

**LUCAS ARTHUR FELGUEIRAS**

**SISTEMA DE GERENCIAMENTO DE  
DISCIPLINAS DE PROJETO DE FORMATURA  
DO PCS**

São Paulo  
2018

**LUCAS ARTHUR FELGUEIRAS**

**SISTEMA DE GERENCIAMENTO DE  
DISCIPLINAS DE PROJETO DE FORMATURA  
DO PCS**

Trabalho apresentado à Escola Politécnica  
da Universidade de São Paulo para obtenção  
do Título de Engenheiro de Computação.

São Paulo  
2018

**LUCAS ARTHUR FELGUEIRAS**

**SISTEMA DE GERENCIAMENTO DE  
DISCIPLINAS DE PROJETO DE FORMATURA  
DO PCS**

Trabalho apresentado à Escola Politécnica  
da Universidade de São Paulo para obtenção  
do Título de Engenheiro de Computação.

Área de Concentração:  
Engenharia de Software

Orientador:  
Prof. Dr. Fábio Levy Siqueira

São Paulo  
2018

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

#### Catálogo-na-publicação

Felgueiras, Lucas Arthur

Sistema de Gerenciamento de Disciplinas de Projeto de Formatura do  
PCS / L. A. Felgueiras -- São Paulo, 2018.  
86 p.

Trabalho de Formatura - Escola Politécnica da Universidade de São  
Paulo. Departamento de Engenharia de Computação e Sistemas Digitais.

1.modelagem de processos 2.casos de uso 3.requisitos 4.visão 5.testes  
I.Universidade de São Paulo. Escola Politécnica. Departamento de  
Engenharia de Computação e Sistemas Digitais II.t.

À minha família,  
que acreditou em mim  
quando nem eu acreditava.

# AGRADECIMENTOS

Agradeço à Deus por toda a força necessária ao longo desses 5 anos de curso para seguir com minha escolha. Agradeço também aos meus familiares: minha irmã Maria Eduarda, minha mãe Márcia, meu pai Francisco, meus avós Helena e Júlio e o cachorro da família Napoleão, por todo o apoio e dedicação que tiveram comigo durante toda minha jornada.

Agradeço imensamente o Prof. Dr. Fábio Levy Siqueira, pela gigantesca ajuda, paciência e dedicação com a orientação deste trabalho. Agradeço também aos Prof. Dr. João Batista Camargo Júnior e Prof. Dr. Paulo Sérgio Cugnasca pelo tempo reservado nas diversas reuniões que constituíram este projeto.

Agradeço também a minha família de consideração, formada pelos meus amigos de cursinho: César Augusto, Rafael Salazar, Bruno Alves, Diego Rodrigues, Airton Júnior e Júnior Castro; e pelos meus amigos de faculdade e estágios: Victor França, João Victor, Daniel Norio, Luiz Gustavo, Willian Wener, Martin Andrade, Alexandre Aguiar e Guilherme Eiras.

Por fim, agradeço a todo o departamento de Engenharia de Computação e Sistemas Digitais, em especial aos técnicos e equipe administrativa, pelo empenho e dedicação na formação de grandes profissionais.

*”Faça ou não faça. Tentativa não há.”*

-Mestre Yoda

# RESUMO

Ao longo dos anos, o processo das disciplinas de projeto de formatura do PCS ocorre manualmente, com entregas descentralizadas, avaliações em papel, cálculo de notas trabalhoso, entre outras características que compõem um problema real no curso.

O projeto consiste no uso de técnicas de Engenharia de Software para modelar o processo, levantar as necessidades que compõem o problema e construir um sistema responsável por automatizar o processo levantado, com foco na verificação e validação do produto desenvolvido.

A implementação do sistema usa tecnologias renomadas no mercado e cobre parte dos requisitos levantados, gerando valor para os coordenadores das disciplinas e viabilizando seu uso pelo departamento a partir de 2019, com expectativas de se tornar um legado para a Escola Politécnica.

**Palavras-Chave** – modelagem de processos, casos de uso, requisitos, testes, verificação, validação, visão, sistemas web.

# ABSTRACT

Over the years, the process of the graduation project disciplines of the PCS occurs manually, with decentralized deliveries, paper assessments, calculation of labor notes, among other characteristics that make up a real problem in the course.

The project consists in the use of software engineering techniques to model the process, to raise the needs that make up the problem and to build a system responsible for automating the raised process, focusing on the verification and validation of the developed product.

The implementation of the system uses renowned technologies in the market and covers part of the requirements raised, generating value for the coordinators of the disciplines and enabling its use by the department from 2019, with expectations of becoming a legacy for the Polytechnic School.

**Keywords** – process modeling, use cases, requirements, tests, verification, validation, vision, web systems.

## LISTA DE FIGURAS

1	Diagrama de Casos de Uso para Atividades . . . . .	35
2	Diagrama de Casos de Uso para Eventos e Avaliações . . . . .	35
3	Diagrama de Casos de Uso para Gerenciamento dos Recursos . . . . .	36
4	Diagrama de Casos de Uso para Base de Dados (Biblioteca Virtual) . . . . .	38
5	Diagrama de Casos de Uso para Infraestrutura . . . . .	38
6	Arquitetura Básica do Sistema . . . . .	39
7	Diagrama Entidade-Relacionamento em Nível Lógico do Sistema Desenvolvido . . . . .	43
8	Diagrama de Navegação de Telas Simplificado para o perfil de Aluno . . . . .	44
9	Diagrama de Navegação de Telas Simplificado para o perfis de Docente e Convidado ( <i>Staff</i> ) . . . . .	45
10	Diagrama de Navegação de Telas Simplificado para o perfil de Coordenador . . . . .	46
11	Tela Inicial do Aluno . . . . .	57
12	Tela Inicial do Professor/Convidado . . . . .	58
13	Tela de Nova Avaliação Teórica . . . . .	59
14	Tela Inicial do Coordenador, com permissões de Professor . . . . .	60
15	Tela de Cadastro Massivo de Alunos . . . . .	61
16	Tela de Listagem de Salas . . . . .	63
17	Tela de Nova Sala, cadastrando como exemplo a B2-08 . . . . .	64
18	Tela de Detalhes da Avaliação . . . . .	65
19	Tela de Notas Calculadas por Grupo de Trabalho . . . . .	66
20	Diagrama BPMN para a disciplina de TCC 1 - Parte 1 . . . . .	69
21	Diagrama BPMN para a disciplina de TCC 1 - Parte 2 . . . . .	70
22	Diagrama BPMN para a disciplina de TCC 2 - Parte 1 . . . . .	71

23	Diagrama BPMN para a disciplina de TCC 2 - Parte 2 . . . . .	72
24	Diagrama BPMN para a Banca e Feira - Parte 1 . . . . .	73
25	Diagrama BPMN para a Banca e Feira - Parte 2 . . . . .	74
26	Diagrama BPMN para a Recuperação - Parte 1 . . . . .	75
27	Diagrama BPMN para a Recuperação - Parte 2 . . . . .	76

# LISTA DE TABELAS

1	Sentença básica de posição do produto . . . . .	28
2	Principais necessidades encontradas . . . . .	31

# SUMÁRIO

<b>1</b>	<b>Introdução</b>	<b>15</b>
1.1	Objetivo . . . . .	15
1.2	Justificativa . . . . .	15
1.2.1	Sistemas Concorrentes . . . . .	16
1.3	Organização do Trabalho . . . . .	16
<b>2</b>	<b>Aspectos Conceituais</b>	<b>17</b>
2.1	Modelagem de Processos . . . . .	17
2.2	Representação de Requisitos . . . . .	18
2.2.1	Histórias de Usuário . . . . .	19
2.2.2	Casos de Uso . . . . .	20
2.3	Requisitos . . . . .	21
2.3.1	Definição dos <i>Stakeholders</i> . . . . .	21
2.3.2	Entendimento dos Problemas . . . . .	22
2.3.3	Documento Visão . . . . .	23
2.4	Verificação e Validação . . . . .	24
2.4.1	Testes . . . . .	24
<b>3</b>	<b>Processo de Desenvolvimento</b>	<b>25</b>
3.1	Levantamento de Requisitos . . . . .	25
3.1.1	Entrevistas Iniciais . . . . .	25
3.1.2	Casos de Uso e Histórias de Usuário . . . . .	25
3.2	Divisão em Iterações . . . . .	26
3.3	Ambientes de Validação . . . . .	26

3.4	Processo de Desenvolvimento e Testes . . . . .	27
<b>4</b>	<b>Especificação de Requisitos do Sistema</b>	<b>28</b>
4.1	Visão do Sistema . . . . .	28
4.1.1	Descrição do Problema . . . . .	28
4.1.2	Descrições dos Envolvidos e Usuários . . . . .	28
4.1.3	Ambiente do Usuário . . . . .	30
4.1.4	Alternativas e Concorrência . . . . .	30
4.1.5	Principais Necessidades . . . . .	30
4.1.6	Perspectiva do Produto . . . . .	32
4.1.7	Suposições e Dependências . . . . .	32
4.2	Requisitos Funcionais . . . . .	32
4.3	Requisitos Não Funcionais . . . . .	33
<b>5</b>	<b>Projeto e Implementação</b>	<b>34</b>
5.1	Elaboração e Estrutura do Sistema . . . . .	34
5.1.1	Atores dos Casos de Uso . . . . .	35
5.1.2	Casos de Uso Implementados . . . . .	35
5.1.3	Casos de Uso Não Implementados . . . . .	37
5.2	Arquitetura e Tecnologias Utilizadas . . . . .	39
5.2.1	Aplicações Construídas . . . . .	41
<b>6</b>	<b>Teste e Validação</b>	<b>47</b>
6.1	Validações . . . . .	47
6.1.1	Reunião de 28/09 . . . . .	47
6.1.2	Reunião de 30/11 . . . . .	49
6.2	Testes Executados . . . . .	50
6.2.1	Cenários de Teste . . . . .	50

6.2.2	Ferramentas de Teste . . . . .	50
<b>7</b>	<b>Considerações Finais</b>	<b>52</b>
7.1	Resultados Alcançados . . . . .	52
7.1.1	Casos de Uso Construídos . . . . .	52
7.2	Próximos Passos . . . . .	52
	<b>Referências</b>	<b>54</b>
	<b>Apêndice A – Manual do Usuário</b>	<b>56</b>
A.1	Aluno . . . . .	57
A.1.1	Tela Inicial . . . . .	57
A.1.2	Meus Grupos . . . . .	58
A.1.3	Minhas Disciplinas . . . . .	58
A.2	Professor/Convidado . . . . .	58
A.2.1	Tela Inicial . . . . .	59
A.2.2	Grupos de Trabalho . . . . .	59
A.2.3	Disciplinas . . . . .	60
A.3	Coordenador . . . . .	60
A.3.1	Gestão de Pessoas . . . . .	61
A.3.1.1	Docentes . . . . .	61
A.3.1.2	Alunos . . . . .	61
A.3.1.3	Convidados Externos . . . . .	62
A.3.2	Gestão de Disciplinas . . . . .	62
A.3.2.1	Disciplinas . . . . .	62
A.3.2.2	Grupos de Trabalho . . . . .	62
A.3.3	Gestão de Eventos e Notas . . . . .	63
A.3.3.1	Salas . . . . .	63

A.3.3.2	Eventos . . . . .	64
A.3.3.3	Fórmulas . . . . .	65
A.4	Funcionalidades Comuns . . . . .	67
A.4.1	Recuperar Senha . . . . .	67
A.4.2	Alterar Senha . . . . .	67
<b>Apêndice B – Diagramas BPMN</b>		<b>68</b>
B.1	Introdução . . . . .	68
B.2	Processo antes do Sistema . . . . .	68
<b>Apêndice C – Casos de Uso</b>		<b>77</b>
C.1	Cadastrar disciplinas . . . . .	77
C.2	Editar disciplinas . . . . .	78
C.3	Cadastrar professores . . . . .	78
C.4	Cadastrar convidados externos . . . . .	79
C.5	Cadastrar grupos de trabalhos . . . . .	79
C.6	Login . . . . .	80
C.7	Entregar atividade . . . . .	81
C.8	Construir bancas práticas . . . . .	82
C.9	Construir bancas teóricas . . . . .	82
C.10	Listar entregas . . . . .	83
C.11	Listar necessidades adicionais . . . . .	84
C.12	Avaliar projetos práticos . . . . .	84
C.13	Avaliar monografias teóricas . . . . .	85
C.14	Calcular nota final dos projetos . . . . .	85

# 1 INTRODUÇÃO

A motivação deste trabalho está em automatizar o processo relacionado às disciplinas de projeto de formatura do Departamento de Engenharia de Computação e Sistemas Digitais (PCS), que atualmente é manual, com avaliações em formulários de papel; descentralizado, dado que as entregas são realizadas em plataformas diferentes, muitas vezes não envolvendo os orientadores; e trabalhoso, dado que todo o processo de cálculo de notas, controle de formulários preenchidos e afins é feito nos dias de avaliação, muitas vezes com tempo escasso.

As disciplinas de projeto de formatura usam diversos sistemas como base para seu processo, como por exemplo o Tidia-AE. Além disso, toda a parte de avaliação e cálculo das notas é realizado de maneira manual, tanto pelos avaliadores, como pelos coordenadores, o que gera trabalho excessivo de todos os envolvidos.

## 1.1 Objetivo

O objetivo deste trabalho é elaborar um sistema responsável por controlar as disciplinas do projeto de formatura do Departamento de Engenharia de Computação e Sistemas Digitais (PCS), automatizando parte do processo manual, centralizando em um único sistema todas as etapas deste processo.

## 1.2 Justificativa

As disciplinas de formatura do PCS são gerenciadas pelos coordenadores da disciplina Prof. Dr. João Batista e Prof. Dr. Paulo Cugnasca. Elas ocorrem com entregas parciais, submetidas no Tidia-AE, sem o acompanhamento direto do orientador. Ao final da segunda disciplina de formatura, há os eventos de avaliação do projeto, com formulários manuais e cálculo das notas no curto espaço de tempo, o que dificulta e torna trabalhoso o gerenciamento das notas finais do processo.

### 1.2.1 Sistemas Concorrentes

Dois sistemas foram analisados como soluções: Tidia-AE <https://ae4.tidia-ae.usp.br> e Moodle <https://moodle.org/>.

O Tidia-AE é usado atualmente pelos coordenadores como sistema de suporte para as entregas intermediárias das disciplinas. Para este aspecto, ele atende bem as necessidades. Porém, não possui uma solução específica para as avaliações, bem como não existe uma maneira de realizar a alocação para os eventos teóricos e práticos

Já o Moodle do E-Disciplinas <https://edisciplinas.usp.br> é uma solução mais robusta, com avaliações e a funcionalidade de repositório. Contudo, ainda não atende o negócio como um todo, bem como a dinâmica de avaliações das disciplinas.

## 1.3 Organização do Trabalho

Este trabalho está organizado da seguinte forma:

- Capítulo 2: aspectos conceituais de Engenharia de Software usados no desenvolvimento deste trabalho, que são Modelagem de Processos, Levantamento de Requisitos, Casos de Uso e Histórias de Usuário.
- Capítulo 3: discute a metodologia usada para a construção do sistema.
- Capítulo 4: mostra os requisitos gerais do sistema.
- Capítulo 5: discorre sobre projeto, arquitetura e implementação.
- Capítulo 6: discute sobre os testes executados e o processo de validação com os coordenadores.
- Capítulo 7: mostra as expectativas do trabalho, próximos passos e resultados obtidos.

## 2 ASPECTOS CONCEITUAIS

Neste capítulo, serão abordados os principais conceitos relacionados à parte teórica do trabalho: modelagem de processos, representação dos requisitos envolvidos, a engenharia por detrás desses requisitos, suas descrições por meio de casos de uso e histórias de usuário e os conceitos de verificação e validação.

### 2.1 Modelagem de Processos

Para entender melhor como o sistema atua no processo, vale compreender como funciona esse processo e estabelecer as possíveis fontes de participação do sistema.

A modelagem de processos de negócio (ou o *Business Process Modeling* - BPM) existe para entender melhor como está estabelecido o processo como um todo atuando em diversos aspectos: torna o processo mais fluido; evidencia os pontos de aumento da produtividade dos recursos humanos; aloca problemas para as pessoas corretas e; simplifica as regularizações necessárias no processo (HAVEY, 2015).

Há duas abordagens para modelagem de um processo, para entender como o sistema atuará (ANGELI, 2018):

- Antes do sistema (*AS IS*): modelar o processo antes do sistema, para entender como ele atuará para automatizá-lo.
- Junto com o sistema (*TO BE*): modelar o processo com o sistema atuante, para evidenciar seus impactos.

Existem ferramentas para representar os processos de negócio, segundo o CBOOK (ABPMP, 2013):

- BPMN (*Business Process Model and Notation*)

- Fluxograma
- EPC (*Event-driven Process Chain*)
- UML (*Unified Modeling Language*)
- IDEF (*Integrated Definition Language*)
- *Value Stream Mapping*

Segundo a OMG (2013, p. 1):

”O objetivo do BPMN é fornecer uma notação que seja prontamente compreensível por todos os usuários de negócios, desde os analistas que criam os rascunhos iniciais dos processos, até os desenvolvedores técnicos responsáveis pela implementação da tecnologia que executará esses processos e, finalmente, para os empresários que irão gerenciar e monitorar os processos.”

Ou seja, o propósito do BPMN é unificar a visão dos processos de negócio usando uma notação universal, compreendida por todos os envolvidos que desejam automatizar etapas deste processo.

Segundo a OMG (2013), o BPMN possui cinco elementos básicos:

- Objeto de Fluxo: descrevem o comportamento do negócio.
- Dados: informações usadas no negócio.
- Conectores: conectam os objetos com outros objetos e/ou informações.
- Piscinas e Raias: representam, graficamente, um envolvido no processo.
- Artefatos: fornecem informações adicionais sobre os processos.

## 2.2 Representação de Requisitos

Ao se desenvolver um sistema, existem diversas técnicas que podem ser usadas para representar os requisitos. Para este trabalho, duas técnicas foram consideradas: Histórias de Usuário e Casos de Uso.

## 2.2.1 Histórias de Usuário

Uma das maneiras de representar os requisitos é a História de Usuário (ou *User Story*). Segundo Rasmusson (2010, p. 98):

”Descrições curtas das funcionalidades que o nosso cliente gostaria de um dia ver em seu software. Elas geralmente são escritas em pequenos cartões de índice (para nos lembrar de não tentar escrever tudo) e estão lá para nos encorajar a ir falar com nossos clientes.”

O essencial em escrever boas histórias de usuário está em agregar valor para os *stakeholders*. Ou seja, escrever histórias de usuário que tangem assuntos como arquitetura do sistema, linguagem de implementação, padrões de código entre outros não são boas histórias de usuário, pois salvo exceções, o cliente não vê valor em histórias desse tipo (RASMUSSON, 2010).

