

UNIVERSIDADE FEDERAL DO PARANÁ

ANGEL MAURICIO SANCHES PEREIRA

**SOCIAL: EVENTOS ONLINE**

CURITIBA

2019

ANGEL MAURICIO SANCHES PEREIRA

**SOCIAL: EVENTOS ONLINE**

Monografia apresentada como requisito parcial à obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas, no Curso de Graduação em Tecnologia em Análise e Desenvolvimento de Sistemas, SEPT, da Universidade Federal do Paraná.

Orientador: Prof. Dr. Razer Anthom Nizer Rojas  
Montaño

CURITIBA

2019

## TERMO DE APROVAÇÃO

Angel Mauricio Sanches Pereira

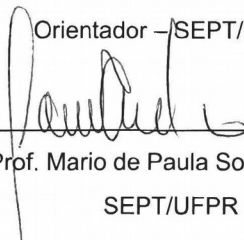
### SOCIAL: EVENTOS ONLINE

Monografia aprovada como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, do Setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná.



Prof. Razer Anthom Nizer Rojas Montañó

Orientador – SEPT/UFPR



Prof. Mario de Paula Soares Filho

SEPT/UFPR

Curitiba, 25 de Junho de 2019.

## RESUMO

O número de ferramentas para diversas ações cotidianas vem crescendo com a era da agilidade, a era onde tudo pode ser feito do seu bolso ou na palma da mão. Muitas dessas ferramentas, apesar de apresentarem um número considerável de concorrentes e até de usos, ainda estão muito imaturas e precisam de novas estratégias para realmente crescerem, antes que outra ferramenta a faça. Eis aqui uma tentativa de explorar as fraquezas das ferramentas atuais para procura e divulgação de eventos e atividades. Serão avaliadas as ferramentas concorrentes dessa área e os pontos fracos que elas apresentam e será desenvolvido um sitio em cima disso, visando fazer as mesmas funções desses sítios, porém focando em seus pontos fracos, tentando expandir o aplicativo para onde essas ferramentas ainda não o fizeram. Para representar as interações entre pessoas que procuram eventos, pessoas que organizam eventos, empresas e os próprios eventos, o melhor nome encontrado para a aplicação foi “Social”.

**Palavras-chave:** Turismo de Eventos, Rede Social, Eventos, Atividades

## **ABSTRACT**

The quantity of tools for various daily actions grow up in the agility age, the age where everything can be done inside your pocket or in your own hands. Many of those tools, despite have a good number of competing applications and until of uses, are a lot immature yet and need new strategies to really grow up, before a competing application do that. This is an attempt to explore the weaknesses of the actual tools to search and divulgation of events and activities. There will be evaluated the competing tools of this area e the weaknesses that they show and there will be a site from this, looking do the same functions of those sites, beside focusing in their weaknesses, trying to expand the application to where those tools don't reached. To represent these actions between people searching for events, people that organize events, companies and the events, the best name we thought for the application was "Social".

**Key-words:** Event Tourism, Social Media, Events, Activities

## LISTA DE FIGURAS

FIGURA 1 - Formulário para cadastro de eventos no <i>Facebook</i> .....	18
FIGURA 2 - Exemplo de evento falso no Facebook.....	20
FIGURA 3 - Gráfico Redes Sociais que os Usuários Pretendem Sair.....	21
FIGURA 4 - Categorias do <i>DiskIngressos</i> .....	24
FIGURA 5 - Servidores Mais Usados no Mercado (Gráfico).....	30
FIGURA 6 - Servidores Mais Usados no Mercado (Tabela).....	30
FIGURA 7 - <i>IntelliSense</i> do <i>Visual Studio</i> .....	32
FIGURA 8 - Exemplo de Sprint.....	37
FIGURA 9 - WBS.....	38
FIGURA 10 - Tela Inicial Quando não Logado.....	59
FIGURA 11 - Tela de Cadastro.....	60
FIGURA 12 - Tela de Login.....	60
FIGURA 13 - Tela de Contato.....	61
FIGURA 14 - Tela Inicial Quando Logado.....	62
FIGURA 15 - Tela de Perfil.....	63
FIGURA 16 - Publicação.....	63
FIGURA 17 - Agenda.....	64
FIGURA 18 - Formulário de Dados da Conta.....	64
FIGURA 19 - Lista de Usuários que Está Seguindo/Lista de Seguidores.....	65
FIGURA 20 - Lista de Certificados.....	65
FIGURA 21 - Menu.....	66
FIGURA 22 - Lista de Empresas.....	66
FIGURA 23 - Formulário de Empresas.....	67
FIGURA 24 - Perfil da Empresa.....	67
FIGURA 25 - Formulário de Eventos - Dados.....	68
FIGURA 26 - Formulário de Eventos - Imagens.....	68
FIGURA 27 - Formulário de Eventos – Perfil do Público-Alvo.....	69
FIGURA 28 - Formulário de Eventos – Previsão do Alcance do Evento.....	69
FIGURA 29 - Perfil de Evento – Imagens e Informações.....	70
FIGURA 30 - Perfil de Evento – Descrição e Publicações.....	71
FIGURA 31 - Perfil de Eventos - Menções.....	72

