

UNIVERSIDADE FEDERAL DO PARANÁ  
RODRIGO PEREZ FURTADO

**SISTEMA PARA CONTROLE EVENTOS**

CURITIBA  
2017

UNIVERSIDADE FEDERAL DO PARANÁ  
RODRIGO PEREZ FURTADO

## **SISTEMA PARA CONTROLE EVENTOS**

Trabalho apresentado para obtenção do título de especialista em Engenharia de Software, no curso Especialização em engenharia de Software, setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná.

Orientador: Professor Dr. Jaime Wojciechowski

CURITIBA

2017



MINISTÉRIO DA EDUCAÇÃO  
UNIVERSIDADE FEDERAL DO PARANÁ  
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO  
Setor SEPT  
Curso de Pós-Graduação ENGENHARIA DE SOFTWARE

### TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em ENGENHARIA DE SOFTWARE da Universidade Federal do Paraná foram convocados para realizar a arguição da Monografia de Especialização de **RODRIGO PEREZ FURTADO** intitulada: **SISTEMA PARA CONTROLE EVENTO**, após terem inquirido o aluno e realizado a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de especialista está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

Curitiba, 07 de Novembro de 2017.

JAIME WOJCIECHOWSKI  
Presidente da Banca Examinadora (UFPR)

RAFAELA MANTOVANI FONTANA  
Avaliador Interno (UFPR)

## RESUMO

Este projeto surgiu da necessidade de gerenciamento do evento anual SIEPE (Semana Integrada de Ensino Pesquisa e Extensão) visto que era feito de forma manual demandando maior número de recursos humanos e abrindo margem para erros operacionais. Tendo em vista essas questões, foi proposto a criação de um software escrito em Java para automatizar e gerenciar as atividades do evento. O sistema permite à comissão organizadora cadastrar o evento, a área temática e o cronograma. Permite também que os participantes submetam seus trabalhos e à comissão avaliadora validar os trabalhos aptos a participar do evento. Na metodologia, é aplicado o processo RUP de engenharia de software durante todo o ciclo de desenvolvimento. Concluiu-se o desenvolvimento do software com sucesso, dentro do prazo e atendendo todos os requisitos acordados. Futuramente pode-se acrescentar novas funcionalidades com o objetivo de proporcionar maior apoio à tarefa de gerenciamento e ainda, ser utilizado para atender outros eventos na UFPR.

Palavras chave: Evento, SIEPE, Submissão de trabalho.

## **ABSTRACT**

This project arose from the need to manage the annual SIEPE (Integrated Week of Teaching Research and Extension) event, as it was done manually requiring a greater number of human resources and opening the margin for operational errors. In view of these issues, it was proposed to create software written in Java to automate and manage the activities of the event. The system allows the organizing committee to register the event, thematic area and the schedule. It also allows the participants to submit their papers and to the evaluation committee to validate the works apt to participate in the event. In the methodology, the software engineering RUP process is applied throughout the development cycle. Software development was successfully completed, on time, and meeting all agreed requirements. In the future, new functionalities can be added in order to provide greater support to the management task and also be used to attend other events at UFPR.

Keywords: Event, SIEPE, Submission of work.

## LISTA DE FIGURAS

Figura 2 – Página inicial .....	25
Figura 3 - Lista de evento.....	26
Figura 4 - Cadastro de evento.....	27
Figura 5 - Indicar participação .....	28
Figura 6 - Lista de comissão .....	28
Figura 7 - Cadastro de comissão .....	29
Figura 8 - Atribuição de avaliador.....	30
Figura 9 - Lista de trabalho.....	30
Figura 10 - Trabalho novo .....	31

## LISTA DE ABREVIATURAS E SIGLAS

CSS: Cascading Style Sheets, mecanismo que adiciona estilo ao HTML.

FRAMEWORK: Conjunto de funcionalidades que une códigos em comum de vários projetos.

HIBERNATE: framework para o mapeamento objeto-relacional escrito na linguagem Java.

HTML: HyperText Markup Language, que significa Linguagem de Marcação de Hipertexto.

JAVA: Linguagem de programação.

JAVASCRIPT: Linguagem de programação interpretada.

SGBD: Sistema de gerenciamento de banco de dados.

XHTML: eXtensible Hypertext Markup Language, é uma reformulação da linguagem de marcação HTML, baseada em XML.

SIEPE: Semana Integrada de Ensino Pesquisa e Extensão

ENEC: Encontro de Extensão e Cultura

ENAF: Encontro de Atividades Formativas

EVINCI: Evento de Iniciação Científica

EINTI: Evento de Inovação Tecnológica

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	<b>10</b>
1.1	OBJETIVOS DO PROJETO.....	10
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b> .....	<b>11</b>
2.1	DESCRIÇÃO DO PROBLEMA.....	11
2.2	SOLUÇÃO DO PROBLEMA.....	11
<b>3</b>	<b>METODOLOGIA</b> .....	<b>12</b>
3.1	UML.....	12
3.1.1	Características da UML.....	12
3.1.2	Aplicações da UML.....	13
3.1.3	Fase do desenvolvimento de um sistema em UML.....	14
3.1.3.1	Análises de requisitos.....	14
3.1.3.2	Análise.....	15
3.1.3.3	Projeto.....	15
3.1.3.4	Implementação (programação).....	16
3.1.4	Diagrama de caso de uso.....	16
3.1.5	Diagrama de classes.....	17
3.1.6	Diagrama de sequência.....	18
3.1.7	Teste.....	18
3.2	HTML.....	19
3.3	XML.....	19
3.4	MVC – MODEL-VIEW-CONTROLLER.....	19
3.5	POSTGRESQL.....	20
3.6	RUP.....	21
3.7	JAVA.....	21
3.8	CASCADING STYLE SHEETS (FOLHAS DE ESTILO) – CSS.....	21
3.9	JAVASCRIPT.....	22
3.10	HIBERNATE.....	23
3.11	PRIMEFACES.....	23
3.12	MATERIAIS.....	23
3.12.1	Hardware.....	23
3.12.2	Software.....	24
<b>4</b>	<b>APRESENTAÇÃO DO SOFTWARE</b> .....	<b>25</b>
4.1	PÁGINA INICIAL.....	25

4.2	LISTA DE EVENTO .....	26
4.3	CADASTRO DE EVENTO.....	27
4.4	INDICAR PARTICIPAÇÃO.....	28
4.5	LISTA DE COMISSÃO .....	28
4.6	CADASTRO DE COMISSÃO .....	29
4.8	ATRIBUIÇÃO DE AVALIADOR NO TRABALHO.....	30
4.9	LISTA DE TRABALHO .....	30
4.10	CADASTRO DE TRABALHO .....	31
<b>5</b>	<b>CONSIDERAÇÕES FINAIS .....</b>	<b>32</b>
	<b>REFERÊNCIAS .....</b>	<b>33</b>
	<b>APÊNDICE A – DOCUMENTO DE VISÃO .....</b>	<b>34</b>
	<b>APÊNDICE B – CASOS DE USO NEGOCIAIS .....</b>	<b>36</b>
	<b>APÊNDICE C – GLOSSÁRIO.....</b>	<b>37</b>
	<b>APÊNDICE D – REGRAS DE NEGÓCIO .....</b>	<b>38</b>
	<b>APÊNDICE E – PROTÓTIPO DE INTERFACES .....</b>	<b>39</b>
	<b>APÊNDICE F – MODELO DE OBJETOS NEGOCIAIS .....</b>	<b>44</b>
	<b>APÊNDICE G – CASOS DE USO.....</b>	<b>45</b>
	<b>APÊNDICE H – MODELO DE OBJETOS.....</b>	<b>54</b>
	<b>APÊNDICE I – DIAGRAMAS DE SEQUÊNCIA .....</b>	<b>55</b>
	<b>APÊNDICE J – MODELO DE OBJETOS .....</b>	<b>59</b>
	<b>APÊNDICE K – MODELO FÍSICO DE DADOS .....</b>	<b>60</b>
	<b>APÊNDICE L – PLANO DE TESTES .....</b>	<b>72</b>
	<b>APÊNDICE M – CASOS DE TESTES .....</b>	<b>76</b>
	<b>APÊNDICE N – LOG DE TESTES.....</b>	<b>81</b>
	<b>APÊNDICE O – SOLICITAÇÃO DE MUDANÇA .....</b>	<b>94</b>
	<b>APÊNDICE P – PLANO DE IMPLANTAÇÃO .....</b>	<b>95</b>

## **1 INTRODUÇÃO**

A Universidade Federal do Paraná promove eventos com o intuito de aproximar a sociedade acadêmica por meio de debates e trocas de experiências. Segundo o site do evento, a SIEPE(Semana Integrada de Ensino Pesquisa e Extensão) é um evento anual que promove e estimula a interação entre docentes, estudantes de graduação, educação profissional, ensino médio, pós graduação, servidores técnicos e comunidade. O objetivo da SIEPE é divulgar trabalhos institucionais orientados por docentes da UFPR nas atividades de Ensino, Pesquisa e Extensão, cumprindo, assim com o que determina o art. 207 da Constituição Brasileira: “as universidades devem obedecer ao princípio de indissociabilidade entre ensino, pesquisa e extensão”.

Com o crescimento do evento surgiu a necessidade de um sistema para gerenciá-lo, deste modo o objetivo do projeto é apresentar um software que possibilite esse gerenciamento oferecendo maior agilidade e eficiência nas atividades dos eventos.

### **1.1 OBJETIVOS DO PROJETO**

O objetivo principal do sistema é gerir as edições anuais da SIEPE realizadas a partir de 2017, garantindo integridade e segurança das informações, além do fácil acesso às informações pertinentes durante o andamento do processo.

## **2 FUNDAMENTAÇÃO TEÓRICA**

No ano de 2017, a SIEPE reunirá o 16º Encontro das Atividades Formativas (ENAF), o 25º Evento de Iniciação Científica (EVINCI), o 10º Evento de Iniciação em Desenvolvimento Tecnológico e Inovação (EINTI) e o 16º Encontro de Extensão e Cultura (ENEC) com o tema: Incertezas em tempos de crise: a Universidade como espaço de debate.

O evento ocorrerá nas cidades de Curitiba, Palotina e Jandaia do Sul com duração de quatro dias sendo um dos maiores eventos organizados pela UFPR contando com o envolvimento de mais ou menos 5500 pessoas

### **2.1 DESCRIÇÃO DO PROBLEMA**

No ano de 2016 a SIEPE contou com a apresentação de 2076 trabalhos tendo toda sua organização feita de forma manual demandando maior número de recursos humanos e abrindo margem para erros operacionais.

### **2.2 SOLUÇÃO DO PROBLEMA**

O desenvolvimento de um sistema capaz de suportar o gerenciamento da SIEPE e de outros eventos da UFPR, beneficiando toda a comunidade acadêmica tornando mais ágil e confiável toda a gestão de eventos, tratando desde o cadastrado, credenciamento, inscrição de trabalhos, ensalamento, cadastrado de comissões e avaliadores os dados armazenados em banco de dados possibilita a geração de relatórios e indicadores mais precisos como nunca feito em edições anteriores.

### 3 METODOLOGIA

Para o desenvolvimento do projeto foi utilizado como base a metodologia RUP (Rational Unified Process), os artefatos utilizam a linguagem de modelagem UML (Unified Modeling Language).

#### 3.1 UML

UML - Unified Modeling Language - é uma linguagem para modelagem de dados orientada a objetos. Com ela podemos fazer uma modelagem visual de maneira que os relacionamentos entre os componentes do sistema sejam mais bem visualizados, compreendidos e documentados.

O modelo de linguagem é a notação que o método usa para descrever o projeto. O processo são os passos que devem ser seguidos para se construir o projeto.

O modelo de linguagem é uma parte muito importante do método, corresponde ao ponto principal da comunicação, se uma pessoa quer conversar sobre o projeto, com outra pessoa, é através do modelo de linguagem que elas se entendem. Nessa hora, o processo não é utilizado.

A UML é baseada em Diagramas: ênfase visual, onde vários aspectos fundamentais na modelagem de sistemas são abordados, tais como:

- Funcional: estrutura estática e interação dinâmica,
- Não funcional: tempo de processamento, confiabilidade e produção,
- Organizacional: organização do trabalho, mapeamento e código.

##### 3.1.1 Características da UML

A UML é uma linguagem de modelagem, não uma metodologia para desenvolvimento de sistemas.

Modelagem de Negócio - Workflow: Sistema onde trabalhos podem ser transferidos de uma pessoa para outra através de conexões em rede seguindo um conjunto de regras. O Sistema Workflow rastreia todos os momentos destes trabalhos onde grava e gera estatísticas sobre eles.

Modelagem de Objetos e Componentes: Na construção do software os modelos podem ser usados na identificação das características e funcionalidades que o software deverá prover e no planejamento de sua construção. A modelagem de softwares normalmente implica na construção de modelos gráficos que

simbolizam os artefatos de software utilizados e seus inter-relacionamentos. Uma forma comum de modelagem de programas procedurais (não orientados a objeto) é através de fluxogramas, enquanto que a modelagem de programas orientados a objeto normalmente usam a linguagem gráfica.

O objetivo da UML é descrever qualquer tipo de sistema em termos de diagramas orientados a objetos. Ela é utilizada para a visualização do problema pela equipe de desenvolvimento de software.

“A UML permite a construção de modelos precisos, não ambíguos e completos para a especificação do software que será construído” (BOOCH, RUMBAUGH, JACOBSON, 2000, p.16).

Os modelos construídos usando a UML têm uma dimensão de “projeto” que é independente de uma linguagem de programação e permitem a tradução entre os modelos e códigos dessas linguagens. Podem ser implementados em qualquer linguagem de programação ou permite também a Engenharia Reversa na qual os modelos são gerados a partir do código-fonte.

Todo projeto de software envolve muita documentação desde a fase inicial até o produto final. Os modelos criados durante o desenvolvimento fazem parte da documentação do software.

“Arquitetura de Sistema de Software é um conjunto de decisões significativas sobre a organização do sistema de software, a seleção de elementos estruturais e suas interfaces, o comportamento do sistema, composição dos elementos estruturais e comportamentais” (BOOCH, RUMBAUGH, JACOBSON, 2000, p. 32).

A arquitetura de um sistema apresenta diferentes visões do sistema que está sendo modelado, diferentes pontos de vista devem ser usados para refletir os aspectos desejados. Cada visão mostra um conjunto de aspectos do sistema numa notação adequada à sua compreensão.

### 3.1.2 Aplicações da UML

**Sistemas de Informação:** Pesquisa, edita, armazena e mostra informações aos usuários com grande quantidade de dados que são armazenado em banco de dados relacionais e orientado a objetos.

**Sistemas Técnicos:** Controlam e mantêm equipamentos de telecomunicação, militares e de processos industriais possuindo uma interface especial e ao contrário

dos Sistemas de Informação possui menos programação sendo assim basicamente um sistema Real-Time.

Sistemas real-time integrados: São executados em hardware, possuem programação de baixo nível e suporte real-time.

Sistemas distribuídos: Distribuídos em máquinas onde os dados são transferidos de forma simples entre as máquinas. Os sistemas de mecanismos são sincronizados onde são mantida a integridade dos dados.

Sistemas de software: Onde são definidas as estruturas que são utilizadas por outros softwares. Sistemas Operacionais, Banco de Dados e ações de baixo nível no hardware.

Sistemas de negócios: Os processos de negócios descrevem seus objetivos e especificações dentro das regras (leis, estratégias de negócios etc.), do atual trabalho desempenhado.

É importante saber que os sistemas não possuem uma ou mais características apresentadas acima, mas sim várias características ao mesmo tempo. Um exemplo disso seria o Sistema de Informações atual que possuem características distribuídas (Real-Time), onde a UML suporta todos os tipos de Sistemas.

### 3.1.3 Fase do desenvolvimento de um sistema em UML

O desenvolvimento de um sistema de software envolve diversas fases onde, em cada uma delas, diversas atividades são desenvolvidas utilizando a UML. As fases de desenvolvimento são: Análise de requisitos, fase de análise, fase de projeto, fase de programação e fase de testes.

#### 3.1.3.1 Análises de requisitos

A análise de requisitos é fundamental para o desenvolvimento de sistemas, pois trata justamente de descobrir o que o cliente quer com o sistema. A análise de requisitos está associada ao processo de descobrir quais são as operações que o sistema deve realizar e quais são as restrições que existem sobre estas operações (WAZLAWICK,2004).

Nessa etapa são coletadas as informações sobre os dados de interesse da aplicação. Nessa etapa são realizadas entrevistas com os futuros usuários do

sistema, análise de documentações fazendo uma descrição dos requisitos da aplicação mais detalhada e clara possível.

Os principais requisitos do sistema são capturados através do desenvolvimento dos diagramas de caso de uso do sistema. Os requisitos são classificados em Funcionais e Não-funcionais. O primeiro requisito define o que o sistema deve fazer, e o segundo define as restrições sobre como o sistema deve desempenhar suas funções.

“É Importante salientar que os requisitos do sistema são informações que o cliente ou usuário solicitam, e não informações planejadas pelo analista” (WAZLAWICK,2004).

### 3.1.3.2 Análise

É a fase de análise dos requisitos obtidos, nessa etapa é realizada a modelagem conceitual que descreve a informação que o sistema vai gerenciar. “A análise de domínio está relacionada à descoberta das informações que são gerenciadas no sistema, ou seja, à representação e transformação da informação modelando-as em classes e objetos” (WAZLAWICK,2004).

“O modelo funcional pode ser representado através dos contratos de operações de sistema” (WAZLAWICK,2004).

O Modelo Conceitual ou Modelo de domínio ilustra as classes conceituais do sistema. Analisa o domínio do problema e não do domínio da solução. Não deve ser confundido com o modelo de dados.

O Modelo Conceitual representa classes conceituais ou objetos do mundo real no domínio de um problema, ou seja, nessa fase são modeladas as principais classes do sistema que estão presentes no domínio do sistema deixando para a fase de projeto a modelagem de classes técnicas que gerenciam banco de dados, interfaces, comunicações.

### 3.1.3.3 Projeto

A fase de projeto enfatiza a proposta de uma solução que atenda os requisitos da análise. “A fase de análise é uma investigação para tentar descobrir o que o cliente quer, o projeto consiste em propor uma solução com base no conhecimento adquirido na análise” (WAZLAWICK,2004).

Na fase de projeto podem-se ter diagramas mais técnicos e mais precisos:

- Diagramas de classes a partir de uma perspectiva de software.
- Diagramas de sequência para cenários comuns, utilizando os cenários mais importantes dos casos de uso e fazendo diagramas de sequência para descobrir como acontecerá no software.
- Diagramas de pacotes para mostrar a organização em larga escala do software.
- Diagramas de estados para as classes com históricos de vida complexos.
- Diagramas de distribuição para mostrar o layout físico do software.

Na fase de Projeto espera-se que a implementação do código acompanhe os diagramas. Essa fase resulta no detalhamento das especificações para a fase de programação do sistema.

#### 3.1.3.4 Implementação (programação)

A utilização de técnicas sistemáticas nas fases de análise e projeto faz com que o processo de geração de código possa ser automatizado. Neste caso, cabe ao programador dominar as características específicas das linguagens, ferramentas, frameworks e estruturas de dados para adaptar o código gerado aos requisitos indicados quando necessário (WAZLAWICK,2004).