Em contrapartida, histórias sobre o comportamento do sistema com valor de produto nelas são histórias de usuário importantes para o processo de desenvolvimento. Muitas vezes, porém, histórias de usuário forçam um viés altamente técnico; nessas situações, cabe buscar a causa raiz que levou à ”solução” escrita na história: histórias de usuário apresentam problemas de negócio, e não soluções técnicas.

Um padrão para escrever histórias de usuário (RASMUSSON, 2010, p. 107):

Eu como <para quem é a história>  
Eu quero <o que ele quer>  
por causa <por que ele quer>

A vantagem de escrever histórias segundo esse modelo é que, naturalmente, elas atendem as características que definem Histórias de Usuário (RASMUSSON, 2010):

- Independentes: elas podem ser implementadas de maneira isolada e distribuída entre a equipe, permitindo maior facilidade para mudanças no projeto (que ocorrem com frequência).
- Negociáveis: podem sofrer alterações e inclusive serem removidas se, com o andamento do projeto, ela perder sua prioridade e tais alterações e remoções não devem afetar o andar geral das outras tarefas.
- Testáveis: cada história pode ser verificada testando o fluxo de negócio e com a escrita de testes automatizados, assim garantimos com facilidade a qualidade do que está sendo entregue

- Pequenas e Estimáveis: o esforço dela pode ser previsto com antecedência, permitindo assim acompanhar o andamento do projeto.

## 2.2.2 Casos de Uso

Outra alternativa para documentar o que será desenvolvido no sistema são os casos de uso. Essencialmente, cada caso de uso possui um ou mais atores interagindo com o sistema. Segundo Bittner (2002, p. 20): "Os **atores** representam as pessoas ou coisas que interagem de alguma forma com o sistema. Já os **casos de uso** representam as coisas de valor que o sistema executa para os atores."

Para escrever casos de uso, foi usada a seguinte estrutura (IBM, 2011):

1. Nome: nome do Caso de Uso
  - (a) Breve Descrição: finalidade do caso de uso
2. Fluxo de Eventos: a descrição dos eventos que ocorrem no sistema, que são divididos em dois conjuntos principais.
  - (a) Fluxo Básico: é o comportamento básico da interação entre os atores e o sistema. Não há situações de contorno nem exceções, que serão listadas nos fluxos alternativos.
  - (b) Fluxos Alternativos: outros comportamentos menos usuais do caso de uso vêm aqui.
3. Requisitos Especiais: requisito, não funcional, que não é contemplado no fluxo de eventos.
4. Pré-condição: estado do sistema antes da realização do caso de uso.
5. Pós-condição: estado do sistema após a execução.

Resumindo, o caso de uso é uma descrição comportamental da funcionalidade descrita, servindo de contrato estabelecido de desenvolvimento do sistema.

Além da representação textual, há também o uso de representações gráficas de casos de uso. A representação gráfica apresenta o nome do caso de uso e os atores que interagem com ele, mas omite sua parte descritiva. Tais padrões de representação são definidos pela UML (OMG, 2017).

## 2.3 Requisitos

Segundo o SWEBOK (2014, cap. 1, p. 1): "Um requisito de software é uma propriedade que representa algo para resolver algum problema no mundo real." Existem dois tipos de requisitos de software, definidos pelo SWEBOK (2014, cap. 1, p. 3):

- Funcionais: descrevem as funcionalidades executadas pelo software.
- Não-funcionais: definem qualidades do software ou restringem seu funcionamento.

Para encontrá-los, existe um exemplo de processo básico, definido pelo Bittner (2002, cap. 6, p. 248-249):

- Estabelecimento da Visão: unificação do entendimento do sistema, por meio do Documento Visão.
- Elaboração do sistema geral: construção da ideia geral do sistema, com base nas necessidades levantadas.
- Consenso do sistema geral: os *stakeholders* concordam com a ideia geral do sistema, caso contrário, reestabelecem-se a Visão e o sistema geral.
- Elaboração dos Casos de Uso: construir os casos de uso do sistema, com base no documento visão levantado.
- Revisão dos Casos de Uso: revisão dos casos de uso escritos, para garantir que a parte da a ideia do sistema está representada na especificação.

Vale lembrar que os requisitos funcionais são representados pelos casos de uso e os requisitos não funcionais são ligados aos casos de uso escritos (BITTNER, 2002).

Além disso, os casos de uso podem capturar um subconjunto dos requisitos funcionais do sistema. Em situações de requisitos ainda não representados pelos casos de uso, eles são capturados por especificações suplementares, definidas pela IBM (BITTNER, 2002).

### 2.3.1 Definição dos *Stakeholders*

Inicialmente, para melhor entender quem será afetado com a existência de um sistema, é necessário encontrar quem são essas pessoas ou grupos. Para isso, existe o conceito de

*stakeholders*. Segundo Bittner (2002, p. 88): "É o indivíduo que é materialmente afetado pelo resultado do sistema ou o(s) projeto(s) que produzem o sistema."

Ou seja, *stakeholders* não são apenas os indivíduos que efetivamente usarão o sistema, mas sim todos os impactados com sua existência. São divididos em:

- Usuários: as pessoas que efetivamente usarão o sistema.
- Patrocinadores: os financiadores do projeto de software que gerará o sistema.
- Desenvolvedores: os responsáveis por desenvolver o sistema.
- Autoridades: órgãos reguladores que determinam regras para o uso do sistema.
- Consumidores: empresas que compram esses softwares para serem usados.

### 2.3.2 Entendimento dos Problemas

Uma vez mapeado os *stakeholders*, cabe entender as dores que cada um possui e o que eles esperam com o produto final a ser desenvolvido. Alguns processos sugeridos pelo Bittner (2002):

- Entrevistas: entrevistar os *stakeholders* e entender diretamente quais são suas dores e expectativas.
- Questionários: sequência de perguntas feitas à todos os *stakeholders* para compreender o que cada um pensa sobre o sistema. São úteis quando há um amplo número de *stakeholders* envolvidos.
- Grupo: reunião com alguns representantes dos grupos de *stakeholders* para entender e construir uma visão única sobre o projeto.
- Quadros de Aviso: um tipo particular de grupo, com o quadro servindo como unificador da visão, com a diferença de não ter todos reunidos ao mesmo tempo.
- *Workshops*: eventos avisados com antecedência para entender melhor sobre o sistema, com a participação dos envolvidos, focando na colaboração entre eles.
- Revisões: reuniões informais com o intuito de revisar documentos gerados com alguns envolvidos.

- Encenação: uma técnica facilitadora usada em conjunto com *workshops* para obter informações mais específicas ou *feedbacks*.

Com essa listagem de problemas levantados, o próximo passo é construir a visão unificada, evidenciando como o software vai atuar no processo. Para isso, há a elaboração de um documento unificando os pontos de vista dos *stakeholders* e estabelecendo o que de fato será o sistema a ser desenvolvido, resultando no Documento Visão.

### 2.3.3 Documento Visão

O Documento Visão é o responsável por unificar as visões de todos os *stakeholders* do sistema, centralizando todas as expectativas e definindo o escopo macro do projeto. Nele também deve constar as justificativas necessárias para a construção do sistema, explicitando possíveis concorrentes e os diferenciais do produto a ser desenvolvido.

Existem diversos modelos de Documento Visão. Um modelo de referência (BITTNER, 2002):

1. Posicionamento: Como o sistema irá se posicionar no quesito de negócios? Há concorrentes que já resolvem o problema? Quais são seus diferenciais em relação a eles?
2. *Stakeholders* e usuários: Quem são os envolvidos direta e indiretamente com o desenvolvimento e a existência do sistema?
3. Necessidades chave: Quais são as demandas que realmente precisam estar nos planos do sistema para satisfazer os envolvidos?
4. Visão geral do produto: O que é o produto de fato? Quais são suas dependências, capacidades e alternativas ao seu desenvolvimento?
5. Funcionalidades: Quais são as funcionalidades em alto nível do sistema, para que elas resolvam as necessidades chave listadas anteriormente?
6. Outros requisitos do produto: Quais são os outros requisitos do sistema que não foram capturados como funcionalidades?

## 2.4 Verificação e Validação

Durante o desenvolvimento, faz parte do processo garantir a qualidade do que está sendo entregue. Para isso, existem os processos de verificação e validação, que segundo o SWEBOK (2014, cap. 10, p. 6): "Servem para ajudar a equipe de desenvolvimento a garantir qualidade no sistema por todo seu ciclo de vida."

Ao se desenvolver um software, é uma boa prática acompanhar se os requisitos funcionais e não funcionais estão sendo atendidos, de preferência com acompanhamento das métricas determinadas nos requisitos. Essa prática é conhecida como verificação, definida de maneira resumida pela seguinte pergunta presente no SWEBOK (2014, cap. 10, p. 6): "Estamos construindo o produto corretamente?"

Além de acompanhar os requisitos propostos, é importante entender se o sistema está de acordo com as expectativas operacionais dos *stakeholders*. Essa análise constante de adequação voltada aos negócios é conhecida como validação, definida pela seguinte questão do SWEBOK (2014, cap. 10, p. 6): "Estamos construindo o produto correto?"

### 2.4.1 Testes

Em ambos os casos descritos acima, são necessárias técnicas para detectar erros e manter a consistência do que já foi entregue. Para isso, existe o conceito de testes, definido no SWEBOK (2014, cap. 4, p. 1): "Consiste na conferência dinâmica de que um programa fornece comportamentos esperados em um conjunto finito de casos de teste, selecionados adequadamente a partir do domínio de execução, geralmente infinito."

Na escala macroscópica, há os testes de aceitação junto aos *stakeholders*, para validar as entregas. Já para testar a arquitetura, costumam ser realizados testes de sistema. Para testar os componentes desenvolvidos do sistema, realizam-se testes de integração entre esses componentes. E, por fim, o código escrito para cada componente elaborado é testado pelos testes de unidade.

Ao desenvolver um sistema, uma boa prática é executar a suíte de testes construída a cada nova funcionalidade adicionada e/ou modificada. Para isso, existe a filosofia de testes automatizados, que são códigos escritos responsáveis por rodar um subconjunto de testes da suíte escolhida. A vantagem dessa filosofia está em reduzir o esforço de trabalho como um todo, dado que é mais custoso refatorar código não-testado problemático do que elaborar as funcionalidades com os testes inclusos (MESZAROS, 2007).

## 3 PROCESSO DE DESENVOLVIMENTO

Neste capítulo, serão discutidos processos de levantamento de requisitos, a divisão do desenvolvimento em iterações, o processo de desenvolvimento, testes e as reuniões de validação com os principais *stakeholders*.

### 3.1 Levantamento de Requisitos

Para o levantamento de requisitos, foram realizadas entrevistas com os *stakeholders* para entender melhor qual é o processo atual, as necessidades encontradas e como o sistema irá ajudar nelas.

#### 3.1.1 Entrevistas Iniciais

A primeira etapa do processo foi a realização de entrevistas com os principais *stakeholders*, de maneira a modelar o processo atual, entender melhor como o sistema irá atuar e estabelecer seus requisitos.

Seguindo os conceitos do capítulo 2, ocorreram entrevistas com os *stakeholders* como forma de levantar requisitos. Por meio dessas entrevistas, o processo geral de negócio (*AS IS*) foi modelado usando BPMN.

Após a modelagem, teve início a construção do Documento Visão, usando como base o processo recém modelado. Com a finalização do Documento Visão, resta entender como será feita a representação de requisitos do sistema.

#### 3.1.2 Casos de Uso e Histórias de Usuário

Realizando um comparativo entre casos de uso e histórias de usuário, tomando como base os conceitos expostos no capítulo 2:

- Casos de uso tendem a ser maiores, mais detalhados, documentando o requisito em

detalhe. Já as histórias de usuário são mais modulares e simples, com os detalhes conversados ao longo do projeto, junto com os *stakeholders*.

- Casos de uso demandam tempo dos *stakeholders* apenas durante a atividade de levantamento de requisitos (caso não tenha problemas na especificação), já as histórias de usuário, demandam tempo dos *stakeholders* durante a implementação, para se obter mais informações sobre as histórias. Casos de uso documentam os requisitos em detalhes, enquanto que não há essa documentação nas histórias do usuário. Os detalhes da história são conversados entre as partes.

Tomando como base essas diferenças, a solução adotada foram os casos de uso, dada a escassez de tempo de muitos dos *stakeholders*, o que dificultaria bastante o desenvolvimento do sistema com o uso de histórias de usuário.

Outro fato relevante na decisão para casos de uso está no fato da equipe possuir apenas um único integrante, o que também dificultaria o uso de histórias de usuário. Além disso, este fato implicou na redução do formalismo do processo, dado que muitas etapas requerem mais de um participante nos processos.

## 3.2 Divisão em Iterações

Após a elaboração do Documento Visão, os Casos de Uso elaborados foram divididos em iterações, cada uma com duas reuniões atreladas:

- Reunião de revisão dos casos de uso: os *stakeholders* revisam os casos de uso a serem desenvolvidos, evidenciaram detalhes faltantes nas descrições e priorizaram quais casos de uso da iteração são os mais importantes.
- Reunião de validação do sistema: os *stakeholders* revisam o fluxo do sistema como um todo, se o que foi desenvolvido está conforme com os casos de usos e se há sugestões de alterações e quais são suas prioridades.

## 3.3 Ambientes de Validação

Nesta seção, será explicado um pouco mais sobre ambientes de validação e como eles são usados no processo de desenvolvimento do sistema.

Para desenvolvimento de sistemas, são criados diversos ambientes de aplicação, para testar as funcionalidades com os diversos *stakeholders*, de acordo com a complexidade de negócios do projeto.

Em geral, são três ambientes de aplicação para realizar os processos de validação do sistema, permitindo testes isolados dos *stakeholders*, sem afetar o fluxo de desenvolvimento. São eles (RAGAN, 2017):

- Desenvolvimento: onde geralmente fica a versão mais recente da aplicação, com as novas funcionalidades desenvolvidas e testadas já integradas no ambiente. Essa versão serve como uma prévia do que será entregue para homologação. As mudanças nesse ambiente são mais frequentes, dado que uma nova funcionalidade completa já pode ir para este ambiente.
- Homologação: já neste ambiente, as mudanças são bem menores e servem como ambiente de validação para os *stakeholders*. No caso do sistema de TCCs, ele serviu como homologação com os coordenadores do curso. Se possível, ele deve ser o mais fiel ao ambiente final de produção, assim o comportamento ideal dele em homologação será o mesmo em produção.
- Produção: este ambiente é onde o sistema irá rodar, de fato. Ele deve ser o mais estável possível, com alterações apenas homologadas pelos *stakeholders*. Salvo raras exceções, como falhas e problemas encontrados, nada deve ser colocado aqui sem a aprovação no ambiente de homologação.

### 3.4 Processo de Desenvolvimento e Testes

Para o processo de desenvolvimento, com os casos de uso definidos e organizados, não há um processo formal, dado o fato da equipe de desenvolvimento ser constituída de apenas um integrante. Cada caso de uso tem seu fluxo básico desenvolvido, junto com os testes relacionados e, ao final do processo, a funcionalidade é enviada para o ambiente de desenvolvimento na nuvem, para testes mais próximos do ambiente de produção.

## 4 ESPECIFICAÇÃO DE REQUISITOS DO SISTEMA

Neste capítulo, serão abordados os requisitos do sistema encontrados, os pontos levantados nas diversas reuniões e os requisitos funcionais e não funcionais encontrados.

### 4.1 Visão do Sistema

De acordo com o capítulo 2, o documento visão é responsável por unificar o entendimento do sistema para os *stakeholders*.

#### 4.1.1 Descrição do Problema

Os problemas estão na descentralização do processo, com os orientadores participando por intermédio de seus orientados, além da parte manual de construção de eventos e de avaliações das bancas. O impacto direto é na demora e esforço para calcular notas, dificuldade de integração entre envolvidos (orientador, alunos e coordenação). Para resolver o problema, o sistema deve automatizar o processo e integrar os envolvidos ao longo das disciplinas.

Tabela 1: Sentença básica de posição do produto

Para	alunos, orientadores, coordenadores e técnicos
Que	cursam ou estão envolvidos diretamente
O sistema	de gerenciamento dos TCC's
Que	armazena os TCC's anteriores e gerencia o TCC atual, realizando avaliação e controlando entregas
Ao contrário do	processo semi-manual de gerenciamento da disciplina, do Tidia-AE e do Site institucional em Wordpress
O sistema	permite um acompanhamento dos envolvidos, bem como serve de histórico para os trabalhos anteriores

#### 4.1.2 Descrições dos Envolvidos e Usuários

Há alguns usuários que não estão diretamente envolvidos com o sistema, mas podem ser beneficiados ou interferem no sistema a ser desenvolvido:

- Infraestrutura: irá aplicar restrições tecnológicas e de negócio dado o ambiente desenvolvido.
- Secretaria do Departamento: nem todos os integrantes irão usar o sistema, mas serão beneficiados pela automação de alguns processos internos da disciplina, diminuindo a demanda recorrente sobre a equipe, vide o fato deles gerenciarem as entregas das avaliações durante o processo.

Por outro lado, também há diversos usuários que serão diretamente beneficiados com o sistema:

- Coordenador: responsáveis por administrarem as disciplinas de TCC 1 e 2 para os cursos de Engenharia de Computação Semestral e Quadrimestral.
- Orientador: responsável por construir com os alunos a monografia.
- Técnico de Operação: responsável por administrar novas funcionalidades futuras do sistema.
- Técnico do Evento: responsável por cuidar da parte de infraestrutura dos eventos a serem realizados.
- Alunos: os cursantes das disciplinas de TCC 1 e 2.
- Avaliador Teórico: avaliam os projetos realizados, durante a banca de defesa.
- Avaliador Prático: avaliam os projetos realizados, durante a feira de projetos de formatura.

Foram escolhidos alguns representantes dos perfis citados acima, de maneira a facilitar conversas durante a especificação do documento.

- João Batista/Paulo Cugnasca: coordenadores das duas disciplinas.
- Nilton: técnico do Evento
- Fábio Levy: responsável pela infraestrutura de hospedagem, professor orientador e avaliador teórico.

### 4.1.3 Ambiente do Usuário

Para o processo em questão, podemos reforçar alguns pontos importantes:

- São 2 professores coordenadores, 2 técnicos, cerca de 50 alunos por ano de TCC e cerca de 20 professores orientadores do departamento de PCS.
- O ciclo das disciplinas de TCC dura 1 ano, sendo metade do tempo de especificação e metade de implementação.
- Atualmente, há a plataforma Tidia-AE existente para administrar as disciplinas. Porém, ela serve apenas como repositório de arquivos, sem a participação direta dos orientadores e sem infraestrutura de automação e comunicação rápida entre os envolvidos.
- Além disso, há o site do departamento, onde ficam as monografias mais recentes, também como simples repositório de arquivos.