FIGURA 32 - Perfil de Evento - Comentar.....	72
FIGURA 33 - Perfil de Evento – Outros Comentários.....	73
FIGURA 34 - Denúncia.....	73
FIGURA 35 - Login.....	74
FIGURA 36 - Menu.....	75
FIGURA 37 - <i>Dashboard</i> Administrador.....	76
FIGURA 38 - Lista de Usuários.....	77
FIGURA 39 - Formulário Usuário.....	78
FIGURA 40 - Imagem de Perfil.....	78
FIGURA 41 - Imagem de Capa.....	79
FIGURA 42 - Estatísticas do Usuário.....	79
FIGURA 43 - Logs do Usuário.....	80
FIGURA 44 - Lista de Empresas.....	81
FIGURA 45 - Lista de Eventos.....	81
FIGURA 46 - Lista de Categorias.....	82
FIGURA 47 - Lista de Administradores.....	82
FIGURA 48 - Logs dos Administradores.....	83
FIGURA 49 - Lista de Denúncias.....	83
FIGURA 50 - Detalhes da Denúncia.....	84

## LISTA DE QUADROS

QUADRO 1 – CRONOGRAMA.....	40
----------------------------	----



## LISTA DE GRÁFICOS

GRÁFICO 1 – GANTT.....	42
------------------------	----

## **LISTA DE TABELAS**

TABELA 1 – REDES SOCIAIS MAIS USADAS NO BRASIL (2018).....	14
--	----

## LISTA DE ABREVIATURAS E SIGLAS

ABEOC	- Associação Brasileira de Empresas e Eventos
AJAX	- <i>Asynchronous JavaScript and XML</i> (XML e JavaScript Assíncronos)
API	- <i>Application Programming Interface</i> (Interface de Programação para Aplicações)
BLOB	- <i>Binary Large Object</i> (Grande Objeto Binário)
CEP	- Código de Endereçamento Postal
CGI	- <i>Common Gateway Interface</i> (Interface Comum de Portões)
CRUD	- <i>Create, Read, Update and Delete</i> (Criar, Ler, Atualizar e Deletar)
CSS	- <i>Cascading Style Sheets</i> (Folhas de Estilo em Cascada)
cURL	- <i>Connect URL</i> (Conectar URL)
DER	- Diagrama Entidade e Relacionamento
DOM	- <i>Document Object Model</i> (Modelo de Objeto do Documento)
GD	- <i>Graphical Drawing</i> (Desenho Gráfico)
HTML	- <i>HyperText Markup Language</i> (Linguagem de Marcação de Hipertexto)
JS	- <i>JavaScript</i>
MER	- Modelo Entidade e Relacionamento
MVC	- <i>Model View Controller</i> (Modelo Visualização Controlador)
PDO	- <i>PHP Data Objects</i> (Objetos de Informação do <i>PHP</i> )
PHP	- <i>Hypertext Processor</i> (Processador de Hipertexto)
PNG	- <i>Portable Network Graphics</i> (Rede de Gráficos Portátil)
QR code	- <i>Quick Response Code</i> (Código de Resposta Rápida)
SGML	- <i>Standard Generalized Markup Language</i> (Linguagem de Marcação de Padrão Generalizado)
SMTP	- <i>Simple Mail Transfer Protocol</i> (Protocolo Simples de Transferência de Emails)
SQL	- <i>Structured Query Language</i> (Linguagem Estruturada de Busca)
SSL	- <i>Secure Sockets Layer</i> (Camada de Soquete Seguro)
TCC	- Trabalho de Conclusão de Curso
UC	- <i>Use Case</i> (Caso de Uso)
UML	- <i>Unified Modeling Language</i> (Linguagem de Modelagem Unificada)
URL	- <i>Uniform Resource Locator</i> (Localizador Uniforme de Recurso)
WBS	- <i>Work Breakdown Structure</i> (Estrutura Analítica do Trabalho)