A Fase de programação é separada das outras fases, pois é nessa fase que os modelos criados e analisados serão convertidos em código, para implementação do software.

#### 3.1.4 Diagrama de caso de uso

Definição: O Diagrama de Caso de Uso dá uma visão geral de todas as funções do sistema mostrando a interação entre atores e Casos de Uso e mesmo entre os Casos de Uso.

Elementos:

Ator é alguém ou alguma coisa que interage com o sistema, é quem ou o que usa o sistema. O ator representa um papel, não um usuário individual do sistema. Atores podem ser humanos ou sistemas automatizados.

Caso de Uso representa uma funcionalidade completa do sistema, conforme percebida por um ator. É um conjunto de seqüências de ações que um sistema executa, gerando um resultado observável que interessa a determinado ator.

A UML não define padrões para descrever os casos de uso, e assim muitas pessoas compreendem erroneamente que a notação gráfica define a natureza do caso de uso; contudo a notação gráfica pode dar a visão geral mais simples de um caso de uso ou de um conjunto destes casos.

O Diagrama de Caso de Uso fornece um modo fácil de descrever a visão externa do sistema e suas interações com o mundo exterior, representando uma visão de alto nível da funcionalidade do sistema mediante uma requisição do usuário. Ele é bastante importante porque vai ser a base do processo de desenvolvimento do sistema.

O diagrama de caso de uso e suas especificações encontram-se no APÊNDICE B e G.

### 3.1.5 Diagrama de classes

“O Diagrama de Classes é o mais utilizado e mais importante da UML. Serve de apoio para a maioria dos demais diagramas.”( Guedes, 2008)

Ele exemplifica a estrutura das classes usadas pelo sistema, determinando seus atributos, métodos e relacionamentos. Um diagrama de classes é um gráfico que mostra a estrutura do sistema, mostra as classes e os relacionamentos entre elas, descreve os aspectos estáticos do sistema e não mostra informações temporais.

Uma classe é um molde para a construção de objetos que define os atributos e operações comuns aos objetos. Uma classe tem nome, uma lista de atributos, comportamentos (operações) e relacionamentos com outras classes. Um objeto é uma instância de uma classe.

Os relacionamentos entre as classes podem ser:

- Associação: É representada através de uma linha simples com um nome, geralmente identificados por verbos na especificação do problema.

- Agregação: É um tipo especial de associação onde o elemento associado corresponde a uma parte do elemento principal. As partes só existem se o todo existir e vice-versa.

- Composição: É representada através de uma linha que conecta a classe agregada com a outra classe através de um losango preenchido ao lado da classe agregada. Esse relacionamento é uma forma restrita de agregação.

- Generalização ou Herança: A classe generalizada é representada por um triângulo unido a classe e as especializações são ligadas a esse triângulo por linhas simples.

O diagrama de classes final encontra-se no APÊNDICE J.

### 3.1.6 Diagrama de sequência

O Diagrama de Sequência demonstra a ordem cronológica que as mensagens são trocadas entre os objetos envolvidos em um determinado processo. Ele baseia-se no Caso de Uso e no Diagrama de Classes para determinar os objetos envolvidos em um processo. Ele também identifica o evento ou ator gerador do processo modelado.

Um Diagrama de Sequência é um diagrama que mostra, para o cenário específico de um caso de uso, os eventos que os atores externos geram sua ordem e os eventos entre sistemas.

Todos os sistemas são tratados como uma caixa-preta, a ênfase do diagrama está nos eventos que atravessam a fronteira do sistema entre atores e outros sistemas.

Deve ser feito o Diagrama de Sequência para a sequência de sucesso principal do caso de uso e para cenários frequentes ou alternativas complexas. “Um Diagrama de Sequência mostra eventos de sistema para uma sequência, ou cenário; portanto, é gerado a partir da inspeção de um caso de uso” (LARMAN,2004, p.144)

Os termos mostrados nos Diagramas de Sequência são bem resumidos e podem necessitar de uma explicação adequada, de modo que, durante o trabalho de projeto fique bem claro o que está entrando e saindo do sistema. Se isto não foi explicado nos casos de uso, então o glossário pode ser usado, no entanto, deve sempre haver um uso significativo do glossário em caso de decisão. Caso contrário, simplesmente teremos um trabalho desnecessário e que não acrescenta valor.

O diagrama de sequência encontra-se no APÊNDICE I.

### 3.1.7 Teste

A fase de testes envolve os testes de unidade, feitos pelo programador, para verificar se os componentes gerados atendem à especificação feita nas fases

anteriores, e aos testes de caso de uso, que visam verificar a adequação do sistema aos requisitos inicialmente levantados (WAZLAWICK,2004).

Os objetivos da fase de teste são:

- Verificar se as especificações da etapa de projeto refletem exatamente os requisitos levantados na fase de análise de requisitos e as definições na fase de análise, ou seja, verificar se o sistema funciona conforme os casos de uso especificados e se as classes implementadas estão se relacionando e cooperando entre si para garantir que o sistema gerado esteja de acordo com o sistema desejado.

- Verificar se não existem erros de lógica no projeto e no código, no fluxo de dados, no entendimento de requisitos, de codificação, de interface em todas as fases do projeto. Nesta fase devemos identificar e corrigir os erros.

O sucesso da fase de testes consiste em descobrir os erros e corrigi-los, para que o usuário final teste e verifique que o sistema está de acordo com o desejado.

Os testes do sistema encontram-se no APÊNDICE L, M e N.

### 3.2 HTML

HTML (HyperText Markup Language, ou linguagem de formatação de hipertexto) trata-se de uma linguagem de marcação utilizada para produzir páginas na internet. Documentos HTML são feitos para prover estrutura lógica da informação destinada à apresentação de páginas da rede mundial de computadores.

### 3.3 XML

XML (eXtensible Markup Language) é uma linguagem universal que permite a troca de informações de forma estruturada através da Internet. Permite que os desenvolvedores transportem dados de um servidor para outro da rede de forma transparente e organizada.

### 3.4 MVC – MODEL-VIEW-CONTROLLER

O design pattern MVC (Model-View-Controller, ou modelo-visão-controlador) permite a separação do modelo de dados das várias formas que o dado pode ser acessado e manipulado. Um sistema MVC é dividido em modelo de dados, conjunto de visões e conjunto de controladores. MVC é útil principalmente para aplicações grandes e distribuídas, em que dados idênticos são visualizados e manipulados de formas variadas. A chave para o MVC é a separação de responsabilidades, as

visões fornecem a interface do usuário, os controladores são responsáveis por selecionar uma visão apropriada e por fazer alterações no modelo de dados, e o modelo é responsável por representar os dados-base da aplicação.

O MVC garante a separação de tarefas, facilitando assim a reescrita de alguma parte, e a manutenção do código.

### 3.5 POSTGRESQL

O PostgreSQL é um sistema de gerenciamento de banco de dados objeto-relacional (SGBDOR) desenvolvido pelo Departamento de Ciência da Computação da Universidade da Califórnia em Berkeley. O POSTGRES foi pioneiro em vários conceitos que somente se tornaram disponíveis muito mais tarde em alguns sistemas de banco de dados comerciais.

O PostgreSQL é um descendente de código fonte aberto deste código original de Berkeley, que suporta grande parte do padrão SQL e oferece muitas funcionalidades modernas, como:

- Comandos complexos
- Chaves estrangeiras
- Gatilhos
- Visões
- Integridade transacional
- Controle de simultaneidade multiversão

Além disso, o PostgreSQL pode ser ampliado pelo usuário de muitas maneiras como, por exemplo, adicionando novos:

- Tipos de dado
- Funções
- Operadores
- Funções de agregação
- Métodos de índice
- Linguagens procedurais

Devido à sua licença liberal, o PostgreSQL pode ser utilizado, modificado e distribuído por qualquer pessoa para qualquer finalidade, seja particular, comercial ou acadêmica, livre de encargos.

### 3.6 RUP

O RUP(Rational Unified Process) é um processo de engenharia de software flexível e configurável, ele pode ser utilizado em projetos de pequeno, médio e grande porte. Organiza o desenvolvimento de software em quatro fases, sendo cada fase composta de uma ou mais iterações:

- Iniciação ou Concepção: ênfase no escopo do sistema;
- Elaboração: ênfase na arquitetura;
- Construção: ênfase no desenvolvimento.
- Transição: ênfase na implantação.

A segmentação por fases tem como objetivo dividir as ações dentro do projeto por competências de cada recurso que o compõe. Para cada uma das fases, há um maior foco de trabalho de acordo com a disciplina que está sendo desenvolvida. A fase é considerada concluída quando atinge seu marco principal.

### 3.7 JAVA

Java uma linguagem de alto nível e orientada a, fortemente tipada que independente de arquitetura, robusta, segura, extensível, bem estruturada, distribuída, multithreaded e com garbage collection(coletor de lixo).

A tecnologia Java está em todo lugar! Ela pode ser encontrada em laptops, datacenters, consoles de jogos, supercomputadores científicos, telefones celulares e na Internet.

### 3.8 CASCADING STYLE SHEETS (FOLHAS DE ESTILO) – CSS

A utilização de CSS permite o desenvolvimento de sites harmônicos e de mais fácil manutenção. Podemos definir elementos de aparência, tais como: fonte (tipo, tamanho e cor), bordas (espessura, cor e tipo), margens, espaçamento, imagens, alteração de estilo de links e qualquer outro elemento da página.

O CSS foi criado com intuito de definir a estrutura semântica de documentos, permitindo a separar o conteúdo das informações de estilo.

O responsável pela definição das especificações do CSS é o W3C( World Wide Web Consortium), que é uma organização não-governamental responsável pela definição de padrões de diversas tecnologias relativas á internet, como o HTML, o protocolo HTTP e o XML. É formada por membros de diversas empresas.

O autor pode definir estilos que se aplicam a toda uma página, ou na maior parte das vezes, a todo um site. Com isso mudança no layout e na apresentação das páginas, que também pode ser feita de forma centralizada.

Muitas das funcionalidades do CSS não são suportadas pelo HTML somente. Com o CSS o código HTML fica mais leve, por não ser necessária a utilização de tags, já que podem ser definidas pelo CSS. As páginas ficam bem menores e a leitura do código-fonte muito mais fácil.

### 3.9 JAVASCRIPT

Inicialmente conhecida por LiveScript, desenvolveu-se no Netscape para tornar seu browser mais poderoso e funcional, proporcionando a seus usuários uma experiência mais rica na web. Atualmente todos os browsers suportam JavaScript, apesar de haver diferenças de sintaxe entre um browser e outro.

Mesmo sendo uma extensão da linguagem HTML, o JavaScript é uma linguagem baseada na linguagem Java. Com isto, o JavaScript suporta a maior parte das sintaxes e comandos da linguagem Java.

Um script JavaScript permite que sejam executadas funções que antes, obrigatoriamente, deveriam ser processadas pelo servidor, como validação de campos, abertura de novas janelas, criação e manipulação de camadas, cálculos simples e complexos, etc.

Diferente da Linguagem HTML, o JavaScript corresponde a programação orientada a objetos, isto significa que todos os elementos de uma página da Web são tratados como objetos. Estes objetos são agrupados de acordo com seu tipo ou finalidade. Dentro da linguagem JavaScript, são criados automaticamente objetos que permitem que o usuário possa criar novos objetos de acordo com sua conveniência. Ao ser carregada uma página da Web, é criado um determinado número de objetos JavaScript, com propriedades e valores próprios que são ajustados pelo conteúdo da própria página. Todos eles seguem uma hierarquia que reflete toda a estrutura de uma página HTML. A linguagem JavaScript pode ser utilizada para a criação de scripts tanto do lado cliente como do lado servidor.

Cada objeto existente na manipulação do JavaScript possui propriedades. Exemplo sabemos que um documento HTML possui título e corpo, estas características do documento são propriedades que existem neste documento. Estas propriedades existem de dois tipos, algumas são os objetos propriamente ditos

enquanto outras não. Um exemplo disto, é o objeto form (formulário) que é uma propriedade do objeto document (documento). Já a propriedade de título da página (title), é pertencente ao objeto document não havendo nenhuma propriedade sobre ela. Concluindo, podemos dizer que a propriedade form do objeto document é um objeto-filho e o objeto document é o objeto-pai. Em geral, as propriedades podem conter valores (string, números, entre outros tipos). A utilização de propriedades se dá acompanhada de seu objeto sendo separados por um ponto apenas.

### 3.10 HIBERNATE

O framework Hibernate simplifica a integração com o banco de dados durante o desenvolvimento. É uma ferramenta open source ORM (Object Relational Mapping) escrito na linguagem Java.

Uma ferramenta ORM simplifica a criação de dados, a manipulação e o acesso. É uma técnica de programação que mapeia o objeto para os dados armazenados no banco de dados.

O Hibernate é um framework de mapeamento objeto relacional para aplicações Java, ou seja, é uma ferramenta para mapear classes Java em tabelas do banco de dados e vice-versa. É bastante poderoso e dá suporte ao mapeamento de associações entre objetos, herança, polimorfismo, composição e coleções. (FERNANDES & LIMA, 2007).

### 3.11 PRIMEFACES

PrimeFaces é uma biblioteca de componentes de interface de código aberto para aplicativos baseados em JavaServer Faces (JSF).

PrimeFaces é uma bibliotecas de componentes ricos em JavaServer Faces. A suíte de componentes inclui diversos campos de entrada, botões, tabelas de dados, árvores, gráficos, diálogos, etc. Os componentes do PrimeFaces possuem funcionalidade de Ajax integrado, baseado na API de Ajax do JSF. (FARIA, 2013)

### 3.12 MATERIAIS

Neste item serão descritos os recursos de hardware e software utilizados para o desenvolvimento e análise do sistema.

#### 3.12.1 Hardware

- Notebook Asus K555:
- Processador Intel Core i5, 8gb de memória RAM
- Disco rígido de 1Tb

### 3.12.2 Software

- Sistema operacional Linux Mint 16.1 64bit
- Astah Community 6.9.0
- IDE Eclipse
- PostgreSQL 9.6
- PgAdmin 3
- Jboss 7.3
- Google Chrome 62

## 4 APRESENTAÇÃO DO SOFTWARE

Este capítulo tem como objetivo fornecer uma visão geral da implementação do sistema e suas funcionalidades.

### 4.1 PÁGINA INICIAL

Ao acessar o sistema, a página inicial é apresentada. A partir da página inicial, o usuário poderá escolher um evento em andamento e o sistema coloca em evidência o título do evento selecionado e disponibiliza as opções de menu para o evento.



Figura 1 – Página inicial

Fonte: O autor (2017)

## 4.2 LISTA DE EVENTO

Página com a lista de todos os eventos cadastrados no sistema, aqui o usuário pode pesquisar os eventos cadastrados, alterar ou iniciar um cadastro.



The screenshot displays the 'Eventos' (Events) management interface. On the left is a sidebar with navigation links: 'Principal', 'Participação', and 'Evento'. The main content area is titled 'Evento - Listar' and includes a search filter section with input fields for 'Título' and 'Ano'. Below the filters are three buttons: 'Pesquisar', 'Limpar', and 'Cadastrar'. A table lists two events: '9ª SIEPE' and '16ª ENEC', both from 2017. Each event has an edit icon in the 'Ação' column. At the bottom, a pagination bar shows 'Página: 1 de 1 | Total: 2 registros', navigation arrows, a page number '1', a dropdown for '10' records per page, and the text 'registros por página'.

Título	Edição	Ano	Ação
9ª SIEPE	9ª	2017	
16ª ENEC	16ª	2017	

Figura 2 - Lista de evento

Fonte: O autor (2017)

### 4.3 CADASTRO DE EVENTO

Página para cadastro de evento, aqui o usuário insere todas as informações do evento, simultaneamente com o cadastro do cronograma, área temática e a comissão organizadora do evento.

**Eventos**

Principal  
Participação  
Evento

**Evento - Novo**

**Evento**

Título: \* Edição: Ano: Link do edital:

Descrição: \*

400

Submissão de trabalho:

**Cronograma**

Descrição: Início: Fim: Localização: \* Tipo

--- Selecione --- Selecione + Adicionar

Descrição	Data inicio	Data fim	Localização	Tipo Cronograma	Ação
Nenhum registro foi encontrado					

**Área Temática**

Descrição: \* + Adicionar

Descrição	Ação
Nenhum registro foi encontrado	

**Comissão Organizadora**

Participante: \* Função: \*

--- Selecione --- Selecione + Adicionar

Membro	Função	Ação
Nenhum registro foi encontrado		

✓ Salvar ✗ Cancelar

Figura 3 - Cadastro de evento

Fonte: O autor (2017)

#### 4.4 INDICAR PARTICIPAÇÃO

Nesta página o usuário indica dentre os eventos em andamento se pretende participar, após indicar sua participação é possível cancelar a mesma.

**Eventos**

- Principal
- Participação
- Evento

### Indicar Participação

16ª - ENEC 2017

"Cronograma"				
Descrição	Data inicio	Data fim	Localização	Tipo Cronograma
Abertura	01/10/2017 00:00	31/10/2017 00:00	Setor de Ciências Sociais Aplicadas	Abertura

✓ Selecionar

9ª - SIEPE 2017

"Cronograma"				
Descrição	Data inicio	Data fim	Localização	Tipo Cronograma
Abertura	01/10/2017 19:00	01/10/2017 21:00	Setor de Ciências Sociais Aplicadas	Abertura
Submissão de Trabalho	09/10/2017 00:00	21/10/2017 00:00	Setor de Ciências da Terra	Apresentação de Trabalho
Fechamento	31/10/2017 11:00	31/10/2017 12:00	Setor de Ciências da Terra	Encerramento

✓ Selecionar

Figura 4 - Indicar participação

Fonte: O autor (2017)

#### 4.5 LISTA DE COMISSÃO

Página com a lista das comissões referente ao evento selecionado, aqui o usuário pode pesquisar as comissões, alterar ou iniciar um cadastro.

**SIEPE 2017**

- Principal
- Participação
- Evento
- Comissão
- Atribuir
- Trabalho

### Comissão - Listar

Filtros de pesquisa

Descrição:  Tipo Comissão: --- Selecione ▾

Descrição	Tipo Comissão	Membros	Ação
Comissão avaliadora	Avaliadora	Rodrigo Perez Furtado	<input type="button" value="✎"/> <input type="button" value="🗑"/>
Comissão organizadora	Organizadora	Rodrigo Perez Furtado Lucimara dos Santos Jose da Silva	<input type="button" value="✎"/> <input type="button" value="🗑"/>

Página: 1 de 1 | Total: 2 registros ⏪ ⏩ 1 ⏪ ⏩ 10 registros por página

Figura 5 - Lista de comissão

Fonte: O autor (2017)

## 4.6 CADASTRO DE COMISSÃO

Página para cadastro de comissão, aqui o usuário insere as informações da comissão, simultaneamente adiciona os membros da comissão.

