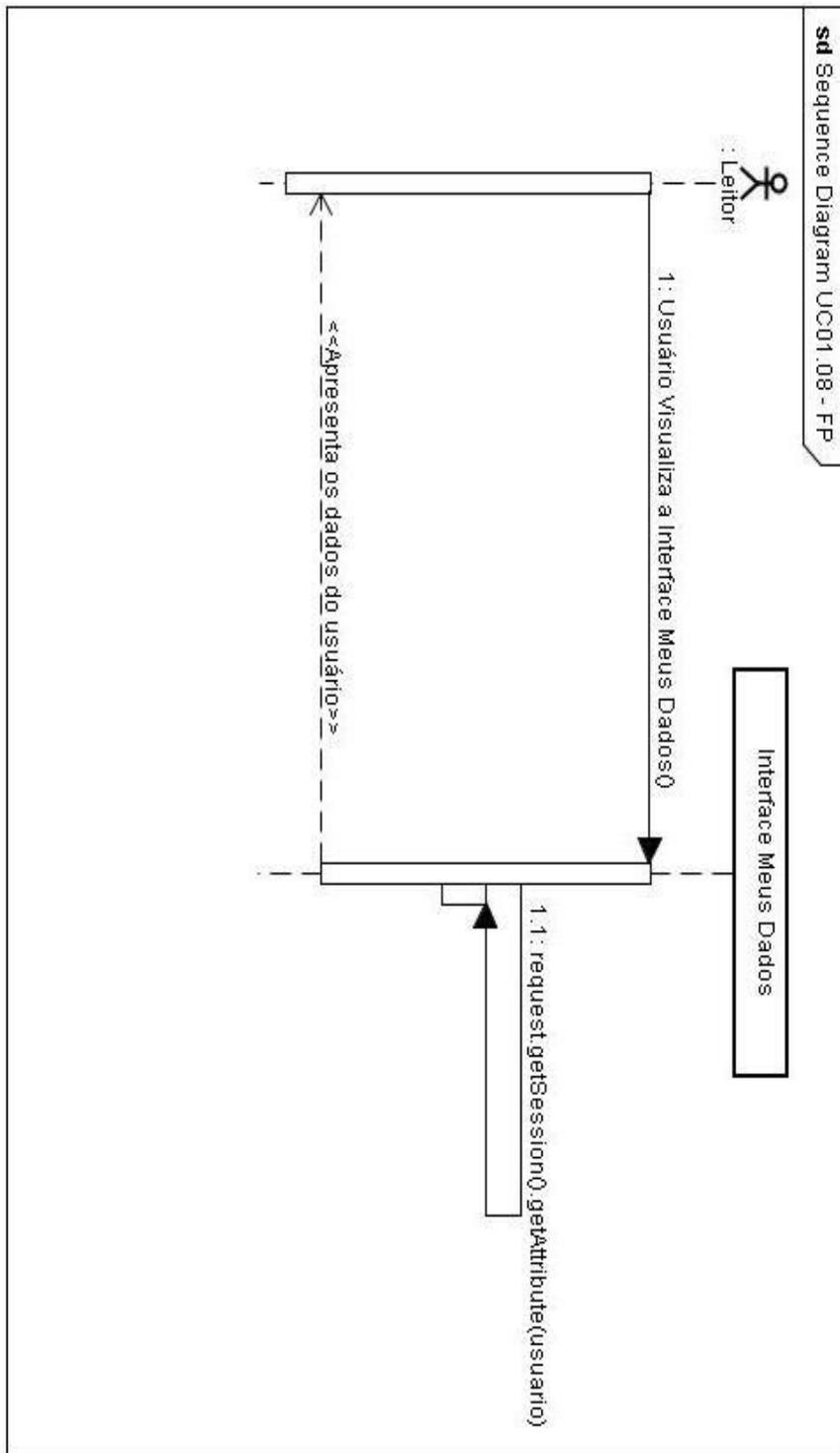
7.7.18 DS01.06.03 – Realizar Buscas de Mensagens dos Fóruns - FP



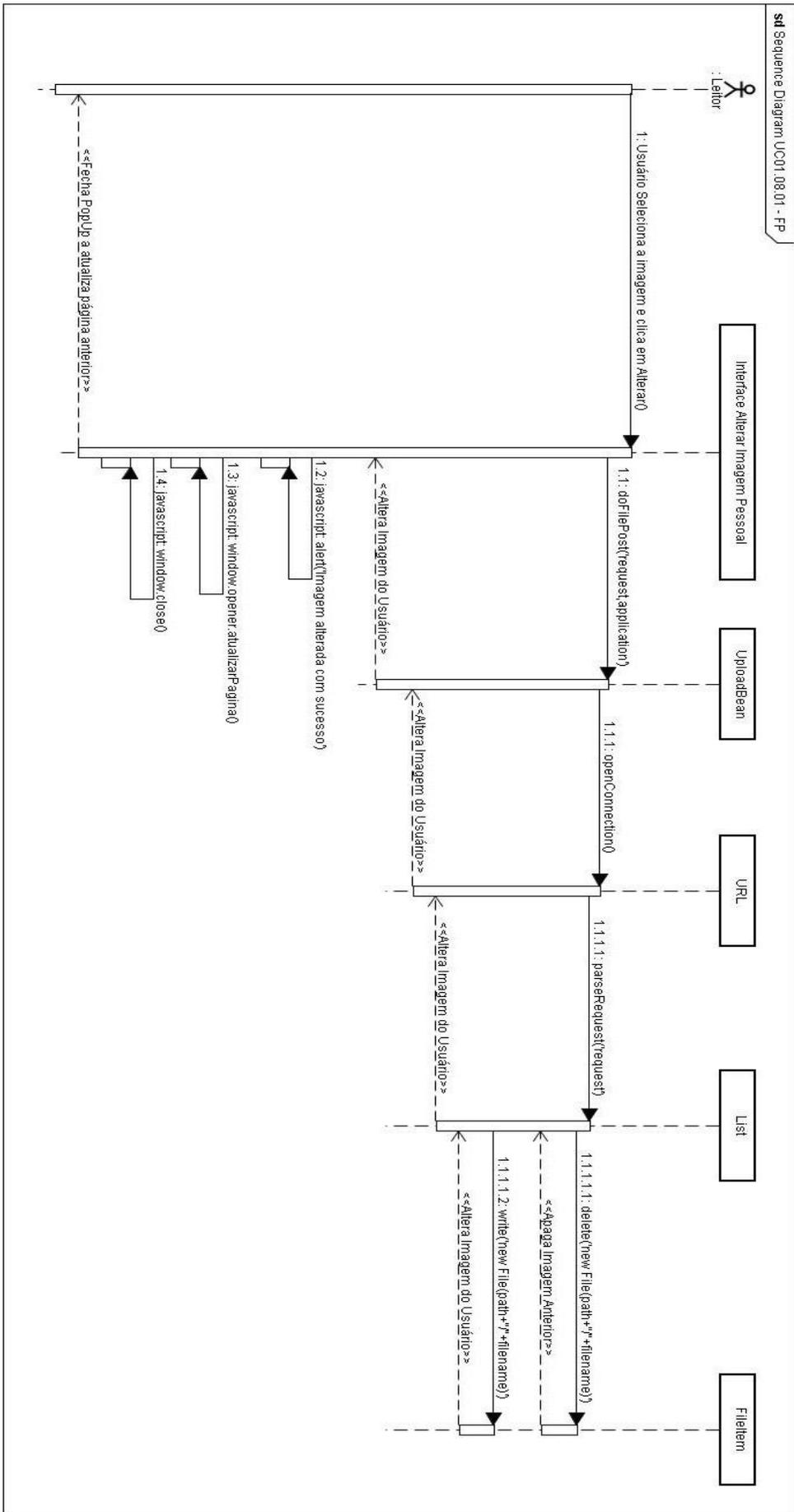
## 7.7.19 DS01.07 – Ver Meus Artigos Favoritos - FP



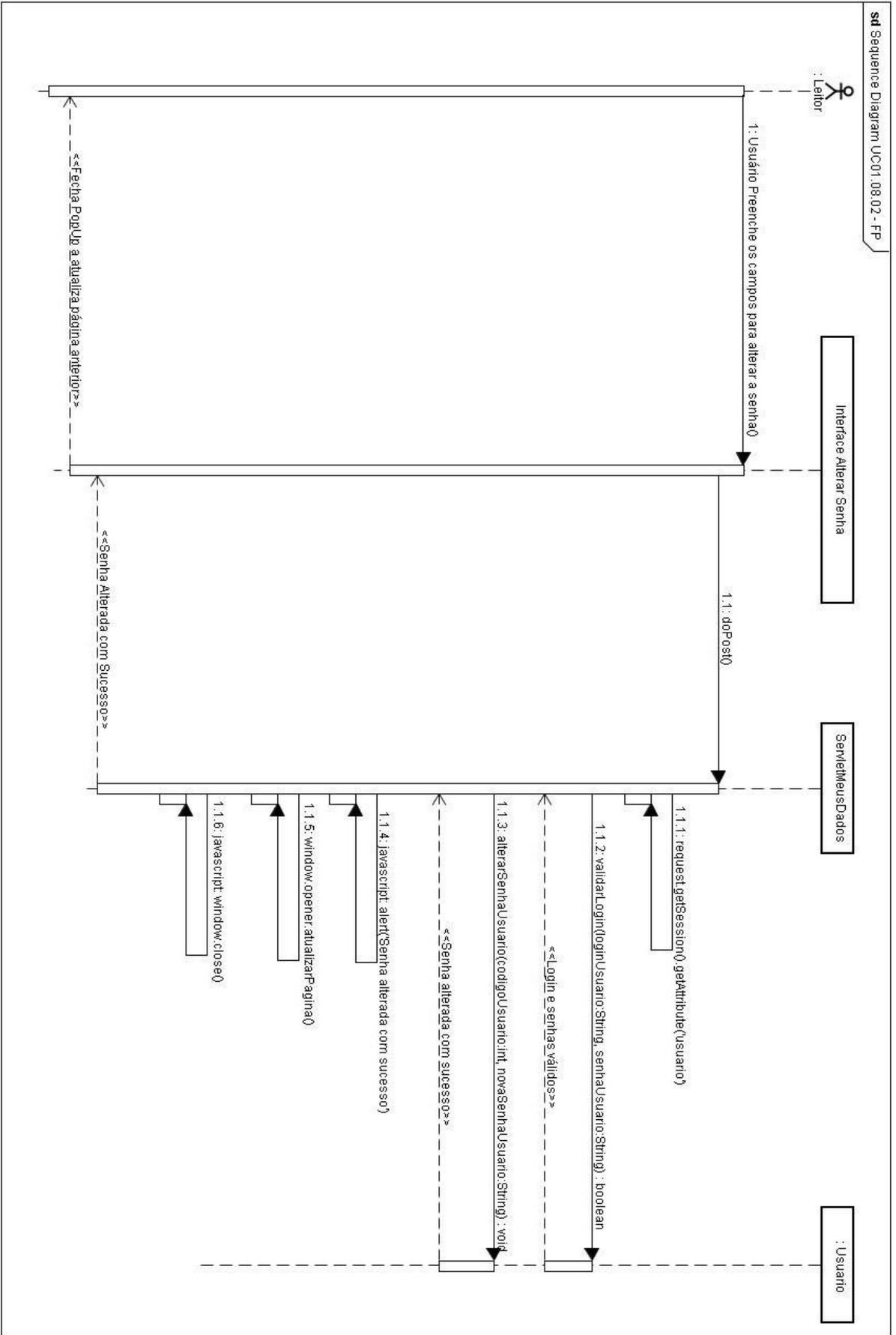
## 7.7.20 DS01.08 – Visualizar Dados Pessoais - FP



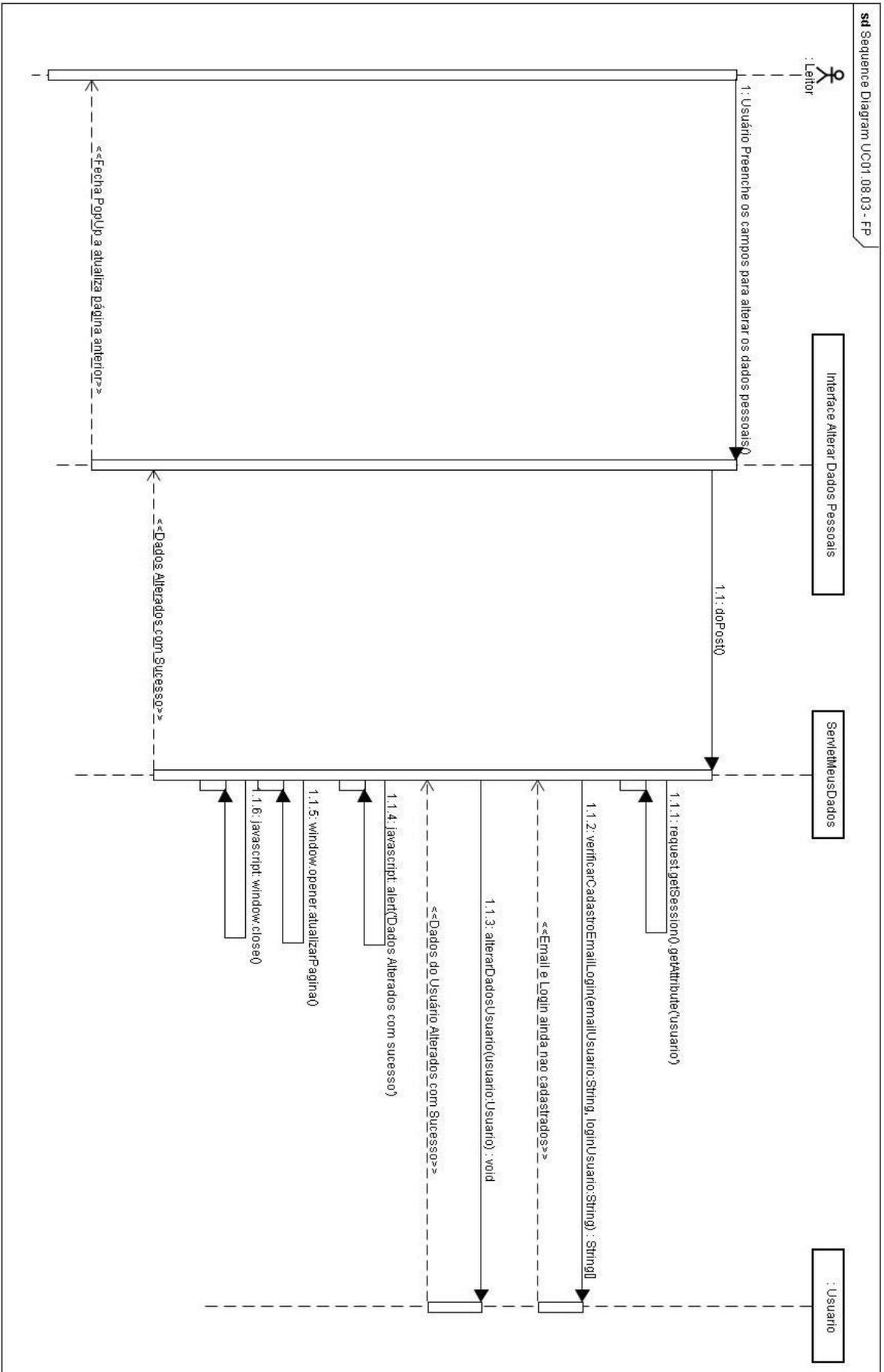
7.7.21 DS01.08.01 – Alterar Imagem Pessoal - FP



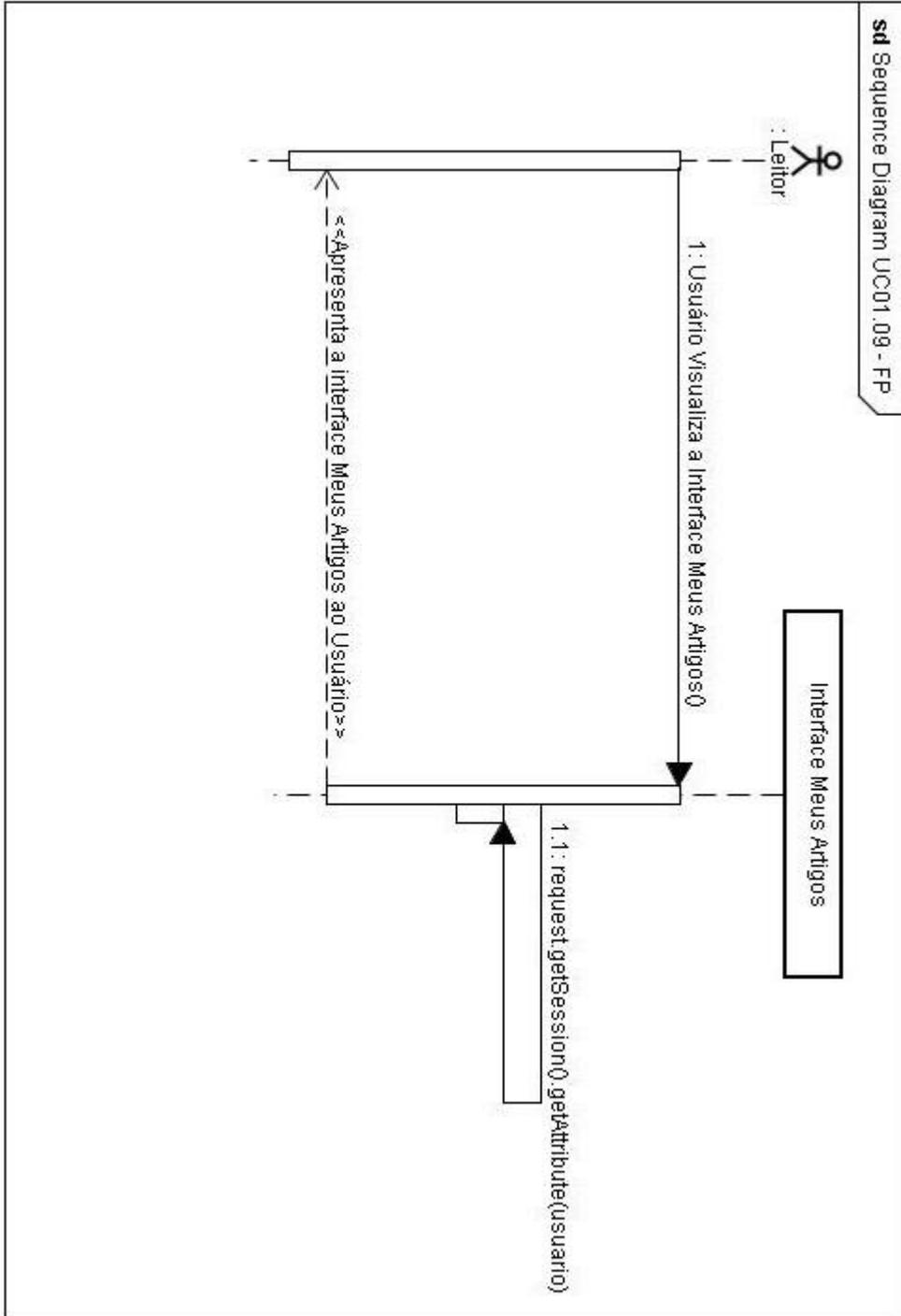
7.7.22 DS01.08.02 – Alterar Senha - FP



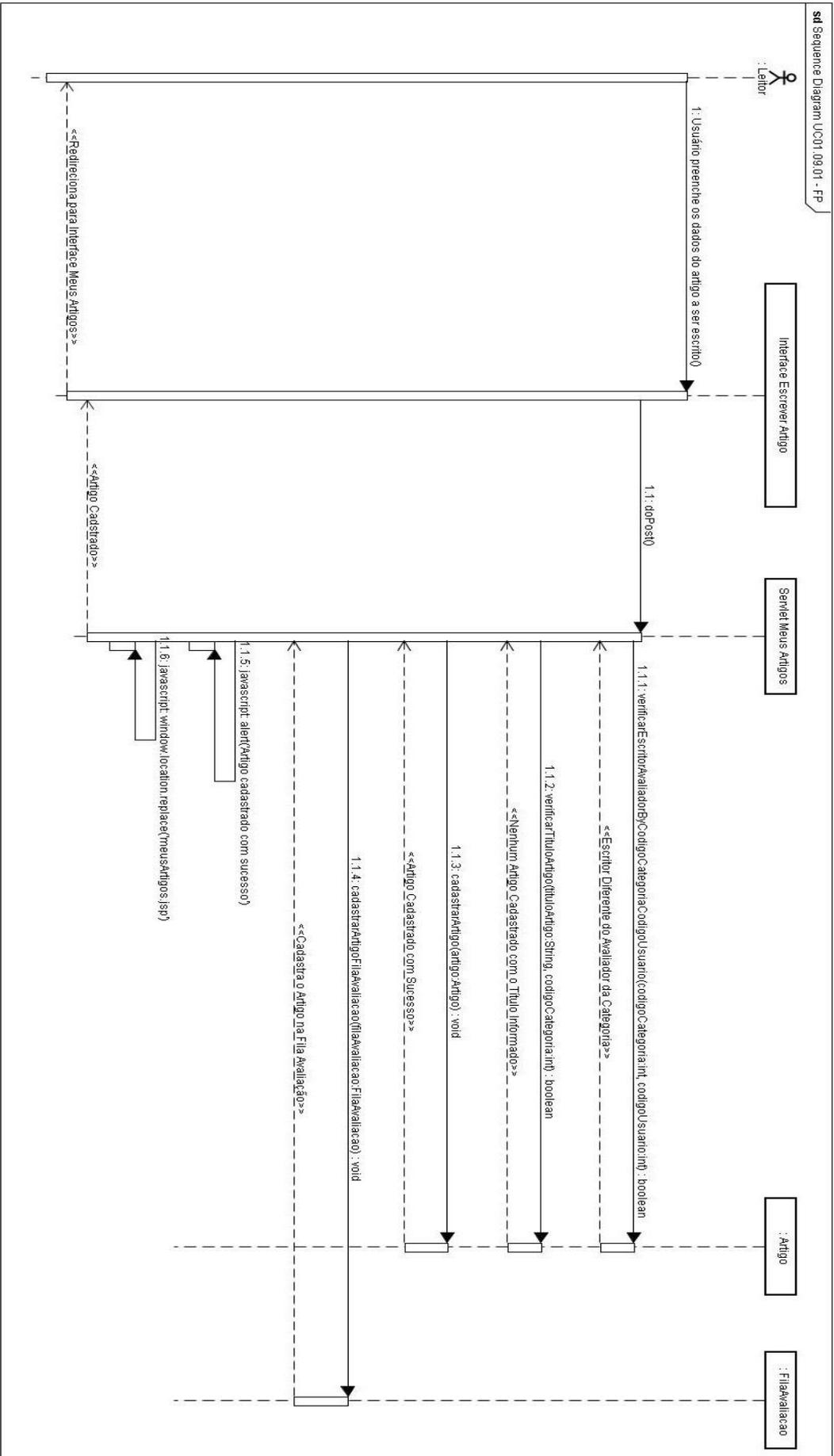
## 7.7.23 DS01.08.03 – Alterar Dados Pessoais - FP