### 4.1.4 Alternativas e Concorrência

Também foram estudados alguns concorrentes do produto, de maneira a compreender se eles não seriam soluções para as necessidades levantadas:

- Tidia-AE: entregas pontuais, com acesso apenas aos integrantes do grupo e aos coordenadores da disciplina. Os orientadores ficam de fora das entregas principais, cabendo aos alunos realizarem essa comunicação de maneira extra-oficial.
- Moodle: plataforma aberta conhecida no mercado para gerenciamento de disciplinas em geral. Serve para construir grupos e até incluir orientadores, porém não é uma alternativa simples e não possui funcionalidades de busca avançada.
- Site PCS (Wordpress): apenas resultado final, de maneira pública, sem integração dos envolvidos no projeto.

### 4.1.5 Principais Necessidades

Diversas necessidades foram encontradas, sendo agrupadas, estudadas e propostas soluções para atendê-las:

Tabela 2: Principais necessidades encontradas

Necessidade	Prior.	Sol.Atual	Sol. Proposta
Orientador, coordenadores e alunos têm dificuldades em manter contato	1	E-mail, reuniões	Colocar os orientadores nas entregas
Busca de monografias antigas é complexa.	2	Site do PCS	Criar repositório de monografias finalizadas
O processo de gerenciar empresas participantes é extremamente manual.	7	Conversas	Integrar empresas às monografias a serem avaliadas
Avaliadores têm dificuldade em acessar monografias.	4	E-mail	Permitir monografia de fácil acesso no sistema
Avaliação dos projetos é de maneira manual.	5	Papel	Permitir avaliação pelo sistema, inclusive de maneira mobile
Encontrar temas e alunos e propor temas é um processo manual.	9	E-mail, conversas	Permitir alunos e orientadores proporem temas no sistema
O gerenciamento de recursos pela parte do técnico é depende dos alunos solicitarem.	8	E-mail, pen drive	Automatizar as demandas para os técnicos providenciarem
O processo de montagem da banca de TCC é manual.	6	E-mail, conversas	Permitir montagem de bancas pelo sistema
Fechar as notas finais é um processo exaustivo e com curto prazo.	3	Papel	Automatizar o cálculo das notas finais

### 4.1.6 Perspectiva do Produto

O produto tem como missão automatizar o processo já existente para a disciplina, pois muitas funcionalidades propostas são realizadas de maneira manual ou simplificada, o que demanda muito tempo e esforço dos coordenadores e orientadores, tornando o processo frágil. Além disso, depende da iniciativa de todos os envolvidos, pois todo o processo de comunicação orientador-aluno é feito de maneira isolada do processo coordenador-aluno e coordenador-orientador.

### 4.1.7 Suposições e Dependências

A principal dependência encontrada é que o serviço deve ser hospedado na infraestrutura interna da Universidade de São Paulo, o que afetará a maneira como o produto será desenvolvido.

## 4.2 Requisitos Funcionais

Com base no Documento Visão construído, resta entender os requisitos macro do sistema. Sendo assim, com base nas necessidades encontradas, foram levantados os seguintes requisitos funcionais:

- Facilitar comunicação orientador/coordenadores, permitindo que eles vejam todas as entregas dos alunos e validem, quando necessário.
- Buscar, de maneira pública, as monografias antigas, usando filtros e busca textual.
- Incluir avaliadores da feira no sistema para acompanhar monografias correntes.
- Permitir avaliação tanto da banca como da feira (com as empresas), permitindo acesso prévio ao conteúdo e facilitando a avaliação (de preferência na plataforma mobile).
- Permitir que orientadores, alunos e empresas proponham temas e consigam combinar grupos para realizar as propostas.
- Permitir aos técnicos gerenciarem recursos necessários para as apresentações (imprimir apresentações sem necessidade do aluno entregar arquivos via pen drive, por exemplo)

- Permitir a montagem de bancas de TCC, convidando os envolvidos e facilitando o acesso ao resultado final do aluno.

### 4.3 Requisitos Não Funcionais

Além dos requisitos funcionais, alguns requisitos não funcionais importantes foram levantados:

- **Segurança:** os acessos às monografias em andamento devem ser apenas aos envolvidos diretos do trabalho. Além disso, as empresas só podem acessar o resultado final não revisado, para fins de avaliação. Foco especial nos requisitos de Confidencialidade e Integridade.
- **Confiabilidade:** o sistema deve suportar situações de falha e lidar bem com as informações, garantindo sua proteção. No caso, foco especial em Recuperabilidade.
- **Usabilidade:** o sistema deve ser intuitivo e de aprendizagem fácil, pelo tempo curto dos envolvidos na feira e pela mobilidade envolvida (avaliações pelo celular, por exemplo). Foco especial na Operacionalidade, Estética/Interface e Proteção contra Erros do Usuário.
- **Manutenibilidade:** o sistema deve permitir realizar alterações, instalações e afins, dado que alunos podem tocar sua manutenção futuramente. Foco especial na Modificabilidade.

## 5 PROJETO E IMPLEMENTAÇÃO

Este capítulo aborda o projeto e a implementação do sistema, com base nos requisitos já levantados, passando pelas tecnologias envolvidas e pela arquitetura do sistema. Além disso, passa pelas etapas que ocorreram, com base no processo comentado no capítulo 3.

### 5.1 Elaboração e Estrutura do Sistema

Inicialmente, para levantar os casos de uso, foram realizadas diversas entrevistas com os *stakeholders*. A entrevista inicial ocorreu no dia 12/03, com os coordenadores da disciplinas de projeto de formatura: Prof. Dr. João Batista e Prof. Dr. Paulo Cugnasca. Nessa reunião, também teve a participação do Prof. Dr. Fabio Levy Siqueira. Esta entrevista teve dois focos: determinar o processo geral das disciplinas de formatura e encontrar os demais envolvidos que podem ser entrevistados.

Após a primeira entrevista, ocorreram outras conversas entre os demais envolvidos do processo. Foram elas:

- Nilton: responsável técnico dos eventos de banca teórica e feira prática. Evidenciou as necessidades técnicas que possui nos dois dias de evento e deu sugestões de como o sistema pode ajudar em seu trabalho.
- Fabio Levy: assumiu três papéis, por motivos de dificuldade de agenda, sendo responsável pela infraestrutura de hospedagem, professor orientador e avaliador técnico. Mostrou com mais detalhes o processo como um todo, mas da visão de docente, as participações durante a orientação e a entrega do trabalho de formatura. Além disso, determinou algumas restrições de construção e da infraestrutura do projeto.

Após cada entrevista, o processo *AS IS* foi revisto e refinado, obtendo como resultado os diagramas do apêndice B. Com base nesses diagramas, foram construídos os casos de uso do sistema, detalhados no apêndice C).

### 5.1.1 Atores dos Casos de Uso

Para o sistema, temos os seguintes atores:

- Alunos: responsáveis por realizar as entregas necessárias das disciplinas.
- Professores: realiza a orientação dos alunos e a avaliação das bancas teóricas.
- Convidados: realiza a co-orientação dos alunos (se necessário) e a avaliação das bancas práticas.
- Técnicos: cuida da gestão dos recursos necessários para os eventos teóricos e práticos.
- Coordenadores: gerenciam tudo das disciplinas de projeto de formatura, as entregas, atividades, eventos e centralização das avaliações.
- Público: acessa os recursos públicos, como as monografias finais.

### 5.1.2 Casos de Uso Implementados

Os casos de uso implementados foram divididos em duas iterações: uma responsável por gerenciar as atividades ao longo das disciplinas e outra para cuidar dos eventos de avaliação.



Figura 1: Diagrama de Casos de Uso para Atividades

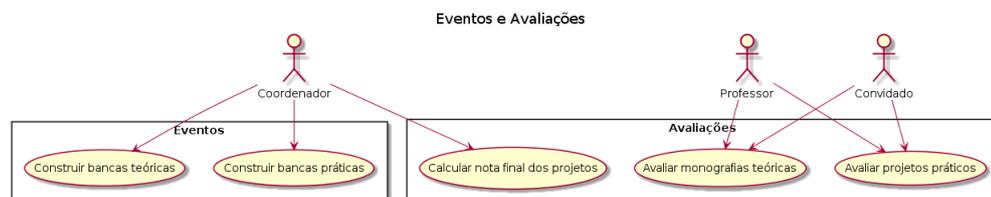


Figura 2: Diagrama de Casos de Uso para Eventos e Avaliações

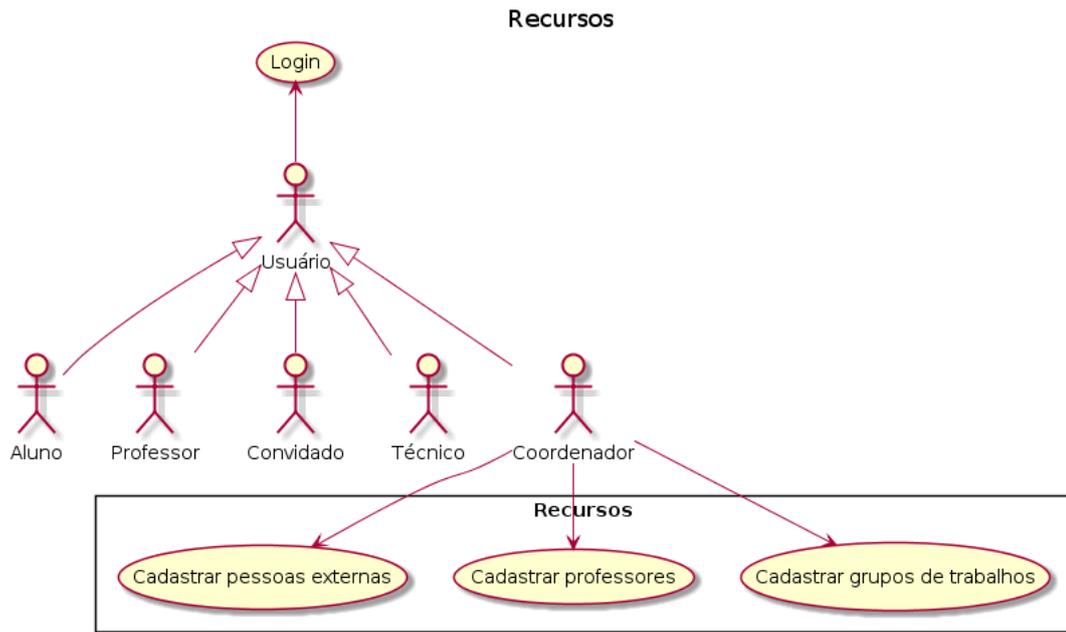


Figura 3: Diagrama de Casos de Uso para Gerenciamento dos Recursos

Nas figuras 1, 2 e 3, são apresentados os casos de uso definidos e implementados do sistema, agrupados de acordo com os casos de uso em comum, para facilitar a visualização: atividades, eventos e avaliações e recursos.

- Primeira Iteração - Julho/Setembro

- Login: alunos, Orientadores, Co-orientadores e Coordenadores acessam sistema de maneira tradicional ou via Senha Única USP (para pertencentes à USP).
- Cadastrar disciplinas: coordenadores cadastram a disciplina, os alunos participantes e as atividades.
- Editar disciplinas: coordenadores editam disciplinas, cadastrando atividades, editando alunos etc.
- Cadastrar grupos de trabalhos: coordenadores cadastram os grupos com os temas e os orientadores, com a confirmação da participação do orientador no grupo.
- Cadastrar professores: coordenadores cadastram professores do departamento que podem orientar/co-orientar.
- Cadastrar pessoas externas: coordenadores cadastram convidados do sistema que podem avaliar projetos nas bancas práticas e/ou co-orientar.

- Entregar atividade: alunos submetem no Google Drive arquivos da atividade para a leitura do orientador, co-orientador e coordenadores.
- Segunda Iteração - Outubro/Novembro
  - Listar entregas: técnicos recebem os arquivos de impressão, com normalização do título, separados por grupo.
  - Listar necessidades adicionais: Técnicos recebem necessidades adicionais revisadas pelos orientadores, separadas por grupo.
  - Construir bancas práticas: coordenadores selecionam os participantes da banca prática, já cadastrados no sistema, e os notifica com comentários sobre o evento.
  - Construir bancas teóricas: coordenadores escolhem participantes da banca teórica do grupo, selecionam o presidente da banca, realizam o agendamento do horário, validando inconsistências (participante já possui horário ocupado) e notificam os participantes por e-mail.
  - Avaliar projetos práticos: participantes da banca prática avaliam os projetos que estão envolvidos, limitando avaliações até o final do dia.
  - Avaliar monografias teóricas: participantes da banca teórica avaliam as monografias que estão envolvidas, gerando comentários e definindo o status do trabalho (aprovado, aprovado com correções, recuperação e reprovado), limitando avaliações até o final do dia.
  - Calcular nota final dos projetos: coordenadores determinam a fórmula para calcular as notas finais, com base nas entregas parciais durante as duas disciplinas e o sistema calcula as notas finais de todos os grupos participantes.

Cada iteração teve sua reunião de validação dos casos de uso: a primeira no dia 4/4 e a segunda no dia 30/10.

### 5.1.3 Casos de Uso Não Implementados

Há também os casos de uso não implementados do sistema, levantados como perspectivas de continuidade do sistema. São eles:

- Importar projetos aprovados: sistema importa monografias revisadas, banners, press-releases, posters e links do site para a base de dados históricos de TCC.

- Cadastrar projetos avulsos: coordenadores cadastram monografias revisadas, banners, press-releases, posters e links do site de trabalhos anteriores para a base de dados históricos de TCC.
- Propor tema de trabalho: alunos e Orientadores propõem temas em busca de orientadores ou alunos dispostos a realizarem.
- Buscar projetos anteriores: usuários externos ao sistema procuram monografias anteriores, de acordo com o nome, ano, turma (semestral ou quadrimestral) e palavras-chave.
- Buscar temas propostos: alunos e orientadores procuram temas propostos por outros alunos e orientadores, de acordo com o nome, palavras-chave e grupos de pesquisa do departamento, podendo se inscrever nas vagas para esses temas (se houverem).

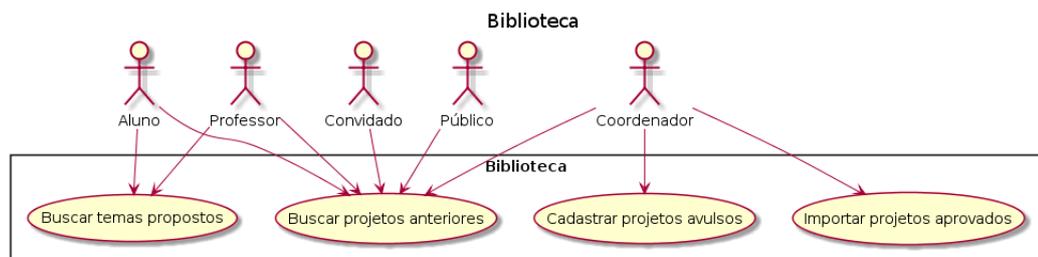


Figura 4: Diagrama de Casos de Uso para Base de Dados (Biblioteca Virtual)

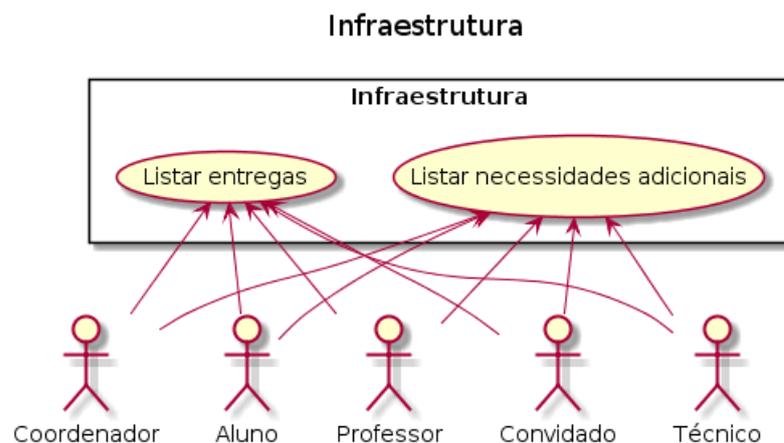


Figura 5: Diagrama de Casos de Uso para Infraestrutura

Nas figuras 4 e 5, são apresentados os casos de uso não implementados do sistema, também agrupados de acordo com os casos de uso em comum: base de dados (projetos anteriores e busca) e infraestrutura (recursos necessários para os eventos).

## 5.2 Arquitetura e Tecnologias Utilizadas

A figura 6 ilustra a arquitetura básica do sistema, composta por quatro componentes básicas: a aplicação em si (Django), o banco de dados principal (PostgreSQL), o banco de dados para cache (Redis) e o banco de dados de arquivos submetidos (Google Drive). O uso de três bancos distintos ocorre pelo fato de um dos requisitos, levantados pelos casos de uso, definir o Google Drive como banco de dados de arquivos, para não sobrecarregar o banco principal. Além disso, por questões de desempenho, informações menores são armazenadas no banco de dados de cache, escolhendo o Redis como solução. A aplicação, o banco de dados principal e o banco de dados de cache estão hospedados no serviço de nuvem Heroku.

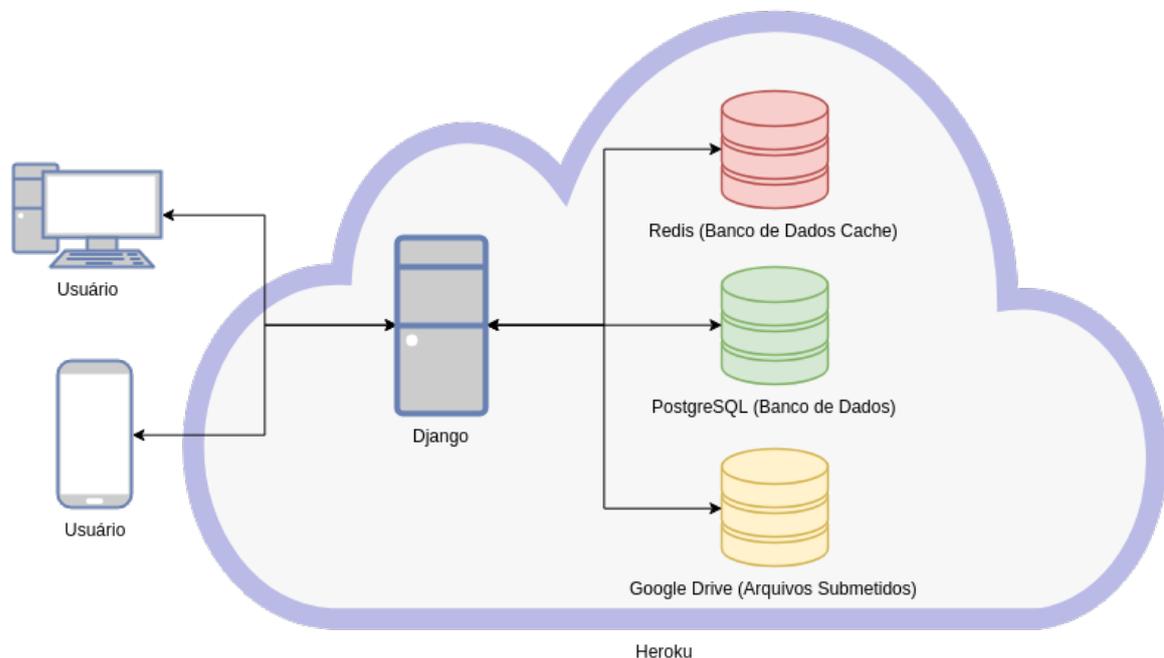


Figura 6: Arquitetura Básica do Sistema

É comum, em desenvolvimento de sistemas, usar soluções prontas como forma de simplificar o desenvolvimento de soluções. Essas soluções são conhecidas como arcabouços (ou *frameworks*), definidos por Sommerville (2011, p. 300) como: "uma estrutura genérica estendida para se criar uma aplicação ou subsistema mais específico". Para este projeto, foi usado um *framework*, em linguagem Python, chamado **Django**. Sua escolha foi pautada na Manutenibilidade, levantada no capítulo 4, dado que alunos farão a manutenção do projeto e que Python, além de ser mais alto nível, é ensinada no primeiro ano dos cursos de Engenharia da Poli.