**SIEPE 2017**

Principal  
Participação  
Evento  
Comissão  
Atribuir  
Trabalho

**Comissão - Novo**

**Comissão**

Descrição: \*

Tipo Comissão: \*  
--- Selecione ▼

**Membros**

Nome ou CPF: \*

Função: \*  
--- Selecione ▼

Comissão		
Membro	Função	Ação
Nenhum registro foi encontrado		

Figura 6 - Cadastro de comissão

Fonte: O autor (2017)

## 4.8 ATRIBUIÇÃO DE AVALIADOR NO TRABALHO

Página que a comissão organizadora atribui o avaliador de cada trabalho cadastrado, nesta tela é possível também fazer a alteração do avaliador cadastrado anteriormente.

**SIEPE 2017**

Principal  
Participação  
Evento  
Comissão  
Atribuir  
Trabalho

**Atribuir - Listar**

**Avaliadores**

Comissão: --- Selecione --- Membro da comissão: --- Selecione ---

Filtros de pesquisa

Número:

Atribuir	Evento	Número	Titulo	Área Temática	Situação	Membro
	9ª - SIEPE	20171	SIEPE	Teste	Adequação	Rodrigo Perez Furtado
	9ª - SIEPE	20172	dsaf	Teste	Atribuido	Rodrigo Perez Furtado

Página: 1 de 1 | Total: 2 registros 1 10 registros por página

Figura 7 - Atribuição de avaliador

Fonte: O autor (2017)

## 4.9 LISTA DE TRABALHO

Página com a lista de trabalho referente ao evento selecionado, aqui o usuário pode pesquisar trabalho, visualizar, avaliar e submeter recurso.

**SIEPE 2017**

Principal  
Participação  
Evento  
Comissão  
Atribuir  
Trabalho

**Trabalho - Listar**

Filtros de pesquisa

Número:  Título:  Situação: --- Selecione --- Área Temática: --- Selecione --- Submissor:

Evento	Número	Titulo	Submissor	Área Temática	Situação	Ação
9ª - SIEPE	20171	SIEPE	Rodrigo Perez Furtado	Teste	Adequação	
9ª - SIEPE	20172	dsaf	Rodrigo Perez Furtado	Teste	Atribuido	

Página: 1 de 1 | Total: 2 registros 1 10 registros por página

Figura 8 - Lista de trabalho

Fonte: O autor (2017)

## 4.10 CADASTRO DE TRABALHO

Página para submissão de trabalho, aqui o usuário insere as informações do trabalho podendo inserir fórmulas matemáticas, adiciona as palavras chave e os membros do trabalho. Antes de efetuar a submissão o usuário deve concordar que leu os termos do edital para efetivar a submissão.

**SIEPE 2017**

Principal  
Participação  
Evento  
Comissão  
Atribuir  
Trabalho

**Trabalho - Novo**

Resumo

Área Temática: \* --- Selecione ---  
Título do Resumo: \*

Resumo: \*

Contagem de caracteres: 0/200

Palavra-chave: ?

Participantes

Nome: \* Email: Atuação no  
+ Adicionar

Cpf	Nome	Email	Atuação no trabalho	Ação
Nenhum registro foi encontrado				

Li e concordo com os termos do Edital. [Link para o edital](#)

✓ Salvar ✗ Cancelar

Figura 9 - Trabalho novo

Fonte: O autor (2017)

## **5 CONSIDERAÇÕES FINAIS**

A monografia apresentada atingiu os objetivos propostos utilizando os conhecimentos no curso de especialização em engenharia de software. A metodologia Scrum mostrou-se eficiente no gerenciamento do projeto e desenvolvimento de um produto final de qualidade.

O software fabricado poderá ser ainda aprimorado e utilizado para o gerenciamento de outros eventos da Universidade Federal do Paraná.

## REFERÊNCIAS

BOOCH, G; RUMBAUGH, J e JACOBSON, I: **UML, Guia do Usuário**: tradução; Fábio Freitas da Silva, Rio de Janeiro, Campus ,2000.

FARIA, Thiago. **Java EE 7 com JSF, PrimeFaces e CDI**. São Paulo: AlgaWorks Softwares, 2013.

FERNANDES, Raphaela Galhardo; LIMA, Gleydson de A. Ferreira. **Hibernate com Anotações**. Natal: Novatec, 2007.

GUEDES, Gilleanes Thorwald Araujo: **UML - uma abordagem prática**. São Paulo: Novatec, 2007.

LARMAN, C.: Utilizando UML e Padrões: **Uma introdução à análise e ao projeto orientados a objetos e ao Processo Unificado**, 2ª edição, Bookman, Porto Alegre.

SCHWABER, K. e SUTHERLAND, J.; **The Scrum Guide**, 2017. Disponível em: <<http://www.scrumguides.org/scrum-guide.html>>. Acesso em: 12 maio 2017

**Scrum: metodologia ágil para gestão e planejamento de projetos**, 2017. Disponível em: <<http://www.desenvolvimentoagil.com.br/scrum/>>. Acesso em: 29 outubro 2017

**SIEPE**, 2017. Disponível em: <<http://www.siepe.ufpr.br/>>. Acesso em: 29 outubro 2017

SOMMERVILLE, Ian. **Engenharia de Software**. 9. ed. São Paulo: Pearson Education do Brasil, 2011.

WAZLAWICK, Raul Sidnei. **Engenharia de software: conceitos e práticas**. Rio de Janeiro: Elsevier, 2013.

## APÊNDICE A – DOCUMENTO DE VISÃO

Este documento tem como objetivo fornecer uma visão geral do sistema para todos os envolvidos. Apresentar os problemas atuais e as funcionalidades que o sistema deverá solucionar.

### 1 DESCRIÇÃO DO NEGÓCIO DA EMPRESA

A SIEPE é um evento anual da Universidade Federal do Paraná (UFPR) que integra o Ensino, a Pesquisa e a Extensão com a finalidade de promover e estimular a interação entre docentes, estudantes de graduação, educação profissional, ensino médio e pós-graduação.

### 2 DESCRIÇÃO DO PROBLEMA

O problema	Dificuldade em gerenciar as inscrições do SIEPE
Afeta	Equipe designada para organizar o evento
Cujo impacto é	Grande volume de informação gerenciada manualmente
Uma boa solução seria	Desenvolver um sistema para auxiliar no gerenciamento das inscrições do evento.

### 3 FUNCIONALIDADES DO SISTEMA

#### 3.1 MANTER EVENTO

O sistema permite um dos membros da comissão organizadora criar o evento no sistema.

##### 3.1.1 Manter cronograma

O sistema permite um dos membros da comissão organizadora cadastrar o cronograma do evento durante o cadastro do evento.

##### 3.1.2 Manter área temática

O sistema permite um dos membros da comissão organizadora cadastrar as áreas temáticas do evento durante o cadastro do evento.

### 3.1.3 Manter comissão organizadora

O sistema permite um dos membros da comissão organizadora manter o cadastro dos outros membros da comissão organizadora do evento durante o cadastro do evento.

### 3.2 MANTER PARTICIPAÇÃO NO EVENTO

O sistema permite o participante indicar que irá participar do evento.

### 3.3 MANTER COMISSÃO AVALIADORA

O sistema permite um dos membros da comissão organizadora manter os membros da comissão avaliadora do evento.

### 3.4 CADASTRAR PARECER

O sistema permite um dos membros da comissão avaliadora cadastrar o parecer do trabalho avaliado.

### 3.5 CADASTRAR RECURSO

O sistema permite o participante cadastrar recurso sobre a avaliação do seu trabalho.

### 3.6 MANTER TRABALHO

O sistema permite ao participante manter o trabalho.

### 3.7 GERAR RELATÓRIOS

O sistema permite a comissão organizadora gerar relatórios de participantes e trabalhos submetidos ao evento.

## **4 VISÃO GERAL DO PRODUTO**

O sistema Evento visa o gerenciamento do evento anual SIEPE (Semana Integrada de Ensino Pesquisa e Extensão), futuramente podendo ser utilizado para atender outros eventos na UFPR.

Desenvolvido pela DSI (Divisão de Sistemas de Informação do Centro de Computação Eletrônica), o sistema entrou em produção no dia 25 de maio de 2017.

## **APÊNDICE B – CASOS DE USO NEGOCIAIS**

Este documento tem como objetivo apresentar as principais funcionalidades do sistema descritas em casos de uso negociais.

### **UC001 - Manter evento**

Esse caso de uso permite um dos membros da comissão organizadora manter o evento no sistema.

### **UC002 - Manter comissão**

Esse caso de uso permite um dos membros da comissão organizadora manter o cadastro de comissão do evento.

### **UC003 - Gerar relatórios**

Esse caso de uso permite que a comissão organizadora gere relatórios de participantes e trabalhos submetidos ao evento.

### **UC004 - Avaliar recurso**

Esse caso de uso permite que a comissão avaliadora avalie os recursos abertos de trabalhos reprovados.

### **UC005 - Cadastrar parecer**

Esse caso de uso permite um dos membros da comissão avaliadora cadastrar o parecer do trabalho avaliado.

### **UC006 - Cadastrar participação no evento**

Esse caso de uso permite ao participante indicar que irá participar do evento.

### **UC007 - Cadastrar recurso**

Esse caso de uso permite o participante cadastrar recurso sobre a avaliação do seu trabalho.

### **UC008 - Manter trabalho**

Esse caso de uso permite ao participante manter o trabalho.

## APÊNDICE C – GLOSSÁRIO

Neste documento consta a relação, em ordem alfabética, de palavras ou expressões técnicas utilizadas no documento, acompanhadas das respectivas definições.

**CSS:** Cascading Style Sheets, mecanismo que adiciona estilo ao HTML.

**FRAMEWORK:** Conjunto de funcionalidades que une códigos em comum de vários projetos.

**HIBERNATE:** framework para o mapeamento objeto-relacional escrito na linguagem Java.

**HTML:** HyperText Markup Language, que significa Linguagem de Marcação de Hipertexto.

**JAVA:** Linguagem de programação.

**JAVASCRIPT:** Linguagem de programação interpretada.

**SGBD:** Sistema de gerenciamento de banco de dados.

**XHTML:** eXtensible Hypertext Markup Language, é uma reformulação da linguagem de marcação HTML, baseada em XML.

## **APÊNDICE D – REGRAS DE NEGÓCIO**

Este documento tem como objetivo apresentar as regras de negócio do sistema.

**RN001:** Para submeter trabalho é obrigatório pelo menos um participante do tipo orientador.

**RN002:** Para submeter trabalho é obrigatório pelo menos um participante do tipo autor.

**RN003:** Para submeter trabalho o campo email é obrigatório para o participante do tipo orientador.

**RN004:** Para submeter trabalho o campo email é obrigatório para o participante do tipo autor.

**RN005:** Para submeter trabalho é obrigatório aceitar os termos do edital.

**RN006:** Validar o tamanho mínimo de caracteres cadastrados para submeter o trabalho.

**RN007:** Validar o tamanho máximo de caracteres cadastrados para submeter o trabalho.

**RN008:** Validar a quantidade mínima de palavras chave cadastradas para submeter o trabalho.

**RN009:** Validar a quantidade máxima de palavras chave cadastradas para submeter o trabalho.

**RN010:** Validar se não existe participante repetido para submeter o trabalho.

**RN011:** O sistema deve permitir o cadastro de fórmulas no formato lattes durante a submissão de trabalhos.

## APÊNDICE E – PROTÓTIPO DE INTERFACES

### Cadastro de evento

#### Evento - Novo

Título: *	Edição:	Ano:	Link do edital:
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Submissão de trabalho: <input checked="" type="checkbox"/>	Qtd min palavra chave: <input type="text"/>	Qtd max. palavra chave: <input type="text"/>	Tamanho min. resumo: <input type="text"/>
Tamanho max. resumo: <input type="text"/>			
Descrição: *			
<input type="text"/>			
400			

<b>Área Temática</b> Descrição: <input type="text"/> <input type="button" value="+ Adicionar"/> <table border="1"> <thead> <tr> <th>Descrição</th> <th>Ação</th> </tr> </thead> <tbody> <tr> <td colspan="2">Nenhum registro foi encontrado</td> </tr> </tbody> </table>	Descrição	Ação	Nenhum registro foi encontrado		<b>Comissão Organizadora</b> Participante: <input type="text"/> Função: <input type="text"/> <input type="button" value="+ Adicionar"/> <table border="1"> <thead> <tr> <th>Membro</th> <th>Função</th> <th>Ação</th> </tr> </thead> <tbody> <tr> <td colspan="3">Nenhum registro foi encontrado</td> </tr> </tbody> </table>	Membro	Função	Ação	Nenhum registro foi encontrado		
Descrição	Ação										
Nenhum registro foi encontrado											
Membro	Função	Ação									
Nenhum registro foi encontrado											

<b>Cronograma</b>											
Descrição: <input type="text"/>	Data inicio: <input type="text"/>	Data fim: <input type="text"/>	<input type="button" value="+ Adicionar"/>								
<table border="1"> <thead> <tr> <th>Descrição</th> <th>Data inicio</th> <th>Data fim</th> <th>Ação</th> </tr> </thead> <tbody> <tr> <td colspan="4">Nenhum registro foi encontrado</td> </tr> </tbody> </table>				Descrição	Data inicio	Data fim	Ação	Nenhum registro foi encontrado			
Descrição	Data inicio	Data fim	Ação								
Nenhum registro foi encontrado											

### Cadastro de comissão

#### Comissão - Novo

<b>Comissão</b>	
Descrição: *	Tipo Comissão: *
<input type="text"/>	--- Selecione ---

<b>Membros</b>													
Nome ou CPF: *	Função: *												
<input type="text"/>	--- Selecione ---												
<input type="button" value="↓ Adicionar"/>													
<table border="1"> <thead> <tr> <th colspan="4">Comissão</th> </tr> <tr> <th>Membro</th> <th>Função</th> <th>Editar</th> <th>Ação</th> </tr> </thead> <tbody> <tr> <td colspan="4">Nenhum registro foi encontrado</td> </tr> </tbody> </table>		Comissão				Membro	Função	Editar	Ação	Nenhum registro foi encontrado			
Comissão													
Membro	Função	Editar	Ação										
Nenhum registro foi encontrado													

## Lista de comissão

### Comissão - Listar

Filtros de pesquisa

Descrição:  Tipo Comissão: --- Selecione --- ▼

🔍 Pesquisar 🧼 Limpar

Descrição	Tipo Comissão	Membros	Ação
Nenhum registro foi encontrado			
Página: 1 de 1   Total: 0 registros <span style="margin-left: 20px;">             ⏪ ⏩ ⏴ ⏵           </span> <span style="margin-left: 20px;">10 ▼</span> registros por página			

## Lista de trabalho

### Trabalho - Listar

Filtros de pesquisa

Número:  Título:  Situação: --- Selecione --- ▼ Área Temática: --- Selecione --- ▼ Participante:

🔍 Pesquisar 🧼 Limpar

Evento	Número	Título	Área Temática	Situação	Ação
Nenhum registro foi encontrado					
Página: 1 de 1   Total: 0 registros <span style="margin-left: 20px;">             ⏪ ⏩ ⏴ ⏵           </span> <span style="margin-left: 20px;">10 ▼</span> registros por página <span style="float: right;">📄 📄</span>					

## Avaliação de recurso

### Avaliação de recurso

Trabalho:  
Título do trabalho

Recurso:  
Recurso do trabalho...

Decisão: \*  
--- Selecione --- ▼

Parecer: \*

2000

✔ Salvar
✘ Cancelar
🔍 Visualizar

## Cadastro de parecer

### Cadastro de Parecer

Trabalho:  
Título do trabalho

Comissão:

Membro:

Situação: \*

Parecer: \*

2000

## Cadastrar participação em evento

### Eventos disponíveis

16ª - ENEC 2017				
"Cronograma"				
Descrição	Data inicio	Data fim	Localização	Tipo Cronograma
Abertura	01/10/2017 00:00	31/10/2017 00:00	Setor de Ciências Sociais Aplicadas	Abertura
<input type="button" value="✓ Selecionar"/>				
9ª - SIEPE 2017				
"Cronograma"				
Descrição	Data inicio	Data fim	Localização	Tipo Cronograma
Abertura	01/10/2017 19:00	01/10/2017 21:00	Setor de Ciências Sociais Aplicadas	Abertura
Submissão de Trabalho	09/10/2017 00:00	21/10/2017 00:00	Setor de Ciências da Terra	Apresentação de Trabalho
Fechamento	31/10/2017 11:00	31/10/2017 12:00	Setor de Ciências da Terra	Encerramento
<input type="button" value="✓ Selecionar"/>				

## Cadastro de recurso

### Cadastro de recurso

**Trabalho:**  
Titulo do trabalho

**Avaliação:**  
Resultado da avaliação pela banca

**Comissão:**

**Membro:**

**Justificativa: \***

2000

Salvar  Cancelar

## Manter trabalho

### Trabalho - Novo

Resumo

Área Temática: \* Título do Resumo: \*

--- Selecione ---

Resumo: \*

📄 📄 | ↶ ↷ | *I* | Ω *fx*

Contagem de caracteres: 0/200

Palavra-chave: ?