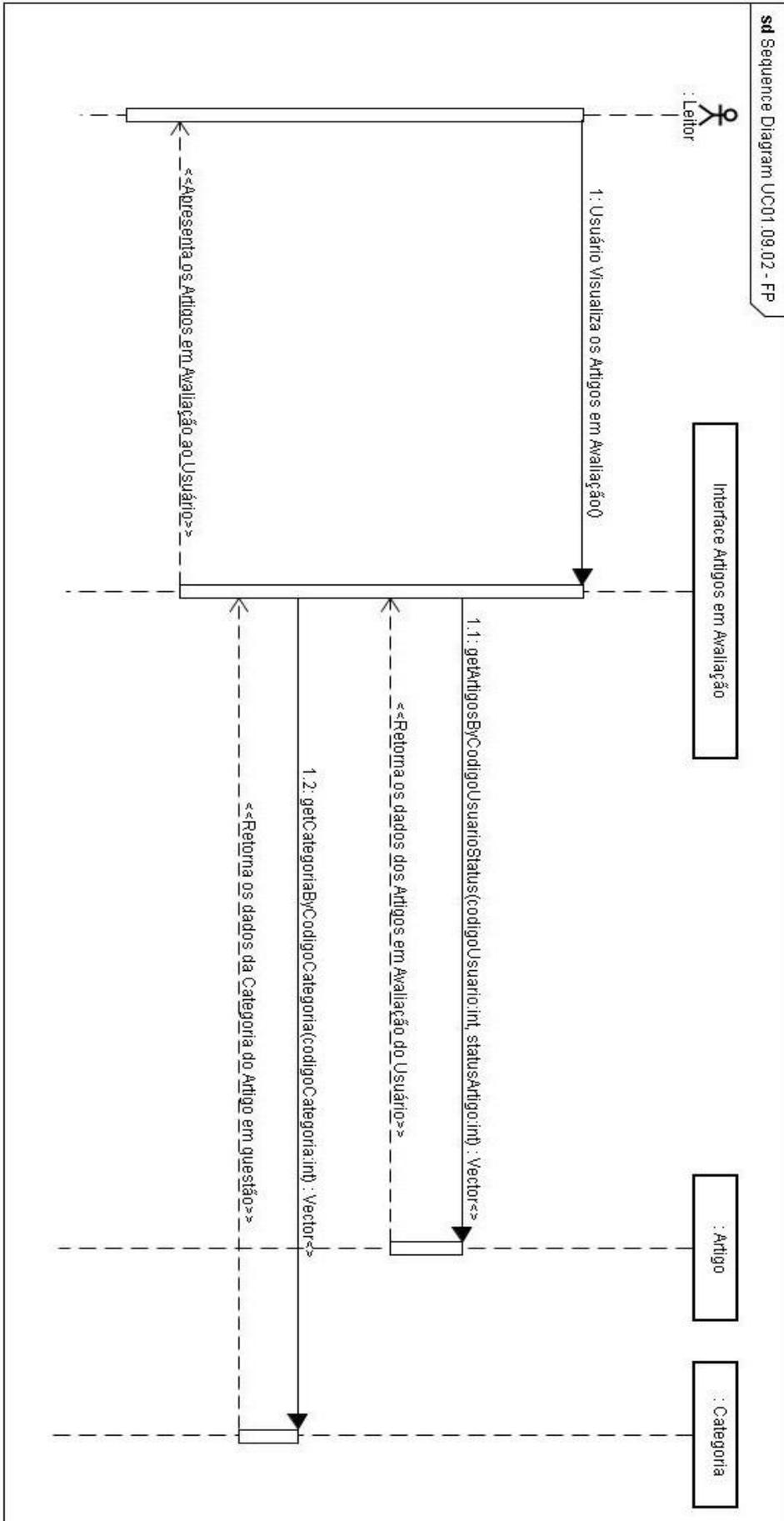
## 7.7.24 DS01.09 – Visualizar Meus Artigos - FP



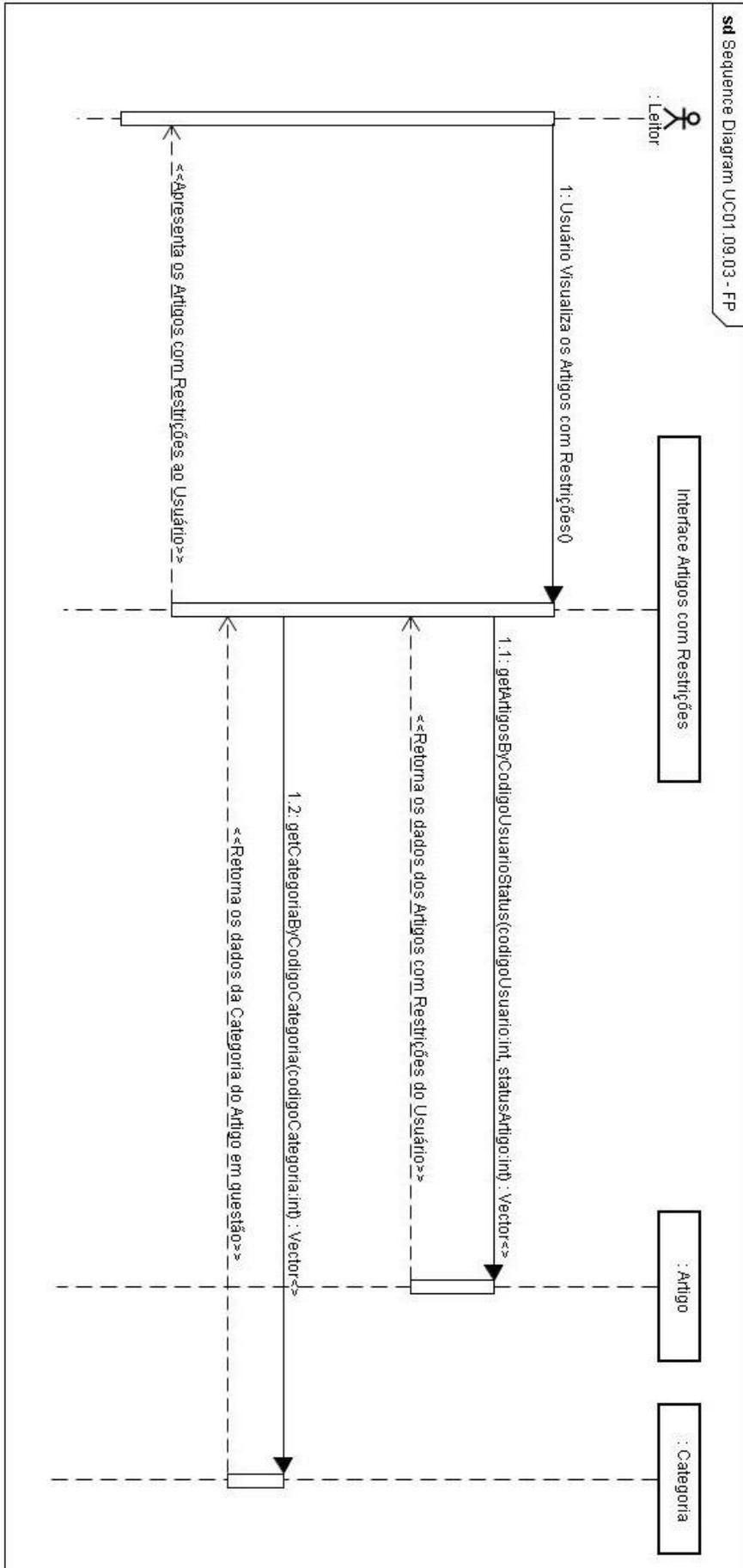
7.7.25 DS01.09.01 – Escrever Artigo - FP



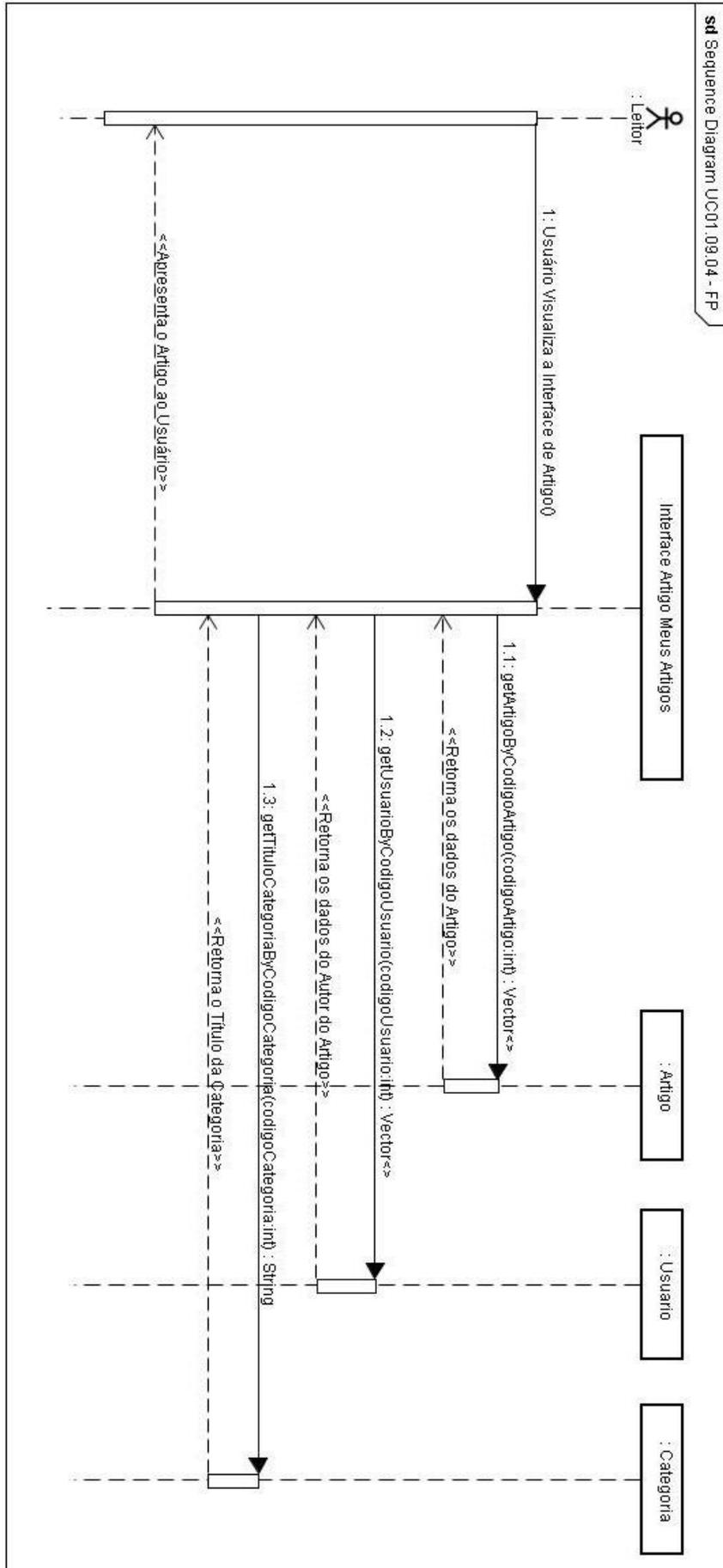
## 7.7.26 DS01.09.02 – Ver Artigos em Avaliação - FP



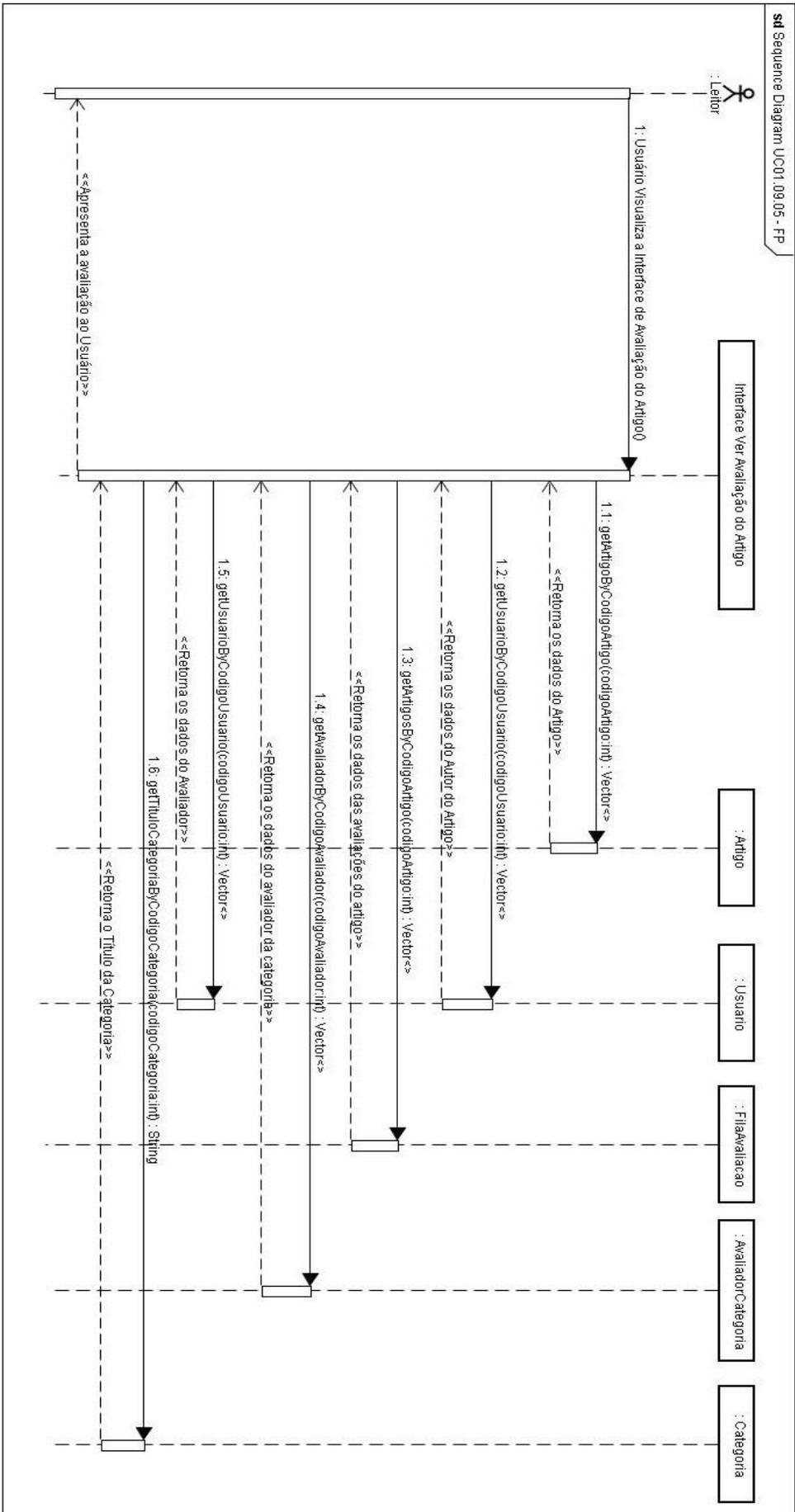
### 7.7.27 DS01.09.03 – Ver Artigos com Restrições - FP



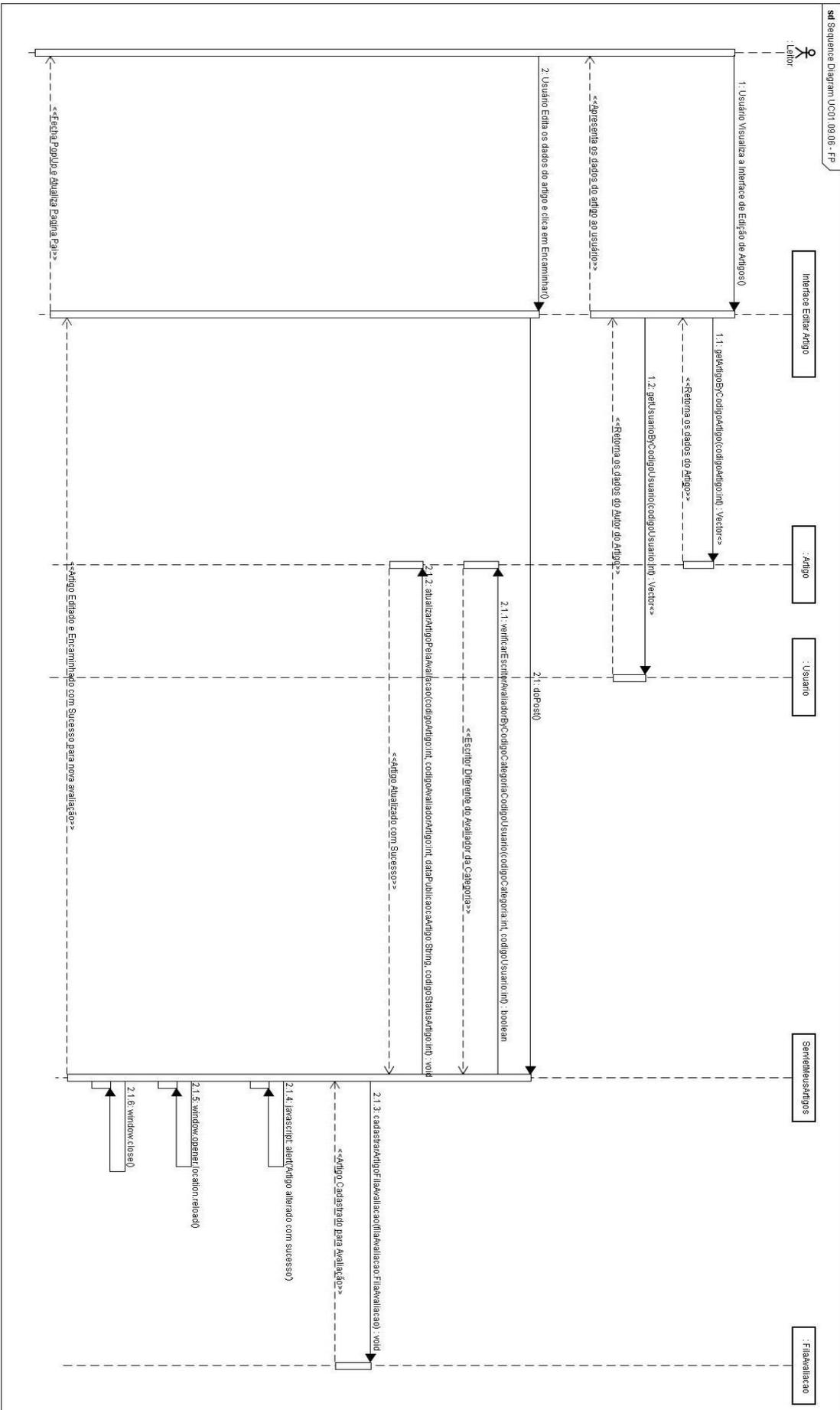
### 7.7.28 DS01.09.04 – Visualizar Artigo Meus Artigos - FP



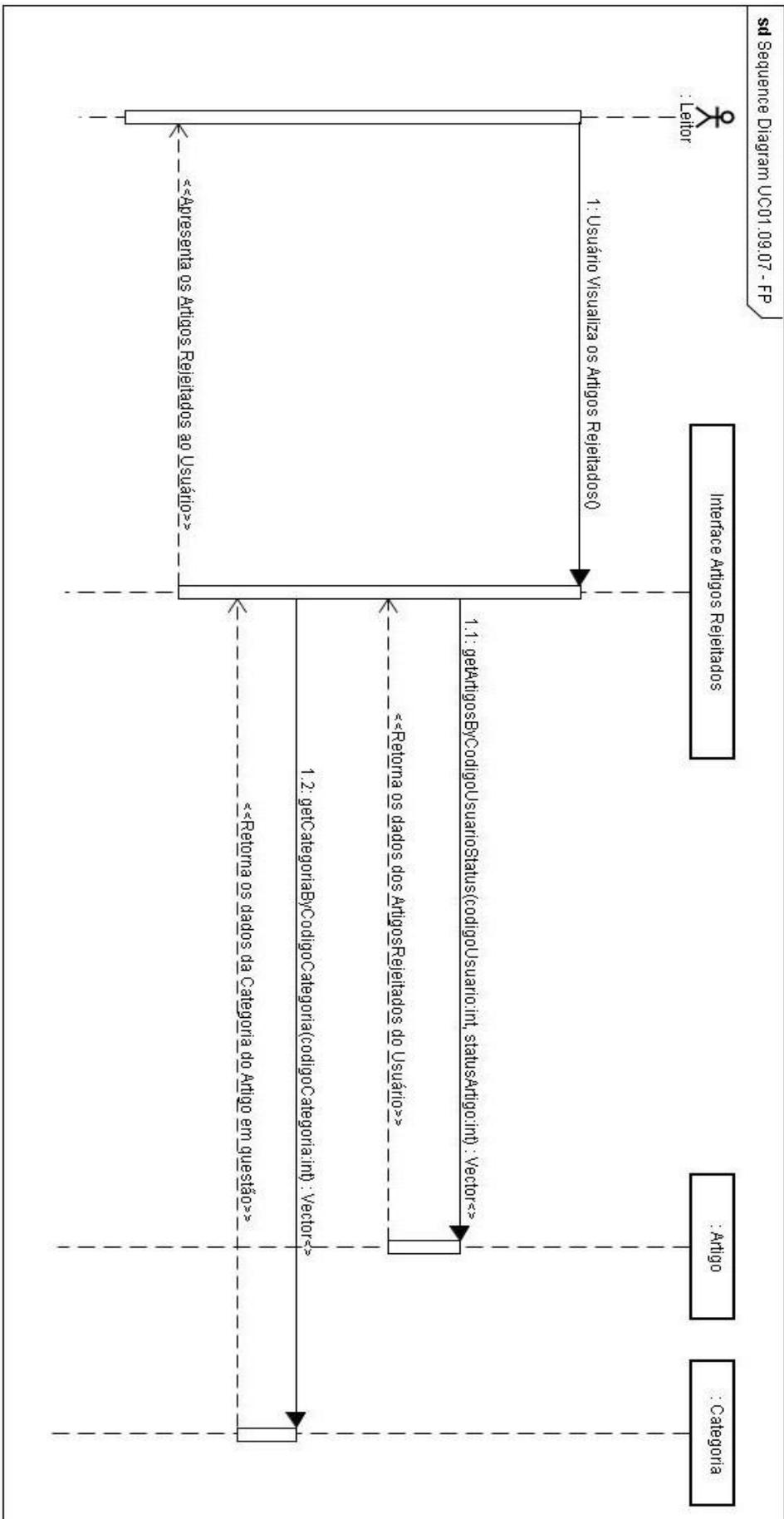
7.7.29 DS01.09.05 – Ver Avaliação do Artigo - FP