O Django é estruturado em aplicações (apps), onde cada aplicação corresponde a

uma parte do sistema, geralmente independente e reciclável (ou seja, pode ser usada em aplicações Django diferentes). Cada aplicação segue o padrão *Model-View-Controller*.

O padrão *Model-View-Controller* - MVC é um padrão arquitetural para organizar os componentes da aplicação. Ele é dividido em três grandes grupos (MASTERINGD-JANGO, 2018):

- Modelo (*Model*): fonte de informação, contendo os dados da aplicação, é onde geralmente fica a lógica de negócio.
- Visualização (*View*): camada de apresentação dos dados da aplicação. No caso do Django, ele é chamado de *Template*.
- Controlador (*Controller*): camada de controle que interliga o modelo com a apresentação dos dados. No caso do Django, ele é conhecido como *View*.

A vantagem do padrão está no fluxo de dados que existe entre os grupos, além de evitar códigos com funcionalidades diferentes em lugares errados, como, por exemplo, lógica de negócio na camada de apresentação.

Já para a hospedagem foi usado como apoio o Heroku, um serviço de plataforma (*Platform as a Service* - **PaaS**) de computação em nuvem conhecida no mercado, com a possibilidade de subir aplicações nas linguagens Ruby, Node.js, Java, Python, Clojure, Scala, Go e PHP. O Heroku foi escolhido pela facilidade em subir uma aplicação gratuitamente, o que possibilita testar e validar ideias básicas. A estrutura básica do Heroku funciona com o uso de *dynos*, que são pequenas máquinas tanto para hospedar sua aplicação principal quanto outras auxiliares para serviços externos e/ou paralelos. Resumidamente, ele possui um plano básico gratuito que possibilita testar aplicações de maneira fácil e sem dificuldades de expansão e, se desejar, há máquinas de performance superior, cobradas com base no conceito de *dynos*-hora.

Vale lembrar que um dos requisitos definidos no capítulo 4 é a hospedagem da aplicação em um serviço de nuvem da USP. Por motivos de tempo e limitações de recursos (dado que não havia, à disposição, mais do que uma máquina, dado os ambientes de validação citados no capítulo 3), foram usados três ambientes do Heroku, com os seguintes domínios:

- Desenvolvimento: <https://tccapp-next-release.herokuapp.com>
- Homologação: <https://tccapp-staging.herokuapp.com>

- Produção: <https://tccapp-production.herokuapp.com>

A migração futura para ambientes da USP não será um problema de alto impacto, dado que, em ambas as máquinas, o ambiente roda sistemas operacionais baseados em Linux.

O Django foi usado na versão mais recente estável (2.1), que corrigiu algumas novidades da versão 2.0 e possui suporte de apoio da equipe até Dezembro de 2020 (DJANGO, 2018b). Ele usa, como padrão, o SQLite, o que atendeu bem durante o desenvolvimento. Já o Heroku tem como banco de dados padrão o PostgreSQL, o que exigiu o chaveamento entre os bancos no ambiente local de desenvolvimento e os ambientes de validação hospedados no serviço. Por isso, o projeto possui configurado dois pacotes de gerenciamento de banco de dados, um para SQLite (nativo do Django), outro para PostgreSQL (o Psycopg (SOUTO, 2017)).

Em relação aos acessos do sistema, o Django possui uma limitação (caso use o serviço nativo de login): apenas um *model* pode realizar o acesso. Como são múltiplos atores, para contornar a limitação, cada pessoa terá um único usuário, com vários perfis diferentes de acesso, cada qual com suas permissões e ações. São quatro perfis implementados: estudante, docente, convidado e coordenador. Cada usuário pode ter um ou mais perfis de acesso simultâneos (é o caso, por exemplo, dos coordenadores da disciplina, que também são docentes e podem orientar e avaliar projetos).

### 5.2.1 Aplicações Construídas

O Django possui uma filosofia em sua documentação (DJANGO, 2018a): "Uma aplicação é um aplicativo da Web que faz alguma coisa - por exemplo, um sistema de blog, um banco de dados de registros públicos ou um aplicativo de pesquisa simples. Um projeto é uma coleção de configurações e aplicações para um site específico." Seguindo essa filosofia, o sistema foi estruturado em aplicações menores (*apps*), cada qual com sua responsabilidade:

- home: aplicação responsável por gerenciar as funções básicas a todos os usuários, como por exemplo login/logout, o conteúdo da página de entrada, entre outros.
- users: gerencia os usuários cadastrados do sistema e os seus perfis (aluno, docente, convidado e coordenador)

- disciplines: gerencia as disciplinas de TCC do curso, duas disciplinas por ano (TCC1 e TCC2).
- activities: gerencia as atividades das diversas disciplinas do sistema.
- workgroups: cuida dos grupos de trabalho criados durante as disciplinas.
- deliveries: cuida das entregas das atividades do curso, cada uma realizada pelos grupos de trabalho e revisadas pelos orientadores/co-orientadores.
- rooms: cuida das salas que serão usadas nos eventos de bancas teóricas e práticas.
- events: gerencia os eventos teóricos e práticos que ocorrem nas disciplinas de TCC2.
- allocations: cuida das alocações de cada grupo, para cada evento teórico ou prático, junto com os convidados e docentes que avaliarão o grupo.
- evaluations: gerencia as avaliações que cada docente ou convidado alocado fará nos eventos.
- rules: determina as disciplinas e os eventos que serão usados para calcular as notas finais.
- score: gerencia as notas finais calculadas para cada situação, por grupo de trabalho.

Com as aplicações estabelecidas, os dados foram organizados na arquitetura de banco de dados, exibida na figura 7.

Para a navegação de telas, foram construídos diagramas de navegação simples, nas figuras 8, 9 e 10, exibindo a arquitetura de informação do sistema para cada perfil de usuário.

O código presente está em um repositório público na plataforma de controle de versão GitHub, onde também se encontram detalhes mais técnicos sobre como realizar manutenções no sistema e como rodar na máquina do desenvolvedor, tudo explicado no repositório: <https://github.com/LucasArthur94/tccapp>.

Há também, no apêndice A, um pequeno manual de instruções de uso do sistema, com algumas capturas de tela para ilustrar o fluxo de navegação para os diversos atores.

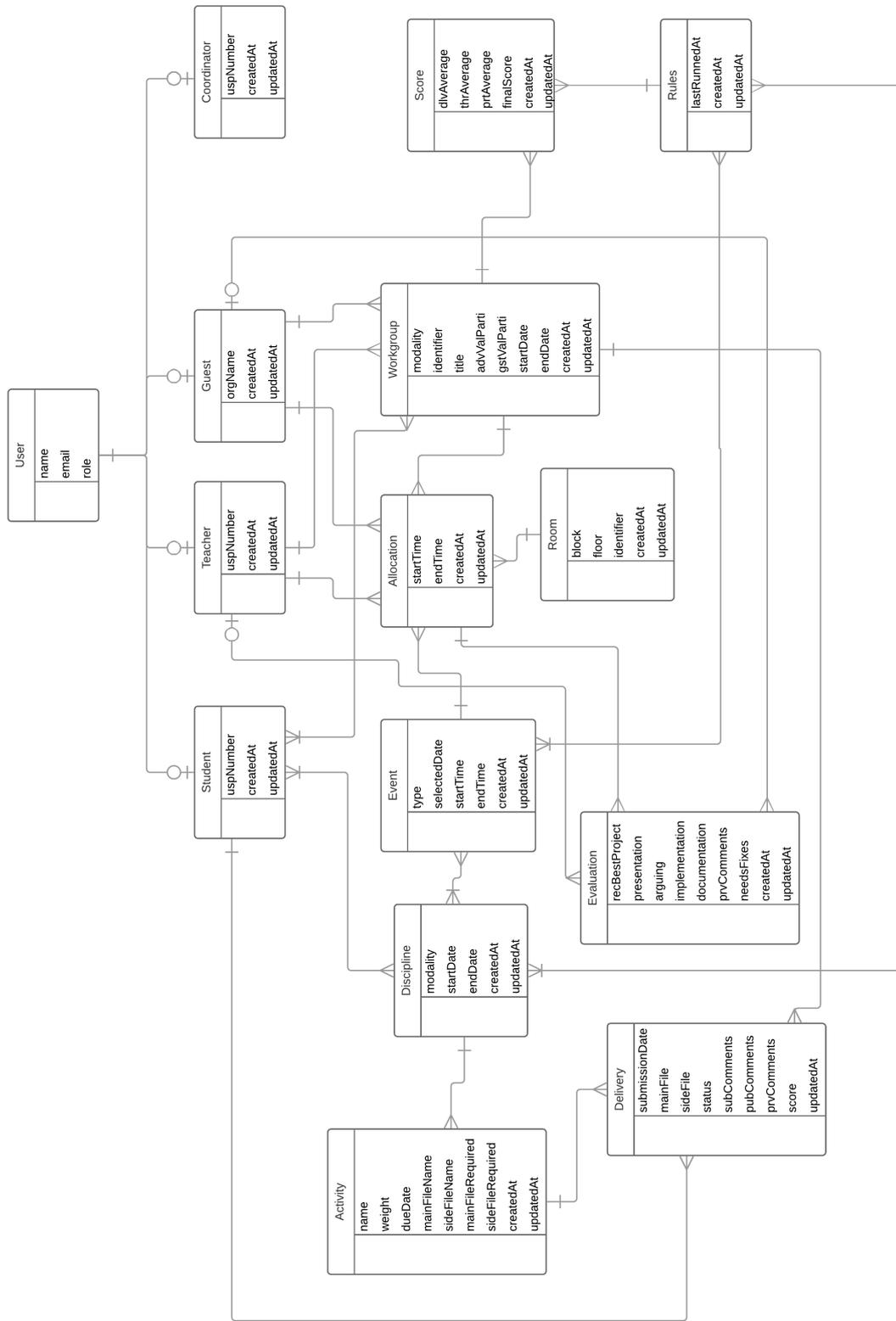


Figura 7: Diagrama Entidade-Relacionamento em Nível Lógico do Sistema Desenvolvido

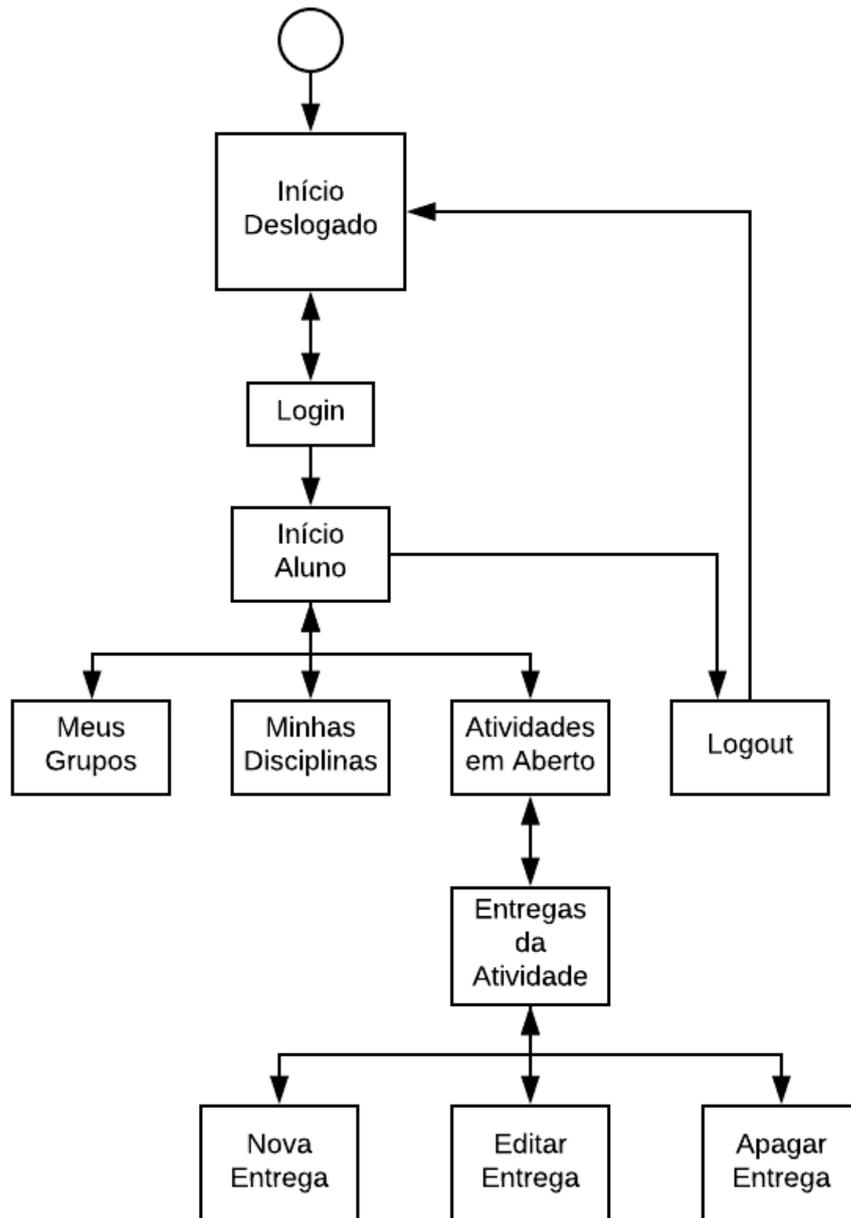


Figura 8: Diagrama de Navegação de Telas Simplificado para o perfil de Aluno

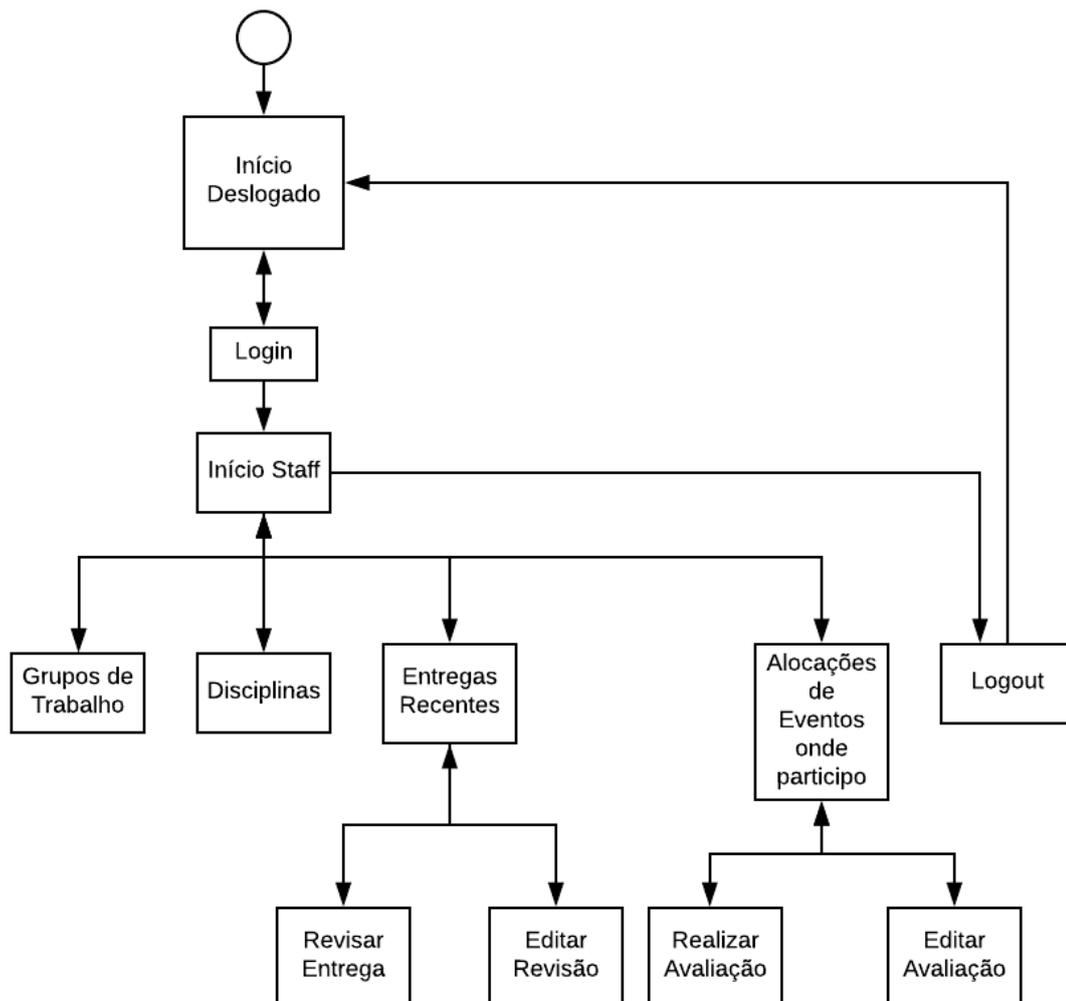


Figura 9: Diagrama de Navegação de Telas Simplificado para o perfil de Docente e Convidado (*Staff*)

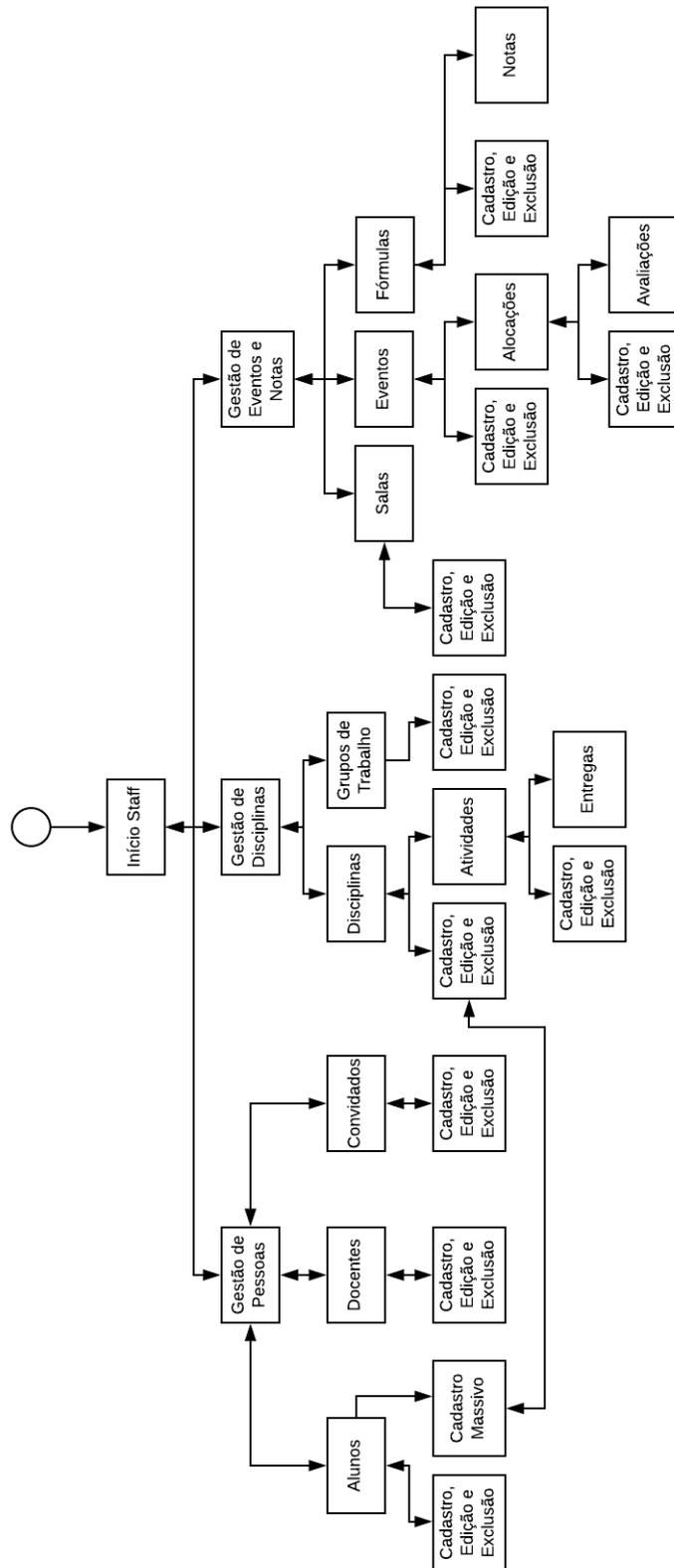


Figura 10: Diagrama de Navegação de Telas Simplificado para o perfil de Coordenador

## 6 TESTE E VALIDAÇÃO

Neste capítulo, serão abordadas as etapas de teste e validação do sistema, as reuniões ocorridas e os testes realizados (automatizados ou não).