Participantes

Nome: \* Email: Atuação no

✎  --- Selecione ▼ [+ Adicionar](#)

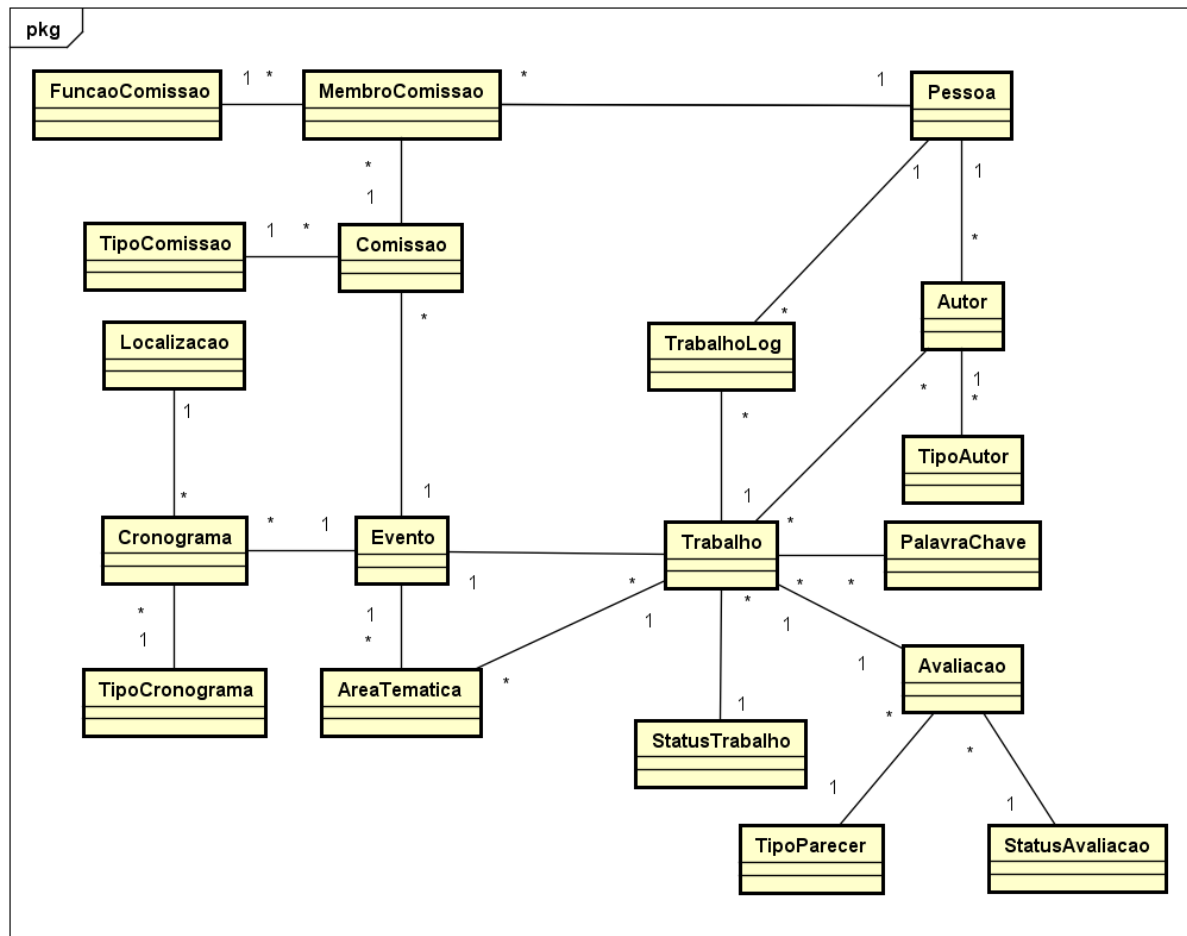
Cpf	Nome	Email	Atuação no trabalho	Ação
Nenhum registro foi encontrado				

Li e concordo com os termos do Edital. [Link para o edital](#)

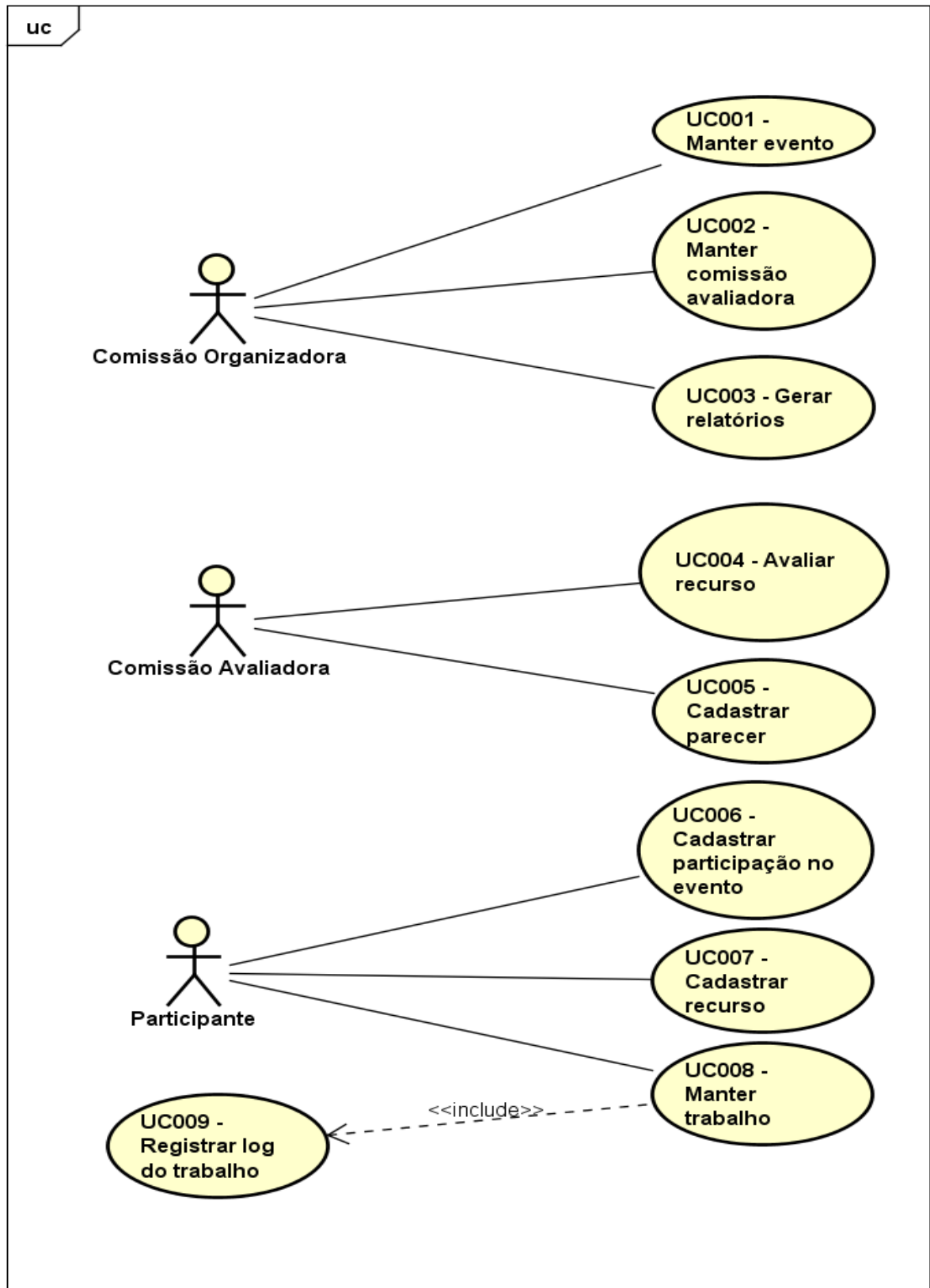
[✓ Salvar](#)

[✗ Cancelar](#)

## APÊNDICE F – MODELO DE OBJETOS NEGOCIAIS



## APÊNDICE G – CASOS DE USO



## 7.1 UC001 - Manter evento

### Descrição

Esse caso de uso permite um dos membros da comissão organizadora manter o evento no sistema.

### Data Views

DV1 - Cadastro de evento (APÊNDICE 5 – PROTÓTIPO DE INTERFACES)

### Pré-condições

Este caso de uso não possui pré-condições

### Pós-condições

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

1. Ter salvo os dados do evento.

### Ator Primário

Comissão Organizadora

### Fluxo de Eventos Principal

1. O sistema apresenta a tela (DV1)
2. O usuário preenche os campos da tela. (A1)
3. O usuário clica no botão “Salvar”
4. O sistema verifica os campos da tela. (E1)
5. O sistema cadastra o evento.
6. O sistema é redirecionado para a tela com a listagem de eventos.
7. O caso de uso é finalizado.

### Fluxos Alternativos

A1. Botão “Cancelar” pressionado.

1. O sistema é redirecionado para a tela com a listagem de eventos.
2. O caso de uso é finalizado

### Fluxos de Exceção

E1. Campos obrigatórios não preenchidos

1. O sistema destaca os campos de preenchimento obrigatório que não foram preenchidos.
2. O caso de uso é reiniciado.

## 7.2 UC002 - Manter comissão

### Descrição

Esse caso de uso permite um dos membros da comissão organizadora manter o cadastro de comissão do evento.

### Data Views

DV2 - Cadastro de comissão (APÊNDICES 5 – PROTÓTIPO DE INTERFACES)

### Pré-condições

Este caso de uso não possui pré-condições

### Pós-condições

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

1. Ter salvo os dados de comissão.

### Ator Primário

Comissão Organizadora

### Fluxo de Eventos Principal

1. O sistema apresenta a tela (DV2)
2. O usuário preenche os campos da tela. (A1)
3. O usuário clica no botão “Salvar”
4. O sistema verifica os campos da tela. (E1)
5. O sistema cadastra a comissão.
6. O sistema é redirecionado para a tela com a listagem de comissões.
7. O caso de uso é finalizado.

### Fluxos Alternativos

A1. Botão “Cancelar” pressionado.

1. O sistema é redirecionado para a tela com a listagem de comissões.
2. O caso de uso é finalizado

### Fluxos de Exceção

E1. Campos obrigatórios não preenchidos

1. O sistema destaca os campos de preenchimento obrigatório que não foram preenchidos.
2. O caso de uso é reiniciado.

### 7.3 UC003 - Gerar relatório de trabalho

#### Descrição

Esse caso de uso permite que a comissão organizadora gere relatórios de trabalhos submetidos ao evento.

#### Data Views

DV3 - Lista de trabalho(APÊNDICES 5 – PROTÓTIPO DE INTERFACES)

#### Pré-condições

Este caso de uso não possui pré-condições.

#### Pós-condições

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

1. Exibir a listagem de trabalhos submetidos e possibilitar gerar relatório segundo os filtros da tela em pdf ou excel.

#### Ator Primário

Comissão Organizadora

#### Fluxo de Eventos Principal

1. O sistema apresenta a tela (DV3)
2. O usuário preenche os filtros na tela. (A1)
3. O usuário clica no botão “Pesquisar”
4. O sistema exibe a tabela com os dados pesquisados
7. O caso de uso é finalizado.

#### Fluxos Alternativos

A1. Botão “Limpar” pressionado

1. O sistema limpa os filtros
2. O caso de uso é reiniciado.

### 7.4 UC004 - Avaliar recurso

#### Descrição

Esse caso de uso permite que a comissão avaliadora avalie os recursos abertos de trabalhos reprovados.

#### Data Views

DV4 - Avaliação de recurso(APÊNDICES 5 – PROTÓTIPO DE INTERFACES)

#### Pré-condições

Selecionar um trabalho com recurso cadastrado.

**Pós-condições**

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

1. Ter salvo os dados da avaliação do recurso.

**Ator Primário**

Comissão Avaliadora

**Fluxo de Eventos Principal**

1. O sistema apresenta a tela (DV4)
2. O usuário preenche os campos da tela. (A1)(A2)
3. O usuário clica no botão “Salvar”
4. O sistema verifica os campos da tela. (E1)
5. O sistema cadastra a avaliação do recurso
6. O sistema é redirecionado para a tela com a listagem de trabalhos
7. O caso de uso é finalizado.

**Fluxos Alternativos**

A1. Botão “Cancelar” pressionado.

1. O sistema é redirecionado para a tela com a listagem de trabalhos
2. O caso de uso é finalizado

A2. Botão “Visualizar” pressionado.

1. O sistema apresenta o recurso aberto pelo participante
2. O caso de uso é reiniciado

**Fluxos de Exceção**

E1. Campos obrigatórios não preenchidos

1. O sistema destaca os campos de preenchimento obrigatório que não foram preenchidos.
2. O caso de uso é reiniciado.

**7.5 UC005 - Cadastrar parecer****Descrição**

Esse caso de uso permite um dos membros da comissão avaliadora cadastrar o parecer do trabalho avaliado.

**Data Views**

DV5 - Cadastro de parecer (APÊNDICES 5 – PROTÓTIPO DE INTERFACES)

**Pré-condições**

Selecionar um trabalho.

**Pós-condições**

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

1. Ter salvo os dados do parecer.

**Ator Primário**

Comissão Avaliadora

**Fluxo de Eventos Principal**

1. O sistema apresenta a tela (DV5)
2. O usuário preenche os campos da tela. (A1)(A2)
3. O usuário clica no botão “Salvar”
4. O sistema verifica os campos da tela. (E1)
5. O sistema cadastra a avaliação do recurso
6. O sistema é redirecionado para a tela com a listagem de trabalhos
7. O caso de uso é finalizado.

**Fluxos Alternativos**

A1. Botão “Cancelar” pressionado.

1. O sistema é redirecionado para a tela com a listagem de trabalhos
2. O caso de uso é finalizado

A2. Botão “Visualizar” pressionado.

1. O sistema apresenta o trabalho selecionado
2. O caso de uso é reiniciado

**Fluxos de Exceção**

E1. Campos obrigatórios não preenchidos

1. O sistema destaca os campos de preenchimento obrigatório que não foram preenchidos.
2. O caso de uso é reiniciado.

**7.6 UC006 - Cadastrar participação no evento****Descrição**

Esse caso de uso permite ao participante indicar que irá participar do evento.

**Data Views**

DV6 - Cadastrar participação no evento(APÊNDICES 5 – PROTÓTIPO DE INTERFACES)

**Pré-condições**

Este caso de uso não possui pré-condições

**Pós-condições**

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

1. Ter salvo a indicação de participação no evento.

**Ator Primário**

Participante

**Fluxo de Eventos Principal**

1. O sistema apresenta a tela (DV6)
2. O usuário clica no evento que pretende participar.
6. O sistema é redirecionado para a tela com a listagem de trabalhos
7. O caso de uso é finalizado.

**7.7 UC007 - Cadastrar recurso****Descrição**

Esse caso de uso permite o participante cadastrar recurso sobre a avaliação do seu trabalho.

**Data Views**

DV7 - Cadastro de recurso(APÊNDICES 5 – PROTÓTIPO DE INTERFACES)

**Pré-condições**

Selecionar um trabalho avaliado.

**Pós-condições**

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

1. Ter salvo os dados do recurso.

**Ator Primário**

Participante

**Fluxo de Eventos Principal**

1. O sistema apresenta a tela (DV7)
2. O usuário preenche os campos da tela. (A1)(A2)
3. O usuário clica no botão “Salvar”
4. O sistema verifica os campos da tela. (E1)
5. O sistema cadastra o recurso
6. O sistema é redirecionado para a tela com a listagem de trabalhos

7. O caso de uso é finalizado.

### **Fluxos Alternativos**

A1. Botão “Cancelar” pressionado.

1. O sistema é redirecionado para a tela com a listagem de trabalhos

2. O caso de uso é finalizado

A2. Botão “Visualizar” pressionado.

1. O sistema apresenta o trabalho selecionado

2. O caso de uso é reiniciado

### **Fluxos de Exceção**

E1. Campos obrigatórios não preenchidos

1. O sistema destaca os campos de preenchimento obrigatório que não foram preenchidos.

2. O caso de uso é reiniciado.

## **7.8 UC008 - Manter trabalho**

### **Descrição**

Esse caso de uso permite ao participante manter o trabalho.

### **Data Views**

DV8 - Manter trabalho(APÊNDICES 5 – PROTÓTIPO DE INTERFACES)

### **Pré-condições**

Este caso de uso não possui pré-condições.

### **Pós-condições**

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

1. Ter salvo os dados do trabalho.

### **Ator Primário**

Participante

### **Fluxo de Eventos Principal**

1. O sistema apresenta a tela (DV8)

2. O usuário preenche os campos da tela. (A1)

3. O usuário clica no botão “Salvar” (RN001 a RN012)

4. O sistema verifica os campos da tela. (E1)

5. O sistema cadastra o trabalho

6. O sistema grava o log do trabalho. (UC009)

7. O sistema é redirecionado para a tela com a listagem de trabalhos

8. O caso de uso é finalizado.

### **Fluxos Alternativos**

A1. Botão “Cancelar” pressionado.

1. O sistema é redirecionado para a tela com a listagem de trabalhos

2. O caso de uso é finalizado

### **Fluxos de Exceção**

E1. Campos obrigatórios não preenchidos

1. O sistema destaca os campos de preenchimento obrigatório que não foram preenchidos.

2. O caso de uso é reiniciado.

## **7.9 UC009 - Manter trabalho**

### **Descrição**

Esse caso de uso grava o log das alterações do trabalho.

### **Pré-condições**

Execução do UC008.

### **Pós-condições**

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

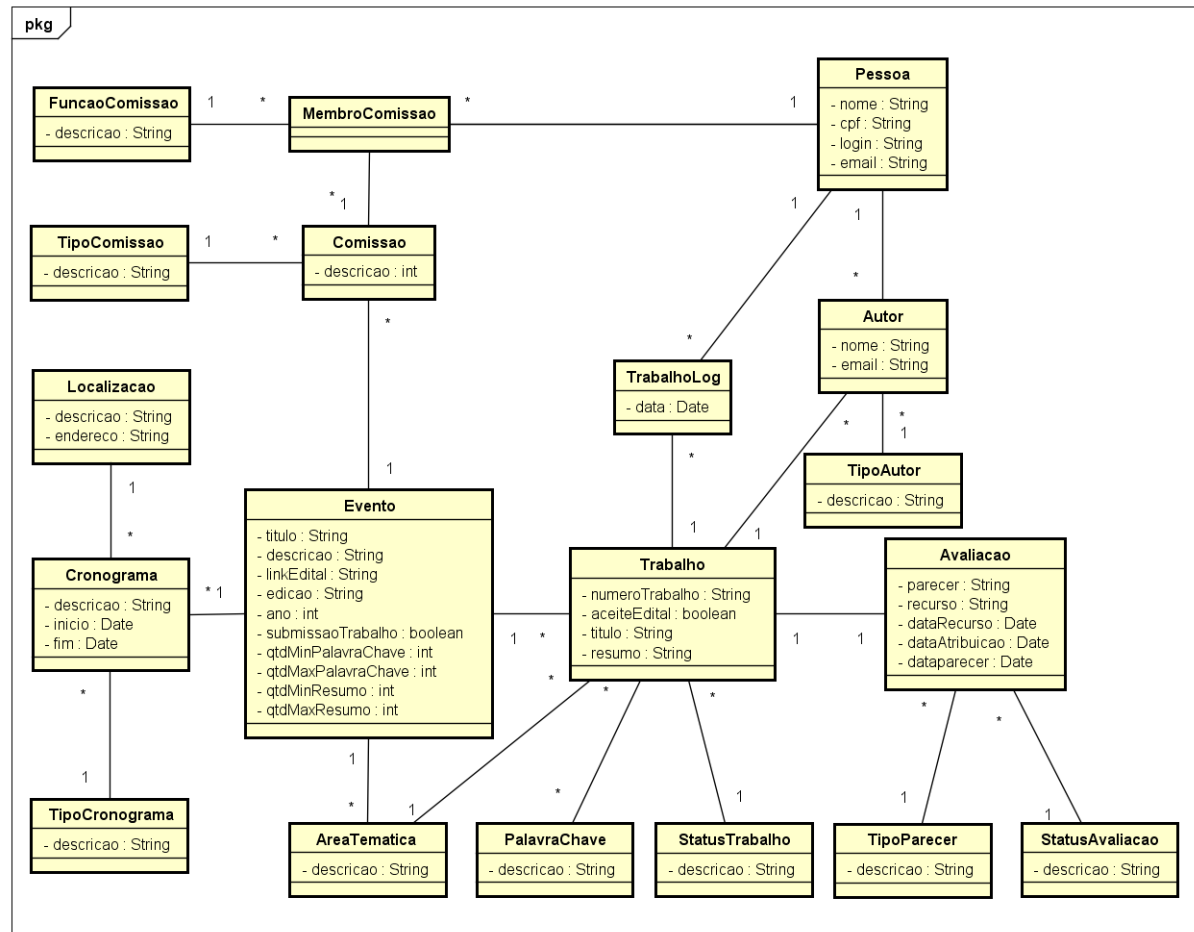
1. Ter salvo o log do trabalho.