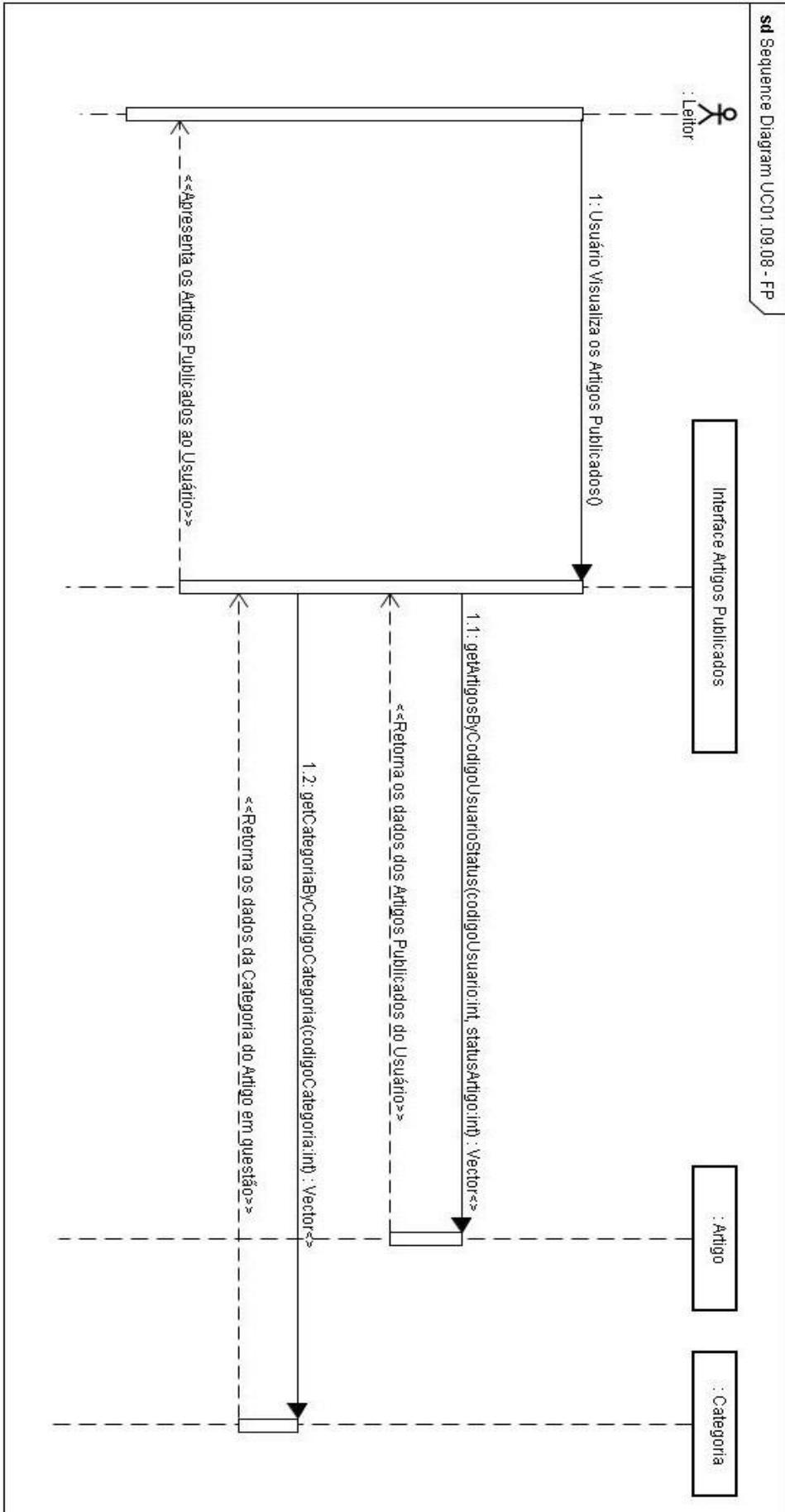
7.7.30 DS01.09.06 – Editar Artigo - FP



## 7.7.31 DS01.09.07 – Ver Artigos Rejeitados - FP

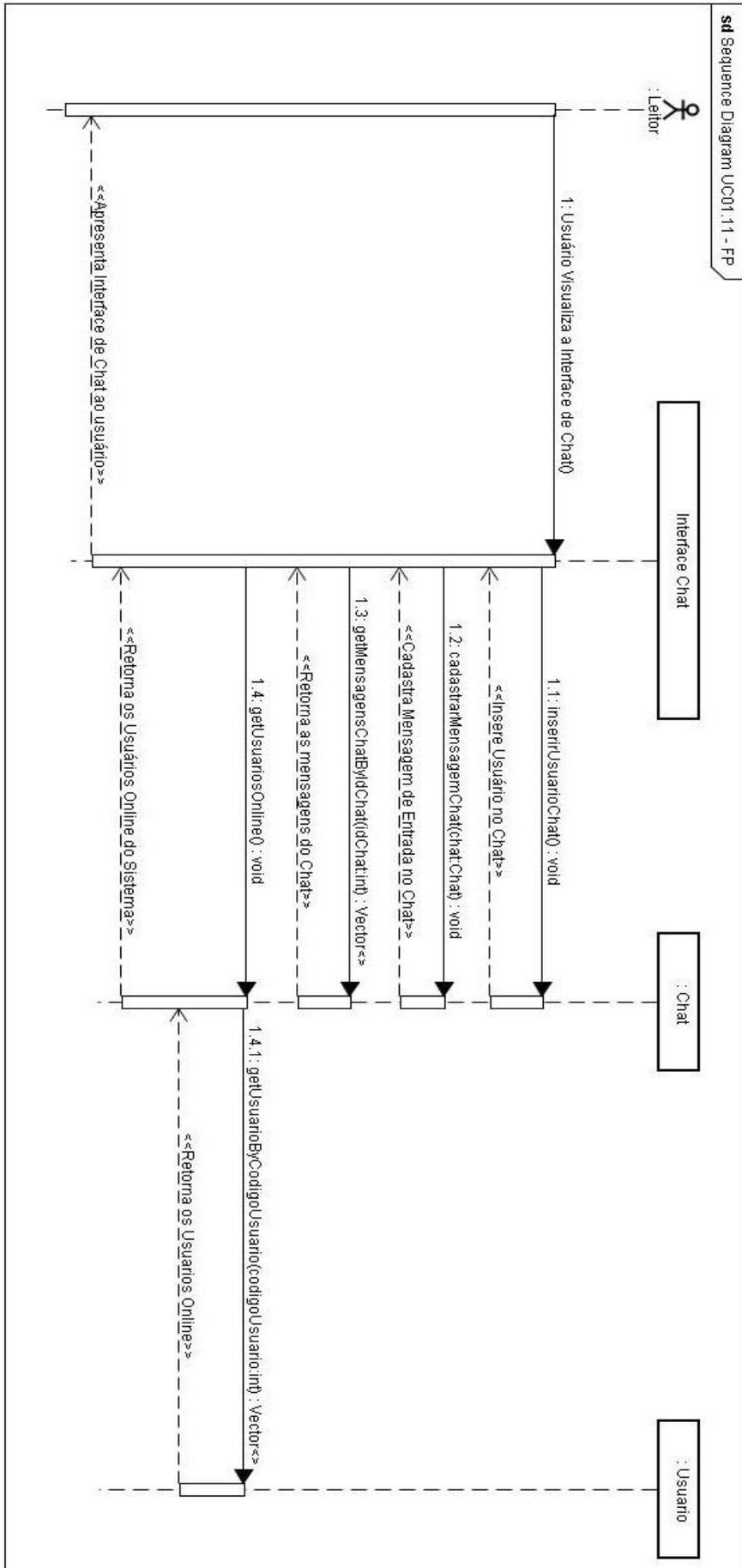


### 7.7.32 DS01.09.08 – Ver Artigos Publicados - FP

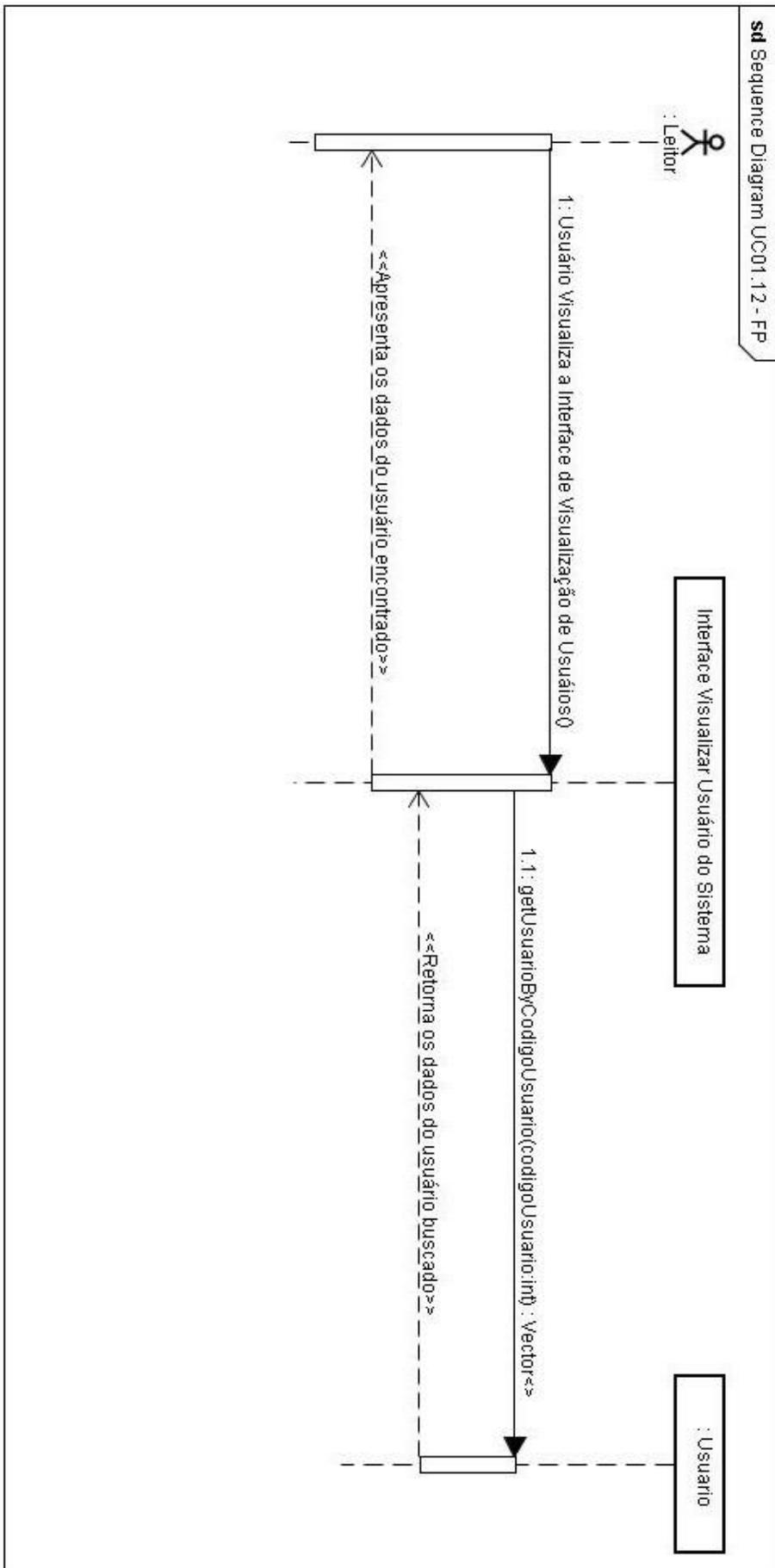




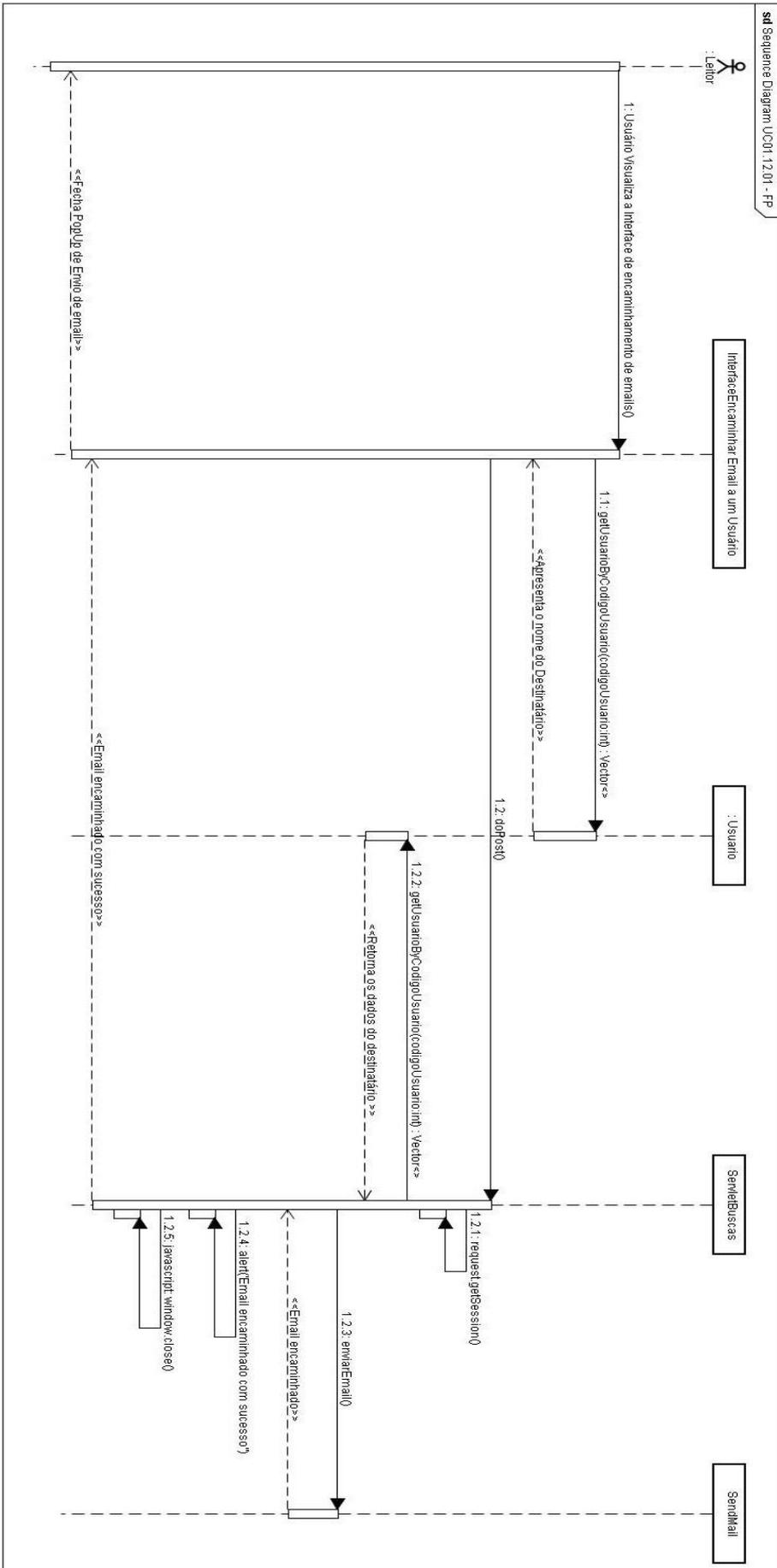
## 7.7.34 DS01.11 – Entrar no Chat do Site - FP



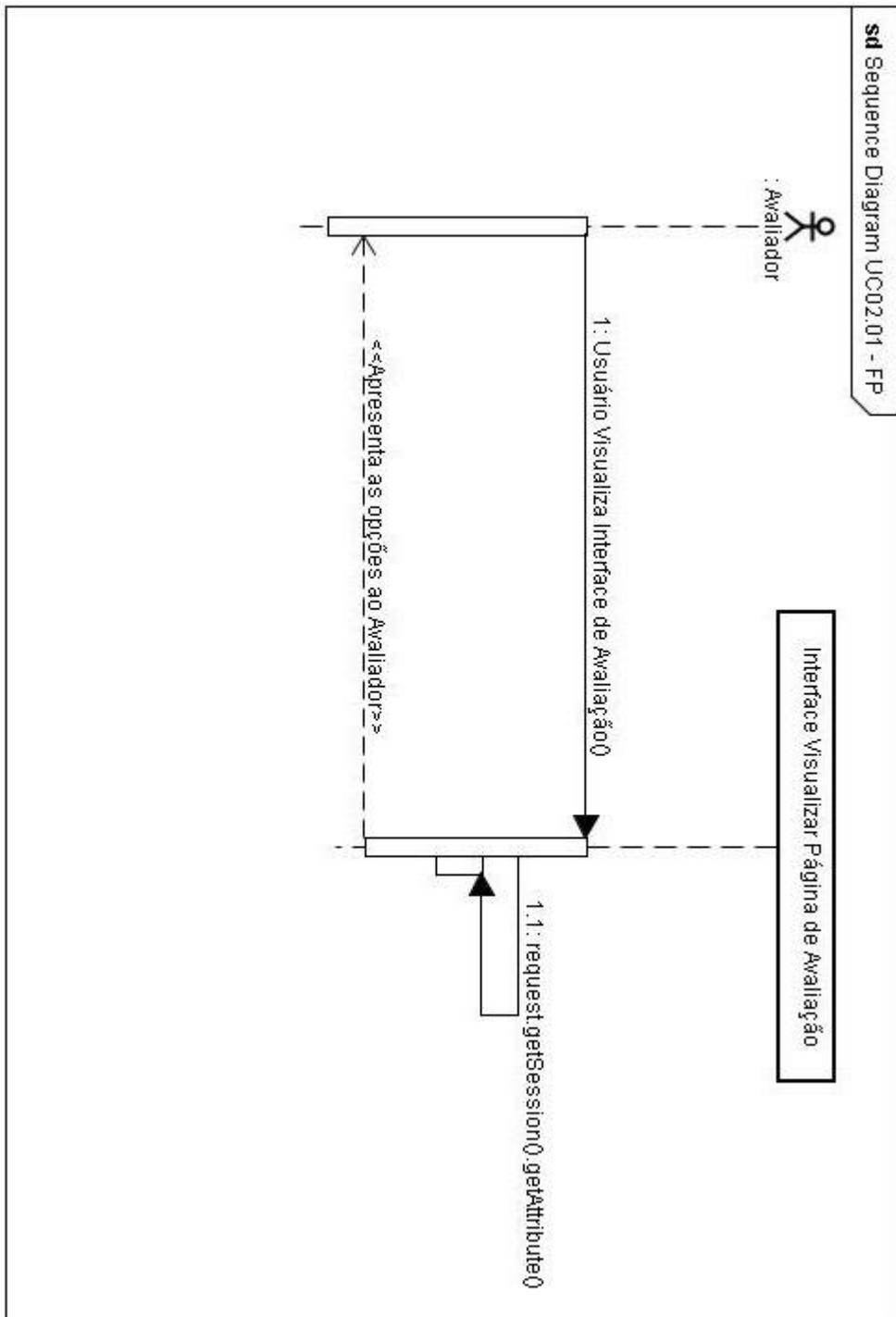
## 7.7.35 DS01.12 – Visualizar Usuário do Sistema - FP



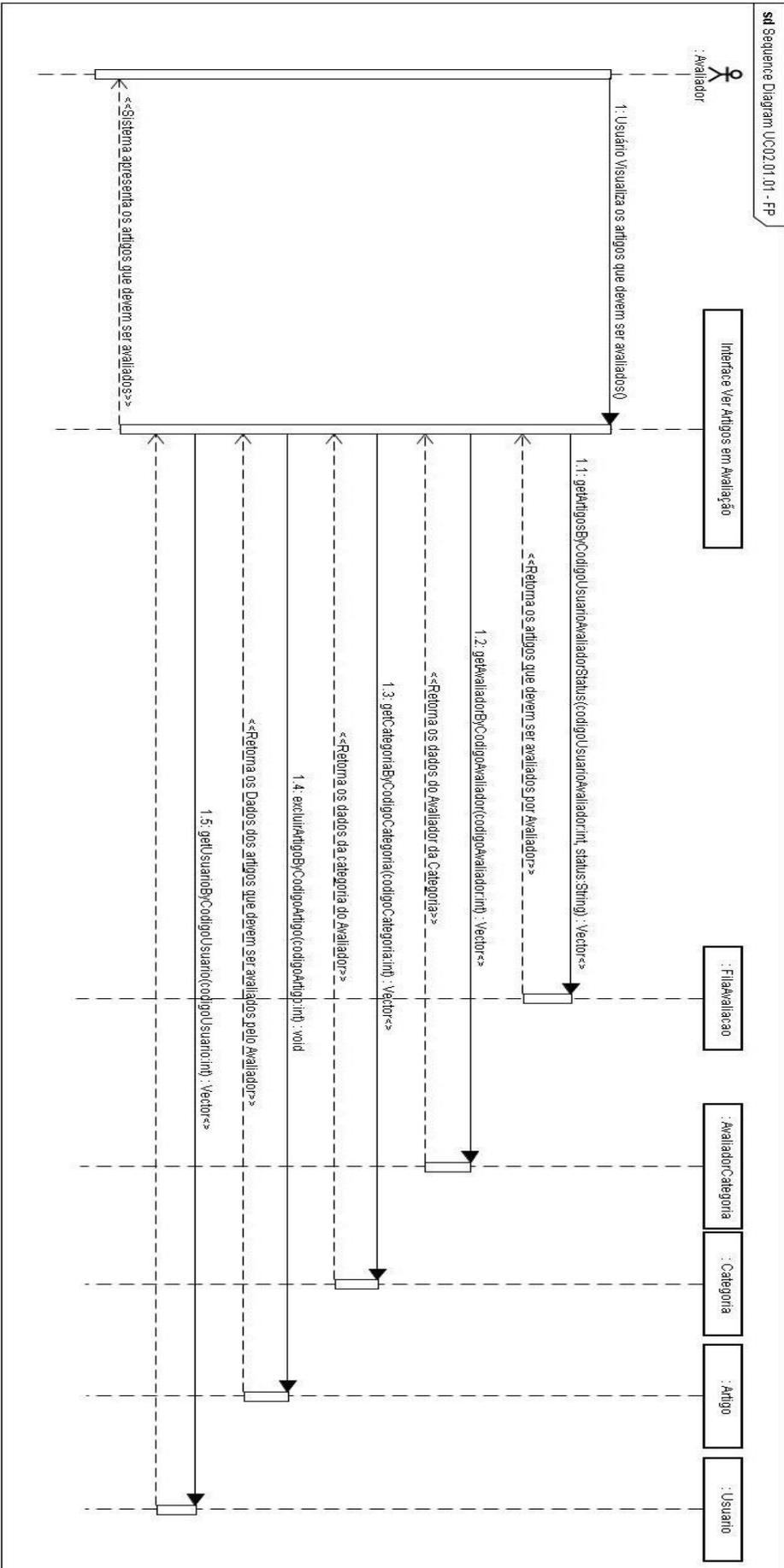
7.7.36 DS01.12.01 – Encaminhar email a um usuário - FP