### 6.1 Validações

Para validar os Casos de Uso do sistema, algumas reuniões com os *stakeholders* ocorreram durante o processo de desenvolvimento, com o objetivo de acompanhar se o produto estava de acordo com as expectativas levantadas no início do projeto. Nesta seção, serão detalhadas as reuniões que ocorreram após o fim de cada iteração e as alterações levantadas.

#### 6.1.1 Reunião de 28/09

A reunião foi marcada por ser a primeira validação dos *stakeholders*, logo, o sistema como um todo foi uma completa novidade. Para a reunião, foi construído um roteiro de testes a ser executado em conjunto com os coordenadores da disciplina (Prof. Dr. João Batista e Prof. Dr. Paulo Cugnasca):

- Logar como administrador (João ou Paulo)
  - Sentir navegação do sistema em todas as partes
  - Cadastrar 1 aluno
  - Cadastrar alunos massivamente
- Logar como aluno recém cadastrado
  - Sentir navegação do sistema em todas as partes
- Logar como administrador (João ou Paulo)

- Cadastrar docente
- Cadastrar convidado externo
- Cadastrar disciplina
- Cadastrar grupo de trabalho
- Logar como orientador (docente)
  - Confirmar participação
- Logar como co-orientador (convidado externo)
  - Confirmar participação
- Logar como administrador (João ou Paulo)
  - Cadastrar atividade
  - Logar como aluno
  - Realizar entrega da atividade
- Logar como co-orientador
  - Revisar atividade
- Logar como orientador
  - Revisar atividade
- Logar como co-orientador
  - Tentar revisar atividade
  - Ver comentários privados
- Logar como aluno
  - Ver comentários públicos

Nesta reunião, a aceitação do sistema foi satisfatória, apenas com as seguintes sugestões e ressalvas:

- Cadastro massivo de pendentes e inscritos no sistema.
- Customização dos textos dos emails.

- Flexibilidade nos arquivos das atividades.
- Comentários dos alunos nas entregas das atividades.
- Alteração/Recuperação de Senha.

### 6.1.2 Reunião de 30/11

Já nesta reunião, o foco da validação era da parte de gerenciamento dos eventos, como alocação de bancas e avaliações em geral. Assim como na primeira reunião, um roteiro de testes foi elaborado para ser usado como base na reunião de validação. Como pré-condição, para facilitar a validação, o ambiente de homologação já terá participantes, grupos, disciplinas, atividades e entregas já presentes no sistema:

- Logar como administrador (João ou Paulo)
  - Criar evento teórico
  - Criar evento prático
  - Alocar grupo para evento teórico
  - Alocar grupo para evento prático
  - Avisar convidados por email do evento teórico
  - Avisar convidados por email do evento prático
- Logar como convidado da banca teórica (docente)
  - Avaliar grupo em qual foi alocado
- Logar como convidado da banca prática (convidado)
  - Avaliar grupo em qual foi alocado
- Logar como administrador (João ou Paulo)
  - Criar fórmula para calcular nota (escolher disciplinas e eventos)

Vale ressaltar que, nessa iteração, não foi possível concluir os Casos de Uso relacionados à listagem de entregas e necessidades adicionais.

Como resultado dessa reunião, a aceitação se manteve positiva, sem grandes ressalvas ao resultado.

## 6.2 Testes Executados

Além do processo de validação, seguindo a filosofia apresentada no capítulo 2, diversos testes foram escritos para reforçar a prática de verificação do sistema, para saber se ele está sendo desenvolvido corretamente. Nesta seção, serão discutidos os cenários de teste escritos e as ferramentas e dificuldades encontradas para escrevê-los.

### 6.2.1 Cenários de Teste

Dado que o conjunto de testes possíveis é infinito, foram selecionados alguns testes a serem escritos. Para o sistema, dois tipos de teste foram adotados: Unidade e Interface.

Nos testes de unidade, foram testados apenas os métodos escritos com função diferenciada e primordial para o sistema, como por exemplo os métodos de classe dos modelos. Já os testes de interface cobrem, por meio de ferramentas automatizadas, execuções de tela do sistema, verificando o fluxo de negócio de cada aplicação.

Cada aplicação escrita foi testada por uma outra aplicação interna a ela, específica pra teste, contendo os seguintes testes:

- Modelo (*test\_model*): Testa os métodos de classe específicos para cada modelo do sistema. Caso não tenha nenhum método escrito, o arquivo fica em branco.
- Formulário (*test\_form*): Testa os formulários relacionados à aplicação, o que é útil para verificar se o cadastro de um novo dado está funcionando corretamente.
- Tela (*test\_view*): Testa a aplicação como um todo, usando a tela como referência.

Todas as telas possuem testes escritos, bem como todos os formulários. Apenas os modelos de alocação, avaliação, usuários e grupos de trabalho não têm testes, dado que não possuem métodos nos modelos. No total, 39 testes foram elaborados para o sistema.

### 6.2.2 Ferramentas de Teste

O Django possui uma biblioteca extensa para realizar testes no sistema. Para testes de unidade, usar a parte nativa resolve o problema. Já para escrever testes de interface, a complexidade foi maior, dado que a aplicação precisaria rodar em um servidor para testes, além de uma ferramenta específica para testes automatizados de tela.

Existem diversas ferramentas específicas para rodar testes automatizados que envolvam interface do sistema. Neste projeto, a ferramenta escolhida foi o Selenium, amplamente utilizada nos mais diversos frameworks. Para o Django, a ferramenta específica usada foi o *Selenium WebDriver*, totalmente integrável com as ferramentas de teste. Em contrapartida, soluções desse aspecto necessitam que a aplicação esteja rodando em um servidor, o que não é o caso nos demais testes.

Para resolver o problema de ter um servidor dedicado para testes, o Django oferece uma solução específica chamada *StaticLiveServerTestCase* (DJANGO, 2018c): uma classe do Django que sobe um servidor exclusivo para os testes escritos na classe que herdá-la. Esta solução atende a necessidade do *Selenium* de ter um servidor dedicado para rodar os testes automatizados.

## 7 CONSIDERAÇÕES FINAIS

O projeto foi parcialmente implementado, cobrindo parte do escopo do problema, usando técnicas de Engenharia de Software: modelagem de processos, levantamento de requisitos, casos de uso e verificação/validação. Usou tecnologias de referência no mercado para solucionar o problema e gerou valor ao automatizar etapas do processo existente, permitindo seu crescimento em projetos futuros.

Espera-se também que o projeto entre em atividade para o departamento de Engenharia de Computação da Escola Politécnica da USP, assim como foi o projeto do Portal de Estágios do PCS. O sistema foi projetado para facilitar um processo fundamental para o curso e permitir seu crescimento, em especial as parcerias com empresas.

### 7.1 Resultados Alcançados

Nesta seção, serão abordados os resultados alcançados pelo sistema, as funcionalidades desenvolvidas e as expectativas alcançadas.

#### 7.1.1 Casos de Uso Construídos

No projeto, doze casos de uso especificados do sistema foram elaborados, construindo toda a parte de gerenciamento de pessoas, disciplinas, atividades, eventos e avaliações, satisfazendo as expectativas dos *stakeholders* e atendendo parte dos problemas encontrados no processo de levantamento de requisitos. Apesar das partes faltantes, o sistema é considerado usável por parte dos coordenadores a partir de 2019.

### 7.2 Próximos Passos

Alguns casos de uso ficaram de fora das iterações do sistema, sendo eles:

- Especificados:
  - Listar entregas
  - Listar necessidades adicionais
  
- Não especificados:
  - Importar projetos aprovados
  - Cadastrar projetos avulsos
  - Buscar projetos anteriores
  - Propor tema de trabalhos
  - Buscar temas propostos

Além dos casos de uso faltantes, para o sistema ser usado pela comunidade politécnica, resta colocar o sistema na infraestrutura da universidade, tarefa não realizada neste projeto.

## REFERÊNCIAS

- ABPMP, A. of B. P. M. P. B. *Business Process Management Common Book of Knowledge - CBOK*. third. [S.l.]: Association of Business Process Management Professionals Brazil - ABPMP, 2013.
- ANGELI, J. *O que é mapeamento de processos AS IS/TO BE?* 2018. Disponível em: <http://www.neomind.com.br:81/blog/mapeamento-de-processos-as-is-to-be/>.
- BITTNER, I. S. K. *Use Case Modeling*. [S.l.]: Addison-Wesley Professional, 2002.
- DJANGO, D. S. F. *Django documentation*. 2018. Disponível em: <https://docs.djangoproject.com/en/2.1/>.
- DJANGO, D. S. F. *Django Download*. 2018. Disponível em: <https://www.djangoproject.com/download/>.
- DJANGO, D. S. F. *The staticfiles app*. 2018. Disponível em: <https://docs.djangoproject.com/pt-br/2.1/ref/contrib/staticfiles/>.
- HAVEY, M. *Essential Business Process Modeling*. [S.l.]: O'Reilly, 2015.
- IBM. *Rational Unified Process - Best Practices for Software Development Teams*. 2011. Disponível em: [https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251\\_bestpractices\\_TP026B.pdf](https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf).
- MASTERINGDJANGO, T. D. B. *The Model-View-Controller Design Pattern*. 2018. Disponível em: <https://djangobook.com/model-view-controller-design-pattern/>.
- MESZAROS, G. *xUnit Test Patterns - Refactoring Test Code*. [S.l.]: Pearson, 2007. (Addison-Wesley).
- OMG, O. M. G. *Business Process Model and Notation (BPMN)*. [S.l.]: Object Management Group®, 2013.
- OMG, O. M. G. *OMG® Unified Modeling Language® (OMG UML®)*. [S.l.]: Object Management Group®, 2017.
- RAGAN, T. *Continuous Delivery vs. Continuous Deployment*. 2017. Disponível em: <https://devops.com/continuous-delivery-vs-continuous-deployment/>.
- RASMUSSEN, J. *The Agile Samurai: How Agile Masters Deliver Great Software*. first. [S.l.]: The Pragmatic Programmers, 2010.
- SELENIUM. *Selenium - Web Browser Automation*. 2018. Disponível em: <https://www.seleniumhq.org/>.
- SOMMERVILLE, I. *Engenharia de Software*. ninth. [S.l.]: Pearson, 2011.

SOUTO, L. *Integrando Django com PostgreSQL*. 2017. Disponível em: ([https://medium.com/@lucas\\_souto/integrando-django-com-postgresql-58b3520ddf6e](https://medium.com/@lucas_souto/integrando-django-com-postgresql-58b3520ddf6e)).

SWEBOK, C. S. I. *Guide to the Software Engineering Body of Knowledge*. third. [S.l.]: Institute of Electrical and Electronics Engineers, 2014.

# APÊNDICE A – MANUAL DO USUÁRIO

Este apêndice contém um manual básico do usuário, mostrando a navegação em algumas telas e elucidando os fluxos de interação com o sistema para os três perfis básicos de usuário: Aluno, Professor/Convidado e Coordenador.

O sistema foi construído usando Django na versão 2.1, roda com um banco de dados (que pode ser SQLite ou PostgreSQL, dependendo do ambiente) e com uma instância do Redis para cache (em especial para suporte do Select2). Para hospedar os arquivos, foi usado o Google Drive. Por fim, para o front-end, foi usado como auxílio o MDBootstrap e também o JQuery. Todas as telas foram construídas pensando na navegação mobile, o que permite que todas as instruções deste manual sejam executadas em *smartphones* sem maiores problemas.

Para entender melhor a navegação no sistema, as funcionalidades serão divididas em três grandes grupos:

- Alunos: Podem verificar seus grupos de trabalho e suas disciplinas inscritas, bem como as atividades em aberto e realizar submissões delas. Podem ver também parte dos resultados das revisões das submissões.
- Professores/Convidados: Possuem maior poder no sistema, podendo ver todas as disciplinas e os grupos de trabalho que são coordenadores. Podem realizar as revisões das entregas dos alunos, como também podem avaliar os projetos em que foram alocados nos eventos.
- Coordenadores: Possuem plenos poderes no sistema, podendo gerenciar todos os recursos existentes.

## A.1 Aluno



Figura 11: Tela Inicial do Aluno

Na tela inicial de aluno, são exibidas as atividades da disciplina as quais o aluno participa. No menu superior, há as opções de ver os grupos de trabalho onde participa e as disciplinas em que está cadastrado.

### A.1.1 Tela Inicial

Para cada atividade listada na tela inicial, é possível ver as entregas já realizadas pelo seu grupo ao selecionar **Ver Entregas**. Na listagem das entregas da atividade escolhida, é possível ver cada entrega com mais detalhes (ao selecionar **Ver Mais**, alterar detalhes da entrega (caso não tenha sido avaliada) ao selecionar **Editar**, bem como realizar novas entregas ao selecionar **Nova Entrega**.

Também é possível, na própria tela inicial, realizar uma entrega para uma atividade da lista, ao selecionar **Realizar Entrega**. Cada entrega alerta, por e-mail, os orientadores/co-orientadores. Vale lembrar que entregas podem ocorrer enquanto a atividade estiver aberta e edições às entregas apenas quando ela não tiver sido avaliada.

## A.1.2 Meus Grupos

Entrando na parte de **Meus Grupos**, é possível ver todos os grupos de trabalho em que o próprio aluno está inscrito. Selecionando o ícone de 👁, é possível ver mais detalhes do grupo, como a participação dos orientadores, o nome do projeto, entre outros.

## A.1.3 Minhas Disciplinas

Já na parte de **Minhas Disciplinas**, é possível ver detalhes das disciplinas em que o aluno está inscrito, selecionando o ícone de 👁 na disciplina desejada, ou ver as atividades da disciplina selecionando o ícone de ☰. Dentro da listagem de atividades, o procedimento de navegação é o mesmo citado anteriormente, para ver detalhes, realizar e editar submissões.

## A.2 Professor/Convidado

The screenshot shows the interface for a Professor/Convidado user. At the top, there is a navigation bar with 'TCC Poli USP', 'Grupos de Trabalho', and 'Disciplinas'. On the right, there are links for 'Alterar Senha' and 'Sair'. The main content area starts with a welcome message: 'Bem vindo, Fábio Levy Siqueira!' and 'Minhas Avaliações em Aberto'. Below this, there is a card for a group evaluation: 'Grupo C10 - Sala AT-AT - Banca Teórica', dated 'Dia: 19 de Dezembro de 2018', with a time of '08:20 - 08:50'. The evaluators listed are 'João Batista Camargo Junior' and 'Fábio Levy Siqueira'. A 'VER AVALIAÇÕES' button is present. Underneath, the 'Entregas' section shows three assignment cards, all with the status 'Status: Avaliada pelo Orientador'. The first card is for 'Atividade: Monografia' by 'Lucas Arthur Felgueiras' with a grade of 7,5. The second is for 'Atividade: Site' by 'Lucas Arthur Felgueiras' with a grade of 6,0. The third is for 'Atividade: Press-Release' by 'Lucas Arthur Felgueiras' with a grade of 6,5. Each card has 'VER MAIS' and 'REAVALIAR' buttons. The footer contains the copyright notice '©2018 PCS Poli USP | Site do PCS'.

Figura 12: Tela Inicial do Professor/Convidado

Já na tela inicial de professor/convidado, são exibidas as informações pertinentes às submissões dos grupos de trabalho em quem é orientador/co-orientador, além dos eventos em que foram alocados como avaliadores teóricos ou práticos. No menu, é possível ver todas as disciplinas existentes no sistema, como também ver todos os grupos de trabalho em que está incluso.

## A.2.1 Tela Inicial

Na tela inicial, a parte de avaliações se assemelha a do aluno, com o diferencial que a listagem é de submissões de atividades. Para cada atividade, é possível ver mais detalhes ao selecionar **Ver Mais**, como também realizar a avaliação selecionando **Avaliar** ou **Reavaliar**, de acordo com o estado da submissão. Vale lembrar que, caso seja um co-orientador, não é possível editar a avaliação feita por um orientador.

Já a parte de eventos, é possível ver quais eventos o indivíduo irá participar, qual o tipo do evento, o grupo relacionado, outros avaliadores, a sala, o dia e o horário correspondentes. Para ver as avaliações do grupo, basta selecionar **Ver Avaliações**.