### **Fluxo de Eventos Principal**

1. O sistema grava o log do trabalho.

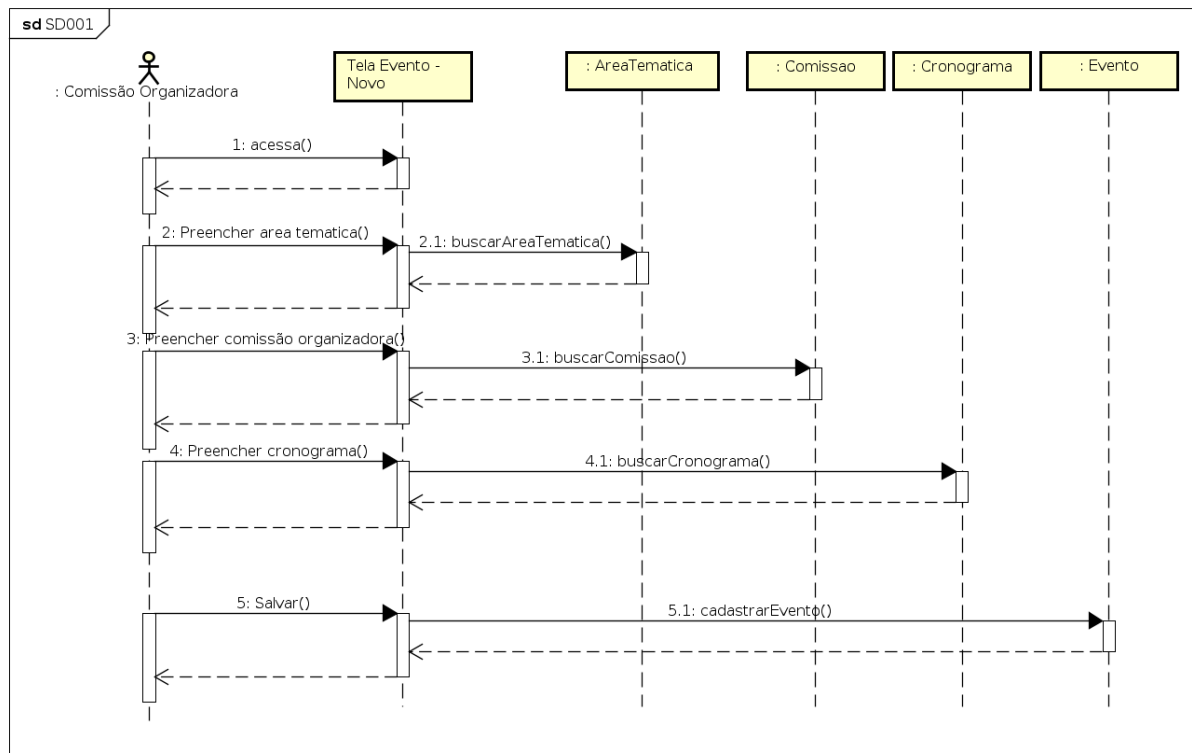
2. O caso de uso é finalizado.