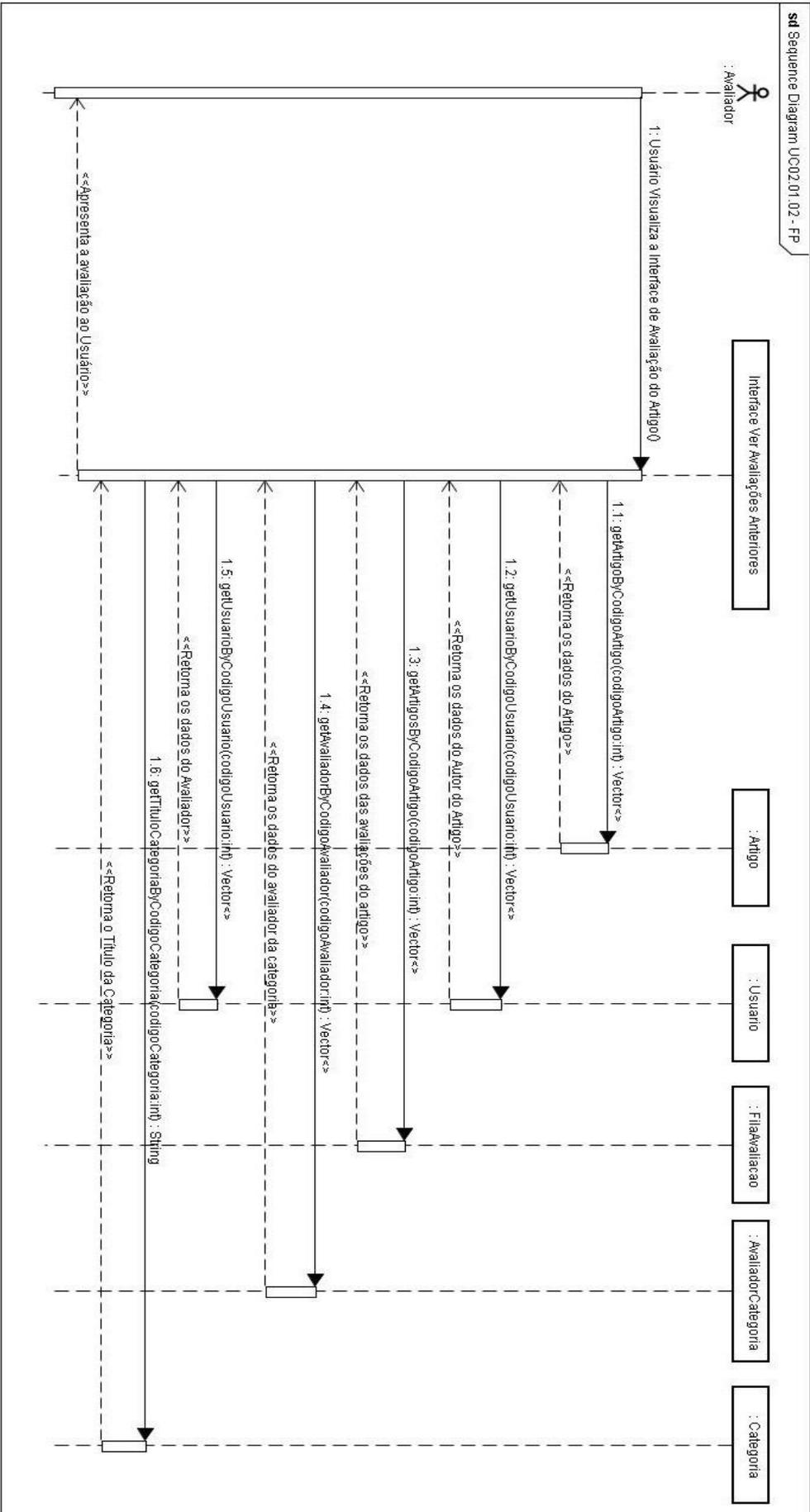
## 7.7.37 DS02.01 – Visualizar Página de Avaliação - FP



7.7.38 DS02.01.01 – Ver Artigos em Avaliação - FP

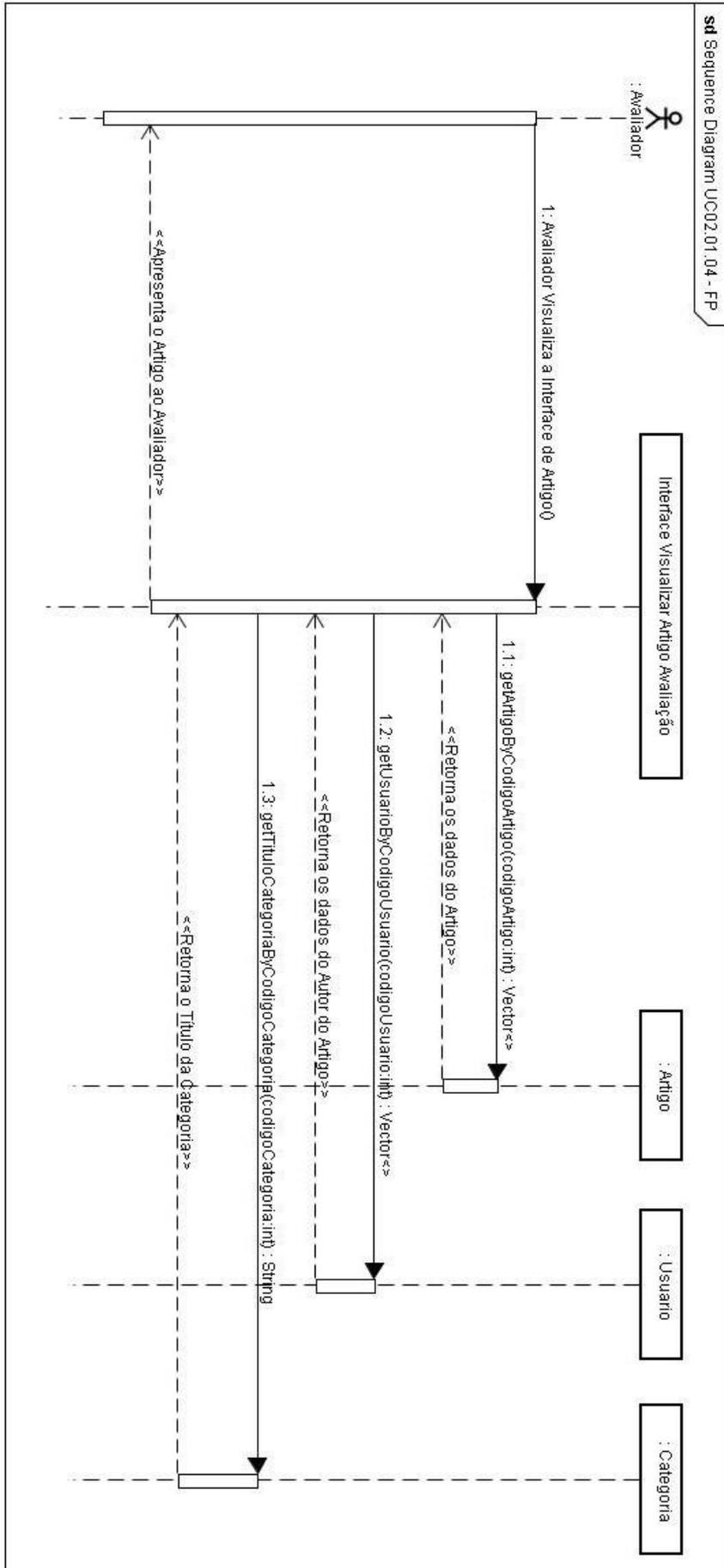


7.7.39 DS02.01.02 – Ver Avaliações Anteriores - FP

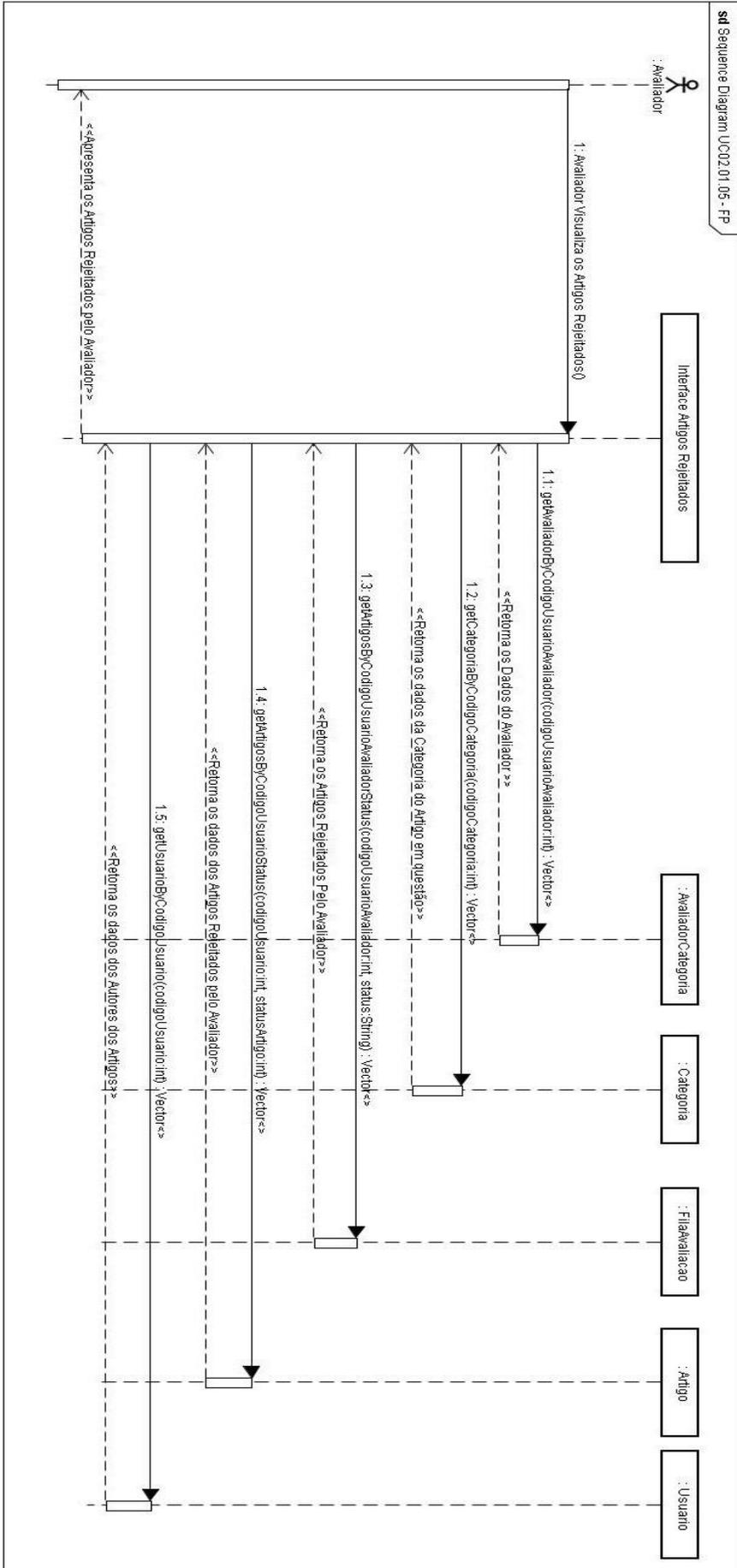




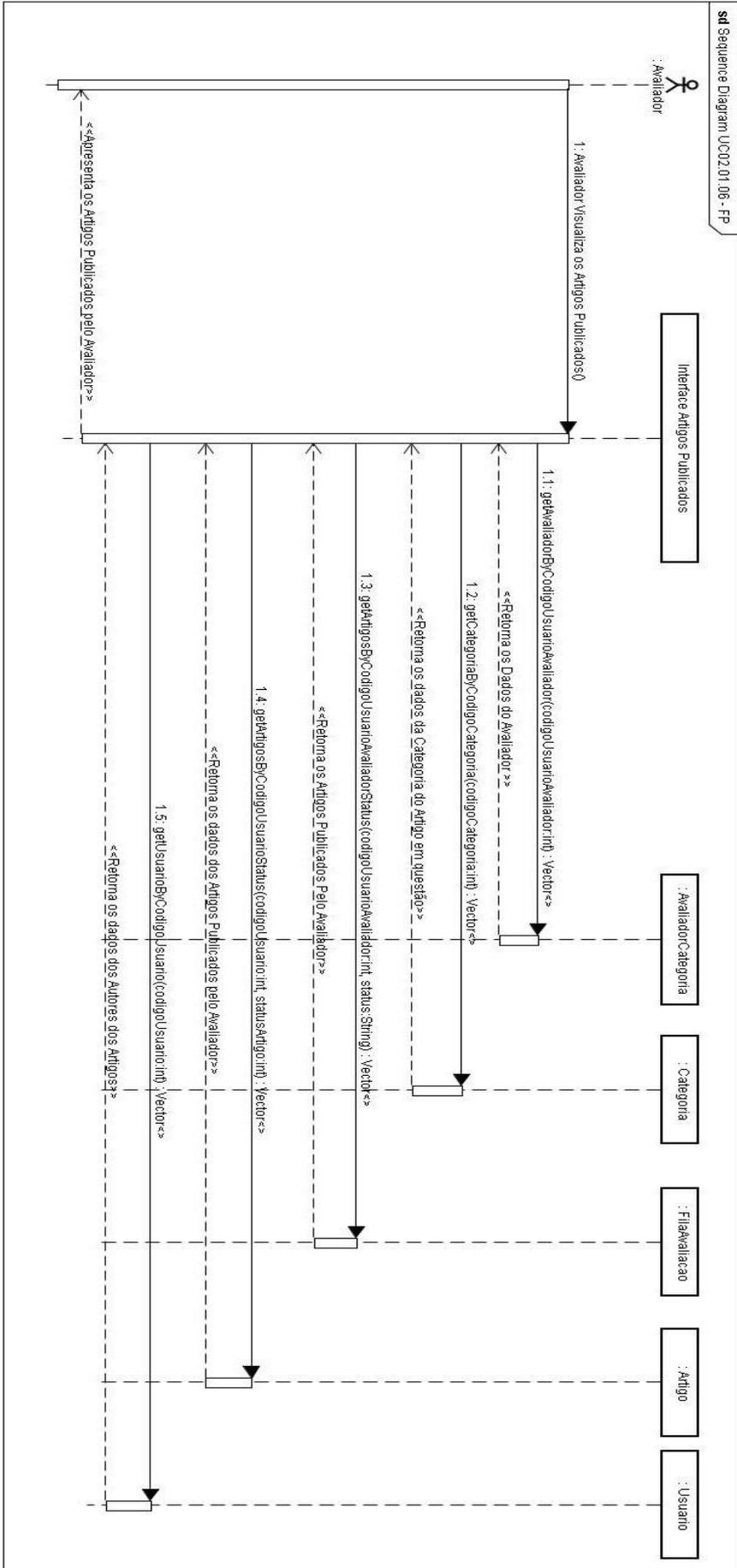
7.7.41 DS02.01.04 – Visualizar Artigo Avaliação - FP



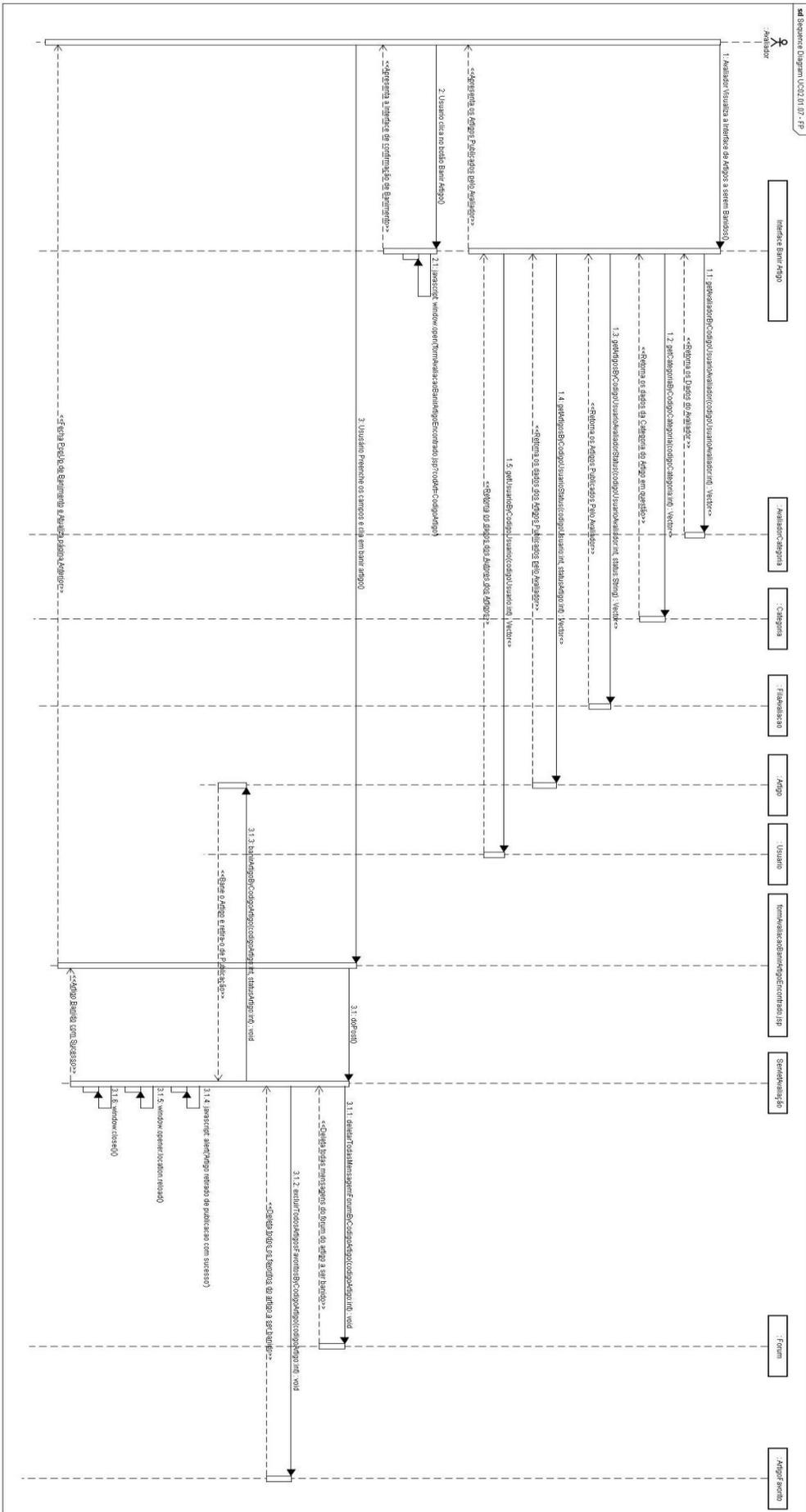
### 7.7.42 DS02.01.05 – Ver Artigos Rejeitados - FP



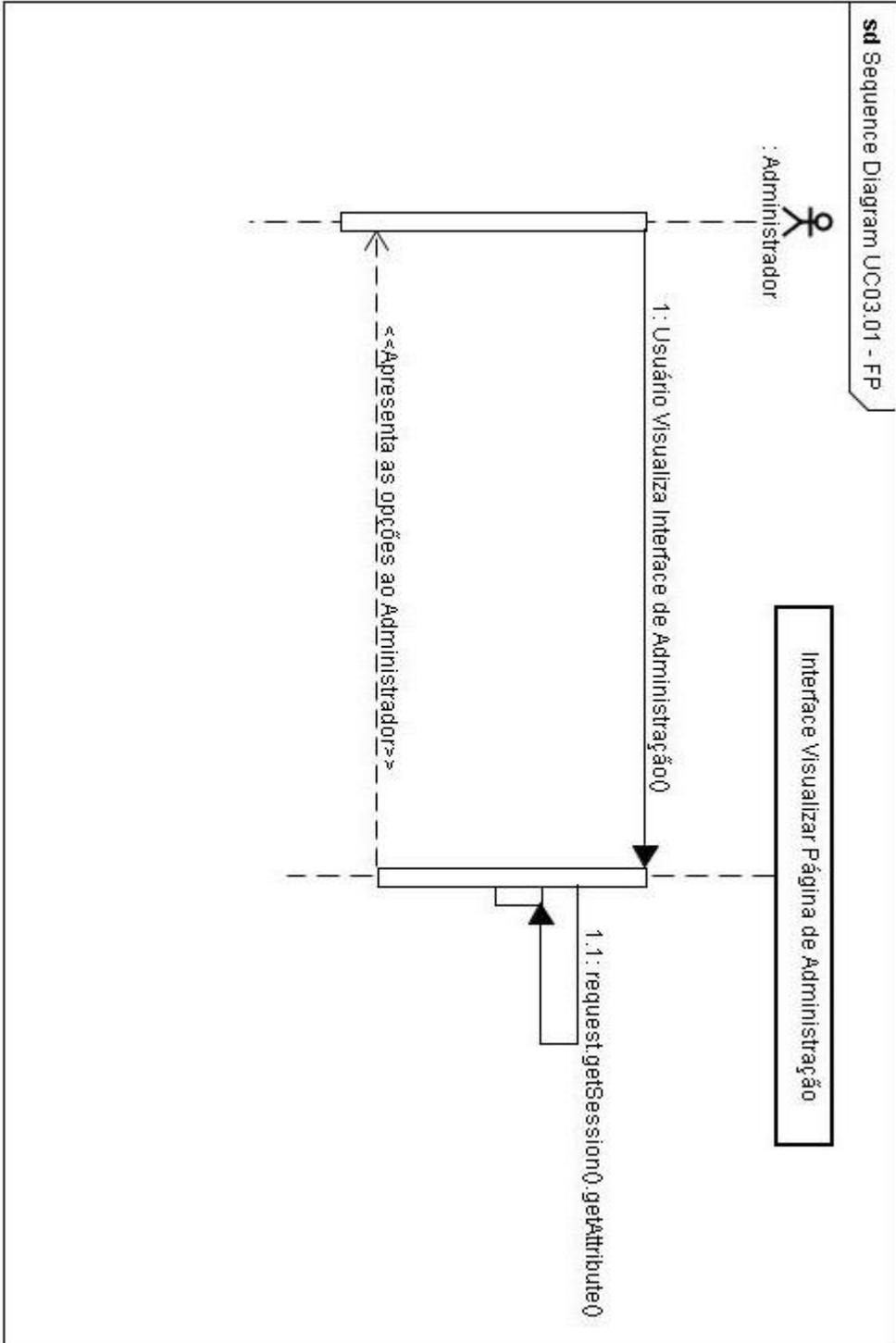
### 7.7.43 DS02.01.06 – Ver Artigos Publicados - FP