TCC Poli USP Grupos de Trabalho Disciplinas Alterar Senha Sair

Avaliar Trabalho

Recomenda para Melhor Projeto da Feira?  
 0 (Não recomendo)  1  2  3  4  5 (Recomendo muito)

Nota da Apresentação (0,0 - 2,0)

Nota da Arguição (0,0 - 2,0)

Nota da Implementação (0,0 - 3,0)

Nota da Documentação (0,0 - 3,0)

Necessita de Correções?

Comentários e Correções

©2018 PCS Poli USP | Site do PCS

Figura 13: Tela de Nova Avaliação Teórica

Na tela de avaliações, apenas as próprias avaliações serão exibidas (caso o docente/convidado não seja também o coordenador da disciplina). Para realizar uma nova avaliação, basta selecionar **Nova Avaliação**. A avaliação ainda é editável, ao selecionar **Editar**, porém apenas enquanto o evento estiver acontecendo, pois ao término do evento, as avaliações não são mais editáveis.

## A.2.2 Grupos de Trabalho

Entrando na parte de **Grupos de Trabalho**, é possível ver todos os grupos de trabalho em que o próprio professor/convidado está inscrito. Semelhante ao comportamento dos alunos, selecionando o ícone de , é possível ver mais detalhes do grupo, como a

participação dos orientadores, o nome do projeto, entre outros. Além dessa funcionalidade, há também a confirmação de participação, selecionando **Confirmar Participação** no grupo desejado.

### A.2.3 Disciplinas

Já na parte de **Disciplinas**, é possível ver detalhes de todas as da disciplina, também selecionando o ícone de , ou ver as atividades de uma disciplina específica selecionando o ícone de . Dentro da listagem de atividades, é possível ver também, para cada uma, as entregas realizadas dos grupos em que orienta, ao selecionar **Ver Entregas**. A listagem de entregas é semelhante a que foi comentada anteriormente na tela inicial, permitindo avaliações e detalhes maiores de cada entrega submetida.

## A.3 Coordenador

TCC Poli USP | Gestão de Pessoas | Gestão de Disciplinas | Gestão de Eventos e Notas | Alterar Senha | Sair

Bem vindo, João Batista Camargo Junior!  
Minhas Avaliações em Aberto

Grupo C10 - Sala ATAT - Banca Teórica	Grupo C10 - Sala ATAT - Feira Prática
<p>Dia: 19 de Dezembro de 2018 Horário: 08:20 - 08:50 Avaliadores: • João Batista Camargo Junior</p> <p>VER AVALIAÇÕES</p> <p>VER MAIS EDITAR APAGAR</p>	<p>Dia: 18 de Dezembro de 2018 Horário: 08:20 - 08:50 Avaliadores: • João Batista Camargo Junior</p> <p>VER AVALIAÇÕES</p> <p>VER MAIS EDITAR APAGAR</p>

Entregas

Status: Avaliada pelo Orientador	Status: Avaliada pelo Orientador	Status: Avaliada pelo Orientador
<p>Atividade: Monografia Autor: Lucas Arthur Felgueiras Grupo de Trabalho: Sistema de Gerenciamento de Disciplinas de TCC Data da Submissão: 29 de Novembro de 2018 Nota: 7,5</p> <p>VER MAIS REAVALIAR APAGAR</p>	<p>Atividade: Site Autor: Lucas Arthur Felgueiras Grupo de Trabalho: Sistema de Gerenciamento de Disciplinas de TCC Data da Submissão: 29 de Novembro de 2018 Nota: 6,0</p> <p>VER MAIS REAVALIAR APAGAR</p>	<p>Atividade: Press-Release Autor: Lucas Arthur Felgueiras Grupo de Trabalho: Sistema de Gerenciamento de Disciplinas de TCC Data da Submissão: 29 de Novembro de 2018 Nota: 6,5</p> <p>VER MAIS REAVALIAR APAGAR</p>

©2018 PCS Poli USP | Site do PCS

Figura 14: Tela Inicial do Coordenador, com permissões de Professor

A tela inicial de coordenador, em geral, não costuma ter nada, além do menu inicial. Como a aplicação permite múltiplos perfis, os coordenadores da disciplina também podem orientar grupos e avaliar projetos como um docente comum. Logo, a tela inicial corresponde a mesma do docente/convidado. Se, futuramente, um coordenador novo for cadastrado e ele não estiver na lista de docentes ou de convidados, a tela inicial terá apenas a mensagem de boas vindas.

### A.3.1 Gestão de Pessoas

Ao selecionar **Gestão de Pessoas**, surgem três opções, todas relacionadas à usuários: **Docentes**, **Alunos** e **Convidados Externos**.

#### A.3.1.1 Docentes

Ao selecionar **Docentes**, surge toda a lista de docentes do sistema. Para cadastrar um docente, basta selecionar **Novo Docente**, enviando um e-mail com instruções de acesso ao novo docente no término do cadastro. Para gerenciar um docente existente, basta selecionar as opções:  para ver mais,  para editar e  para excluí-lo.

#### A.3.1.2 Alunos

Ao selecionar **Alunos**, o comportamento é semelhante, com a lista de alunos do sistema. Para cadastrar um aluno, basta selecionar **Novo Aluno**, enviando um e-mail com instruções de acesso ao novo aluno no término do cadastro. Para gerenciar um aluno existente, basta selecionar as opções:  para ver mais,  para editar e  para excluí-lo (lembrando que é impossível excluir seu próprio usuário).

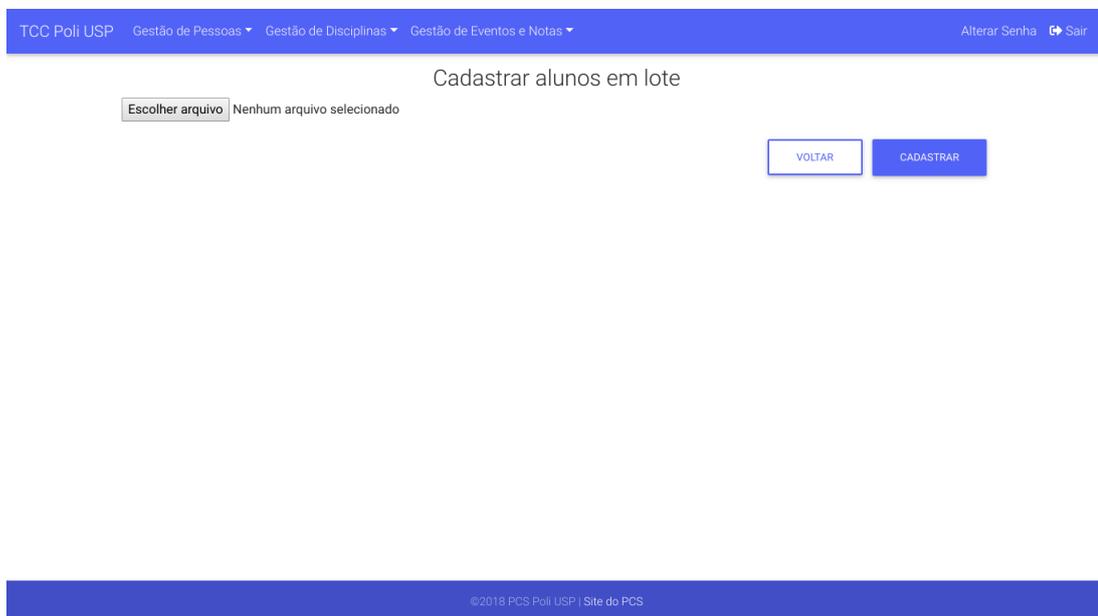


Figura 15: Tela de Cadastro Massivo de Alunos

O diferencial desta parte está no cadastro massivo de alunos (usando como entrada a planilha de nomes gerada pelo JúpiterWeb). Ao selecionar **Cadastrar Alunos Massivamente**, será possível carregar a planilha e, para cada nome encontrado, ele será

cadastrado e um e-mail com instruções será disparado. Ao término do cadastro, você será redirecionado à tela de cadastro de disciplinas, dado que é usual realizar o cadastro massivo na hora de cadastrar uma disciplina.

### A.3.1.3 Convidados Externos

Por fim, ao selecionar **Convidados Externos**, surge toda a lista de convidados do sistema. Novamente, para cadastrar um convidado, basta selecionar **Novo Convidado Externo**, enviando um e-mail com instruções de acesso ao novo convidado no término do cadastro. Para gerenciar um convidado existente, basta selecionar as opções:  para ver mais,  para editar e  para excluí-lo.

## A.3.2 Gestão de Disciplinas

Já ao selecionar **Gestão de Disciplinas**, surgem duas opções relacionadas ao dia-a-dia das disciplinas de formatura: **Disciplinas** e **Grupos de Trabalho**.

### A.3.2.1 Disciplinas

Ao selecionar **Disciplinas**, o comportamento é bem semelhante ao dos outros perfis de usuário, porém listando todas as disciplinas. Para cadastrar uma nova disciplina, selecione **Nova Disciplina**, onde dentro do formulário há uma opção também para cadastro massivo de alunos (**Cadastrar Alunos Massivamente**). Para ver mais sobre uma em específico, basta selecionar o ícone de , para editá-la, o ícone de , para apagá-la, . Para ver as atividades de uma disciplina específica, deve-se selecionar o ícone de .

Dentro da listagem de atividades, é possível cadastrar uma nova atividade em **Nova Atividade da Disciplina**. Para gerenciar uma atividade criada, basta selecionar a opção desejada: **Ver mais**, **Editar** ou **Apagar**. É possível ver também, para cada uma, as entregas realizadas de todos os grupos, ao selecionar **Ver Entregas**. A listagem de entregas é semelhante a que foi comentada anteriormente na tela inicial, permitindo avaliações (caso seja orientador/co-orientador) e detalhes maiores de cada entrega submetida.

### A.3.2.2 Grupos de Trabalho

Agora, ao selecionar **Grupos de Trabalho**, a gestão é semelhante ao que foi comentado anteriormente, porém com a visualização de todos os grupos de trabalho. Para cadastrar um novo grupo, basta selecionar **Novo Grupo de Trabalho**, enviando e-mail

para todos os envolvidos no final do cadastro. Selecionando o ícone de 👁, é possível ver mais detalhes do grupo, selecionando ✎, é possível editá-lo, selecionando 🗑, é possível apagá-lo. Se o coordenador estiver em um grupo de trabalho, também há a confirmação de participação no grupo ao selecionar **Confirmar Participação**.

### A.3.3 Gestão de Eventos e Notas

A última seção do menu **Gestão de Eventos e Notas** possui três opções relacionadas aos eventos e notas finais: **Salas**, **Eventos** e **Fórmulas**.

#### A.3.3.1 Salas

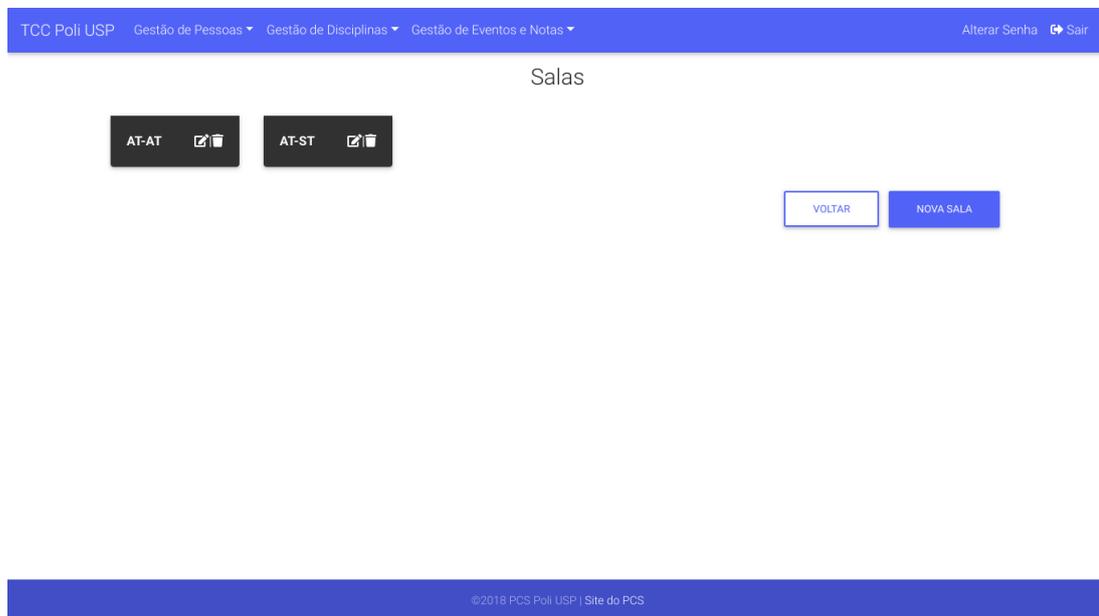


Figura 16: Tela de Listagem de Salas

Ao selecionar **Salas**, surge a lista de salas do sistema. Para cadastrar uma nova sala, basta selecionar **Nova Sala** e escolher as opções de bloco, andar e identificador. Para editar uma sala da listagem, selecione ✎, para apagar uma sala, 🗑.

TCC Poli USP Gestão de Pessoas Gestão de Disciplinas Gestão de Eventos e Notas Alterar Senha Sair

Informações da Sala

Bloco  
Bloco B

Andar  
Segundo

Identificador da Sala  
08

VOLTAR SALVAR

©2018 PCS Poli USP | Site do PCS

Figura 17: Tela de Nova Sala, cadastrando como exemplo a B2-08

### A.3.3.2 Eventos

Agora, ao selecionar **Eventos**, surge a lista de eventos totais do sistema, tanto bancas teóricas como feiras práticas. Para cadastrar um novo evento, basta selecionar **Novo Evento**, lembrando que para cada evento são necessárias duas disciplinas: uma quadri-mestral e outra semestral. Para cada evento, é possível realizar seu gerenciamento: **Ver mais**, **Editar** ou **Apagar**. E em cada evento, é possível alocar os grupos de trabalho que participarão da feira ou da banca, selecionando **Alocar Grupos**.

Na listagem de alocações de grupos, é possível ver todos os grupos alocados, com dia, horário, sala e avaliadores, permitindo seu gerenciamento com as opções: **Ver mais**, **Editar** ou **Apagar**. Para realizar uma nova alocação, basta selecionar **Nova Alocação** e informar os detalhes, lembrando que o horário deve estar dentro da janela do evento, bem como não deve ter nenhuma sala ocupada no horário desejado.



Figura 18: Tela de Detalhes da Avaliação

Em cada alocação, é possível ver as avaliações já realizadas, selecionando **Ver Avaliações**. Na listagem de avaliações, pode-se ver todas as avaliações do grupo e, para cada uma, é possível realizar seu gerenciamento: **Ver mais**, **Editar** ou **Apagar**, lembrando que as duas últimas opções são possíveis apenas se o coordenador estiver alocado para avaliar, podendo excluir e editar apenas sua avaliação.

### A.3.3.3 Fórmulas

Por fim, ao selecionar **Fórmulas**, surge a lista de fórmulas totais. O conceito de fórmula consiste em juntar componentes de avaliação e, com base nelas, calcular as notas. As componentes da fórmula são: disciplina semestral, disciplina quadrimestral, banca teórica e feira prática. O cálculo geral é realizado da seguinte forma:

Sendo  $n$  a quantidade de atividades:

$$mediaAtividades = \frac{\sum_{i=1}^n (pesoAtv_i * notaAtv_i)}{\sum_{i=1}^n (pesoAtv_i)} \quad (A.1)$$

Sendo  $p$  a quantidade de avaliadores práticos:

$$mediaPratica = \frac{\sum_{k=1}^p (notaAvaliador_k)}{p} \quad (A.2)$$

Sendo  $t$  a quantidade de avaliadores teóricos (em geral são três):

$$mediaTeorica = \frac{\sum_{j=1}^t (notaAvaliador_j)}{t} \quad (A.3)$$

Por fim, a média final:

$$mediaFinal = \frac{mediaAtividades + pesoBanca * mediaTeorica + pesoFeira * mediaPratica}{1 + pesoBanca + pesoFeira} \quad (A.4)$$

Na hora de criar uma fórmula (selecionando **Nova Fórmula**), deve-se selecionar as disciplinas relacionadas para obter as atividades e os eventos para obter as notas dos avaliadores. Para calcular a média de atividades, apenas a última submissão revisada por orientador/co-orientador é levada em consideração para cada atividade. A vantagem de usar este método está na generalização do cálculo, tornando possível estabelecer fórmulas também para a primeira disciplina de TCC, permitindo eventos teóricos e práticos para a parte de especificação do projeto.

Grupo de Trabalho	Média das Atividades	Média da Banca Teórica	Média da Feira Prática	Média Final
C10	7,2	9,0	8,0	8,2

Figura 19: Tela de Notas Calculadas por Grupo de Trabalho

Após salvar a nova fórmula, é possível realizar seu gerenciamento: ver mais selecionando , editar selecionando  e apagar selecionando . Para efetuar o cálculo das notas com base nas fórmulas acima, selecione . Após o cálculo, para ver as notas dos grupos, basta selecionar . Em cada nota, há o grupo de trabalho relacionado, as componentes individuais e a média final. O cálculo pode ser feito quantas vezes desejar, pois a cada novo cálculo a pontuação é recalculada e salva na componente do grupo de trabalho.

## A.4 Funcionalidades Comuns

Para todos os perfis de usuário, há funcionalidades básicas e que são importantes para o funcionamento do sistema. São elas:

### A.4.1 Recuperar Senha

No menu sem acessar o sistema, ao selecionar **Recuperar Senha**, a pessoa é redirecionada a um formulário onde, ao entrar com o e-mail, se ele estiver no sistema, ele recebe um e-mail com instruções para redefinir a senha, de acordo com as regras do formulário.

### A.4.2 Alterar Senha

No menu, entrando no sistema, ao selecionar **Alterar Senha**, a pessoa é redirecionada a um formulário onde tem a possibilidade de redefinir a senha, inserindo a senha antiga e a nova senha, de acordo com as regras do formulário.

# APÊNDICE B – DIAGRAMAS BPMN

## B.1 Introdução

Os diagramas de Modelo e Notação de Processos de Negócio (Business Process Model and Notation - BPMN) servem para modelar um processo de negócio de maneira a unificar a visão sobre aquele processo e encontrar possíveis otimizações para o mesmo processo, evidenciando os pontos onde um futuro sistema pode agir.

## B.2 Processo antes do Sistema

Os diagramas BPMN foram gerados com base nas entrevistas realizadas durante o processo de levantamento de requisitos. Vale lembrar que eles foram divididos de maneira a facilitar a visualização neste documento.

As figuras 20 e 21 mostram a modelagem BPMN para a disciplina de TCC1.

Já as figuras 22 e 23 mostram a modelagem BPMN para a disciplina de TCC2, porém desconsiderando os eventos de feira prática e banca teórica, para reduzir a complexidade e facilitar a visualização.

As figuras 24 e 25 mostram a modelagem BPMN os eventos de feira prática e banca teórica, responsáveis pela etapa final da disciplina de TCC2. A parte de recuperação de TCC2 foi desconsiderada nessas figuras, também para facilitar a visualização.