## APÊNDICE H – MODELO DE OBJETOS



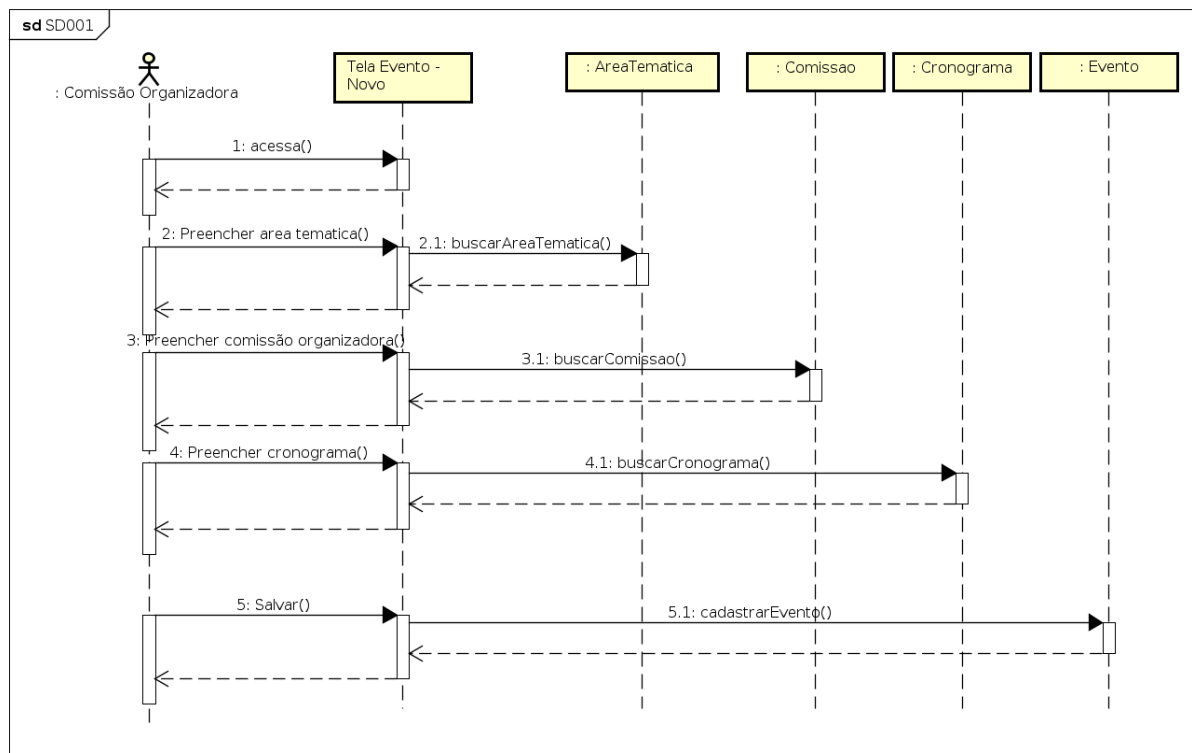
## APÊNDICE I – DIAGRAMAS DE SEQUÊNCIA

### SD001 - Manter evento



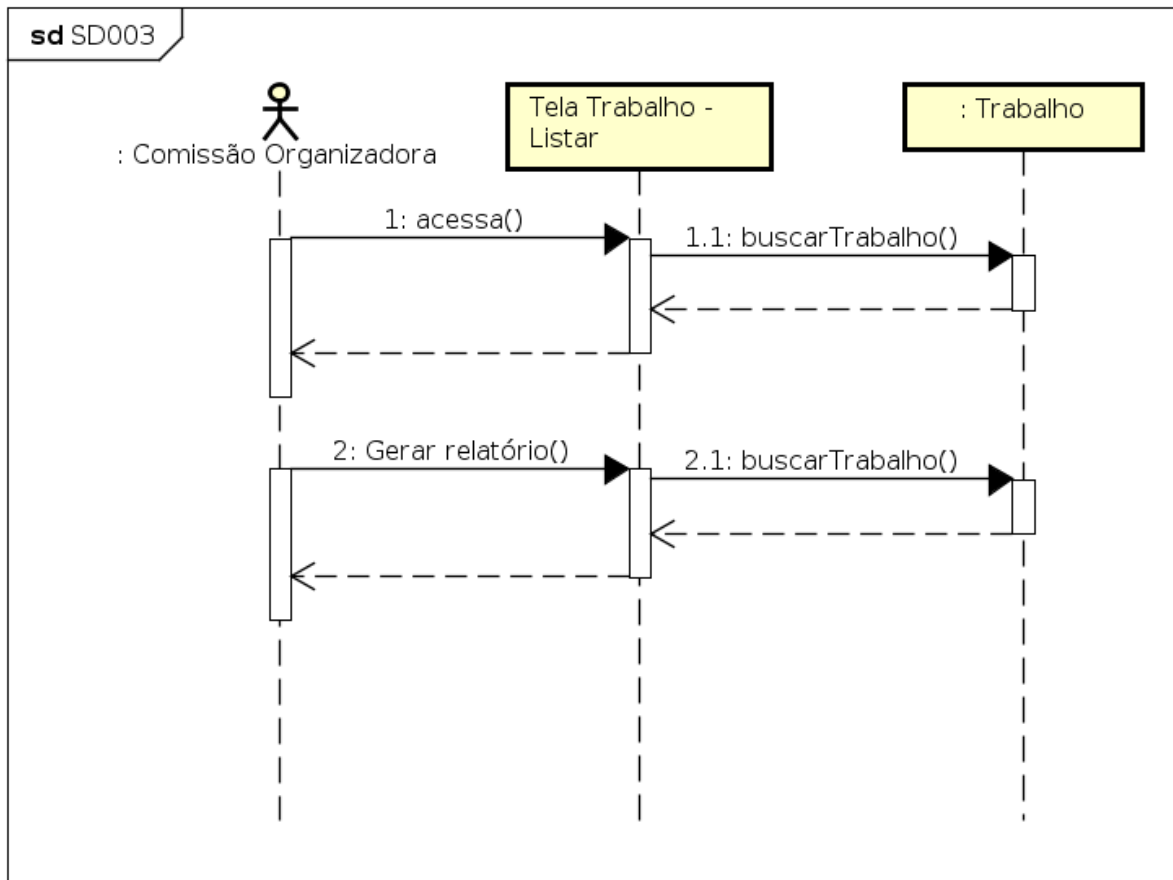
powered by Astah

### SD002 - Manter comissão avaliadora



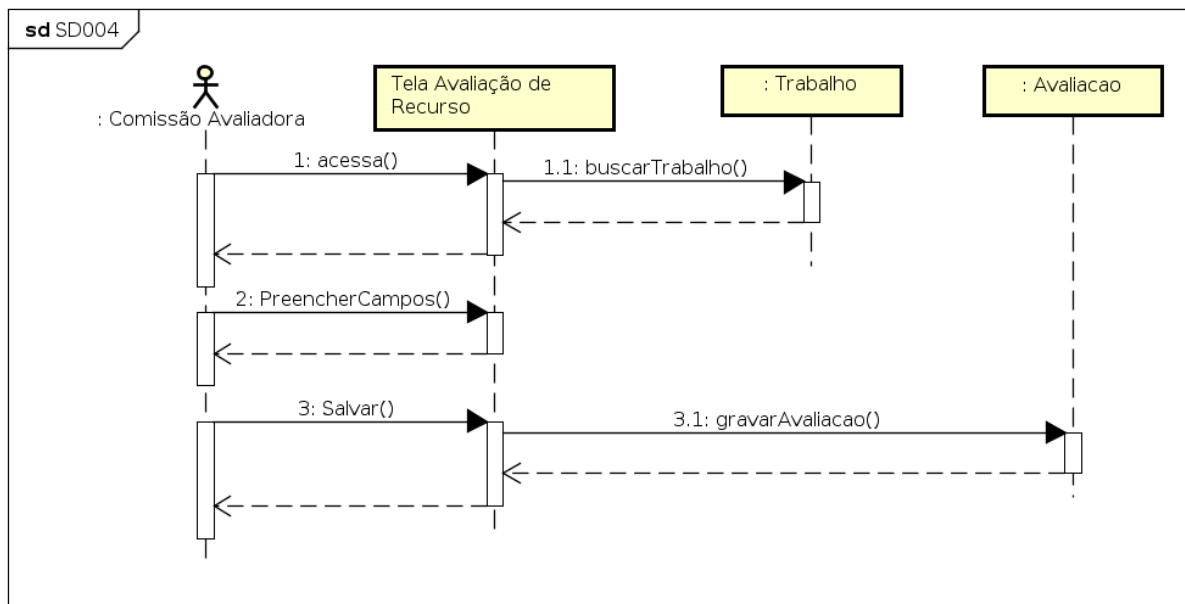
powered by Astah

### SD003 - Gerar relatórios



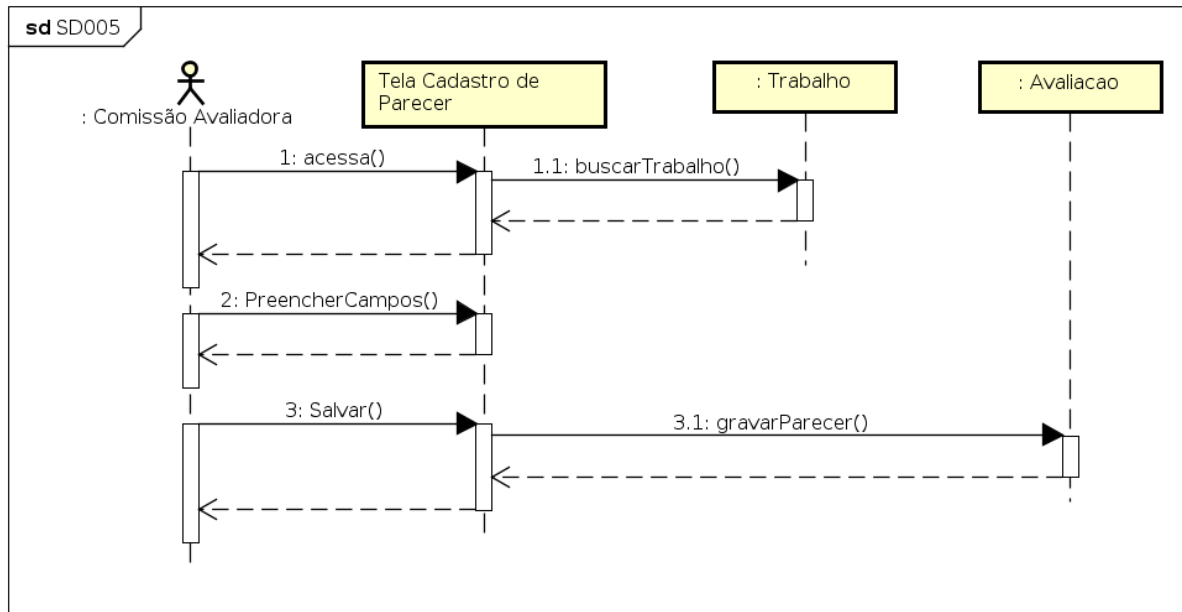
powered by Astah

### SD004 - Avaliar recurso



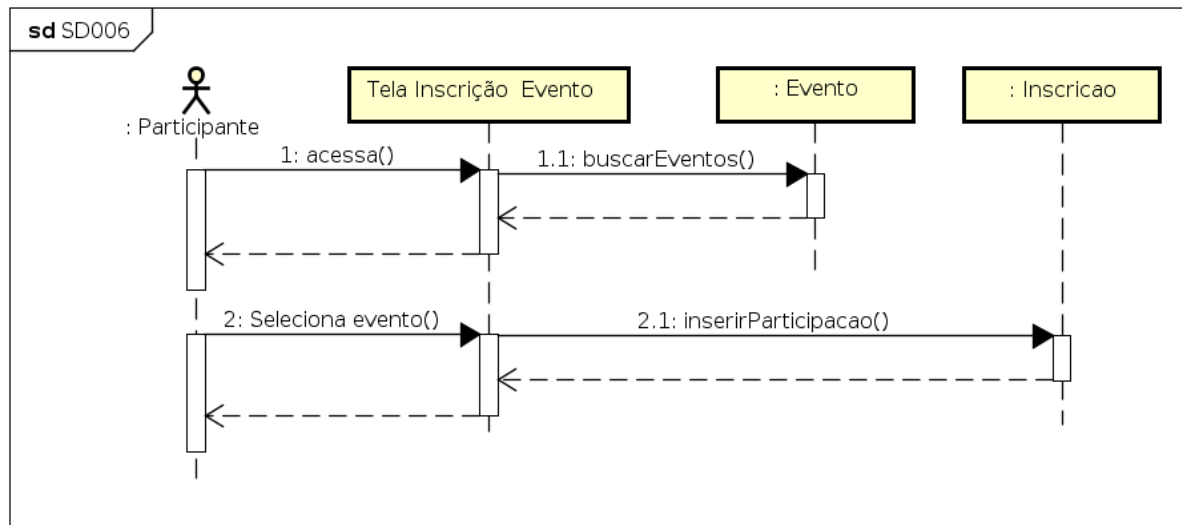
powered by Astah

### SD005 - Cadastrar parecer



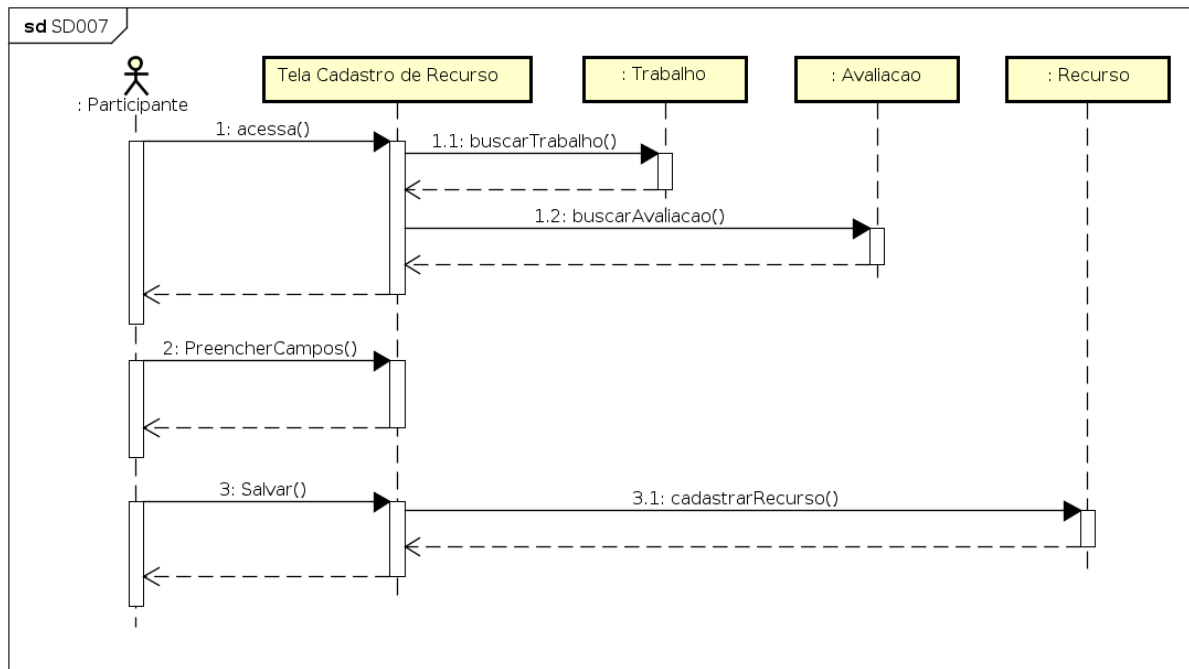
powered by Astah

### SD006 - Cadastrar participação no evento



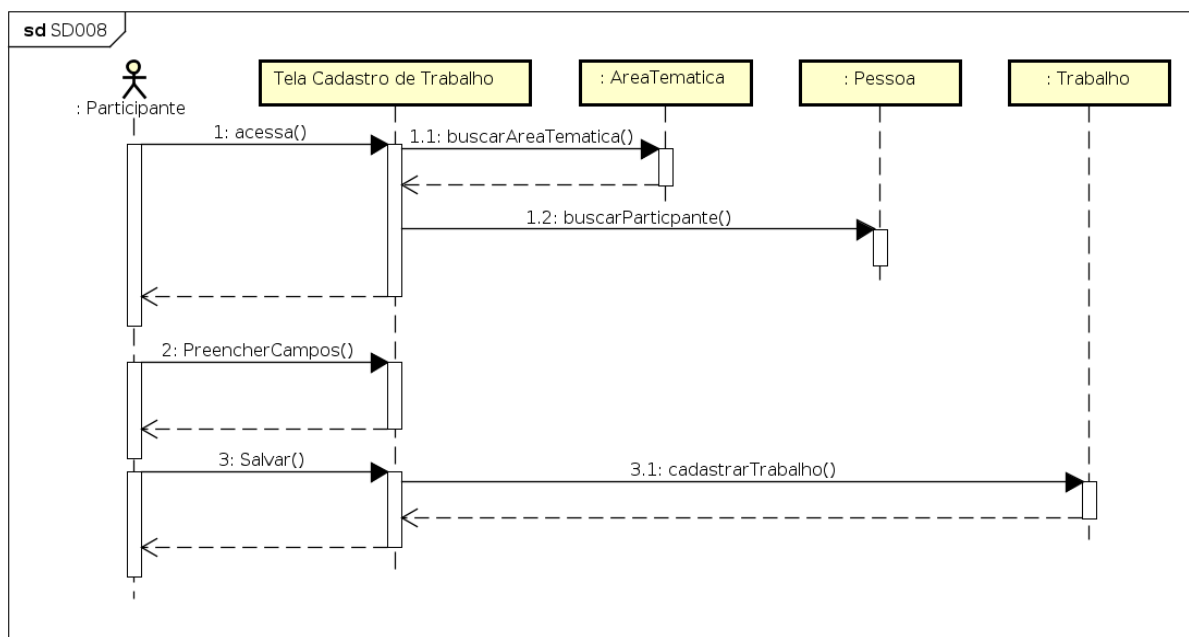
powered by Astah

## SD007 - Cadastrar recurso



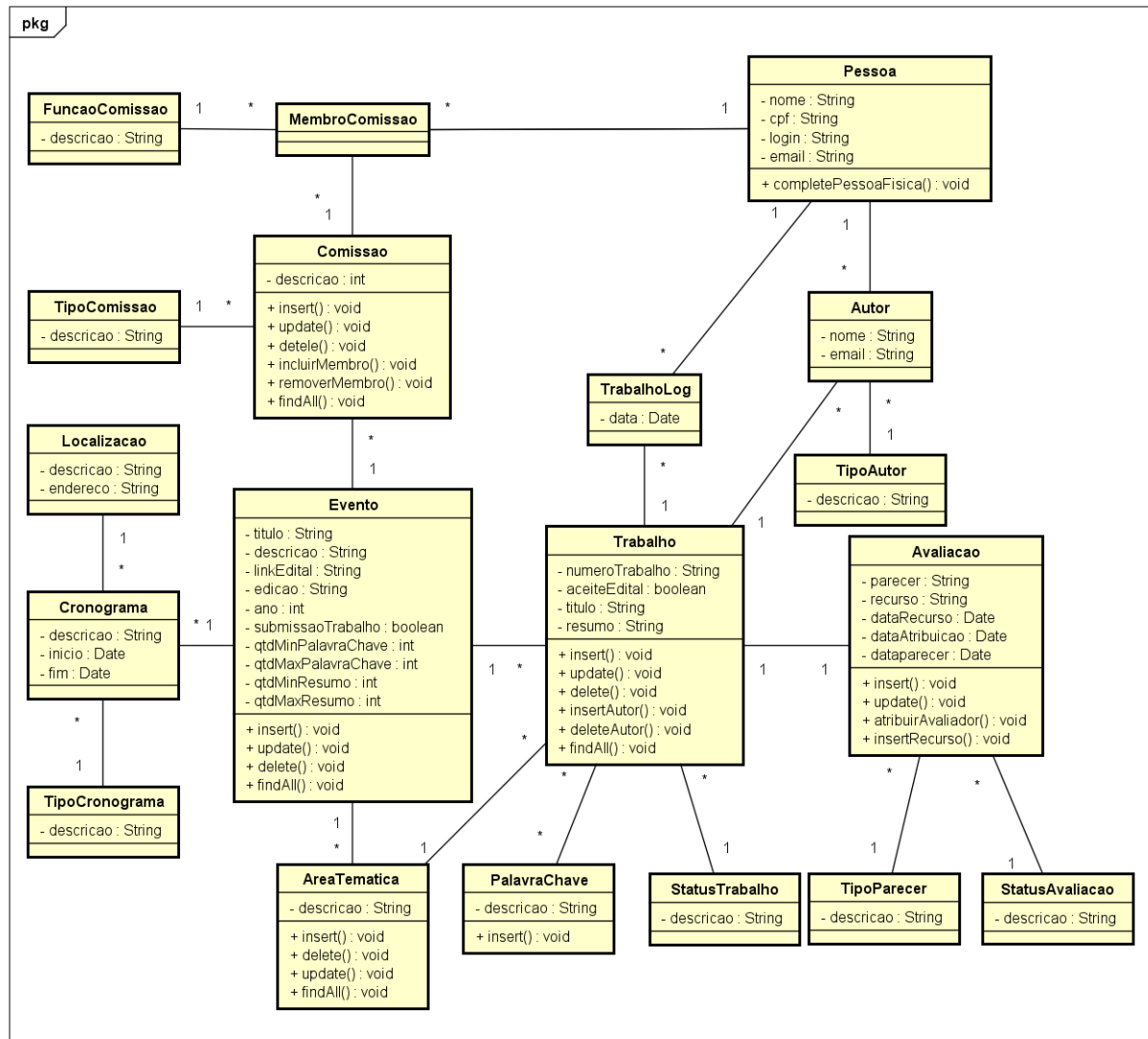
powered by Astah

## SD008 - Manter trabalho

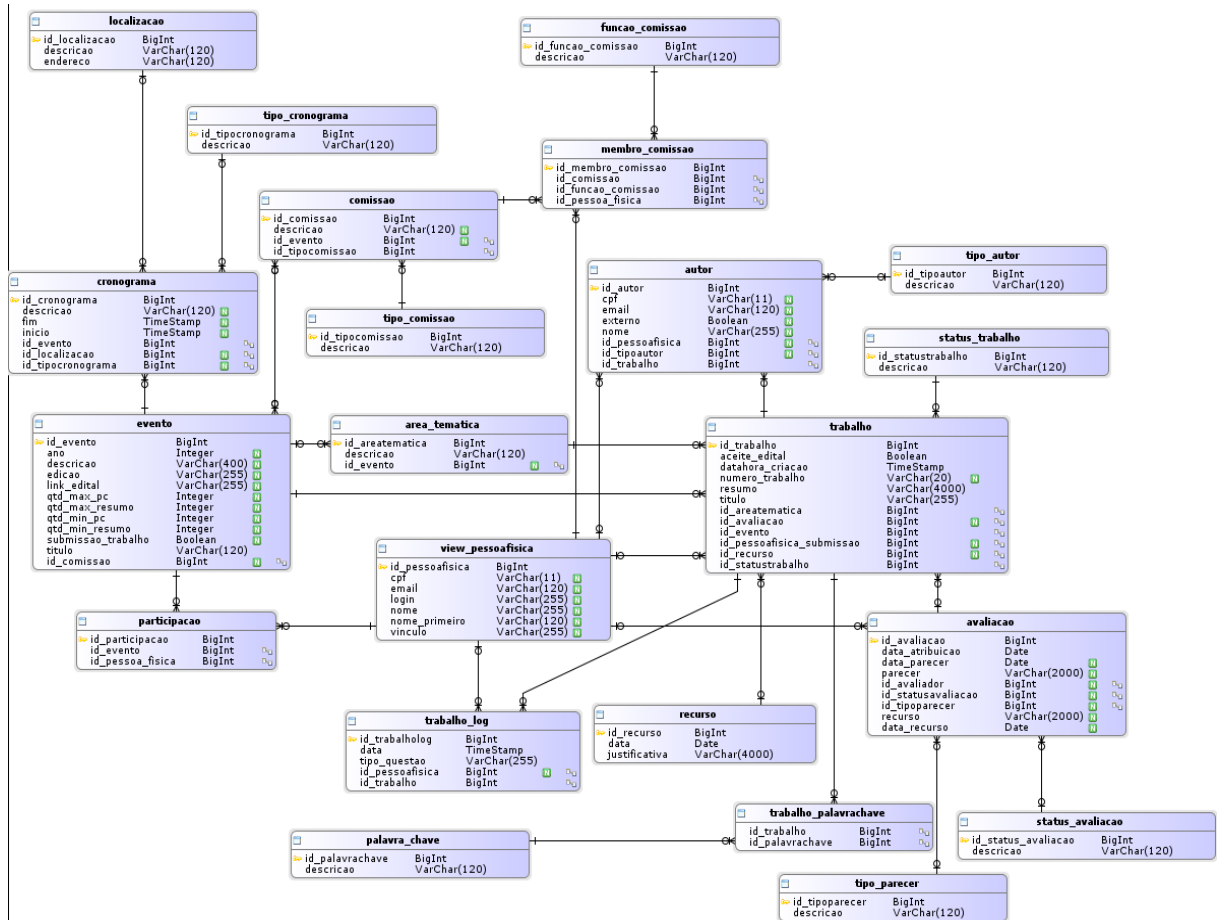


powered by Astah

## APÊNDICE J – MODELO DE OBJETOS



## APÊNDICE K – MODELO FÍSICO DE DADOS



### 11.1 - SCRIPT

```
create table EVENTO.AREA_TEMATICA (
    ID_AREATEMATICA int8 not null,
    descricao varchar(120) not null,
    ID_EVENTO int8,
    primary key (ID_AREATEMATICA)
);
```

```
create table EVENTO.AUTOR (
    ID_AUTOR int8 not null,
    CPF varchar(11),
    EMAIL varchar(120),
    EXTERNO boolean,
    NOME varchar(255),
```

```
ID_PESSOA FISICA int8,  
ID_TIPOAUTOR int8,  
ID_TRABALHO int8 not null,  
primary key (ID_AUTOR)  
);
```

```
create table EVENTO.AVALIACAO (  
  ID_AVALIACAO int8 not null,  
  DATA_ATRIBUICAO date not null,  
  DATA_PARECER date,  
  DATA_RECURSO date,  
  parecer varchar(2000),  
  recurso varchar(2000),  
  ID_AVALIADOR int8,  
  ID_STATUSAVALIACAO int8,  
  ID_TIPO_PARECER int8,  
  primary key (ID_AVALIACAO)  
);
```

```
create table EVENTO.COMISSAO (  
  ID_COMISSAO int8 not null,  
  descricao varchar(120),  
  ID_EVENTO int8 not null,  
  ID_TIPOCOMISSAO int8 not null,  
  primary key (ID_COMISSAO)  
);
```

```
create table EVENTO.CRONOGRAMA (  
  ID_CRONOGRAMA int8 not null,  
  descricao varchar(120),  
  FIM timestamp,  
  INICIO timestamp,  
  ID_EVENTO int8 not null,  
  ID_LOCALIZACAO int8,
```

```
ID_TIPOCRONOGRAMA int8,  
primary key (ID_CRONOGRAMA)  
);
```

```
create table EVENTO.EVENTO (  
  ID_EVENTO int8 not null,  
  ano int4,  
  descricao varchar(400),  
  edicao varchar(255),  
  LINK_EDITAL varchar(255),  
  QTD_MAX_PC int4,  
  QTD_MAX_RESUMO int4,  
  QTD_MIN_PC int4,  
  QTD_MIN_RESUMO int4,  
  SUBMISSAO_TRABALHO boolean,  
  titulo varchar(120) not null,  
  ID_COMISSAO int8,  
  primary key (ID_EVENTO)  
);
```

```
create table EVENTO.FUNCAO_COMISSAO (  
  ID_FUNCAO_COMISSAO int8 not null,  
  descricao varchar(120) not null,  
  primary key (ID_FUNCAO_COMISSAO)  
);
```

```
create table EVENTO.LOCALIZACAO (  
  ID_LOCALIZACAO int8 not null,  
  descricao varchar(120) not null,  
  endereco varchar(120) not null,  
  primary key (ID_LOCALIZACAO)  
);
```

```
create table EVENTO.MEMBRO_COMISSAO (  
  ID_MEMBRO_COMISSAO int8 not null,  
  descricao varchar(120) not null,  
  primary key (ID_MEMBRO_COMISSAO)  
);
```

```
ID_MEMBRO_COMISSAO int8 not null,  
ID_COMISSAO int8 not null,  
ID_FUNCAO_COMISSAO int8 not null,  
ID_PESSOA_FISICA int8 not null,  
primary key (ID_MEMBRO_COMISSAO)  
);
```

```
create table EVENTO.PALAVRA_CHAVE (  
  ID_PALAVRACHAVE int8 not null,  
  descricao varchar(120) not null,  
  primary key (ID_PALAVRACHAVE)  
);
```

```
create table EVENTO.PARTICIPACAO (  
  ID_PARTICIPACAO int8 not null,  
  ID_EVENTO int8 not null,  
  ID_PESSOA_FISICA int8 not null,  
  primary key (ID_PARTICIPACAO)  
);
```

```
create table EVENTO.RECURSO (  
  ID_RECURSO int8 not null,  
  data date not null,  
  justificativa varchar(4000) not null,  
  primary key (ID_RECURSO)  
);
```

```
create table EVENTO.STATUS_AVALIACAO (  
  ID_STATUS_AVALIACAO int8 not null,  
  descricao varchar(120) not null,  
  primary key (ID_STATUS_AVALIACAO)  
);
```

```
create table EVENTO.STATUS_TRABALHO (  
  ID_STATUS_TRABALHO int8 not null,  
  descricao varchar(120) not null,  
  primary key (ID_STATUS_TRABALHO)  
);
```

```
ID_STATUSTRABALHO int8 not null,  
descricao varchar(120) not null,  
primary key (ID_STATUSTRABALHO)  
);
```

```
create table EVENTO.TIPO_AUTOR (  
  ID_TIPOAUTOR int8 not null,  
  descricao varchar(120) not null,  
  primary key (ID_TIPOAUTOR)  
);
```

```
create table EVENTO.TIPO_COMISSAO (  
  ID_TIPOCOMISSAO int8 not null,  
  descricao varchar(120) not null,  
  primary key (ID_TIPOCOMISSAO)  
);
```

```
create table EVENTO.TIPO_CRONOGRAMA (  
  ID_TIPOCRONOGRAMA int8 not null,  
  descricao varchar(120) not null,  
  primary key (ID_TIPOCRONOGRAMA)  
);
```

```
create table EVENTO.TIPO_PARECER (  
  ID_TIPOPARECER int8 not null,  
  descricao varchar(120) not null,  
  primary key (ID_TIPOPARECER)  
);
```

```
create table EVENTO.TRABALHO (  
  ID_TRABALHO int8 not null,  
  ACEITE_EDITAL boolean not null,  
  DATAHORA_CRIACAO timestamp not null,  
  NUMERO_TRABALHO varchar(20),
```

```
resumo varchar(4000) not null,  
titulo varchar(255) not null,  
ID_AREATEMATICA int8 not null,  
ID_AVALIACAO int8,  
ID_EVENTO int8 not null,  
ID_PESSOAFISICA_SUBMISSAO int8,  
ID_RECURSO int8,  
ID_STATUSTRABALHO int8 not null,  
primary key (ID_TRABALHO)  
);
```

```
create table EVENTO.TRABALHO_ARQUIVO (  
  ID_TRABALHOARQUIVO int8 not null,  
  arquivo oid,  
  nome varchar(120),  
  tamanho int4,  
  TIPO_ARQUIVO varchar(120),  
  primary key (ID_TRABALHOARQUIVO)  
);
```

```
create table EVENTO.TRABALHO_LOG (  
  ID_TRABALHOLOG int8 not null,  
  DATA timestamp not null,  
  TIPO_QUESTAO varchar(255) not null,  
  ID_PESSOAFISICA int8,  
  ID_TRABALHO int8 not null,  
  primary key (ID_TRABALHOLOG)  
);
```

```
create table EVENTO.TRABALHO_PALAVRACHAVE (  
  ID_TRABALHO int8 not null,  
  ID_PALAVRACHAVE int8 not null,  
  unique (ID_TRABALHO, ID_PALAVRACHAVE)  
);
```

```
create table EVENTO.VIEW_PESSOAFISICA (  
  ID_PESSOAFISICA int8 not null,  
  CPF varchar(11),  
  EMAIL varchar(120),  
  LOGIN varchar(255),  
  NOME varchar(255),  
  NOME_PRIMEIRO varchar(120),  
  vinculo varchar(255),  
  primary key (ID_PESSOAFISICA)  
);
```

```
alter table EVENTO.AREA_TEMATICA  
  add constraint FK_AREATEMATICA_EVENTO  
  foreign key (ID_EVENTO)  
  references EVENTO.EVENTO;
```

```
alter table EVENTO.AUTOR  
  add constraint FK_AUTOR_PESSOAFISICA  
  foreign key (ID_PESSOAFISICA)  
  references EVENTO.VIEW_PESSOAFISICA;
```

```
alter table EVENTO.AUTOR  
  add constraint FK_AUTOR_TIPO_AUTOR  
  foreign key (ID_TIPOAUTOR)  
  references EVENTO.TIPO_AUTOR;
```

```
alter table EVENTO.AUTOR  
  add constraint FK_AUTOR_TRABALHO  
  foreign key (ID_TRABALHO)  
  references EVENTO.TRABALHO;
```

```
alter table EVENTO.AVALIACAO  
  add constraint FK_AVALIACAO_AVALIADOR
```

```
foreign key (ID_AVALIADOR)
references EVENTO.VIEW_PESSOAFISICA;
```

```
alter table EVENTO.AVALIACAO
add constraint FK_AVALIACAO_TIPO_PARECER
foreign key (ID_TIPOPARECER)
references EVENTO.TIPO_PARECER;
```

```
alter table EVENTO.AVALIACAO
add constraint FK_AVALIACAO_STATUS_AVALIACAO
foreign key (ID_STATUSAVALIACAO)
references EVENTO.STATUS_AVALIACAO;
```

```
alter table EVENTO.COMISSAO
add constraint FK_COMISSAO_TIPOCOMISSAO
foreign key (ID_TIPOCOMISSAO)
references EVENTO.TIPO_COMISSAO;
```

```
alter table EVENTO.COMISSAO
add constraint FK_COMISSAO_EVENTO
foreign key (ID_EVENTO)
references EVENTO.EVENTO;
```

```
alter table EVENTO.CRONOGRAMA
add constraint FK_CRONOGRAMA_LOCALIZACAO
foreign key (ID_LOCALIZACAO)
references EVENTO.LOCALIZACAO;
```

```
alter table EVENTO.CRONOGRAMA
add constraint FK_CRONOGRAMA_EVENTO
foreign key (ID_EVENTO)
references EVENTO.EVENTO;
```

```
alter table EVENTO.CRONOGRAMA
```

```
add constraint FK_CRONOGRAMA_TIPO_CRONOGRAMA
foreign key (ID_TIPOCRONOGRAMA)
references EVENTO.TIPO_CRONOGRAMA;
```

```
alter table EVENTO.EVENTO
add constraint FK_EVENTO_COMISSAO
foreign key (ID_COMISSAO)
references EVENTO.COMISSAO;
```

```
alter table EVENTO.MEMBRO_COMISSAO
add constraint FK_MEMBROCOMISSAO_PESSOAFISICA
foreign key (ID_PESSOA_FISICA)
references EVENTO.VIEW_PESSOAFISICA;
```

```
alter table EVENTO.MEMBRO_COMISSAO
add constraint FK_MEMBROCOMISS_FUNCAOCOMISSAO
foreign key (ID_FUNCAO_COMISSAO)
references EVENTO.FUNCAO_COMISSAO;
```

```
alter table EVENTO.MEMBRO_COMISSAO
add constraint FK_MEMBROCOMISSAO_COMISSAO
foreign key (ID_COMISSAO)
references EVENTO.COMISSAO;
```

```
alter table EVENTO.PARTICIPACAO
add constraint FK_PARTICIPACAO_PESSOAFISICA
foreign key (ID_PESSOA_FISICA)
references EVENTO.VIEW_PESSOAFISICA;
```

```
alter table EVENTO.PARTICIPACAO
add constraint FK_PARTICIPACAL_EVENTO
foreign key (ID_EVENTO)
references EVENTO.EVENTO;
```

```
alter table EVENTO.TRABALHO
  add constraint FK_TRABALHO_AREATEMATICA
  foreign key (ID_AREATEMATICA)
  references EVENTO.AREA_TEMATICA;
```

```
alter table EVENTO.TRABALHO
  add constraint FK_TRABALHO_STATUS
  foreign key (ID_STATUSTRABALHO)
  references EVENTO.STATUS_TRABALHO;
```

```
alter table EVENTO.TRABALHO
  add constraint FK_TRABALHO_RECURSO
  foreign key (ID_RECURSO)
  references EVENTO.RECURSO;
```

```
alter table EVENTO.TRABALHO
  add constraint FK_TRABALHO_AVALIACAO
  foreign key (ID_AVALIACAO)
  references EVENTO.AVALIACAO;
```

```
alter table EVENTO.TRABALHO
  add constraint FK_TRABALHO_EVENTO
  foreign key (ID_EVENTO)
  references EVENTO.EVENTO;
```

```
alter table EVENTO.TRABALHO
  add constraint FK_TRABALHO_PESSOA
  foreign key (ID_PESSOAFISICA_SUBMISSAO)
  references EVENTO.VIEW_PESSOAFISICA;
```

```
alter table EVENTO.TRABALHO_LOG
  add constraint FK_TRABALHOLOG_PESSOAFISICA
  foreign key (ID_PESSOAFISICA)
  references EVENTO.VIEW_PESSOAFISICA;
```

```
alter table EVENTO.TRABALHO_LOG
  add constraint FK_TRABALHOLOG_TRABALHO
  foreign key (ID_TRABALHO)
  references EVENTO.TRABALHO;

alter table EVENTO.TRABALHO_PALAVRACHAVE
  add constraint FKD2FFDADA61AF8AEA
  foreign key (ID_PALAVRACHAVE)
  references EVENTO.PALAVRA_CHAVE;

alter table EVENTO.TRABALHO_PALAVRACHAVE
  add constraint FKD2FFDADAC395B954
  foreign key (ID_TRABALHO)
  references EVENTO.TRABALHO;

create sequence EVENTO.AREA_TEMATICA_SEQ;

create sequence EVENTO.AUTOR_SEQ;

create sequence EVENTO.AVALIACAO_SEQ;

create sequence EVENTO.COMISSAO_SEQ;

create sequence EVENTO.CRONOGRAMA_SEQ;

create sequence EVENTO.EVENTO_SEQ;

create sequence EVENTO.FUNCAO_COMISSAO_SEQ;

create sequence EVENTO.LOCALIZACAO_SEQ;

create sequence EVENTO.MEMBRO_COMISSAO_SEQ;
```

```
create sequence EVENTO.PALAVRA_CHAVE_SEQ;
```

```
create sequence EVENTO.PARTICIPACAO_SEQ;
```

```
create sequence EVENTO.RECURSO_SEQ;
```

```
create sequence EVENTO.STATUS_AVALIACAO_SEQ;
```

```
create sequence EVENTO.TIPO_AUTOR_SEQ;
```

```
create sequence EVENTO.TIPO_COMISSAO_SEQ;
```

```
create sequence EVENTO.TIPO_CRONOGRAMA_SEQ;
```

```
create sequence EVENTO.TIPO_PARECER_SEQ;
```

```
create sequence EVENTO.TRABALHO_ARQUIVO_SEQ;
```

```
create sequence EVENTO.TRABALHO_LOG_SEQ;
```

```
create sequence EVENTO.TRABALHO_SEQ;
```