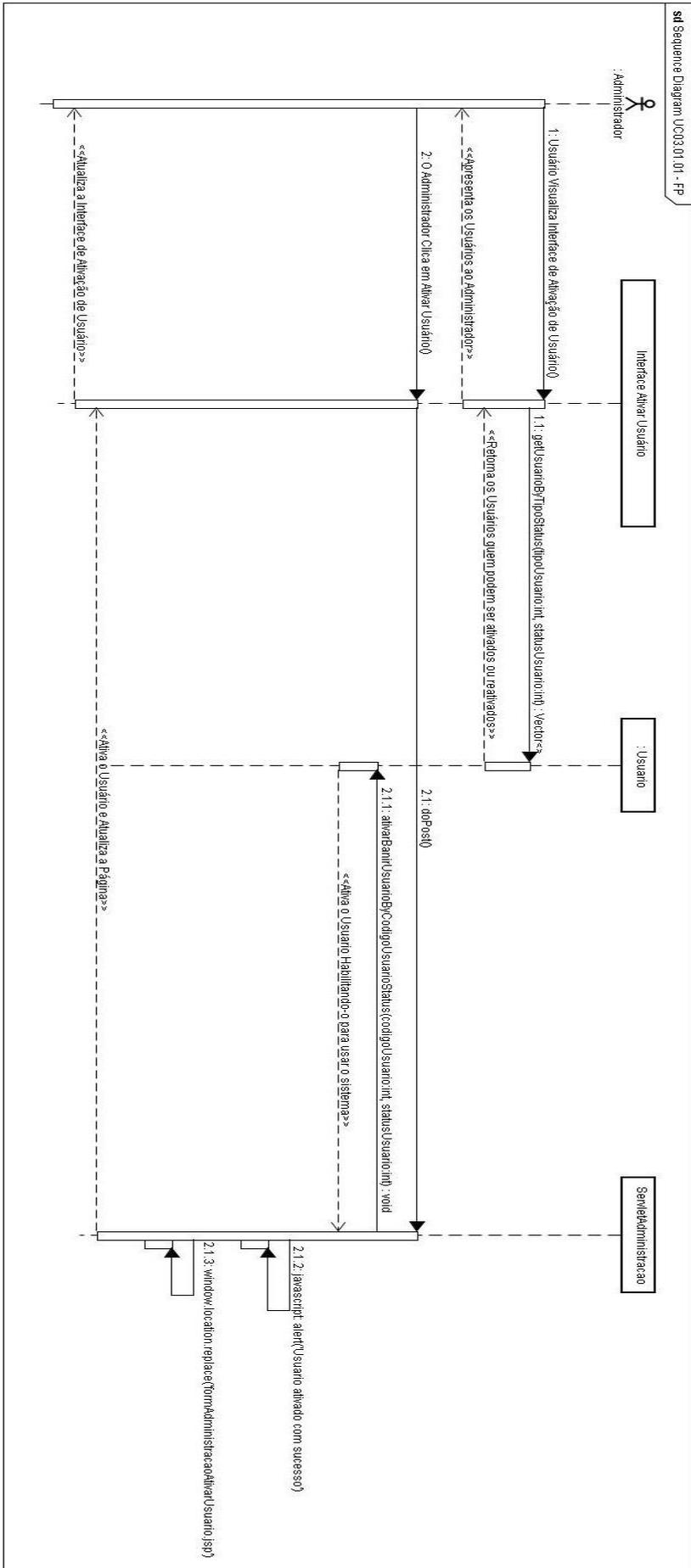
7.7.44 DS02.01.07 – Banir Artigo - FP



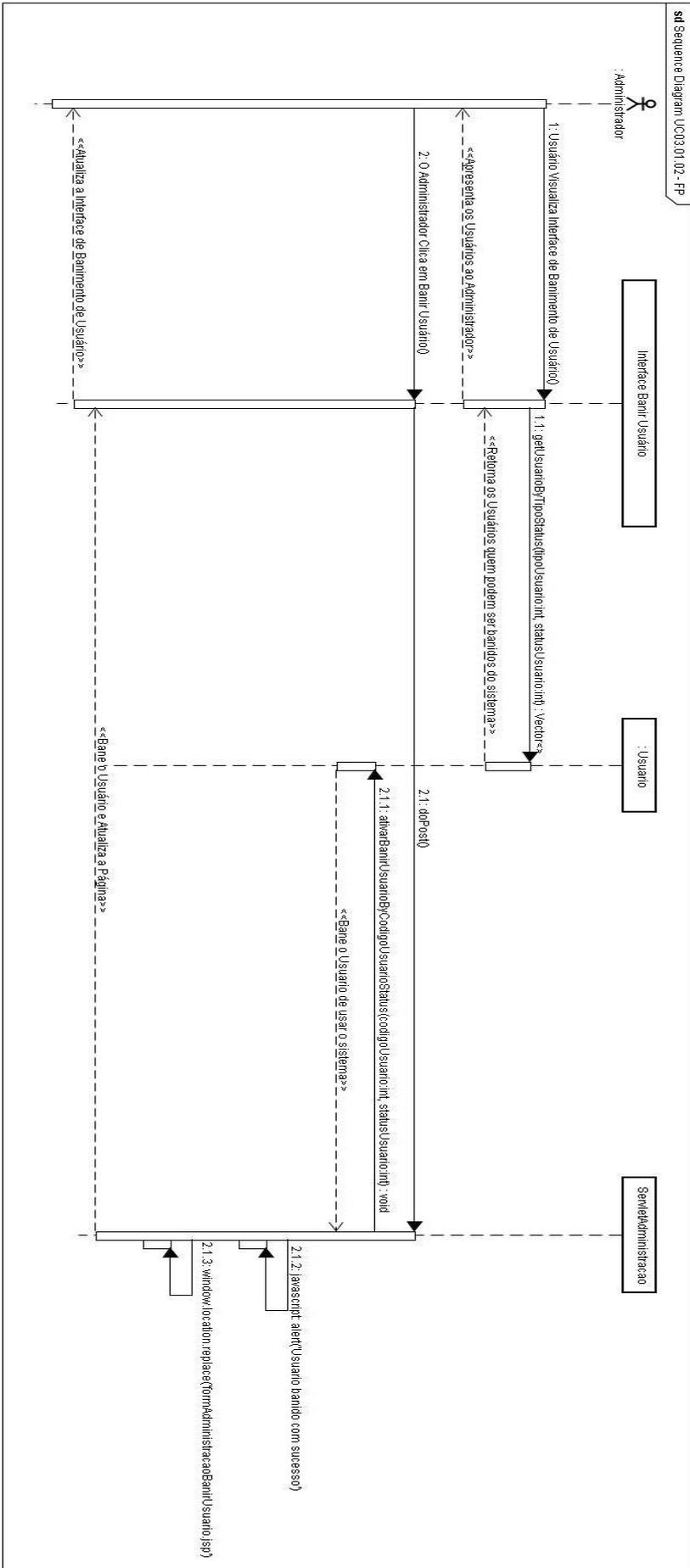
### 7.7.45 DS03.01 – Visualizar Página de Administração - FP



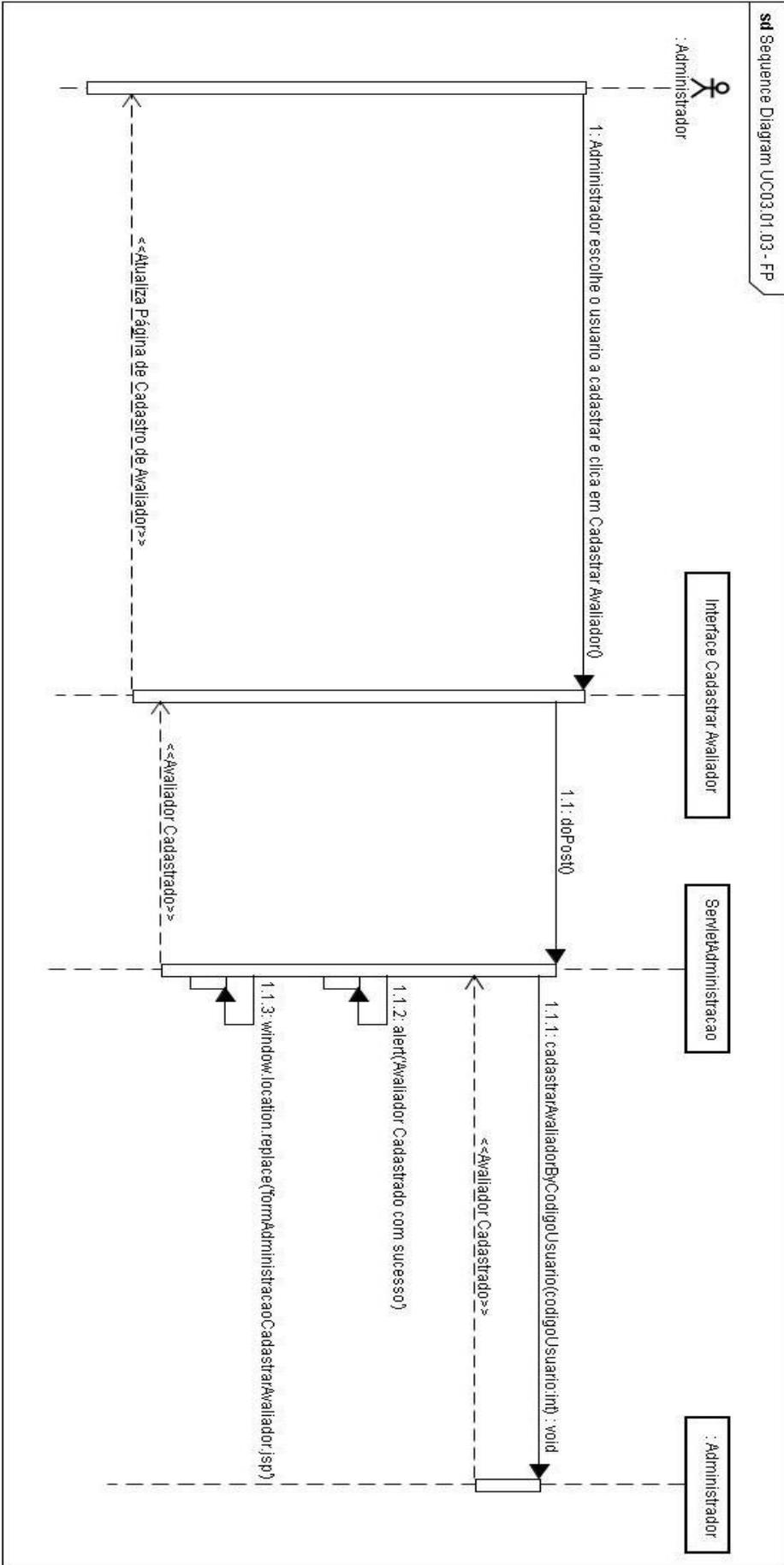
7.7.46 DS03.01.01 – Ativar Usuário - FP



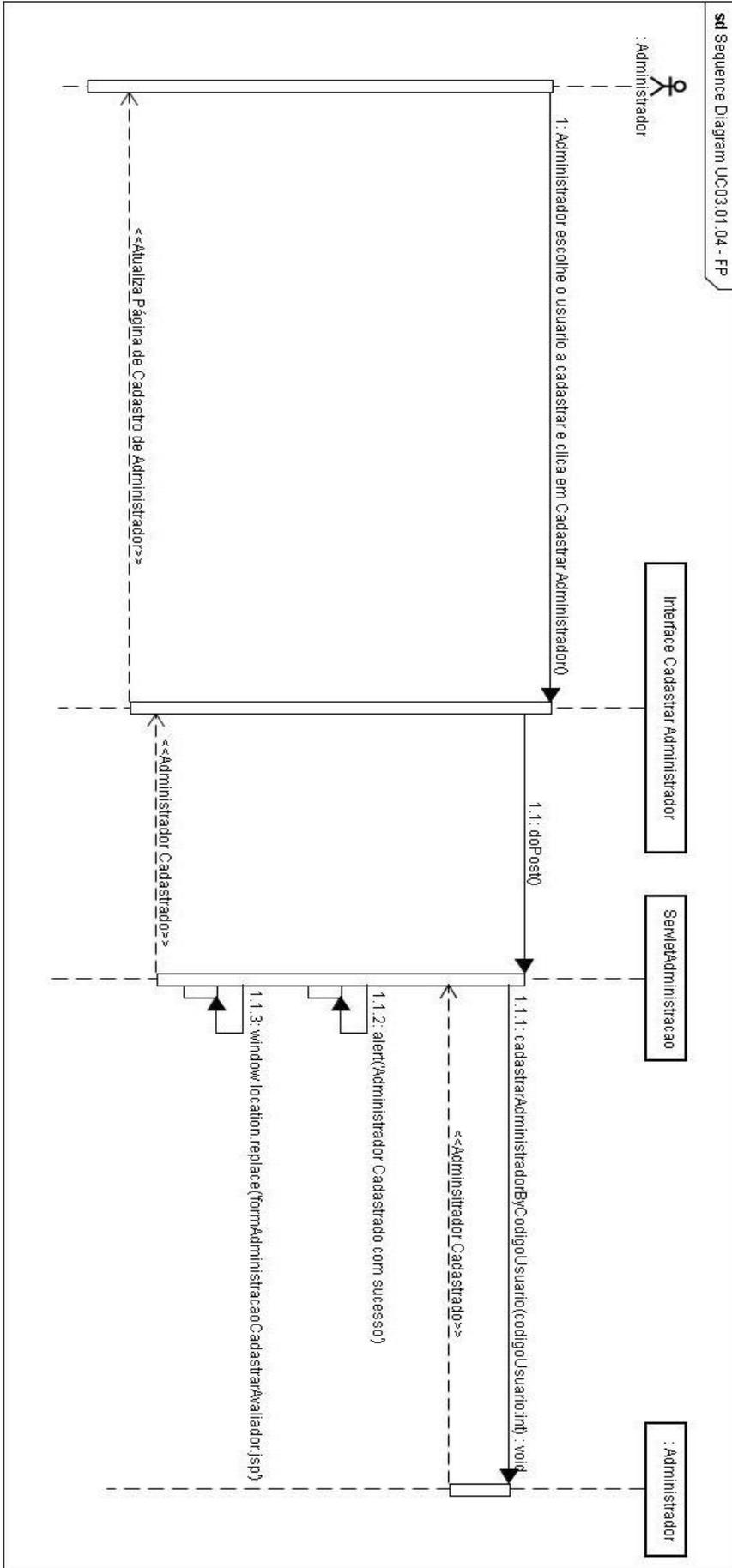
7.7.47 DS03.01.02 – Banir Usuário - FP



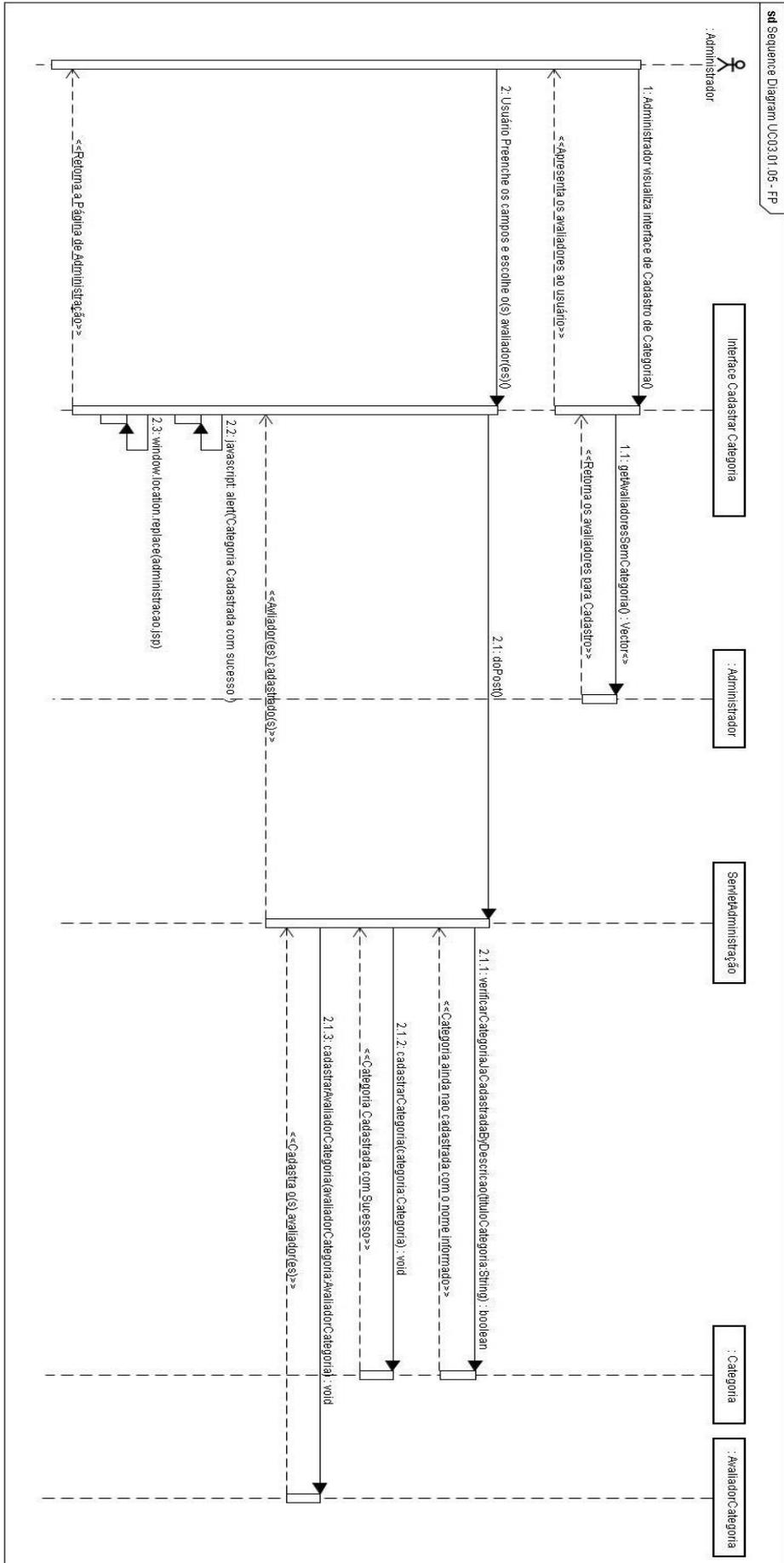
### 7.7.48 DS03.01.03 – Cadastrar Avaliador – FP



7.7.49 DS03.01.04 – Cadastrar Administrador - FP



7.7.50 DS03.01.05 – Cadastrar Categoria - FP



## 8. CONCLUSAO

O principal objetivo do trabalho, além do desenvolvimento de um sistema gerenciador de artigos utilizando a tecnologia JavaWeb, foi mostrar a eficiência de se empregar a Engenharia de Software, mais precisamente o RUP, no acompanhamento do projeto.

O RUP mostrou ser uma metodologia que auxilia e muito o andamento do trabalho, não apenas na parte de modelagem e especificações, mas também no desenvolvimento. Destaco os documentos de regras e visão de negócios, que quando bem elaborados e estruturados, ajudam o analista e programador durante todas as disciplinas e etapas conseguintes, mostrando claramente os principais termos e definições que serão empregados no software a ser desenvolvido.

A principal dificuldade em se empregar a metodologia é a grande quantidade de documentos e artefatos que devem ser gerados, podendo assim ocasionar em um atraso no cronograma e entrega do Projeto. As fases e disciplinas do RUP são extensas e requerem certo grau de atenção e disponibilidade, para que se consiga atingir um resultado eficiente à frente.

Apesar de todo esse esforço, o RUP mostrou ser a ferramenta ideal para o acompanhamento de desenvolvimento de projetos de software, pois mesmo com sua estrutura pesada e de difícil aceitação por parte de muitos usuários, quando bem aplicado, o RUP produz um resultado sem igual, empregando objetivos bem específicos e eficientes no sistema desenvolvido e gerando documentos que auxiliam e ajudam nas especificações da documentação do software quando se faz necessária à realização de manutenções e releases futuros.

## REFERENCIAS BIBLIOGRÁFICAS

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML - Guia do usuario segunda edição**: o mais avançado tutorial sobre Unified Modeling Language (UML). Rio de Janeiro: Campus, 2005.

DALL'OGGIO, Pablo. **PHP: Programando com Orientação a Objetos** (Inclui Design Patterns). São Paulo: Novatec, 2007.

FERREIRA, Aline Rodrigues. **Contribuição ao Estudo da Sumarização Automática de Textos**: relações semânticas entre elementos textuais. Dissertação de Mestrado. Departamento de Computação - PPGIA, PUC-PR, 2004.