Por fim, para o processo de recuperação, foram modelados os diagramas das figuras 26 e 27, que encerram o fluxo completo da disciplina de TCC2.

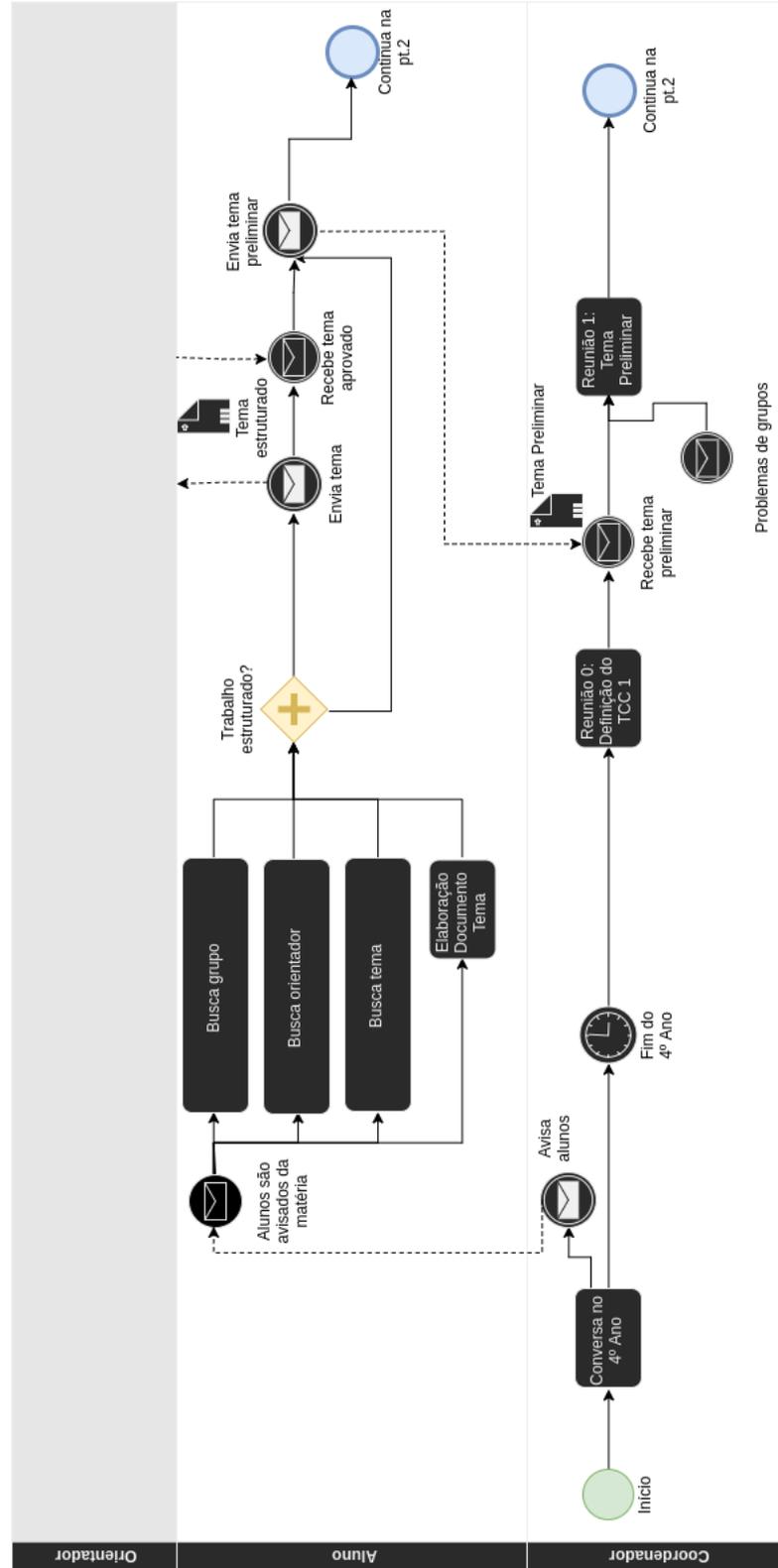


Figura 20: Diagrama BPMN para a disciplina de TCC 1 - Parte 1

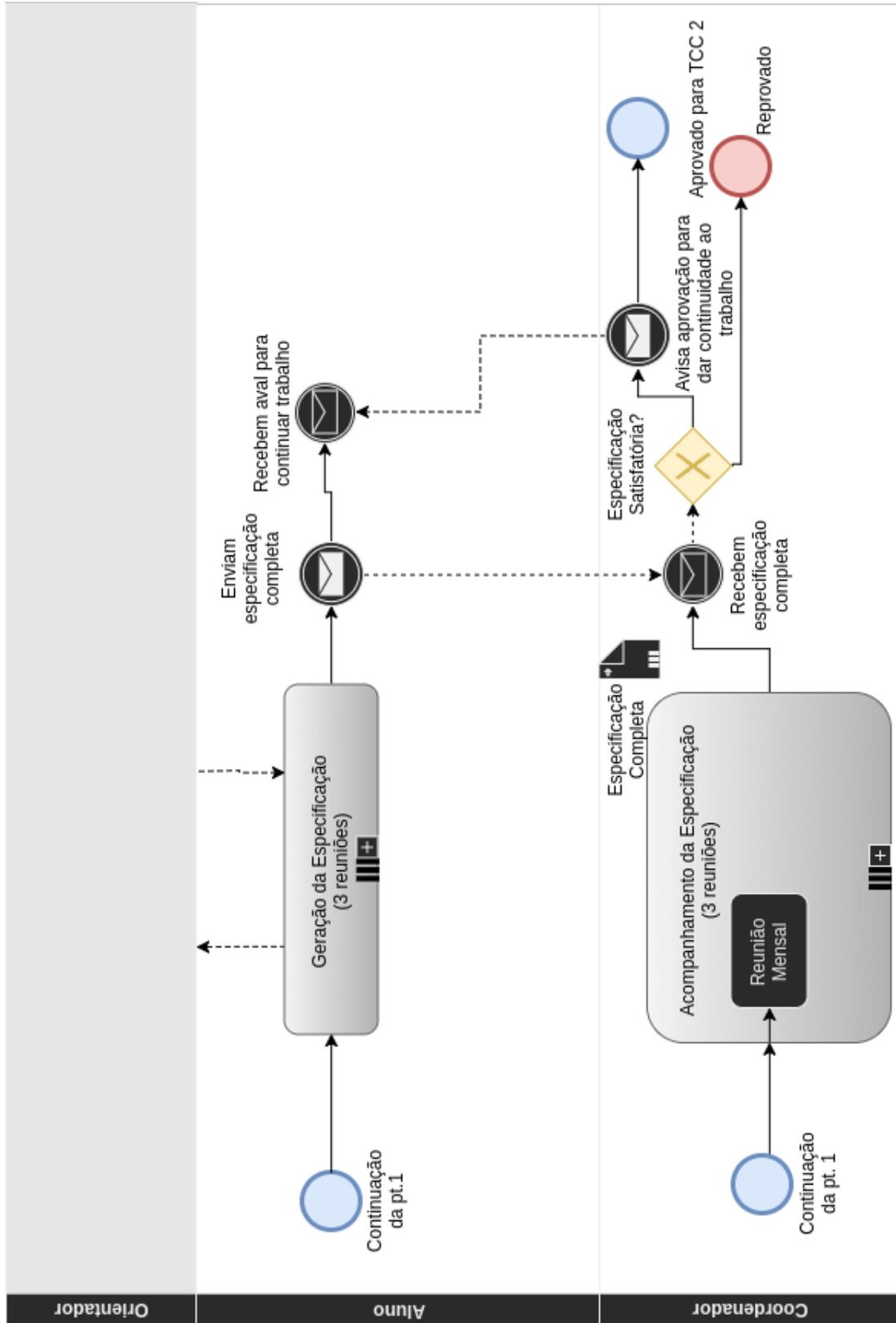


Figura 21: Diagrama BPMN para a disciplina de TCC 1 - Parte 2

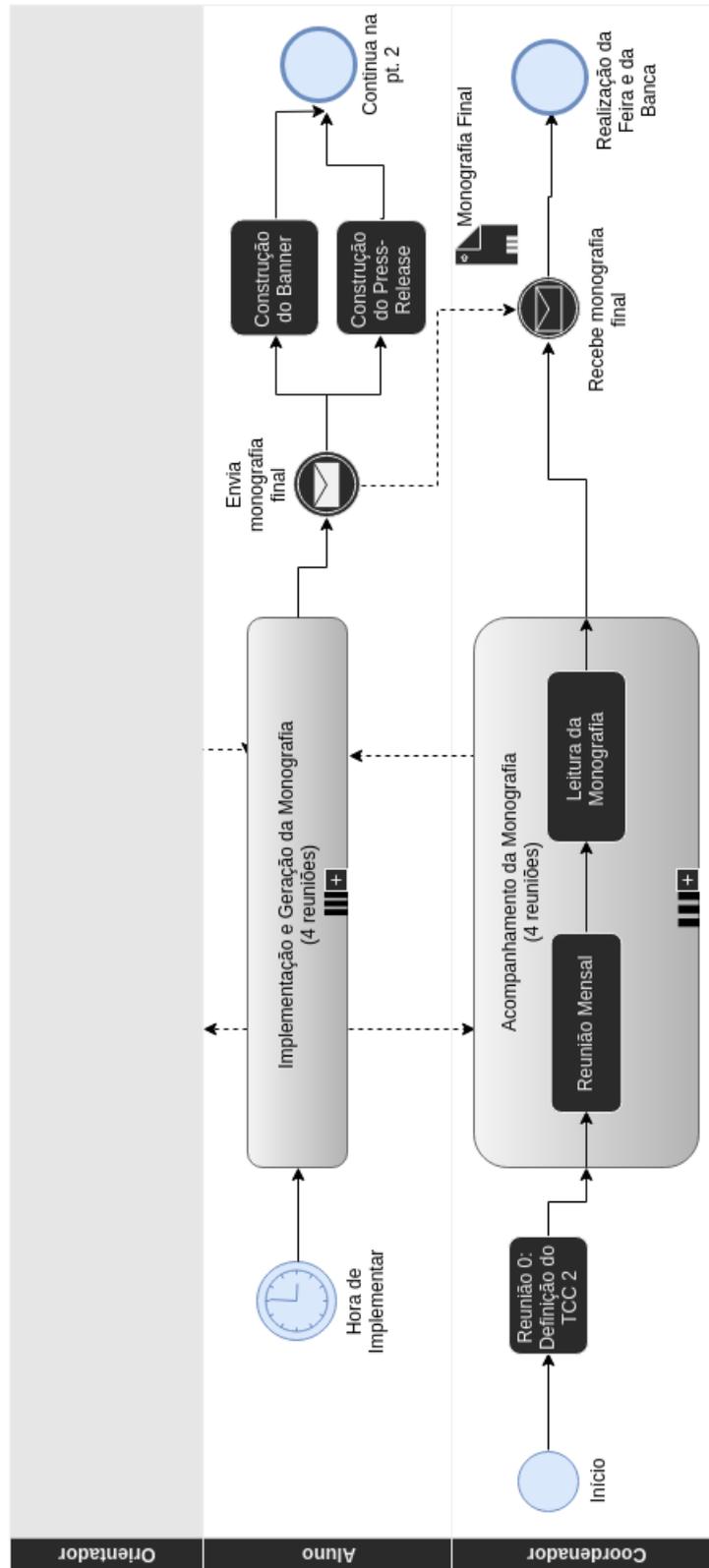


Figura 22: Diagrama BPMN para a disciplina de TCC 2 - Parte 1

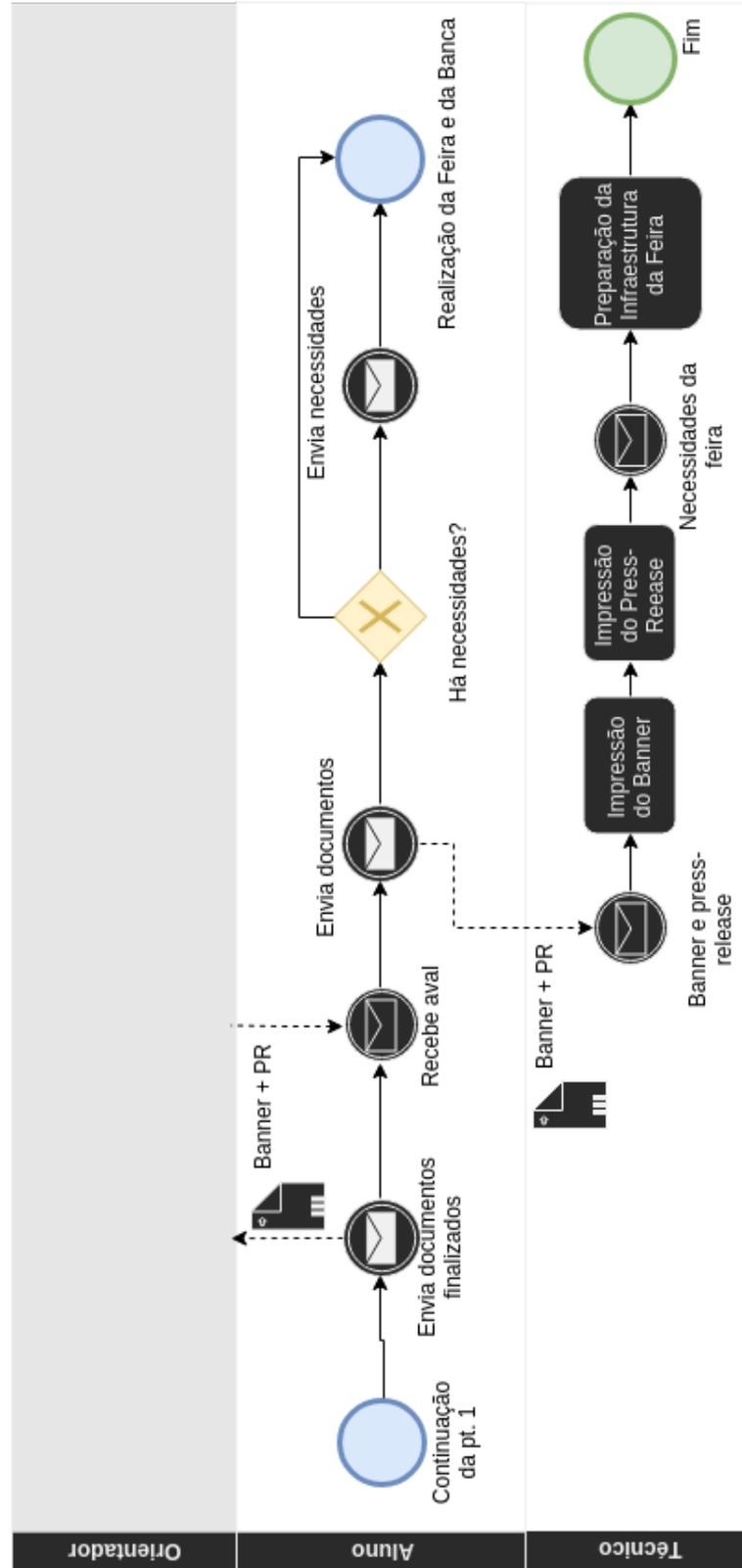


Figura 23: Diagrama BPMN para a disciplina de TCC 2 - Parte 2



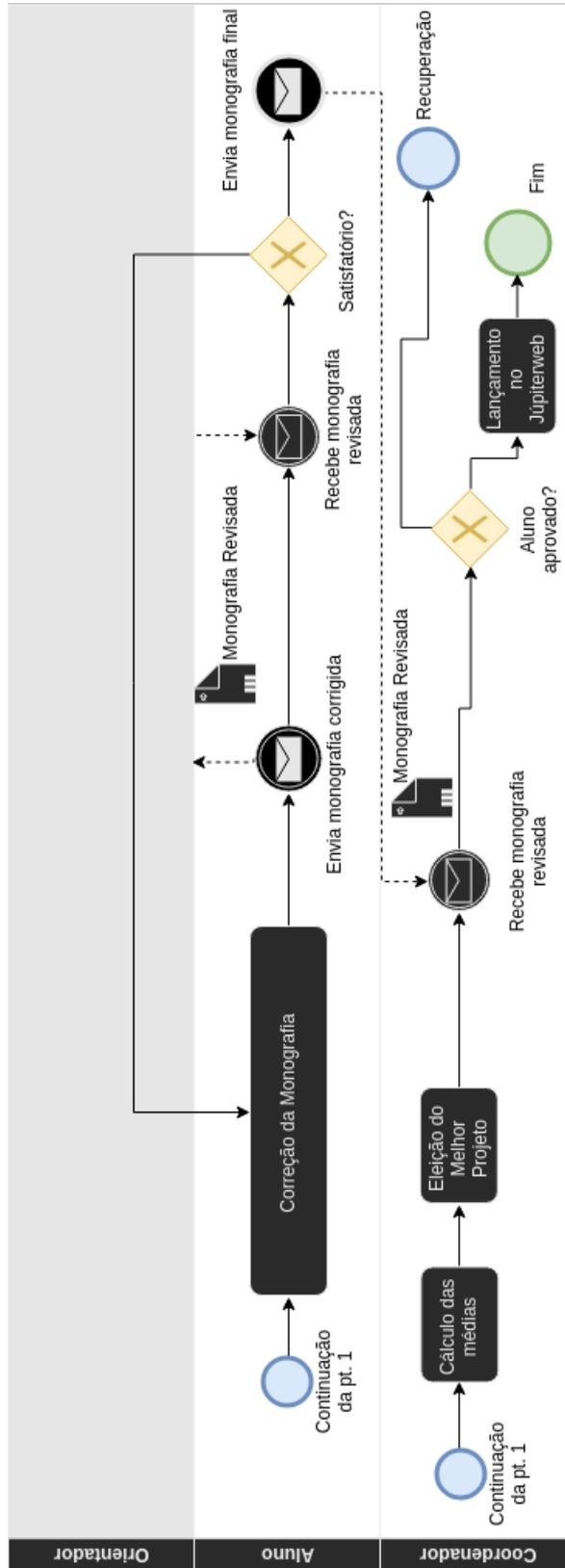


Figura 25: Diagrama BPMN para a Banca e Feira - Parte 2

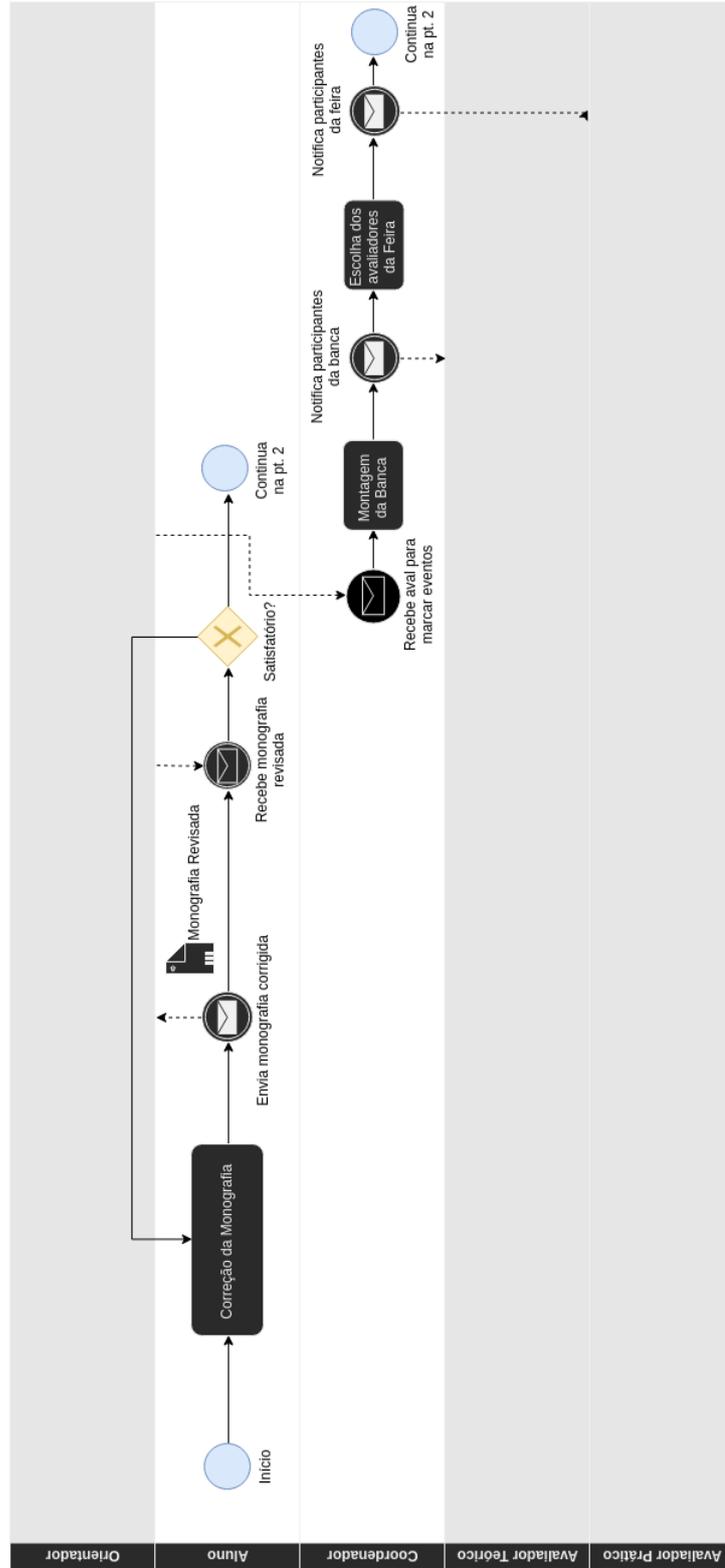


Figura 26: Diagrama BPMN para a Recuperação - Parte 1

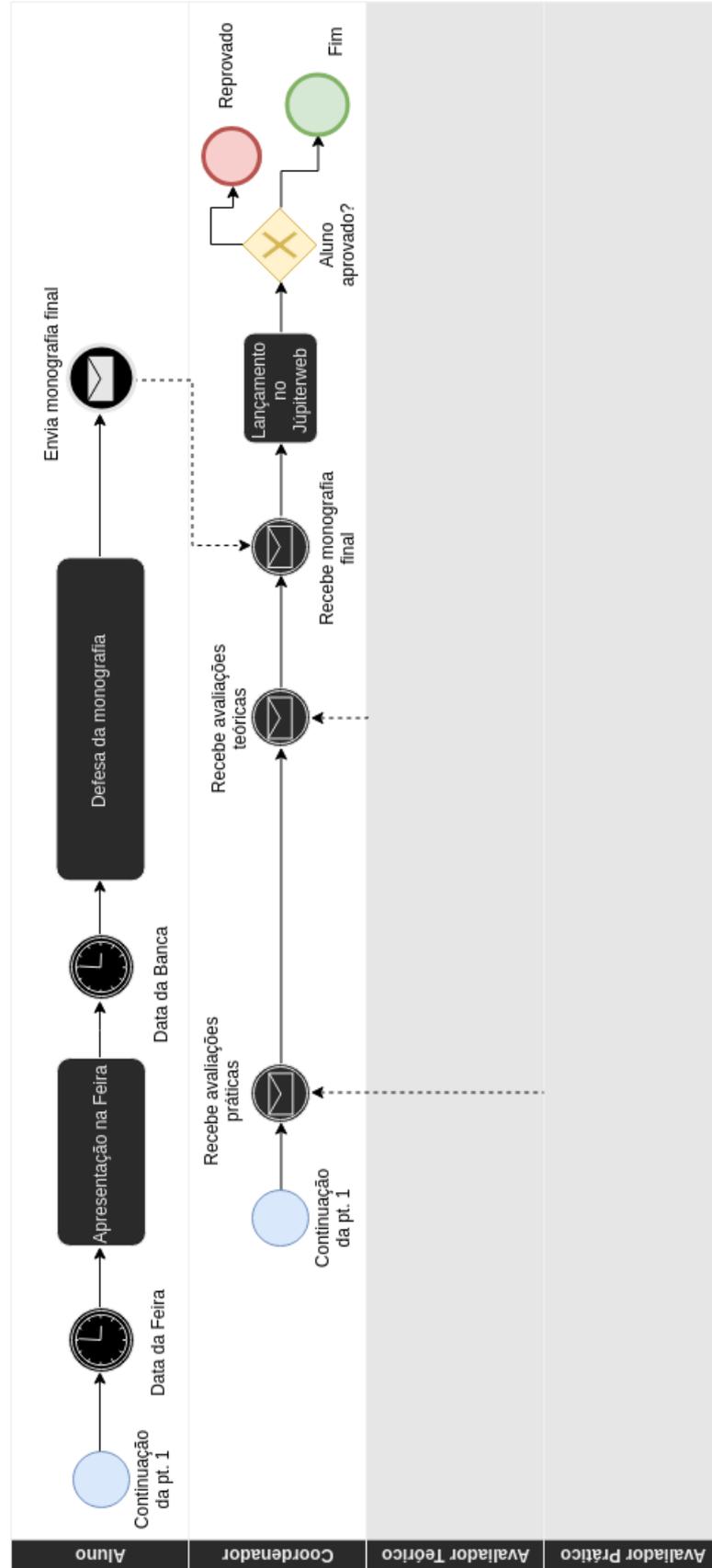


Figura 27: Diagrama BPMN para a Recuperação - Parte 2

## APÊNDICE C – CASOS DE USO

Neste apêndice consta os casos de uso escritos para o sistema em questão, usando o padrão explicado no capítulo de casos de uso (IBM, 2011). Para economizar espaços, campos ausentes nos casos de uso não foram explicitados.

### C.1 Cadastrar disciplinas

1. Breve Descrição: Coordenadores cadastram a disciplina, os alunos participantes e as atividades.
2. Fluxo Básico:
  - Coordenador insere disciplina, com os seguintes dados:
    - Modalidade (Sem/Quad)
    - Data de início e data de término
  - Coordenador importa planilha com alunos participantes, com os seguintes dados dos alunos
    - Nome
    - E-mail
    - N° USP
  - Coordenador insere uma nova atividade da disciplina, com os seguintes dados
    - Nome
    - Data de entrega e arquivos relacionados
    - Peso da atividade
  - Sistema salva alunos novos, dispara e-mail ao aluno com seu acesso (login e senha) e vincula os existentes à disciplina e salva a disciplina
3. Fluxos Alternativos:

- (a) Data de início é posterior a de término
    - i. Sistema exibe novamente tela de cadastro da disciplina, alertando sobre erro
  - (b) E-mail é inválido
    - i. Sistema exibe novamente tela de cadastro da disciplina, alertando sobre erro
  - (c) E-mail retornou
    - i. Sistema envia e-mail ao coordenador com o aluno problemático
4. Pré-condição: Coordenador deve estar logado

## C.2 Editar disciplinas

1. Breve Descrição: Coordenadores editam disciplinas, cadastrando atividades, editando alunos etc.
2. Fluxo Básico:
  - Coordenador edita início e término da disciplina, alunos participantes, com novos alunos ou removendo os já participantes
  - Coordenador insere uma nova atividade da disciplina, com os seguintes dados
    - Nome
    - Data de entrega
    - Arquivos relacionados
    - Peso da atividade
  - Sistema salva novas atividades e as mudanças da disciplina
3. Pré-condição: Coordenador deve estar logado

## C.3 Cadastrar professores

1. Breve Descrição: Coordenadores cadastram professores do departamento que podem orientar/co-orientar.
2. Fluxo Básico:

- Coordenador insere os dados do professor
    - Nome
    - Número USP
    - E-mail
  - Sistema salva o professor, disparando e-mail para o professor cadastrado
3. Pré-condição: Coordenador deve estar logado

## C.4 Cadastrar convidados externos

1. Breve Descrição: Coordenadores cadastram convidados externos do departamento que podem co-orientar.
2. Fluxo Básico:
  - Coordenador insere os dados do convidado externo
    - Nome
    - Empresa
    - E-mail
  - Sistema salva o convidado externo, disparando e-mail para o convidado cadastrado
3. Pré-condição: Coordenador deve estar logado

## C.5 Cadastrar grupos de trabalhos

1. Breve Descrição: Coordenadores cadastram os grupos com os temas e os orientadores, com a confirmação da participação do orientador no grupo.
2. Fluxo Básico:
  - Coordenador insere os dados do grupo
    - Título
    - Alunos
    - Orientador
    - Co-orientador

- Sistema salva grupo e envia e-mail para o orientador, co-orientador e alunos
- Orientador valida participação no grupo
- Co-orientador valida participação no grupo

3. Fluxos Alternativos:

(a) Grupo não tem orientador

- i. Sistema cadastra grupo, enviando e-mail para os alunos com o aviso de urgência na escolha do orientador

(b) Grupo não tem aluno

- i. Sistema retorna para a tela de cadastro de grupo, avisando sobre o erro de ausência de alunos

4. Pré-condição: Coordenador deve estar logado, alunos, orientadores e co-orientadores cadastrados

5. Pós-condição: Grupo confirmado

## C.6 Login

1. Breve Descrição: Alunos, Orientadores, Co-orientadores e Coordenadores acessam sistema de maneira tradicional ou via Senha Única USP (para pertencentes à USP).

2. Fluxo Básico:

- Sistema mostra campos de login e senha
- Usuário insere seu e-mail e sua senha
- Sistema valida e-mail e senha
- Usuário acessa sistema

3. Fluxos Alternativos:

(a) Usuário erra credenciais

- i. Sistema retorna para tela de acesso ao sistema, exibindo mensagem de erro
- ii. Retorna normalmente à situação de mostrar campos de login e senha

(b) Usuário realiza login pela Senha Única da USP

- i. Usuário seleciona “Acessar pela Senha Única USP”

- ii. Usuário é redirecionado para os sistemas USP
  - iii. Retorna para a situação de acesso ao sistema
4. Requisitos Especiais: Integração via Shibboleth com os Sistemas USP
  5. Pós-condição: Usuário dentro do sistema

## C.7 Entregar atividade

1. Breve Descrição: Alunos submetem no Google Drive arquivos da atividade para a leitura do orientador, co-orientador e coordenadores. Orientador e co-orientador revisam e dão seu aval de aprovação com a documentação.
2. Fluxo Básico:
  - Aluno submete arquivos nos respectivos espaços de atividades, que são carregados no Google Drive e deixa comentários adicionais sobre a entrega
  - Sistema salva a entrega com o status da entrega da atividade para “Não avaliada”
  - Sistema envia e-mail para Orientador e Co-orientador avisando de submissão
  - Orientador baixa documentos submetidos
  - Orientador avalia a entrega, faz comentários aos alunos e comentários exclusivos à coordenação com notas
  - Sistema salva entrega e dispara e-mail com o resultado da avaliação para os alunos
3. Fluxos Alternativos:
  - (a) Data de submissão expirou (1)
    - i. Aluno não consegue interagir com atividade, encerrando fluxo
  - (b) Co-orientador realiza fluxo de revisão, antes do Orientador (4)
    - i. Passos 5 - 7 ocorrem normalmente, trocando Orientador por Co-orientador
  - (c) Co-orientador realiza fluxo de revisão, após Orientador (8)
    - i. Sistema exibe detalhes da entrega, porém não permite edições do lado do Co-orientador, encerrando fluxo