## APÊNDICE L – PLANO DE TESTES

Este documento define o planejamento de testes do sistema.

### 1 Planejamento do teste - UC001

Os testes serão feitos em 3 baterias, a fim de verificar o fluxo principal, os fluxos alternativos e os de exceção.

**Bateria 1** – Cadastrar um evento e produzir todos os fluxos de exceção.

**Bateria 2** – Alterar os dados cadastrados do evento.

#### 1.1 DETALHAMENTO DAS BATERIAS

##### **Bateria 1**

1. Simular o cadastro de um evento sem preencher os campos obrigatórios.
2. Cadastrar um evento preenchendo todos os campos.

##### **Bateria 2**

1. Editar um evento alterando todos os campos.

### 2 Planejamento do teste - UC002

Os testes serão feitos em 3 baterias, a fim de verificar o fluxo principal, os fluxos alternativos e os de exceção.

**Bateria 1** – Cadastrar uma comissão e produzir todos os fluxos de exceção.

**Bateria 2** – Alterar os dados cadastrados da comissão.

**Bateria 3** – Excluir a comissão cadastrada.

#### 1.1 DETALHAMENTO DAS BATERIAS

##### **Bateria 1**

1. Simular o cadastro de uma comissão sem preencher os campos obrigatórios.

2. Cadastrar uma comissão preenchendo todos os campos.

##### **Bateria 2**

1. Editar uma comissão alterando todos os campos.

##### **Bateria 3**

1. Excluir uma comissão.

### 3 Planejamento do teste - UC003

Os testes serão feitos em 2 baterias, a fim de verificar o fluxo principal e o fluxo alternativo.

**Bateria 1** – Gerar relatório dos trabalhos submetidos.

**Bateria 2** – Limpar os filtros de pesquisa.

#### 1.1 DETALHAMENTO DAS BATERIAS

##### **Bateria 1**

1. Exibir a listagem de trabalhos submetidos e gerar relatório segundo os filtros da tela em pdf ou excel.

##### **Bateria 2**

1. Limpar os filtros de pesquisa.

### 4 Planejamento do teste - UC004

Os testes serão feitos em 2 baterias, a fim de verificar o fluxo principal, os fluxos alternativos e os de exceção.

**Bateria 1** – Avaliar um recurso e produzir todos os fluxos de exceção.

**Bateria 2** – Alterar os dados cadastrados na avaliação.

#### 1.1 DETALHAMENTO DAS BATERIAS

##### **Bateria 1**

1. Simular a avaliação de um recurso sem preencher os campos obrigatórios.
2. Avaliar um recurso preenchendo todos os campos.

##### **Bateria 2**

1. Editar uma avaliação alterando todos os campos.

### 5 Planejamento do teste - UC005

Os testes serão feitos em x baterias, a fim de verificar o fluxo principal, os fluxos alternativos e os de exceção.

**Bateria 1** – Cadastrar um parecer e produzir todos os fluxos de exceção.

**Bateria 2** – Alterar os dados cadastrados do parecer.

## 1.1 DETALHAMENTO DAS BATERIAS

### **Bateria 1**

1. Simular o cadastro de parecer sem preencher os campos obrigatórios.
2. Cadastrar um parecer preenchendo todos os campos.

### **Bateria 2**

1. Editar um parecer alterando todos os campos.

## **6 Planejamento do teste - UC006**

Os testes serão feitos em 1 bateria, a fim de verificar o fluxo principal.

**Bateria 1** – Cadastrar participação no evento.

## 1.1 DETALHAMENTO DAS BATERIAS

### **Bateria 1**

1. Cadastrar participação no evento.

## **7 Planejamento do teste - UC007**

Os testes serão feitos em x baterias, a fim de verificar o fluxo principal, os fluxos alternativos e os de exceção.

**Bateria 1** – Cadastrar um recurso e produzir todos os fluxos de exceção.

**Bateria 2** – Alterar os dados cadastrados do recurso.

## 1.1 DETALHAMENTO DAS BATERIAS

### **Bateria 1**

1. Simular o cadastro de um recurso sem preencher os campos obrigatórios.
2. Cadastrar um recurso preenchendo todos os campos.

### **Bateria 2**

1. Editar um recurso alterando todos os campos.

## **8 Planejamento do teste - UC008**

Os testes serão feitos em x baterias, a fim de verificar o fluxo principal, os fluxos alternativos e os de exceção.

**Bateria 1** – Cadastrar um trabalho e produzir todos os fluxos de exceção.

**Bateria 2** – Alterar os dados cadastrados do trabalho.

## 1.1 DETALHAMENTO DAS BATERIAS

### **Bateria 1**

1. Simular o cadastro de um trabalho sem preencher os campos obrigatórios.
2. Cadastrar um trabalho preenchendo todos os campos.

### **Bateria 2**

1. Editar um trabalho alterando todos os campos.

## APÊNDICE M – CASOS DE TESTES

Este documento tem como objetivo apresentar os casos de teste do sistema.

### 1 Caso de teste - UC001

Nº	Pré-condições	Entrada	Ação	Resultado Esperado
1	Os campos do formulário deverão estar vazios.		Clicar no botão Salvar	O sistema exibe mensagem de erro para todos os campos obrigatórios.
2	Os campos do formulário deverão estar vazios.	Preencher todos os campos do formulário	Clicar no botão Salvar	O sistema exibe mensagem de sucesso e salva as informações no banco de dados.
3	Os campos deverão estar preenchidos conforme o teste 2	Alterar todos os campos do formulário	Clicar no botão Salvar	O sistema exibe mensagem de sucesso e altera as informações no banco de dados.

### 2 Caso de teste - UC002

Nº	Pré-condições	Entrada	Ação	Resultado Esperado
1	Os campos do formulário deverão estar vazios.		Clicar no botão Salvar	O sistema exibe mensagem de erro para todos os campos obrigatórios.
2	Os campos do formulário deverão estar vazios.	Preencher todos os campos do formulário	Clicar no botão Salvar	O sistema exibe mensagem de sucesso e salva as informações no banco de dados.
3	Os campos deverão estar	Alterar todos os campos do	Clicar no botão Salvar	O sistema exibe mensagem de sucesso

	preenchidos conforme o teste 2	formulário		e altera as informações no banco de dados.
4			Clicar no botão Excluir e confirmar a exclusão	O sistema exibe mensagem de sucesso e exclui as informações no banco de dados.

### 3 Caso de teste - UC003

Nº	Pré-condições	Entrada	Ação	Resultado Esperado
1	Ter pelo menos um trabalho cadastrado.		Clicar no botão Pesquisar	O sistema a lista com os trabalhos cadastrados
2	Ter pelo menos um trabalho cadastrado.		Clicar no botão Pesquisar. Clicar no botão PDF	O sistema exibe a lista de trabalhos em um arquivo de formato pdf.
3	Ter pelo menos um trabalho cadastrado.		Clicar no botão Pesquisar. Clicar no botão EXCEL	O sistema exibe a lista de trabalhos em um arquivo de formato excel.
4	Os filtros de busca deverão estar vazios.	Preencher todos os filtros de busca	Clicar no botão Limpar	O sistema retorna todos os filtros para o estado inicial

#### 4 Caso de teste - UC004

Nº	Pre-condições	Entrada	Ação	Resultado Esperado
1	Os campos do formulário deverão estar vazios.		Clicar no botão Salvar	O sistema exibe mensagem de erro para todos os campos obrigatórios.
2	Os campos do formulário deverão estar vazios.	Preencher todos os campos do formulário	Clicar no botão Salvar	O sistema exibe mensagem de sucesso e salva as informações no banco de dados.
3	Os campos deverão estar preenchidos conforme o teste 2	Alterar todos os campos do formulário	Clicar no botão Salvar	O sistema exibe mensagem de sucesso e altera as informações no banco de dados.

#### 5 Caso de teste - UC005

Nº	Pre-condições	Entrada	Ação	Resultado Esperado
1	Os campos do formulário deverão estar vazios.		Clicar no botão Salvar	O sistema exibe mensagem de erro para todos os campos obrigatórios.
2	Os campos do formulário deverão estar vazios.	Preencher todos os campos do formulário	Clicar no botão Salvar	O sistema exibe mensagem de sucesso e salva as informações no banco de dados.
3	Os campos deverão estar preenchidos conforme o teste 2	Alterar todos os campos do formulário	Clicar no botão Salvar	O sistema exibe mensagem de sucesso e altera as informações no banco de dados.

**6 Caso de teste - UC006**

Nº	Pre-condições	Entrada	Ação	Resultado Esperado
1	Ter evento cadastrado no banco de dados		Clicar no botão Inscrição do evento pretendido	O sistema exibe mensagem de sucesso e salva as informações no banco de dados.

**7 Caso de teste - UC007**

Nº	Pre-condições	Entrada	Ação	Resultado Esperado
1	Os campos do formulário deverão estar vazios.		Clicar no botão Salvar	O sistema exibe mensagem de erro para todos os campos obrigatórios.  O sistema destaca todos os campos de preenchimento obrigatório.
2	Os campos do formulário deverão estar vazios.	Preencher todos os campos do formulário	Clicar no botão Salvar	O sistema exibe mensagem de sucesso e salva as informações no banco de dados.

**8 Caso de teste - UC008**

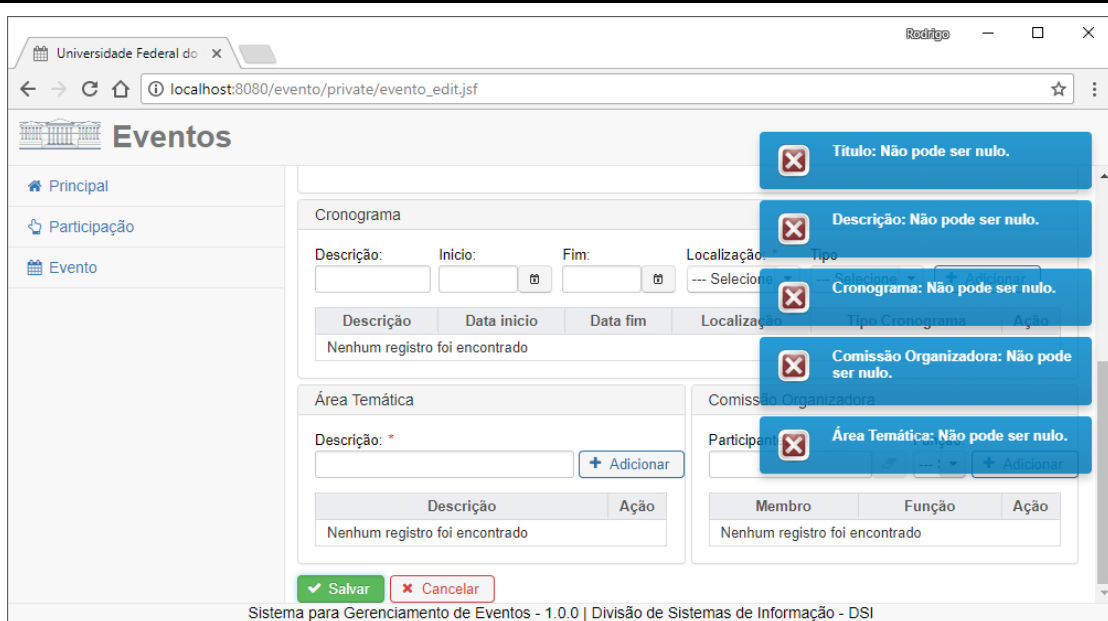
Nº	Pre-condições	Entrada	Ação	Resultado Esperado
1	Os campos do formulário deverão estar vazios.		Clicar no botão Salvar	O sistema exibe mensagem de erro para todos os campos obrigatórios.
2	Os campos do	Preencher todos	Clicar no	O sistema exibe mensagem


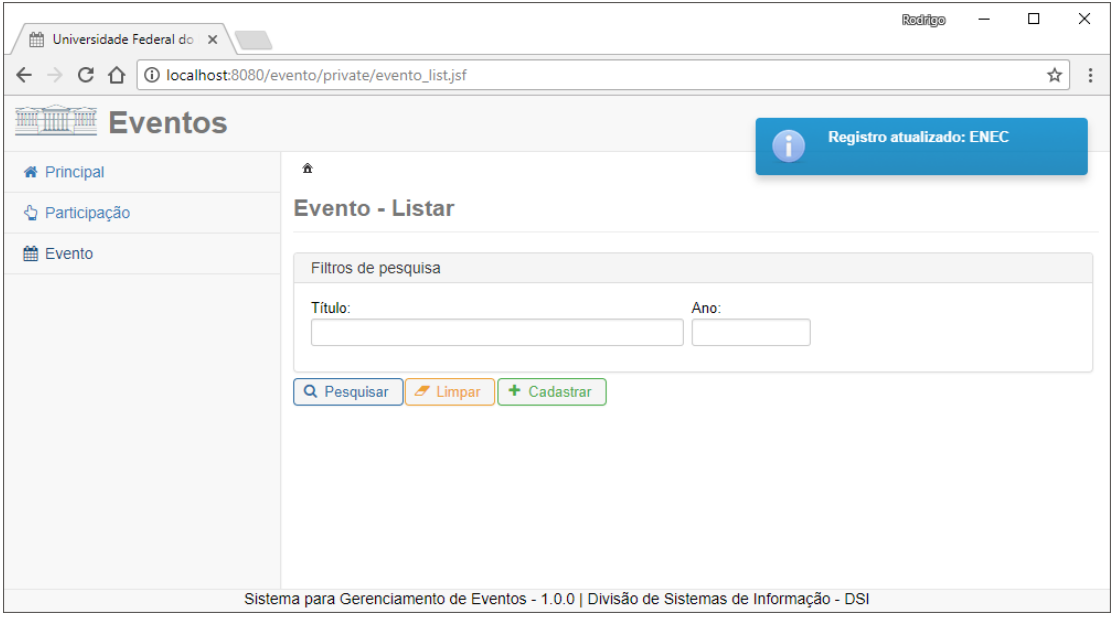
	formulário deverão estar vazios.	os campos do formulário	botão Salvar	de sucesso e salva as informações no banco de dados.
--	--	----------------------------	-----------------	--

## APÊNDICE N – LOG DE TESTES



### 1 Log de teste - UC001

N°	Log de teste	Resultado
1	<p>Título: Não pode ser nulo.</p> <p>Descrição: Não pode ser nulo.</p> <p>Cronograma: Não pode ser nulo.</p> <p>Comissão Organizadora: Não pode ser nulo.</p> <p>Área Temática: Não pode ser nulo</p>	SUCESSO
2	Registro inserido: SIEPE	SUCESSO