FURLAN, José Davi. **Modelagem de objetos através da UML – The Unified Modeling Language**. São Paulo: Person Makron Books, 1998.

GONÇALVES, Edson. **Desenvolvendo aplicações Web com JSP, Servlets, JavaServer Faces, Hibernate, EJB3 Persistence e Ajax**. Rio de Janeiro: Ciência Moderna Ltda, 2007.

GUEDES , G. T. A. **UML: Uma Abordagem Prática**. 3.ed. São Paulo: Novatec Editora, 2008.

KRUCHTEN, Phillippe. **Introdução ao RUP - Rational Unified Process**. 2.ed. Rio de Janeiro: Ciência Moderna Ltda, 2003.

LARROCA, Joel Neto. **Contribuição ao Estudo de Técnicas para Sumarização Automática de Textos**. Dissertação de Mestrado. Departamento de Computação - PPGIA, PUC-PR, 2001.

PEREIRA, Alice T. Cybis. **Ambientes Virtuais de Aprendizagem – Em Diferentes Contextos**. Rio de Janeiro: Ciência Moderna Ltda, 2007.

POMPILHO, S. **Análise Essencial – Guia Prático de Análise de Sistemas**. Rio de Janeiro: Infobook, 1994.

PRESSMAN, Roger S. **Engenharia de Software**. 6.ed. São Paulo: Mcgraw-Hill, 2006.

SALTON, G; Buckley, C. **Term-weighting approaches in automatic text retrieval. Information Processing and Management**. 24, 513-523 1988. Reprint in "Readings in Information Retrieval", Karen Spark-Jones and Peter Willet, Morgan-Kaufmann 1997, pp.323-328.

## ANEXO – PLANO GLOBAL DO PROJETO

### 1. TEMA DE ABERTURA

#### Nome do Projeto

SISTEMA GERENCIADOR DE ARTIGOS

#### Gerente do Projeto

Designou-se, para este projeto, como gerente Sérgio Patrique Zotto.

#### Introdução

Para conclusão do curso de pós-graduação lato sensu em Engenharia de Software, será desenvolvido um projeto para elaboração de um sistema que gerencie e controle a edição de artigos eletrônicos.

O tema foi escolhido devido à dificuldade em se encontrar sistemas de matérias publicadas que tenham uma interface de fácil utilização e amigável. As principais funcionalidades do sistema serão:

- Cadastro de Artigos separados por categoria
- Sistema de avaliação de artigos
- Mecanismo de busca inteligente
- Sistema de notícias por categoria
- Fórum de discussões por artigo

#### Objetivo ou Justificativa do Projeto

Este projeto tem como objetivo a criação de um novo sistema de informação, que contemple as funcionalidades de controle e gerenciamento de artigos, de acordo com as necessidades específicas do trabalho de conclusão.

#### Marcos do Cronograma do Projeto

Atividade	Data inicial estimada	Data final estimada
Adaptação à linguagem JAVA	20/08/2008	20/12/2008
Levantamento de Requisitos	20/08/2008	30/10/2008

Análise e Design	01/11/2008	31/01/2009
Desenvolvimento	08/01/2009	30/05/2009
Testes	01/06/2009	30/06/2009
Entrega	15/07/2009	30/07/2009

**Quadro 1 – Cronograma básico do projeto**

### **Resumo do Orçamento Estimado**

Devido o motivo de o sistema ser de desenvolvimento acadêmico, sem fins lucrativos, estima-se um investimento de R\$ 600,00 durante todo o desenvolvimento deste projeto por parte do Gerente, para aquisição de livros, despesas quanto transporte, luz e outros custos, podendo este valor ser suscetível a variações a serem devidamente informadas.

### **Grau de Influência dos Stakeholders**

Como partes interessadas no projeto destacam-se:

- O gerente do projeto, na pessoa do Sr. Sérgio Patrique Zotto, que tem por objetivo cumprir todas as metas propostas para conclusão do sistema e obtenção do título de especialista. Ele será o responsável pelo desenvolvimento de todas as fases.
- O orientador do projeto, na pessoa do Professor Jaime Wojciechowski, que irá instruir todo o andamento do projeto e ajudar para que o mesmo finalize com qualidade.
- Os outros professores do curso, para orientação dos mais diversos temas a serem desenvolvidos no projeto, bem como orientação para elaboração da monografia, linguagem a ser usada e RUP.

### **Premissas do Projeto**

Considera-se, para efeitos de planejamento e gerenciamento do projeto:

- Disponibilidade de 1 hora quinzenal, no mínimo, do orientador do projeto, para atualização, e discussão das etapas do projeto.
- Disponibilidade de atendimento quando necessário do Professor Douglas, para auxiliar nos assuntos referentes à linguagem Java, para o desenvolvimento do sistema.

- Disponibilidade de atendimento quando necessário do Professor Jaime, para esclarecimento e auxílio nos documentos referentes ao RUP.
- Disponibilidade de atendimento quando necessário da Professora Maria Valéria, para sanar dúvidas referentes à elaboração da monografia.
- Disponibilidade de atendimento quando necessário da Professora Rafaela Fontana, para auxílio referente à estrutura e gestão do projeto a serem desenvolvidos.
- Disponibilidade do laboratório e biblioteca da instituição quando necessários.

### **Restrições do Projeto**

Como restrição principal tem-se o prazo para entrega do sistema, no início do segundo semestre de 2009. Destaca-se, como outra restrição do projeto, a necessidade de aprendizagem da linguagem de programação JAVA, por ser requisito para o desenvolvimento do sistema.

### **Riscos Iniciais Conhecidos**

Consideram-se riscos do projeto, a demora para aquisição de conhecimento referente à linguagem de programação JAVA e assim o atraso no cronograma, e a quantidade de horas trabalhadas semanalmente, podendo ser insuficiente para uma conclusão de qualidade.

### **Limites do Projeto**

Por se tratar de um projeto acadêmico e sem fins lucrativos, o mesmo não contempla atividades de suporte ao usuário após a entrega.

### **Assinaturas**

Curitiba, 01 de setembro de 2008.

---

Orientador do Projeto  
PROFº JAIME WOJCIECHOWSKI

---

Representante da Equipe de Projeto  
SÉRGIO PATRIQUE ZOTTO

## 2. DEFINIÇÕES DO ESCOPO DO PRODUTO

### Descrição do Produto

#### SISTEMA GERENCIADOR DE ARTIGOS

O sistema gerenciador de artigos será desenvolvido segundo as necessidades informadas para elaboração do trabalho de conclusão de curso e deve contemplar as seguintes funcionalidades:

- **Cadastro de Usuários:** Os interessados no sistema de artigos devem previamente realizar um cadastro para poderem navegar no site. Esse cadastro está aberto a todas as pessoas e é através dele que o usuário informa previamente seus dados pessoais, que serão usados durante todas as suas ações no sistema. Após informar seus dados pessoais, o sistema enviará para o email do usuário uma confirmação indicando que ele realizou um cadastro no site. Apenas após o usuário realizar a efetuação de confirmação através do seu email é que o sistema ficará disponível para utilização por parte do usuário.
- **Cadastro de Artigos:** Todos os usuários leitores e avaliadores podem escrever artigos nos mais diversos temas abordados no site. Para cadastrar um artigo, o usuário deve primeiramente indicar a qual categoria que o mesmo pertence, e após irá escolher o tema do artigo e escrever o desenvolvimento.
- **Encaminhamento para Avaliação:** Após escrever um novo artigo, o usuário manda o artigo para avaliação, e o mesmo será enviado para um avaliador fazer a análise e dar o seu parecer sobre o artigo.
- **Escolha dos Avaliadores:** O administrador do site é quem cria as categorias e escolhe quem serão os seus avaliadores. Cada categoria deve ter no mínimo um e no máximo três avaliadores, assim ao se criarem categorias, o administrador deve informar quem são os avaliadores respectivos a aquela categoria. Cada avaliador pode realizar a avaliação também de apenas uma categoria.
- **Quantidade de Avaliação:** Cada avaliador deve receber no máximo cinco artigos para avaliação, assim sendo, se um avaliador tiver 5 artigos em avaliação, o próximo artigo escrito para avaliação será encaminhado para

outro avaliador da categoria, e se este também estiver com 5 artigos, o artigo entrará em uma fila de espera esperando até que um dos avaliadores fique com pelo menos 4 artigos.

- **Avaliação de Artigos:** O avaliador abre o artigo para avaliação e após análise, escolhe uma nota do artigo e depois seleciona uma das opções de avaliação “Publicar Artigo / Retornar Para Revisão / Cancelar Artigo”, e após escolhe uma chamada “Vista de Artigo” que é um breve comentário do avaliador sobre o artigo. Os artigos que forem escolhidos com “Publicar Artigo” vão para publicação imediata, os escolhidos como “Retornar para Revisão” devem ser acrescidos de um comentário do avaliador e após, voltam para o autor do artigo realizar as alterações. Os artigos selecionados com “Cancelar Artigo” são cancelados momentaneamente e não podem mais ser alterados.
- **Edição de Artigos:** Um artigo avaliado e finalizado com a opção de “Retornar Para Revisão” pelo avaliador retorna ao usuário para eventuais alterações. O autor pode fazer as mudanças necessárias no artigo e reencaminhá-lo para uma nova avaliação.
- **Meus Favoritos:** Todos os usuários podem ao encontrar um artigo de seu interesse, incluí-los na opção de “Meus Favoritos”, assim, toda vez que o usuário realizar o login no sistema, na página principal, os artigos escolhidos como meus favoritos aparecerão para o usuário sem a necessidade de o mesmo ter de procurá-los no sistema.
- **Notas do Artigo:** Cada usuário pode dar uma nota variando de 0 - 10 por artigo, assim será feita uma média geral de notas de cada artigo publicado. Cada usuário pode dar apenas uma nota por artigo.
- **Fórum de Discussões:** Cada artigo terá um fórum de discussões onde os usuários poderão trocar informações a respeito da matéria publicada e também sanar dúvidas a respeito do artigo.
- **Banir Usuários:** Função específica do Administrador do sistema. Ele pode realizar o banimento de usuários que desrespeitarem as regras e normas do fórum. Usuários banidos nunca mais podem retornar ao sistema.

- **Cadastro de Administrador:** Apenas administradores podem cadastrar novos administradores. O cadastro de um novo administrador pode ocorrer apenas se o usuário não possuir nenhum artigo publicado.

Por mais simples que se apresente, o sistema tende a ter conceitos de Ambientes Virtuais de Aprendizagem, pois visa proporcionar uma melhor forma de exposição e controle de conhecimento.

É importante salientar ainda que o sistema estará disponível em ambiente web, podendo ser acessado através da internet.

A Figura 1 apresenta as funcionalidades gerais a serem oferecidas pelo sistema, bem como as possibilidades de uso do sistema de cada tipo de usuário.



Figura 1 - Definição das funcionalidades do Sistema de Gerenciamento de Artigos

## 2.1 Especificações de Casos de Uso

### 2.1.1 UC01 - REALIZAR CADASTRO NO SISTEMA

Especificação de Caso de Uso  
**UC01 – Realizar Cadastro no Sistema**

#### Controle do Documento

Versão	Autor	Data	Descrição
1.0	Sérgio Patrique Zotto	27/08/2008	Elaboração

#### Descrição

Este caso de uso serve para descrever o cadastro de usuários no sistema

#### Pré-condições

Este caso de uso pode iniciar somente se:

1. O usuário tiver acessado o sistema web de gerencia de artigos

#### Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter cadastrado um novo usuário e enviado por email uma confirmação de cadastro finalização do mesmo.

#### Ator Primário

Leitor

#### Fluxo de Eventos Principal

1. Sistema apresenta a tela de cadastro de usuários
2. O usuário preenche os campos com os dados pessoais
3. O usuário clica no botão “Salvar”.

4. O sistema salva os dados do usuário e encaminha um email com a confirmação da inscrição
5. O sistema direciona o usuário para a página de “Cadastro realizado com sucesso, acesse seu email para efetivar sua participação no site”.
6. O caso de uso é finalizado.

### 2.1.2 UC02 - CADASTRAR ARTIGO

## Especificação de Caso de Uso

### UC02 – Cadastrar Artigo

#### Controle do Documento

Versão	Autor	Data	Descrição
1.0	Sérgio Patrique Zotto	27/08/2008	Elaboração

#### Descrição

Este caso de descreve os passos para os usuários cadastrarem novos artigos.

#### Pré-condições

Este caso de uso pode iniciar somente se:

1. O usuário tiver efetuado o login no sistema

#### Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ter cadastrado um novo artigo e salvo no menu “Artigos Salvos”.

#### Ator Primário

Leitor, Avaliador

#### Fluxo de Eventos Principal

1. O sistema apresenta a tela de cadastro de artigos
2. O usuário preenche os campos com os dados do artigo
3. O usuário clica no botão “Salvar”.
4. O sistema salva o artigo.
5. O sistema emite a mensagem “Artigo salvo com sucesso, verifique o menu ‘Artigos Salvos””.
6. O usuário seleciona o botão “ok”;
7. O sistema encaminha o usuário para sua home page.
8. O caso de uso é finalizado.

### 2.1.3 UC03 - ALTERAR ARTIGO

#### Especificação de Caso de Uso

#### **UC03 – Alterar Artigo**

#### **Controle do Documento**

<b>Versão</b>	<b>Autor</b>	<b>Data</b>	<b>Descrição</b>
1.0	Sérgio Patrique Zotto	27/08/2008	Elaboração

#### **Descrição**

Este caso de descreve os passos para os usuários editarem e alterarem artigos já escritos.

#### **Pré-condições**

Este caso de uso pode iniciar somente se:

1. O usuário tiver efetuado o login no sistema
2. O usuário ter selecionado o menu “Artigos Salvos”

#### **Pós-condições**

Após o fim normal deste caso de uso o sistema deve:

1. Ter salvo a edição do artigo e coloca-lo no menu “Artigos Salvos”.

### **Ator Primário**

Leitor, Avaliador

### **Fluxo de Eventos Principal**

1. O sistema apresenta a tela de edição de artigos
2. O usuário seleciona o artigo a editar
3. O sistema apresenta o artigo a editar
4. O usuário realiza as modificações no artigo
5. O usuário clica no botão “Salvar”.
6. O sistema salva o artigo e coloca-o no menu “Artigos Salvos”.
7. O sistema emite a mensagem “Artigo salvo com sucesso, verifique o menu Artigos Salvos”.
8. O usuário seleciona o botão “ok”;
9. O sistema encaminha o usuário para sua home page.
10. O caso de uso é finalizado.

#### 2.1.4 UC04 - ENVIAR ARTIGO PARA AVALIAÇÃO

### Especificação de Caso de Uso UC04 – Enviar Artigo para Avaliação

#### **Controle do Documento**

<b>Versão</b>	<b>Autor</b>	<b>Data</b>	<b>Descrição</b>
1.0	Sérgio Patrique Zotto	27/08/2008	Elaboração

#### **Descrição**

Este caso de descreve os passos para os usuários enviarem artigos para avaliação.

#### **Pré-condições**

Este caso de uso pode iniciar somente se:

- 1 O usuário tiver efetuado o login no sistema
- 2 O usuário ter escrito algum artigo

### **Pós-condições**

Após o fim normal deste caso de uso o sistema deve:

- 1 Ter enviado o artigo para avaliação.

### **Ator Primário**

Leitor, Avaliador

### **Fluxo de Eventos Principal**

1. O sistema apresenta a tela de “Artigos Salvos”
2. O usuário seleciona o botão para envio do artigo
3. O sistema apresenta a mensagem “Deseja realmente encaminhar o artigo para avaliação”
4. O usuário seleciona a opção “Sim”.
5. O sistema encaminha o artigo para avaliação e muda seu status para “Em avaliação”
6. O sistema tira o artigo do menu “Artigos Salvos” e o coloca no menu “Artigos em Avaliação”
7. O sistema encaminha o usuário para sua home page.
8. O caso de uso é finalizado.

#### 2.1.5 UC05 - VISUALIZAR ARTIGO

Especificação de Caso de Uso

### **UC05 – Visualizar Artigo**

#### **Controle do Documento**

<b>Versão</b>	<b>Autor</b>	<b>Data</b>	<b>Descrição</b>
1.0	Sérgio Patrique Zotto	27/08/2008	Elaboração


### Descrição

Este caso de descreve os passos para os usuários enviarem artigos para avaliação.

### Pré-condições

Este caso de uso pode iniciar somente se:

1. O usuário tiver efetuado o login no sistema

### Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Apresentar o artigo selecionado pelo usuário.

### Ator Primário

Leitor, Avaliador, Administrador

### Fluxo de Eventos Principal

1. O usuário seleciona um artigo a visualizar
2. O sistema apresenta o artigo ao usuário
3. O caso de uso é finalizado.

### 2.1.6 UC06 - BUSCAR ARTIGO

Especificação de Caso de Uso

### UC06 – Buscar Artigo

### Controle do Documento

Versão	Autor	Data	Descrição
1.0	Sérgio Patrique Zotto	28/08/2008	Elaboração

**Descrição**

Este caso de descreve os passos para os usuários procurarem artigos no sistema.

**Pré-condições**

Este caso de uso pode iniciar somente se:

- 1 O usuário tiver efetuado o login no sistema

**Pós-condições**

Após o fim normal deste caso de uso o sistema deve:

- 1 Apresentar os artigos encontrados na busca, com um grau de relevância dos mesmos.

**Ator Primário**

Leitor, Avaliador, Administrador

**Fluxo de Eventos Principal**

- 1 O sistema apresenta a tela de busca de artigos
- 2 O usuário escolhe a forma de procurar artigos, se por título, data ou autor
- 3 O usuário escreve a descrição da busca no campo descrição
- 4 O usuário seleciona o botão pesquisar
- 5 O sistema realiza a busca
- 6 O sistema apresenta os resultados encontrados
- 7 Encerra caso de uso

## 2.1.7 UC07 - AVALIAR ARTIGO

Especificação de Caso de Uso

**UC07 – Avaliar Artigo****Controle do Documento**

<b>Versão</b>	<b>Autor</b>	<b>Data</b>	<b>Descrição</b>
1.0	Sérgio Patrique Zotto	28/08/2008	Elaboração


### **Descrição**

Este caso de descreve os passos para o avaliador realizar a análise do artigo.

### **Pré-condições**

Este caso de uso pode iniciar somente se:

- 1 O usuário tiver efetuado o login no sistema
- 2 O usuário em questão tiver permissão de Avaliador

### **Pós-condições**

Após o fim normal deste caso de uso o sistema deve:

- 1 Alterar o status do artigo conforme avaliação do avaliador.
- 2 Chamar UC08 – Definir Vista do Artigo

### **Ator Primário**

Avaliador

### **Fluxo de Eventos Principal**

- 1 O usuário seleciona o artigo a avaliar
- 2 O sistema apresenta o artigo ao usuário
- 3 O usuário realiza a conferência do artigo.
- 4 O usuário escolhe a opção de avaliação “Publicar Artigo”
- 5 O usuário escolhe uma nota para o Artigo
- 6 O usuário escreve no campo comentário, uma breve análise sobre o artigo
- 7 O usuário seleciona o botão “Vista do Artigo”
- 8 Chama UC08 – Definir vista do artigo

#### **2.1.8 UC08 - DEFINIR VISTA DO ARTIGO**

Especificação de Caso de Uso  
**UC08 – Definir Vista do Artigo**

**Controle do Documento**

<b>Versão</b>	<b>Autor</b>	<b>Data</b>	<b>Descrição</b>
1.0	Sérgio Patrique Zotto	28/08/2008	Elaboração

**Descrição**

Este caso de descreve os passos para o avaliador realizar a vista do artigo

**Pré-condições**

Este caso de uso pode iniciar somente se:

- 1 O sistema tiver realizado o UC-07 Avaliar artigo com fluxo normal de eventos

**Pós-condições**

Após o fim normal deste caso de uso o sistema deve:

- 1 Salvar a vista do artigo, alterar o status e encaminha-lo para publicação.

**Ator Primário**

Avaliador

**Fluxo de Eventos Principal**

- 1 O usuário seleciona uma vista de artigo
- 2 O usuário seleciona o botão salvar
- 3 O sistema salva os dados da vista e encaminha o artigo para publicação
- 4 O caso de uso é finalizado.

2.1.9 UC09 - INCLUIR NOTÍCIAS

## Especificação de Caso de Uso

**UC09 – Incluir Notícias****Controle do Documento**

<b>Versão</b>	<b>Autor</b>	<b>Data</b>	<b>Descrição</b>
1.0	Sérgio Patrique Zotto	28/08/2008	Elaboração

**Descrição**

Este caso de descreve os passos para os administradores realizarem a inclusão de notícias no site.

**Pré-condições**

Este caso de uso pode iniciar somente se:

- 1 O usuário tiver efetuado o login no sistema
- 2 O usuário tiver permissão de administrador

**Pós-condições**

Após o fim normal deste caso de uso o sistema deve:

- 1 Salvar a notícia e coloca-la na base RSS o site.

**Ator Primário**

Administrador

**Fluxo de Eventos Principal**

- 1 O usuário seleciona a opção de nova notícia
- 2 O usuário escreve o título da notícia
- 3 O usuário escreve o desenvolvimento da notícia
- 4 O usuário seleciona o botão “Salvar Notícia”
- 5 O sistema salva a notícia e atualiza a base RSS
- 6 Encerra caso de uso

## 2.1.10 UC10 - CADASTRAR CATEGORIA

Especificação de Caso de Uso

### UC10 – Cadastrar Categoria

#### Controle do Documento

Versão	Autor	Data	Descrição
1.0	Sérgio Patrique Zotto	28/08/2008	Elaboração

#### Descrição

Este caso de descreve os passos para os administradores cadastrarem novas categorias de artigos.

#### Pré-condições

Este caso de uso pode iniciar somente se:

- 1 O usuário tiver efetuado o login no sistema
- 2 O usuário tiver permissão de administrador

#### Pós-condições

Após o fim normal deste caso de uso o sistema deve:

- 1 Salvar a nova categoria criada.

#### Ator Primário

Administrador

#### Fluxo de Eventos Principal

- 1 O usuário seleciona a opção de cadastrar categoria.
- 2 O sistema apresenta ao usuário as categorias já existentes e a opção de criar nova categoria.

- 3 O usuário escreve o nome da nova categoria.
- 4 O usuário seleciona o botão “Salvar categoria”.
- 5 O sistema verifica que não há nenhuma categoria com mesmo nome.
- 6 O sistema salva a nova categoria.
- 7 Encerra caso de uso.