  - (d) Aluno submete nova entrega da atividade após ter feito uma submissão (1)

- i. Aluno vê status da avaliação
  - ii. Aluno realiza nova entrega da atividade, repetindo o caso de uso
- (e) Aluno submete entrega da atividade quando alguém do grupo já submeteu (1)
  - i. Sistema exibe detalhes da entrega já realizada
  - ii. Aluno pode realizar nova entrega da atividade, passando por cima da entrega anterior e repetindo o caso de uso
- 4. Pré-condição: Atores devem estar autenticados e atividade deve estar cadastrada no sistema

## C.8 Construir bancas práticas

1. Breve Descrição: Coordenadores selecionam os participantes da banca prática, já cadastrados no sistema, e os notifica com comentários sobre o evento.
2. Fluxo Básico:
  - Coordenador informa o dia da feira, a disciplina correspondente e as salas disponíveis para o evento, além de determinar o peso da avaliação da banca
  - Sistema salva evento
  - Coordenador seleciona evento recém criado.
  - Sistema lista grupos do evento.
  - Coordenador seleciona o grupo que deseja atribuir a sala e os convidados.
  - Coordenador seleciona os convidados que avaliarão o grupo
  - Sistema salva banca e envia e-mail para os convidados externos, avisando-os sobre sua participação
  - Convidado acessa sistema e confirma participação na banca
3. Pré-condição: Atores devem estar autenticados e grupo de trabalho deve estar cadastrado no sistema

## C.9 Construir bancas teóricas

1. Breve Descrição: Coordenadores escolhem participantes da banca teórica do grupo, selecionam o presidente da banca, realizam o agendamento do horário, validando

inconsistências (participante já possui horário ocupado) e notificam os participantes por e-mail.

## 2. Fluxo Básico:

- Coordenador informa o dia da banca, a disciplina correspondente e as salas disponíveis para o evento
- Sistema salva evento
- Coordenador seleciona evento recém criado.
- Sistema lista grupos do evento.
- Coordenador seleciona o grupo que deseja atribuir a sala e os convidados.
- Coordenador seleciona os avaliadores da banca e o horário da avaliação. O orientador é um dos pré-selecionados por padrão

## 3. Fluxos Alternativos:

- (a) Convidado externo já possui banca nesse dia e horário
  - i. Sistema barra criação de banca, alertando sobre qual convidado já possui agenda ocupada
- (b) Sala está ocupada no horário selecionado
  - i. Sistema barra criação de banca, alertando sobre qual sala já possui agenda ocupada

## 4. Pré-condição: Atores devem estar autenticados e grupo de trabalho deve estar cadastrado no sistema

## C.10 Listar entregas

### 1. Breve Descrição: Técnicos recebem os arquivos de impressão, com normalização do título, separados por grupo.

### 2. Fluxo Básico:

- Coordenador lista todas as entregas finais de impressão (*banner e press-release*)
- Sistema salva as listas de arquivos finais, com o nome normalizado e envia por e-mail para o técnico responsável

### 3. Fluxos Alternativos:

- (a) Grupo de trabalho está com arquivo faltante
  - i. Sistema envia e-mail para o grupo com entrega faltante, avisando para regularizar a situação com urgência.
- 4. Pré-condição: Atores devem estar autenticados e grupo de trabalho deve estar cadastrado no sistema

## C.11 Listar necessidades adicionais

1. Breve Descrição: Técnicos recebem necessidades adicionais revisadas pelos orientadores, separadas por grupo.
2. Fluxo Básico:
  - Coordenador lista todos os comentários das entregas finais
  - Sistema salva a lista de comentários e a envia por e-mail para o técnico responsável
3. Pré-condição: Atores devem estar autenticados e grupo de trabalho deve estar cadastrado no sistema

## C.12 Avaliar projetos práticos

1. Breve Descrição: Participantes da banca prática avaliam os projetos que estão envolvidos, limitando avaliações até o final do dia.
2. Fluxo Básico:
  - Convidado externo acessa espaço da banca, com detalhes do grupo e as entregas finais
  - Convidado preenche os comentários e as notas de acordo com cada critério estabelecido para avaliação de bancas práticas
  - Convidado salva avaliação
3. Fluxos Alternativos:
  - (a) Convidado tenta submeter avaliação em dia diferente ao da banca prática
    - i. Avaliação é barrada, avisando o convidado de que a avaliação só pode ser feita exclusivamente no dia

4. Pré-condição: Atores devem estar autenticados e grupo de trabalho deve estar cadastrado no sistema

### C.13 Avaliar monografias teóricas

1. Breve Descrição: Participantes da banca teórica avaliam as monografias que estão envolvidas, gerando comentários e definindo o status do trabalho (aprovado, aprovado com correções, recuperação e reprovado), limitando avaliações até o final do dia.
2. Fluxo Básico:
  - Participante da banca acessa espaço da banca, com detalhes do grupo e as entregas parciais e finais.
  - Participante preenche os comentários e as notas de acordo com cada critério estabelecido para avaliação de bancas teóricas.
  - Participante salva avaliação, com seu parecer para aprovação da monografia.
  - Sistema usa a média das avaliações da banca para determinar o estado do grupo.
3. Pré-condição: Atores devem estar autenticados e grupo de trabalho deve estar cadastrado no sistema

### C.14 Calcular nota final dos projetos

1. Breve Descrição: Coordenadores determinam a fórmula para calcular as notas finais, com base nas entregas parciais durante as duas disciplinas e o sistema calcula as notas finais de todos os grupos participantes.
2. Fluxo Básico:
  - Coordenador escolhe a disciplina de TCC2, a banca teórica e a feira prática que deseja obter as entregas
  - Sistema salva fórmula de avaliação
  - Sistema lista todos os grupos, com as notas calculadas e os estados de avaliação de cada grupo

### 3. Fluxos Alternativos:

- (a) Coordenador não preenche algum dos campos necessários
  - i. Sistema barra criação de fórmula de avaliação, avisando os campos faltantes
- (b) Grupo tem alguma avaliação faltante
  - i. Sistema exhibe grupo, mas com campo de Não Avaliado
- (c) Grupo vai para recuperação na avaliação da banca teórica
  - i. A nota calculada é a nota da banca teórica, passando todas as outras avaliações realizadas ao longo das disciplinas
- (d) Grupo é reprovado na avaliação da banca teórica
  - i. A nota calculada é a nota da banca teórica, passando todas as outras avaliações realizadas ao longo das disciplinas

### 4. Pré-condição: Atores devem estar autenticados e grupo de trabalho deve estar cadastrado no sistema