	 <p>Universidade Federal do x</p> <p>localhost:8080/evento/private/evento_list.jsf</p> <p><b>Eventos</b></p> <p>Principal   Participação   Evento</p> <p><b>Evento - Listar</b></p> <p>Filtros de pesquisa</p> <p>Título: <input type="text"/> Ano: <input type="text"/></p> <p>Pesquisar   Limpar   Cadastrar</p> <p>Registro inserido: SIEPE</p> <p>Sistema para Gerenciamento de Eventos - 1.0.0   Divisão de Sistemas de Informação - DSI</p>	
3	Registro atualizado: ENEC	SUCESSO
	 <p>Universidade Federal do x</p> <p>localhost:8080/evento/private/evento_list.jsf</p> <p><b>Eventos</b></p> <p>Principal   Participação   Evento</p> <p><b>Evento - Listar</b></p> <p>Filtros de pesquisa</p> <p>Título: <input type="text"/> Ano: <input type="text"/></p> <p>Pesquisar   Limpar   Cadastrar</p> <p>Registro atualizado: ENEC</p> <p>Sistema para Gerenciamento de Eventos - 1.0.0   Divisão de Sistemas de Informação - DSI</p>	

## 2 Log de teste - UC002

Nº	Log de teste	Resultado
1	<p>Descrição: Não pode ser nulo.</p> <p>Tipo Comissão: Não pode ser nulo.</p> <p>Membros: Não pode ser nulo.</p>	SUCESSO
		
2	Registro inserido:	SUCESSO
		
3	Registro atualizado:	SUCESSO

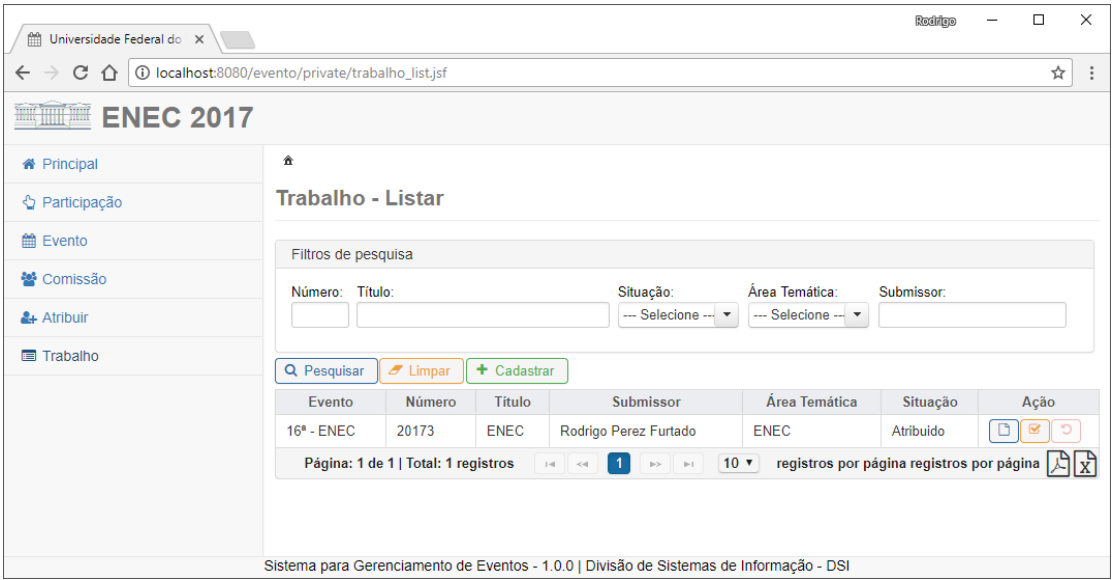
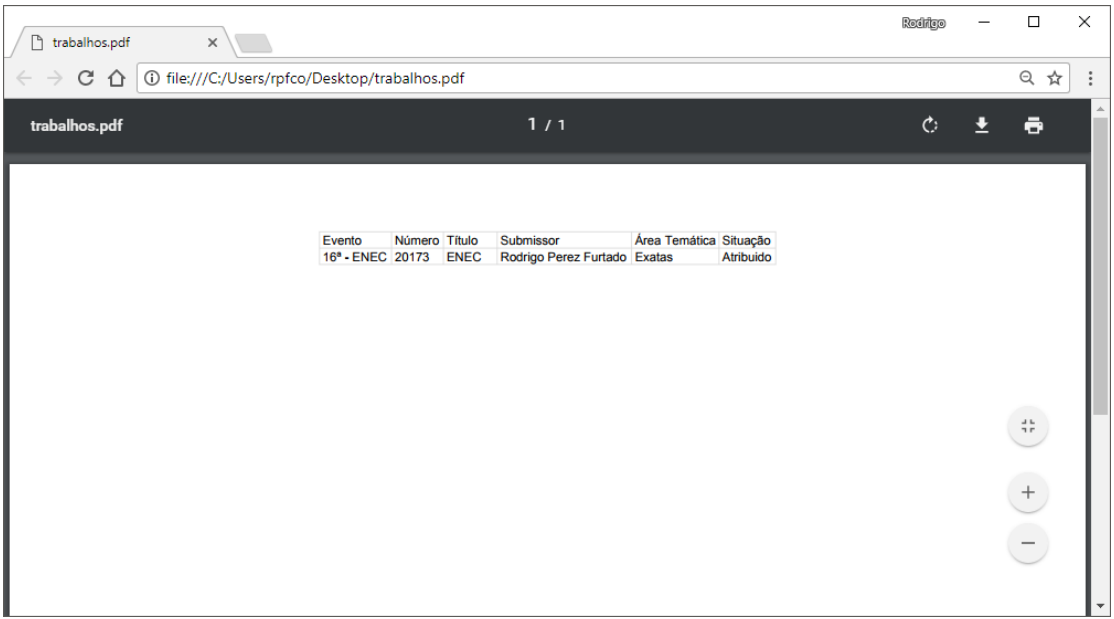
4 Registro excluído:

SUCESSO

Descrição	Tipo Comissão	Membros	Ação
Comissão organizadora	Organizadora	Rodrigo Perez Furtado	 

Página: 1 de 1 | Total: 1 registros | 10 registros por página

## 3 Log de teste - UC003

N°	Log de teste	Resultado														
1	 <p>The screenshot shows a web browser window with the URL localhost:8080/evento/private/trabalho_list.jsf. The page title is 'Trabalho - Listar'. It features a search filter section with fields for 'Número', 'Título', 'Situação', 'Área Temática', and 'Submissor'. Below the filter is a table with the following data:</p> <table border="1"> <thead> <tr> <th>Evento</th> <th>Número</th> <th>Título</th> <th>Submissor</th> <th>Área Temática</th> <th>Situação</th> <th>Ação</th> </tr> </thead> <tbody> <tr> <td>16ª - ENEC</td> <td>20173</td> <td>ENEC</td> <td>Rodrigo Perez Furtado</td> <td>ENEC</td> <td>Atribuído</td> <td>[Icons]</td> </tr> </tbody> </table> <p>At the bottom of the page, it says 'Sistema para Gerenciamento de Eventos - 1.0.0   Divisão de Sistemas de Informação - DSI'.</p>	Evento	Número	Título	Submissor	Área Temática	Situação	Ação	16ª - ENEC	20173	ENEC	Rodrigo Perez Furtado	ENEC	Atribuído	[Icons]	SUCESSO
Evento	Número	Título	Submissor	Área Temática	Situação	Ação										
16ª - ENEC	20173	ENEC	Rodrigo Perez Furtado	ENEC	Atribuído	[Icons]										
2	 <p>The screenshot shows a PDF viewer window with the file path file:///C:/Users/rpfc/Desktop/trabalhos.pdf. The document contains a table with the following data:</p> <table border="1"> <thead> <tr> <th>Evento</th> <th>Número</th> <th>Título</th> <th>Submissor</th> <th>Área Temática</th> <th>Situação</th> </tr> </thead> <tbody> <tr> <td>16ª - ENEC</td> <td>20173</td> <td>ENEC</td> <td>Rodrigo Perez Furtado</td> <td>Exatas</td> <td>Atribuído</td> </tr> </tbody> </table>	Evento	Número	Título	Submissor	Área Temática	Situação	16ª - ENEC	20173	ENEC	Rodrigo Perez Furtado	Exatas	Atribuído	SUCESSO		
Evento	Número	Título	Submissor	Área Temática	Situação											
16ª - ENEC	20173	ENEC	Rodrigo Perez Furtado	Exatas	Atribuído											
3		SUCESSO														

trabalhos.xls - Planilhas

Seguro | https://docs.google.com/spreadsheets/d/1Er3e-ch2xrrveuyKrTITXF9bQJj-1qsDPA-KDceui8/edit#gid=666701933

rpfcool@gmail.com

Comentários Compartilhar

Arquivo Editar Visualizar Inserir Formatar Dados Ferramentas Complementos Ajuda Todas as alter...

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Evento	Número	Título	Submissor	Área Temática	Situação								
2	16ª - ENEC	20173	ENEC	Rodrigo Perez Furtado	Exatas	Atribuído								
3														
4														
5														
6														
7														
8														
9														
10														
11														
12														
13														
14														
15														
16														

listaDataTable

4

SUCESSO

Universidade Federal do

localhost:8080/evento/private/trabalho\_list.jsf

ENEC 2017

Principal Participação Evento Comissão Atribuir Trabalho

Trabalho - Listar

Filtros de pesquisa

Número:  Título:  Situação:  Área Temática:  Submissor:



Pesquisar Limpar Cadastrar

Evento	Número	Título	Submissor	Área Temática	Situação	Ação
16ª - ENEC	20173	ENEC	Rodrigo Perez Furtado	ENEC	Atribuído	

Página: 1 de 1 | Total: 1 registros | 10 registros por página

Sistema para Gerenciamento de Eventos - 1.0.0 | Divisão de Sistemas de Informação - DSI



## 4 Log de teste - UC004

N°	Log de teste	Resultado
1	Situação: Não pode ser nulo. Parecer: Não pode ser nulo.	SUCESSO
		
2	Parecer registrado com sucesso	SUCESSO
		
3	Parecer registrado com sucesso	SUCESSO



### 5 Log de teste - UC005

Nº	Log de teste	Resultado
1	Situação: Não pode ser nulo. Parecer: Não pode ser nulo.	SUCESSO
2	Parecer registrado com sucesso	SUCESSO

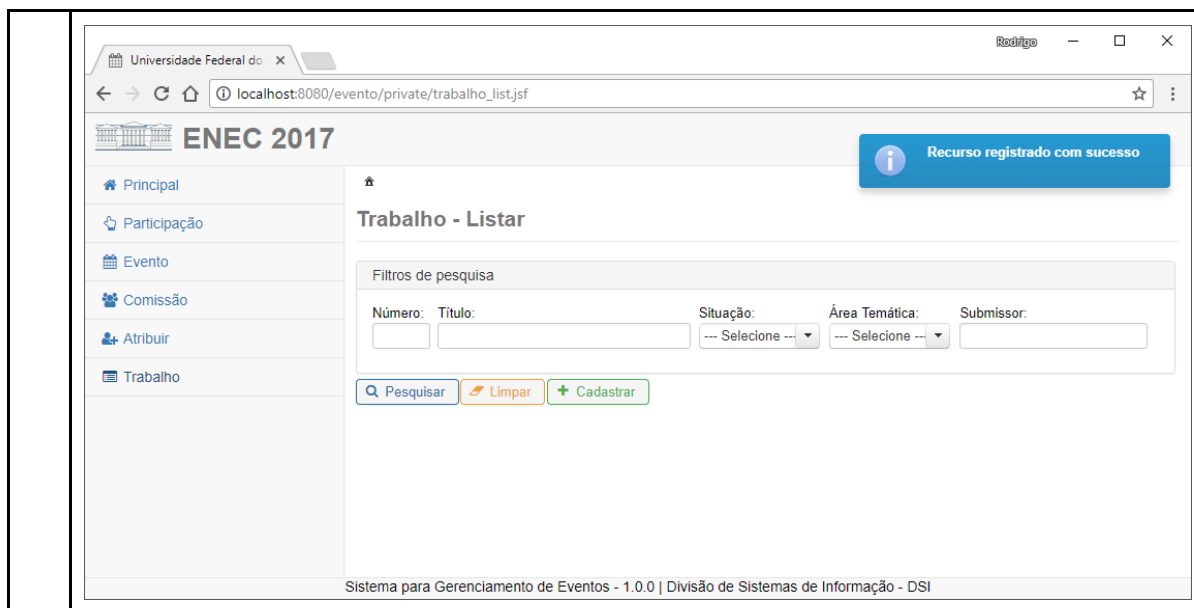
		
3	Parecer registrado com sucesso	SUCESSO
		

## 6 Log de teste - UC006

Nº	Log de teste	Resultado
1	Participação inserida:	SUCESSO

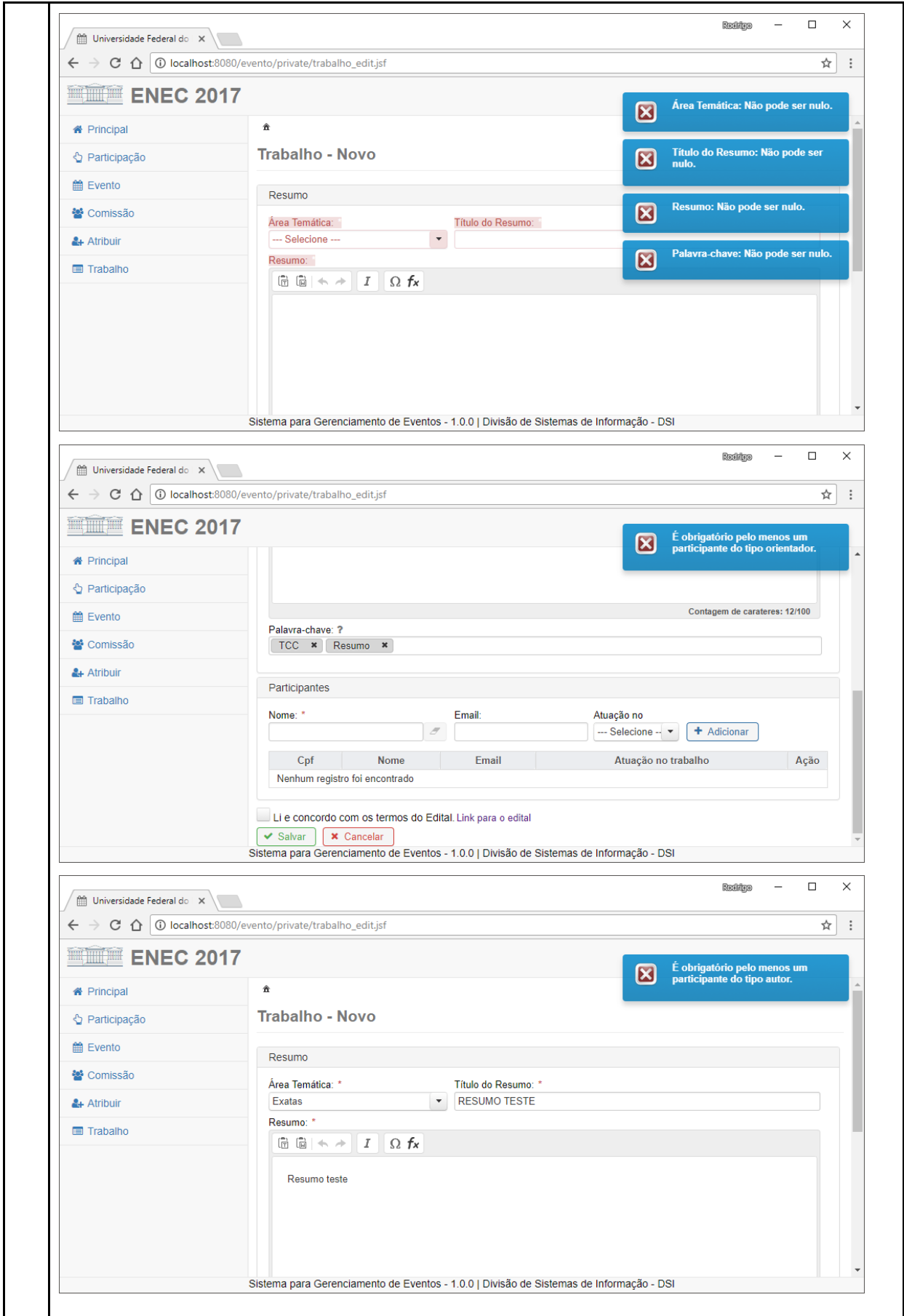
## 7 Log de teste - UC007

Nº	Log de teste	Resultado
1	Justificativa: Não pode ser nulo.	SUCESSO
2	Recurso registrado com sucesso	SUCESSO



## 8 Log de teste - UC008

Nº	Log de teste	Resultado
1	<p>Área Temática: Não pode ser nulo.</p> <p>Título do Resumo: Não pode ser nulo.</p> <p>Resumo: Não pode ser nulo.</p> <p>Palavra-chave: Não pode ser nulo.</p> <p>É obrigatório pelo menos um participante do tipo orientador.</p> <p>É obrigatório pelo menos um participante do tipo autor.</p> <p>Para submeter trabalho é obrigatório aceitar os termos do edital.</p>	SUCESSO



Universidade Federal do x

localhost:8080/evento/private/trabalho\_edit.jsf

**ENEC 2017**

Principal  
Participação  
Evento  
Comissão  
Atribuir  
Trabalho

Para submeter trabalho é obrigatório aceitar os termos do edital.

Palavra-chave: ?  
TCC x Resumo x

Participantes

Nome: \* Email: Atuação no

--- Selecione -- + Adicionar

Cpf	Nome	Email	Atuação no trabalho	Ação
04063250903	Rodrigo Perez Furtado	rodrigo@ufpr.br	Orientador	
22222222222	Maria da Silva	maria@ufpr.br	Autor	

Li e concordo com os termos do Edital. [Link para o edital](#)

Salvar Cancelar

Sistema para Gerenciamento de Eventos - 1.0.0 | Divisão de Sistemas de Informação - DSI

2

Trabalho submetido:

SUCESSO

Universidade Federal do x

localhost:8080/evento/private/trabalho\_list.jsf

**ENEC 2017**

Principal  
Participação  
Evento  
Comissão  
Atribuir  
Trabalho

Trabalho submetido: Resumo teste

Trabalho - Listar

Filtros de pesquisa

Número: Título: Situação: Área Temática: Submissor:

--- Selecione -- --- Selecione --

Pesquisar Limpar Cadastrar

Sistema para Gerenciamento de Eventos - 1.0.0 | Divisão de Sistemas de Informação - DSI

## **APÊNDICE O – SOLICITAÇÃO DE MUDANÇA**

Visando estender o aproveitamento da solução oferecida para toda a comunidade da UFPR se faz necessário continuar o desenvolvimento da solução mesmo depois da apresentação da monografia, visando ampliar a utilização deste sistema para gerir não apenas a SIEPE, mas os demais eventos de ensino, pesquisa e extensão da Universidade, tais como semanas acadêmicas e congressos científicos.

## **APÊNDICE P – PLANO DE IMPLANTAÇÃO**

Este documento tem como objetivo apresentar o conjunto de tarefas necessárias para a implantação do sistema de modo que alcance seus objetivos com sucesso.

### **1 PLANEJAMENTO DA IMPLANTAÇÃO**

As seguintes etapas devem ser realizadas para a implantação do sistema em ambiente de produção.

- Execução do script de banco de dados no servidor de produção.
- Implantação do arquivo .war no servidor de aplicação.
- Criação de perfis no SCA(Sistema de Controle de Acesso).
- Treinamento dos usuários para utilização do sistema.