### 2.1.11 UC11 - CADASTRAR AVALIADOR

#### Especificação de Caso de Uso **UC11 – Cadastrar Avaliador**

#### Controle do Documento

<b>Versão</b>	<b>Autor</b>	<b>Data</b>	<b>Descrição</b>
1.0	Sérgio Patrique Zotto	29/08/2008	Elaboração

#### Descrição

Este caso de descreve os passos para os administradores cadastrarem novos avaliadores.

#### Pré-condições

Este caso de uso pode iniciar somente se:

- 1 O usuário tiver efetuado o login no sistema
- 2 O usuário tiver permissão de administrador

#### Pós-condições

Após o fim normal deste caso de uso o sistema deve:

- 1 Salvar o novo avaliador.

#### Ator Primário

Administrador

### Fluxo de Eventos Principal

- 1 O usuário seleciona a opção de cadastrar avaliador.
- 2 O sistema apresenta a tela de busca de usuários do sistema.
- 3 O usuário preenche o campo nome com o nome do usuário a procurar.
- 4 O usuário seleciona o botão “Procurar”.
- 5 O sistema retorna com os resultados encontrados.
- 6 O usuário seleciona o leitor que procurava
- 7 O usuário seleciona a opção “Salvar como Avaliador”.
- 8 O sistema salva o novo avaliador e altera o status de “Leitor” para “Avaliador”.
- 9 Encerra caso de uso.

### 2.1.12 UC12 - CADASTRAR ADMINISTRADORES

#### Especificação de Caso de Uso UC12 – Cadastrar Administradores

#### Controle do Documento

Versão	Autor	Data	Descrição
1.0	Sérgio Patrique Zotto	29/08/2008	Elaboração

#### Descrição

Este caso de descreve os passos para os administradores cadastrarem novos administradores no sistema.

#### Pré-condições

Este caso de uso pode iniciar somente se:

- 1 O usuário tiver efetuado o login no sistema
- 2 O usuário tiver permissão de administrador

**Pós-condições**

Após o fim normal deste caso de uso o sistema deve:

- 1 Salvar um novo administrador.

**Ator Primário**

Administrador

**Fluxo de Eventos Principal**

- 1 O usuário seleciona a opção de cadastrar administrador.
- 2 O sistema apresenta a tela de busca de usuários do sistema.
- 3 O usuário preenche o campo nome com o nome do usuário a procurar.
- 4 O usuário seleciona o botão “Procurar”.
- 5 O sistema retorna com os resultados encontrados.
- 6 O usuário seleciona o usuário que procurava
- 7 O usuário seleciona a opção “Salvar como Administrador”.
- 8 O sistema salva o novo administrador e altera o status de usuário para “Administrador”
- 9 Encerra caso de uso.

## 2.1.13 UC13 - BANIR USUÁRIOS

## Especificação de Caso de Uso

**UC13 – Banir Usuários****Controle do Documento**

<b>Versão</b>	<b>Autor</b>	<b>Data</b>	<b>Descrição</b>
1.0	Sérgio Patrique Zotto	29/08/2008	Elaboração

**Descrição**

Este caso de descreve os passos para os administradores banirem usuários do sistema.

### **Pré-condições**

Este caso de uso pode iniciar somente se:

- 1 O usuário tiver efetuado o login no sistema
- 2 O usuário tiver permissão de administrador

### **Pós-condições**

Após o fim normal deste caso de uso o sistema deve:

- 1 Banir totalmente o usuário de todas as funções do sistema.

### **Ator Primário**

Administrador

### **Fluxo de Eventos Principal**

- 1 O usuário seleciona a opção de banir usuários.
- 2 O sistema apresenta a tela de busca de usuários do sistema.
- 3 O usuário preenche o campo nome com o nome do usuário a procurar.
- 4 O usuário seleciona o botão "Procurar".
- 5 O sistema retorna com os resultados encontrados.
- 6 O usuário seleciona o usuário que procurava
- 7 O usuário preenche o campo "Motivo" com o motivo do banimento.
- 8 O usuário seleciona a opção "Banir Usuário".
- 9 O sistema salva o status do usuário para "Banido"
- 10 Encerra caso de uso.

### 3. PONTOS POR CASO DE USO

#### Atores:

Leitor, Avaliador, Administrador.

#### Complexidade dos atores:

Ator	Complexidade	Peso
Leitor	Médio	2
Avaliador	Complexo	3
Administrador	Complexo	3

#### Cálculo do TPNAA (Total de pesos não ajustados dos atores)

Complexidade	Qtidade	Peso	Total
Simples	0	1	0
Médio	1	2	2
Complexo	2	3	6
Total			8

#### Complexidade dos Casos de Uso

Caso de Uso	Complexidade	Peso
1. Realizar Cadastro no sistema	Médio	2
2. Cadastrar Artigo	Médio	2
3. Alterar Artigo	Simples	1
4. Enviar Artigo para Avaliação	Médio	2
5. Visualizar Artigo	Simples	1
6. Buscar Artigo	Médio	2
7. Avaliar Artigo	Complexo	3
8. Definir Vista do Artigo	Complexo	3
9. Incluir Notícias	Médio	2

10. Cadastrar Categoria	Médio	2
11. Cadastrar Avaliador	Complexo	3
12. Cadastrar Administradores	Médio	2
13. Banir Usuários	Complexo	3

### Cálculo do TPNAUC (Total de Pesos não ajustados por Caso de Uso)

Complexidade	Qtidade	Peso	Total
Simple	2	1	3
Médio	7	2	12
Complexo	4	3	12
Total			28

### Cálculo do PTNA (Pontos totais não ajustados)

$$PTNA = TPNA + TPNAUC$$

$$PTNA = 8 + 28$$

$$PTNA = 36$$

### Tabela de FCA (Fatores de Complexidade Ambiental)

Fator	Descrição	Peso	Valor	Efator
F1	Familiaridade com o Processo Interativo Unificado	1,5	2	3
F2	Experiência na Aplicação	0,5	2	1
F3	Experiência em Orientação a Objetos	1	4	4
F4	Capacidade de Liderança de Análise	0,5	3	1,5
F5	Motivação	1	4	4
F6	Estabilidade de Requisitos	2	3	6
F7	Consultores Part-Time	-1	0	0
F8	Dificuldade de Programação na Linguagem	-1	4	-4
Total				15,5

$$E_{\text{fator}} = 15,5$$

$$FCA = 1,4 + (-0,03 * E_{\text{fator}})$$

$$FCA = 1,4 + (-0,03 * 15,5)$$

$$FCA = 1,4 + (-0,465)$$

$$FCA = 0,935$$

**Tabela de FCT (Fatores de Complexidade Técnica)**

Fator	Descrição	Peso	Valor	Tfator
T1	Distribuição do sistema	2	0	0
T2	Resposta aos objetivos de desempenho	1	5	5
T3	Eficiência do usuário final	1	4	4
T4	Complexidade do processo interno	1	5	5
T5	Código deve ser reutilizado	1	4	4
T6	Facilidade de instalação	0,5	0	0
T7	Facilidade de uso	0,5	4	2
T8	Portabilidade	2	2	4
T9	Facilidade de Alterar	1	4	4
T10	Concorrência	1	0	0
T11	Features de Segurança	1	2	2
T12	Acesso direto a dispositivos de parceiros	1	0	0
T13	Treinamento especial aos usuários	1	0	0
Total				30

$$T_{\text{fator}} = 30$$

$$FCT = 0,6 + (0,01 * T_{\text{fator}})$$

$$FCT = 0,6 + (0,01 * 30)$$

$$FCT = 0,6 + (0,3)$$

$$FCT = 0,9$$

**PTUC (Pontos totais de casos de uso)**

$$PTUC = PTNA * FCA * FCT$$

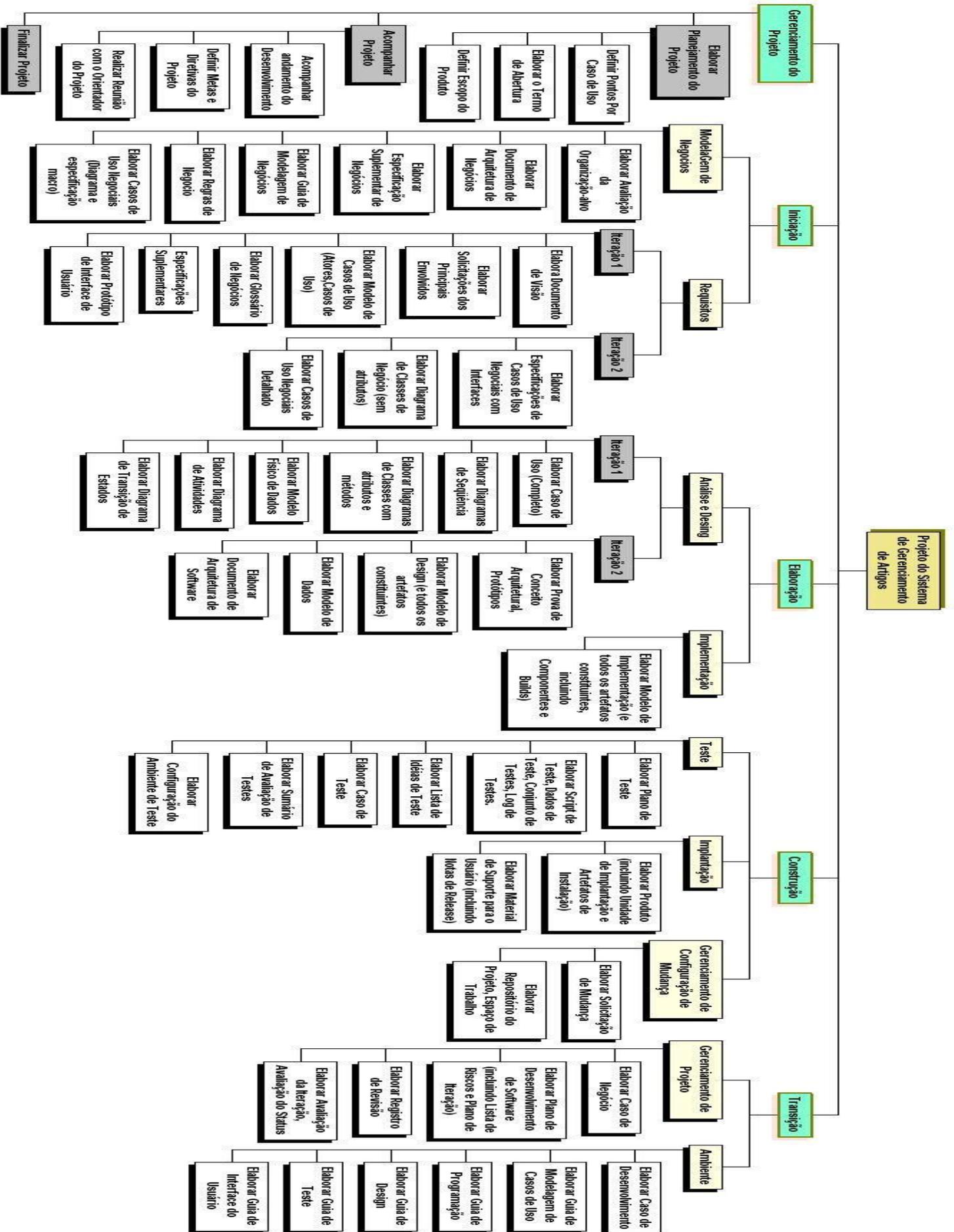
$$PTUC = 36 * 0,935 * 0,9$$

$$PTUC = 30,294$$

**Estimativas**

$$\text{Sugestão de Karner: } 30,294 * 20 = \mathbf{605 \text{ horas}}$$

### 4. WBS - WORK BREAKDOWN STRUCTURE

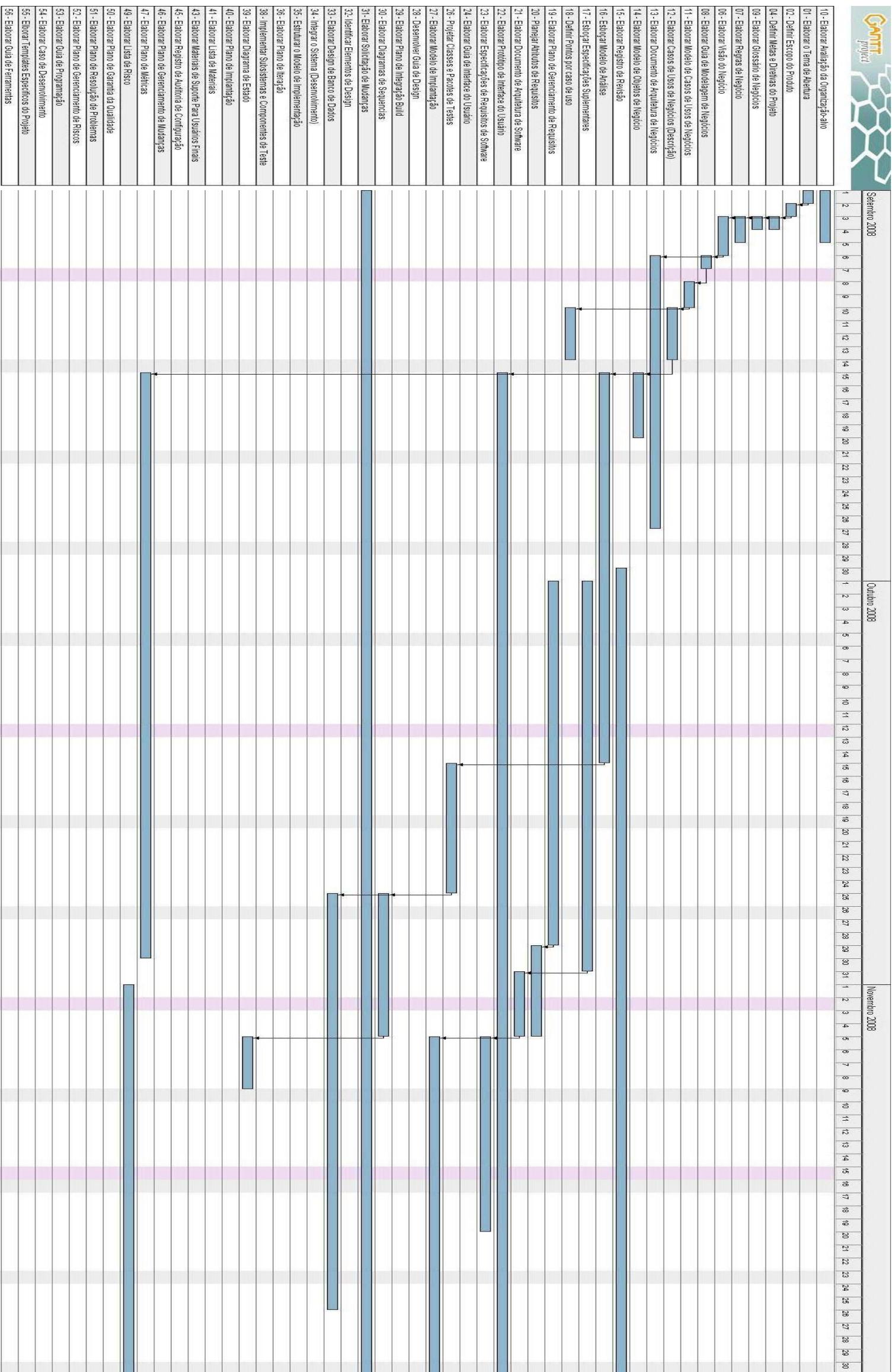


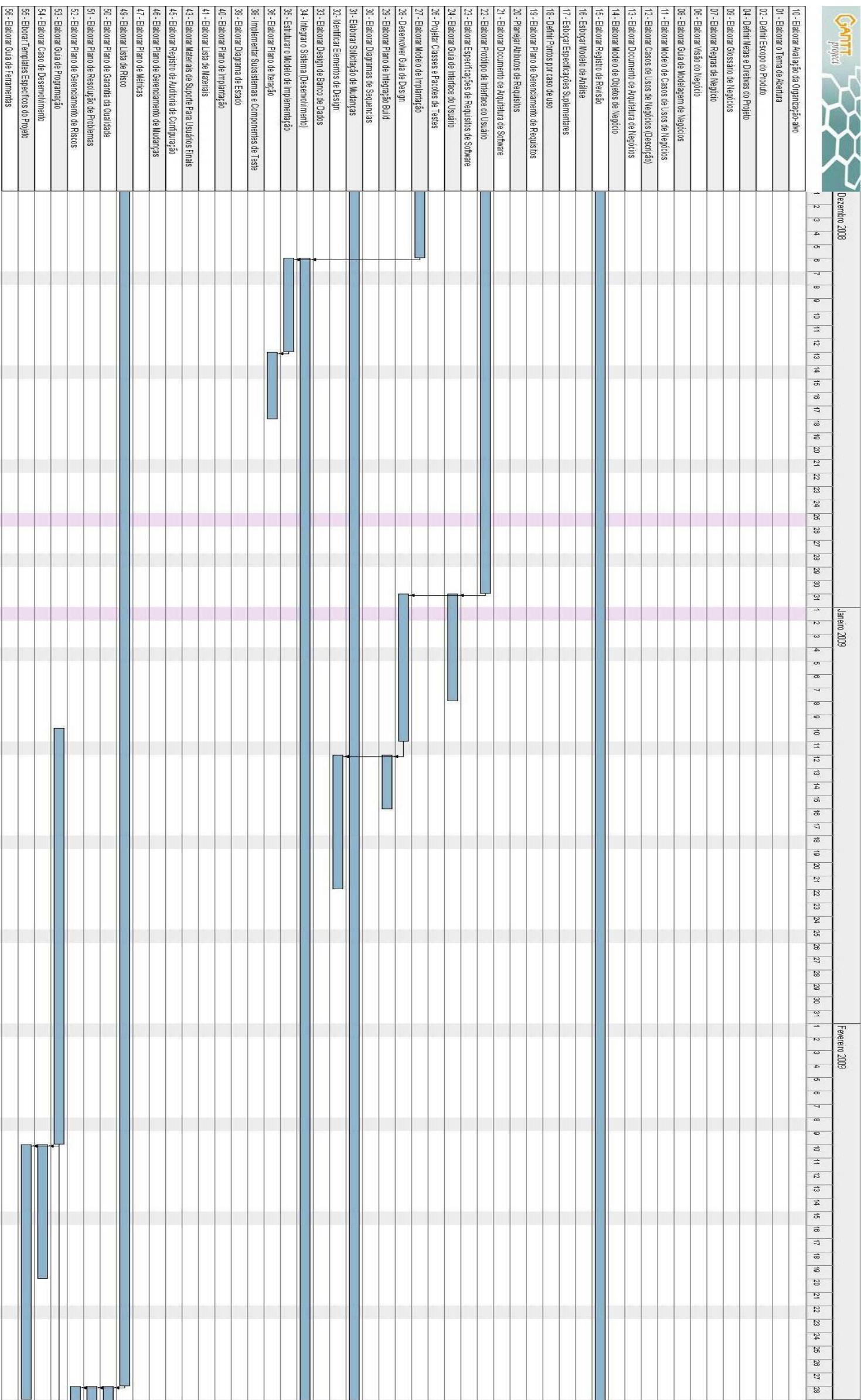
## 5. TABELA DE PRECEDÊNCIA

WBS TABELA DE PRECEDÊNCIA		
SISTEMA GERENCIADOR DE ARTIGOS		
Número	Atividade	Depende de
1	Elaborar o Tema de Abertura	
2	Definir Escopo do Produto	1
3	Acompanhar Andamento do Projeto	2
4	Definir Metas e Diretivas do Projeto	2
5	Realizar Reunião com o Orientador do Projeto	1
6	Elaborar Visão do Negócio	2
7	Elaborar Regras de Negócio	2
8	Elaborar Guia de Modelagem de Negócios	6
9	Elaborar Glossário de Negócios	2
10	Elaborar Avaliação da Organização-alvo	
11	Elaborar Modelo de Casos de Usos de Negócios	8
12	Elaborar Casos de Usos de Negócios (Descrição)	11
13	Elaborar Documento de Arquitetura de Negócios	6
14	Elaborar Modelo de Objetos de Negócio	12
15	Elaborar Registro de Revisão	
16	Esboçar Modelo de Análise	12
17	Esboçar Especificações Suplementares	
18	Definir Pontos por caso de uso	11
19	Elaborar Plano de Gerenciamento de Requisitos	
20	Planejar Atributos de Requisitos	19
21	Elaborar Documento de Arquitetura de Software	17
22	Elaborar Protótipo de Interface do Usuário	12
23	Elaborar Especificações de Requisitos de Software	21
24	Elaborar Guia de Interface do Usuário	22
25	Elaborar Prova de Conceito Arquitetural	
26	Projetar Classes e Pacotes de Testes	16
27	Elaborar Modelo de Implantação	21
28	Desenvolver Guia de Design	22
29	Elaborar Plano de Integração Build	28
30	Elaborar Diagramas de Seqüência	26
31	Elaborar Solicitação de Mudanças	
32	Identificar Elementos de Design	28
33	Elaborar Design de Banco de Dados	26
34	Integrar o Sistema	27
35	Estruturar o Modelo de Implementação	27
36	Elaborar Plano de Iteração	35
37	Integrar o SubSistema	34
38	Implementar Subsistemas e Componentes de Teste	34
39	Elaborar Diagrama de Estado	30
40	Elaborar Plano de Implantação	34
41	Elaborar Lista de Materiais	40
42	Elaborar Manuais de Treinamento	41
43	Elaborar Materiais de Suporte Para Usuários Finais	41
44	Elaborar Arte-Final do Produto	42
45	Elaborar Registro de Auditoria de Configuração	15

<b>46</b>	Elaborar Plano de Gerenciamento de Mudanças	45
<b>47</b>	Elaborar Plano de Métricas	12
<b>48</b>	Elaborar Caso de Negócio	12
<b>49</b>	Elaborar Lista de Risco	10
<b>50</b>	Elaborar Plano de Garantia da Qualidade	49
<b>51</b>	Elaborar Plano de Resolução de Problemas	49
<b>52</b>	Elaborar Plano de Gerenciamento de Riscos	49
<b>53</b>	Elaborar Guia de Programação	48
<b>54</b>	Elaborar Caso de Desenvolvimento	53
<b>55</b>	Elaborar Templates Específicos do Projeto	53
<b>56</b>	Elaborar Guia de Ferramentas	53

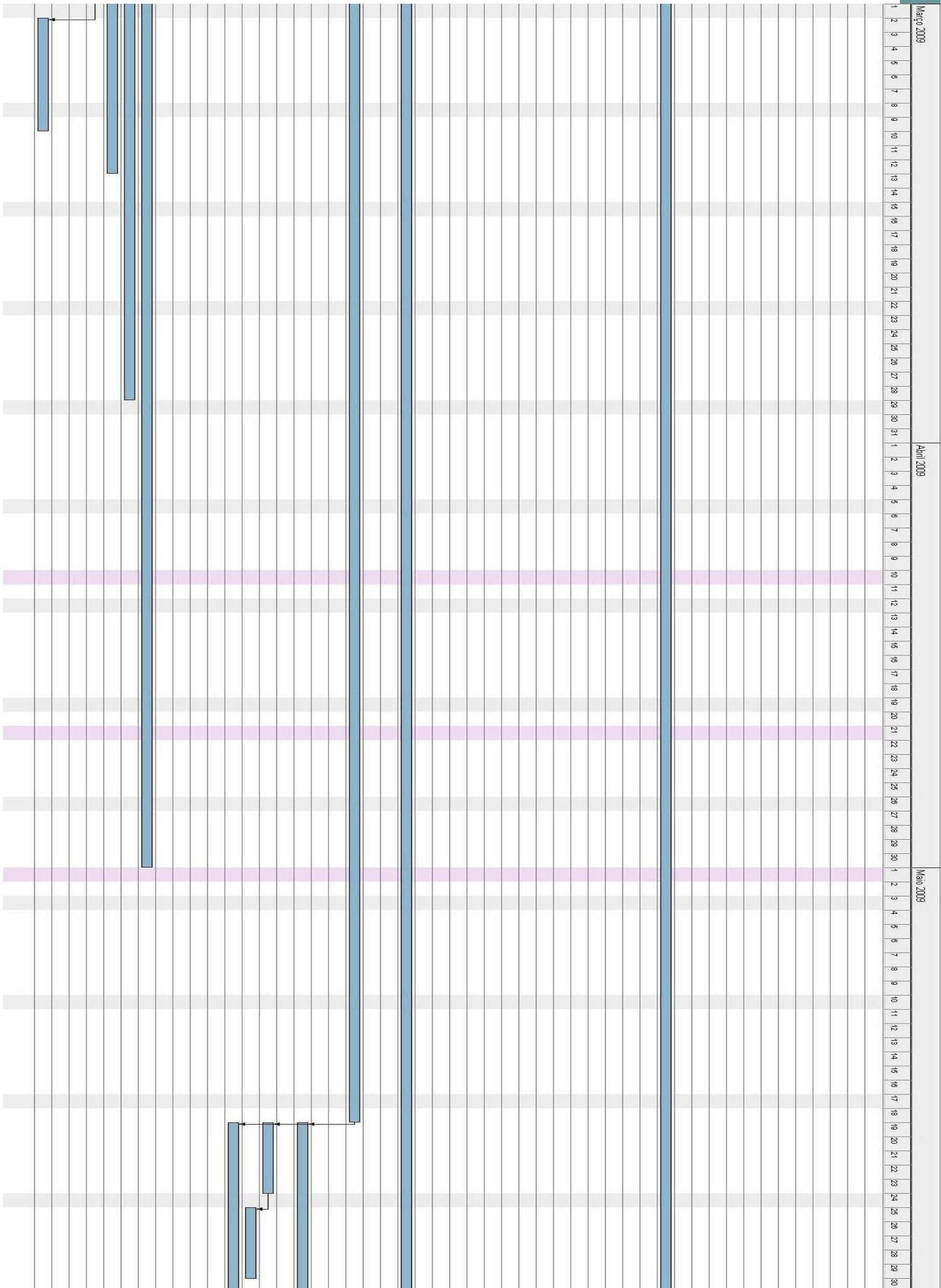
## 6. GRÁFICO DE GANTT





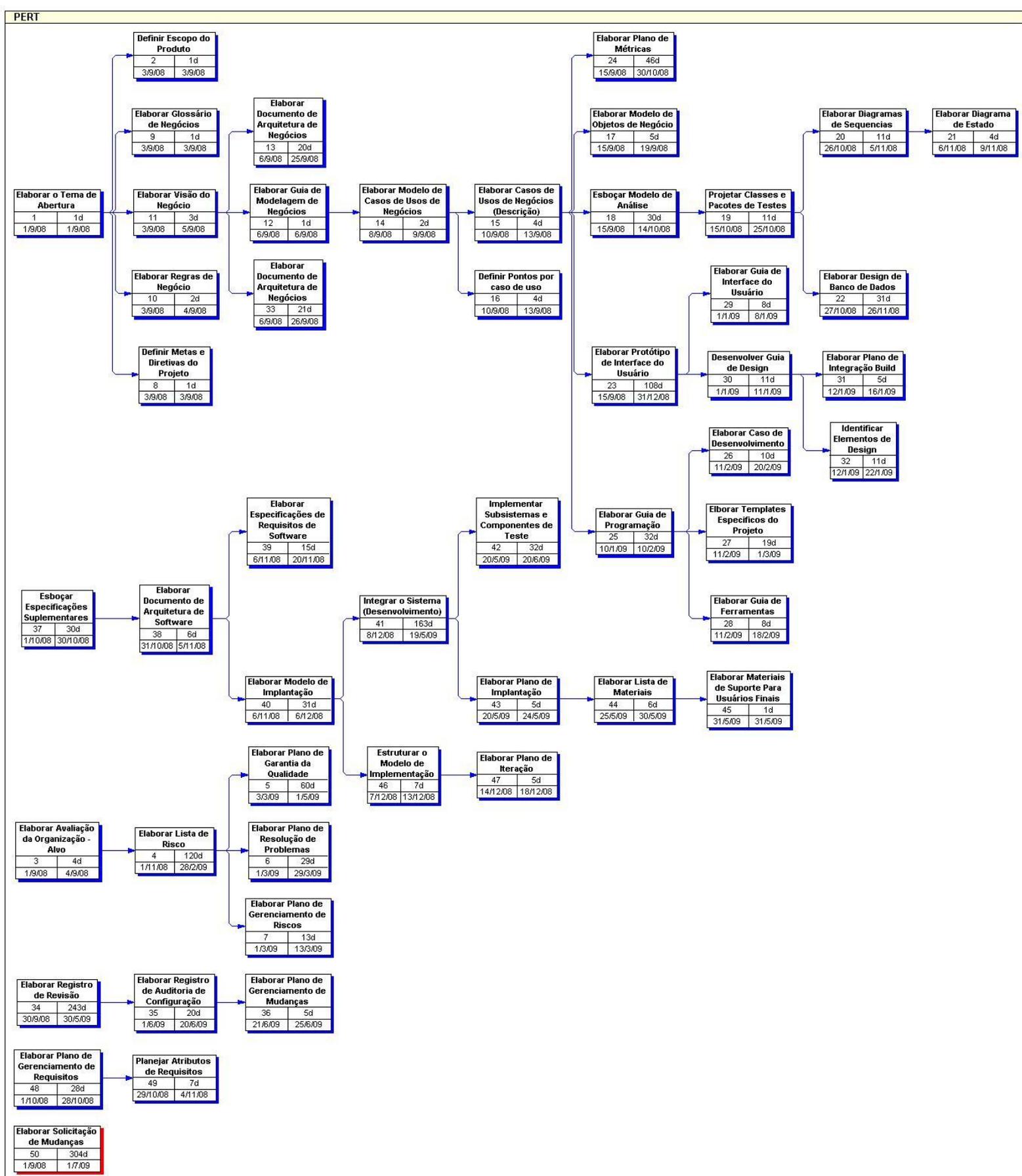


10 - Elaborar Avaliação da Organização-alvo
01 - Elaborar o Tema de Abertura
02 - Definir Escopo do Produto
04 - Definir Metas e Diretrizes do Projeto
09 - Elaborar Glossário de Negócios
07 - Elaborar Regras de Negócio
06 - Elaborar Visão do Negócio
08 - Elaborar Guia de Modelagem de Negócios
11 - Elaborar Modelo de Casos de Usos de Negócios
12 - Elaborar Casos de Usos de Negócios (Descrição)
13 - Elaborar Documento de Arquitetura de Negócios
14 - Elaborar Modelo de Objetivos de Negócio
15 - Elaborar Registro de Requisitos
16 - Estimar Modelo de Análise
17 - Estimar Especificações Suplementares
18 - Definir Pontos por caso de uso
19 - Elaborar Plano de Gerenciamento de Requisitos
20 - Planejar Atributos de Requisitos
21 - Elaborar Documento de Arquitetura de Software
22 - Elaborar Protótipo de Interface do Usuário
23 - Elaborar Especificações de Requisitos de Software
24 - Elaborar Guia de Interface do Usuário
26 - Projetar Classes e Pacotes de Testes
27 - Elaborar Modelo de Implantação
28 - Desenvolver Guia de Design
29 - Elaborar Plano de Integração Build
30 - Elaborar Diagramas de Sequências
31 - Elaborar Solicitação de Mudanças
32 - Identificar Elementos de Design
33 - Elaborar Design de Banco de Dados
34 - Integrar o sistema (Desenvolvimento)
35 - Estruturar o Modelo de Implementação
36 - Elaborar Plano de Integração
38 - Implementar Subsystemas e Componentes de Teste
39 - Elaborar Diagrama de Estado
40 - Elaborar Plano de Implantação
41 - Elaborar Lista de Materiais
43 - Elaborar Materiais de Suporte Para Usuários Finais
45 - Elaborar Registro de Auditoria de Configuração
46 - Elaborar Plano de Gerenciamento de Mudanças
47 - Elaborar Plano de Métricas
49 - Elaborar Lista de Risco
50 - Elaborar Plano de Garantia da Qualidade
51 - Elaborar Plano de Resolução de Problemas
52 - Elaborar Plano de Gerenciamento de Riscos
53 - Elaborar Guia de Programação
54 - Elaborar Caso de Desenvolvimento
55 - Elaborar Templates Específicos do Projeto
56 - Elaborar Guia de Ferramentas





## 7. PERT CHART



## 8. PLANO DE CUSTOS

## PLANO DE CUSTOS - SISTEMA GERENCIADOR DE ARTIGOS

N	Dados da Atividade		Alocação e Respectivos Custos dos Recursos		Total por Atividade
	Atividade	Esforço estimado (hh)	Analista Pleno (Aluno)		
			Valor Unit	R\$ 10,00	
			Qtidade Hh	R\$	
1	Modelagem do negócio	2	2	R\$ 20,00	R\$ 20,00
2	Elaborar Casos de Usos de Negócios	1	1	R\$ 10,00	R\$ 10,00
3	Elaborar Casos de Usos de Negócios (Descrição)	2	2	R\$ 20,00	R\$ 20,00
4	Elaborar Documento de Arquitetura de Negócios	2	2	R\$ 20,00	R\$ 20,00
5	Esboçar Modelo de Análise	1	1	R\$ 10,00	R\$ 10,00
6	Esboçar Especificações Suplementares	2	2	R\$ 20,00	R\$ 20,00
7	Elaborar Plano de Gerenciamento de Requisitos	3	3	R\$ 30,00	R\$ 30,00
8	Planejar Atributos de Requisitos	2	2	R\$ 20,00	R\$ 20,00
9	Elaborar Protótipo de Interface do Usuário	1	1	R\$ 10,00	R\$ 10,00
10	Projetar Classes e Pacotes de Testes	1	1	R\$ 10,00	R\$ 10,00
11	Elaborar Diagramas de Seqüência	1	1	R\$ 10,00	R\$ 10,00
12	Elaborar Modelo de Implantação	2	2	R\$ 20,00	R\$ 20,00
13	Elaborar Design de Banco de Dados	1	1	R\$ 10,00	R\$ 10,00
14	Integrar o Sistema (Desenvolvimento)	15	15	R\$ 150,00	R\$ 150,00
15	Elaborar Diagrama de Estado	5	5	R\$ 50,00	R\$ 50,00
16	Elaborar Plano de Implantação	3	3	R\$ 30,00	R\$ 30,00
17	Equipamentos	-	-	R\$ -	R\$ -
18	Recursos (Luz, água, Telefone, Transporte)	-	-	R\$ 160,00	R\$ 160,00
<b>TOTAL</b>		<b>44</b>	<b>44</b>	<b>600</b>	<b>R\$ 600,00</b>

\* Obs: Não incluso na planilha de custo a mensalidade da Pós Graduação

## 9. PLANO DE COMUNICAÇÃO

### PLANO DE COMUNICAÇÃO - SISTEMA GERENCIADOR DE ARTIGOS

Grupo de Interessados	Foco	O que este Grupo Precisa Saber	Método	Quando
<b>INTERNO AO PROJETO</b>				
Equipe do Projeto (Aluno e Orientador)	Informar sobre o andamento do projeto e compartilhar conhecimento técnico	Cronograma atualizado, eventuais problemas e soluções referentes aos diversos tópicos do projeto, compartilhar informações	Reunião de aproximadamente 30 minutos	Quinzenalmente
Equipe do Projeto (Aluno e Orientador)	Informar sobre o andamento do projeto e sanar dúvidas em relação ao desenvolvimento do mesmo	Cronograma atualizado, eventuais problemas e soluções referentes aos diversos tópicos do projeto, compartilhar informações	E-mail	Diariamente, quando necessário
<b>EXTERNO AO PROJETO</b>				
Equipe do Projeto (Aluno)	Sanar Dúvida referente a linguagem de programação utilizada para o desenvolvimento do projeto	Conhecimento técnico em JAVA	E-mail	Quando necessário

## 10. PLANO DE RISCOS

## PLANO DE RISCOS - SISTEMA GERENCIADOR DE ARTIGOS

N.	Condição	Data Limite	Conseqüência	Ação	Monitoramento	Probabilidade	Impacto	Classificação
1	Deficiência na capacidade técnica da equipe	nov/08	Incapacidade de realização de tarefas, atraso no cronograma, possibilidade de realização de cursos	Atualização constante nas tecnologias presentes, reuniões com o orientador do projeto	Aluno, através de reuniões com o orientador	Moderado	Alto	6
2	Excesso de mudança nos requisitos	sem data	Atraso no cronograma, dificuldade de elaboração do plano de solicitação de mudanças, impossibilidade de especificar datas limites para o desenvolvimento	Total interação com o plano de gerenciamento de mudanças, colocando datas limites para cada alteração e impactos provenientes	Aluno, através de acompanhamento e constante atualização do plano de solicitação de mudanças	Alto	Moderado	6
3	Requisitos pouco precisos	dez/08	Falta de qualidade na execução de tarefas	Definir bem os requisitos necessários no plano de requisitos	Aluno, através de acompanhamento e constante atualização do plano de requisitos	Baixo	Baixo	3
4	Indisponibilidade de recursos	dez/08	Falta de qualidade no desenvolvimento	Definir no início do projeto todos os recursos para desenvolvimento e acompanhamento	Aluno, através de acompanhamento e avaliação dos recursos a serem	Baixo	Baixo	3

					utilizados			
5	Decisões externas causam mudanças forçadas no projeto	fev/09	Atraso no cronograma	Definir o tempo necessário para cada tarefa do projeto e datas limites, para que eventuais mudanças não venham a interferir a entrega e finalização do sistema	Aluno, através de constante atualização e acompanhamento do gráfico de Gannt	Muito Baixo	Baixo	2
6	Não consegue atingir os requisitos de performance	fev/09	Dificuldade nos testes do sistema. Atraso para implantação	Realizar testes de benchmark para auxiliar na definição de pontos e performance do sistema	Aluno, através de testes de stress para garantir a performance e agilidade do sistema	Moderado	Moderado	5
7	Cronograma não realista	jun/09	Confusão entre as etapas, atraso na entrega do projeto	Definir semanalmente uma data para verificar se as tarefas estão realmente dentro do prazo estipulado	Aluno, através de acompanhamento semanal do cumprimento do cronograma	Alto	Muito Alto	8
8	Uso de novas tecnologias de hardware e software	abr/09	Atraso no cronograma, dificuldade de desenvolvimento, demora para aprendizagem por parte do desenvolvedor	Conhecer bem a tecnologia a ser utilizada, estar atento as mudanças nas linguagens e novas técnicas, registrar as mudanças no plano de mudanças	Aluno, através da adaptação a tecnologia a ser utilizada e solicitar ajuda ao orientador e professore do curso quando necessário	Alto	Alto	7

9	Conhecimento insuficiente do negócio	fev/09	Atraso no cronograma, dificuldade em planejar as regras e metas do negócio	Definir claramente os documentos de regras, visão e glossário de negócios	Aluno, através da inserção de mudanças constantes nos documentos de regras, visão e glossário de negócios	Moderado	Alto	6
10	Falhas de comunicação ou baixa performance	fev/09	Atraso no cronograma, e dificuldade na implementação e testes do sistema	Definir claramente os tipos de comunicação e performance esperada e através de testes verificar a qualidade obtida	Aluno, através de testes de comunicação e performance	Moderado	Baixo	4

## 11. PLANO DE MONITORAMENTO E CONTROLE

### PLANO DE MONITORAMENTO E CONTROLE SISTEMA GERENCIADOR DE ARTIGOS

#### 11.1. Monitorar e controlar o trabalho do projeto

Coletar, medir e disseminar informações sobre o desempenho e avaliar as medições e as tendências para efetuar melhorias no processo, incluindo o monitoramento de riscos para garantir que os mesmos sejam identificados no início, e que o andamento seja relatado e que planos de risco adequados estejam sendo executados.

Realizar a emissão de relatórios de andamento, medição do progresso e previsão com o objetivo de fornecer informações sobre o desempenho do projeto em relação a escopo, cronograma, custo, recursos, qualidade e risco.

Acompanhar rigorosamente o andamento do projeto com base no RUP, não deixando a metodologia de lado e avaliar a sua progressão.

Ações a serem realizadas nesta etapa:

- Ações corretivas encontradas
- Ações preventivas encontradas
- Previsões em geral
- Reparo de defeitos
- Mudanças solicitadas

#### 11.2. Controle Integrado de mudanças

Controlar os fatores que criam mudanças através do plano de solicitação de mudanças, para garantir que essas mudanças sejam benéficas, determinando se ocorreu uma mudança e gerenciar as alterações aprovadas, inclusive o momento em que ocorrem.

Realizar o processo durante todo o projeto, desde a iniciação até o encerramento do mesmo.

Ações a serem realizadas nesta etapa:

- Solicitações de mudança aprovadas
- Plano de gerenciamento do projeto

- Declaração do escopo do projeto
- Ações corretivas
- Ações preventivas
- Reparo de defeitos encontrados

### **11.3. Verificação e controle do escopo**

É a fase de verificação do projeto que contém os processos necessários para formalizar a aceitação formalizada das entregas.

Ações a serem realizadas nesta etapa:

- Entregas aceitas pelo Orientador
- Mudanças solicitadas e documentadas
- Ações corretivas recomendadas

É o processo necessário para controlar as mudanças feitas no escopo do projeto.

Ações a serem realizadas nesta etapa:

- Declaração bem definida do escopo do projeto
- Estrutura analítica do projeto
- Dicionário da EAP
- Linha de base do escopo
- Mudanças solicitadas
- Ações corretivas recomendadas
- Plano de gerenciamento do projeto

### **11.4. Controle do cronograma**

Processo necessário para controlar as mudanças feitas no cronograma do projeto, uma das etapas mais críticas, pois um impacto nesta fase pode acarretar em grandes prejuízos ao andamento do projeto.

Ações a serem realizadas nesta etapa:

- Dados do modelo de cronograma (Gráfico de Gannt)
- Linha de base do cronograma
- Medições de desempenho
- Mudanças solicitadas
- Ações corretivas recomendadas

- Lista de atividades
- Atributos da atividade
- Plano de gerenciamento do projeto

### **11.5. Controle de custos**

Processo de influenciar os fatores que criam as variações e controlar as mudanças no orçamento do projeto. A estrutura deve ser feita por base no plano de custos, mas devido ao fato de se tratar de um projeto acadêmico, o controle de custos não será de grande importância nem de grande impacto.

Ações a serem realizadas nesta etapa:

- Elaboração do Plano de Custos
- Relatórios de desempenho
- Informações sobre o desempenho do trabalho
- Solicitações de mudança aprovadas
- Plano de gerenciamento do projeto

### **11.6. Controle da qualidade**

Processo necessário para monitorar resultados específicos do projeto a fim de determinar se eles estão de acordo com os padrões relevantes de qualidade e identificar maneiras de eliminar as causas de um desempenho insatisfatório.

Ações a serem realizadas nesta etapa:

- Plano de gerenciamento da qualidade
- Métricas de qualidade
- Listas de verificação da qualidade
- Ativos de processos organizacionais
- Informações sobre o desempenho do trabalho
- Solicitações de mudança aprovadas

### **11.7. Gerenciar a equipe do projeto**

Processo necessário para acompanhar o desempenho de membros da equipe (orientador e aluno), fornecer feedback, resolver problemas e coordenar mudanças para melhorar o desempenho do projeto.

Ações a serem realizadas nesta etapa:

- Ativos de processos organizacionais
- Funções e responsabilidades
- Organogramas do projeto
- Plano de gerenciamento de pessoal
- Avaliação do desempenho do aluno
- Informações sobre o desempenho do trabalho
- Relatórios de desempenho

#### **11.8. Fornecer relatórios de desempenho e gerenciar partes interessadas**

Processo necessário para coletar e distribuir informações sobre o desempenho. Isso inclui relatório de andamento, medição do progresso e previsão.

Ações a serem realizadas nesta etapa:

- Informações sobre o desempenho do trabalho
- Medições de desempenho
- Previsão de término
- Medições de controle da qualidade
- Plano de gerenciamento do projeto
- Solicitações de mudança aprovadas

#### **11.9. Monitoramento e controle de riscos**

Processo necessário para acompanhar os riscos identificados, monitorar os riscos residuais, identificar novos riscos, executar planos de respostas a riscos e avaliar sua eficiência durante todo o ciclo de vida do projeto.

- Plano de gerenciamento de riscos
- Registro de riscos
- Solicitações de mudança
- Informações sobre o desempenho do trabalho
- Relatórios de desempenho

#### **11.10. Administração de contratos**

Processo necessário para gerenciar o contrato e a relação entre o aluno e projeto acadêmico a ser desenvolvido com a Universidade, analisar e documentar o

desempenho atual e, quando adequado, gerenciar a relação contratual com o orientador do projeto.

- Termo de abertura
- Plano de gerenciamento do projeto
- Relatórios de desempenho
- Solicitações de mudança aprovadas
- Informações sobre o desempenho do trabalho