# Sumário

<b>1 INTRODUÇÃO.....</b>	<b>13</b>
1.1 JUSTIFICATIVA.....	13
1.2 OBJETIVOS.....	14
1.2.1 Objetivo Geral.....	15
1.2.2 Objetivos Específicos.....	15
1.3 METODOLOGIA.....	15
<b>2 REVISÃO DE LITERATURA.....</b>	<b>16</b>
2.1 CRIAÇÃO DE EVENTOS.....	16
2.1.1 Marketing.....	16
2.1.2 Público Alvo.....	17
2.1.3 Pós-Evento.....	17
2.2 FERRAMENTAS CONCORRENTES.....	18
2.2.1 Facebook.....	18
2.2.2 Eventbrite.....	21
2.2.3 Sympla.....	22
2.2.4 Doity.....	22
2.2.5 Disk Ingressos.....	23
2.2.6 Outgo.....	24
2.2.7 Resumo da Concorrência.....	25
2.3 TECNOLOGIAS DE DESENVOLVIMENTO.....	25
2.3.1 Front-End.....	25
2.3.1.1 HTML.....	26
2.3.1.2 CSS.....	26
2.3.1.3 JavaScript.....	27
2.3.1.4 Material Design.....	27
2.3.1.5 Materializecss.....	28
2.3.1.6 jQuery.....	28
2.3.2 Back-End.....	29
2.3.2.1 Apache.....	29
2.3.2.2 MySQL.....	30
2.3.2.3 PHP.....	31

2.3.3 Documentação.....	31
2.3.4 Produção.....	32
2.3.4.1 Visual Studio.....	32
2.3.4.2 MySQL Workbench.....	33
2.3.4.3 LibreOffice.....	33
2.3.4.4 Xmind.....	33
2.3.4.5 Astah UML.....	34
2.3.4.6 UML.....	34
2.4 METODOLOGIA DE DESENVOLVIMENTO.....	35
<b>3 MATERIAL E MÉTODOS.....</b>	<b>37</b>
3.1 IMPLEMENTAÇÃO DO SCRUM NO DESENVOLVIMENTO DO PROJETO.....	37
3.2 DESCRIÇÃO DAS SPRINTS.....	43
3.2.1 Pré-Sprints.....	43
3.2.2 Sprint 1.....	43
3.2.3 Sprint 2.....	44
3.2.4 Sprint 3.....	44
3.2.5 Sprint 4.....	44
3.2.6 Sprint 5.....	44
3.2.7 Sprint 6.....	45
3.2.8 Sprint 7.....	45
3.2.9 Sprint 8.....	45
3.2.10 Sprint 9.....	46
3.2.11 Sprint 10.....	46
3.2.12 Sprint 11.....	46
3.2.13 Sprint 12.....	46
3.2.14 Sprint 13.....	47
3.2.15 Sprint 14.....	48
3.2.16 Sprint 15.....	48
3.2.17 Sprint 16.....	48
3.2.18 Sprint 17.....	49
3.2.19 Sprint 18.....	50
3.2.20 Sprint 19.....	51
3.3 DESENVOLVIMENTO DA APLICAÇÃO.....	51
3.3.1 Apache.....	51

3.3.2 MySQL.....	51
3.3.3 PHP.....	52
3.3.4 Front-End.....	54
<b>4 APRESENTAÇÃO DOS RESULTADOS.....</b>	<b>57</b>
4.1 FRAMEWORK.....	57
4.2 USUÁRIO NÃO REGISTRADO OU LOGADO.....	58
4.3 USUÁRIOS REGISTRADOS E LOGADOS.....	61
4.4 ADMINISTRADOR.....	73
<b>5 CONSIDERAÇÕES FINAIS.....</b>	<b>85</b>
5.1 RECOMENDAÇÕES PARA TRABALHOS FUTUROS.....	85
<b>REFERÊNCIAS.....</b>	<b>88</b>
<b>APÊNDICE A – DIAGRAMA DE CASOS DE USO DO SÍTIO.....</b>	<b>91</b>
<b>APÊNDICE B – DIAGRAMA DE CASOS DE USOS DO ADMINISTRADOR.....</b>	<b>109</b>
<b>APÊNDICE C – DIAGRAMA DE CLASSES.....</b>	<b>124</b>
<b>APÊNDICE D – DIAGRAMAS DE SEQUÊNCIA.....</b>	<b>125</b>
<b>APÊNDICE F – MODELO LÓGICO.....</b>	<b>136</b>
<b>APÊNDICE G – EXEMPLO DE MODELO E CONTROLADOR.....</b>	<b>137</b>

## 1 INTRODUÇÃO

Em 2016, o mercado de eventos no Brasil registrou um faturamento de R\$ 17.000.000 (dezesete milhões de reais), com expectativas para que o mercado cresça (ABRAFESTA, 2017).

Os eventos adquiriram com o passar dos tempo características econômicas, históricas, sociais e políticas das sociedades representativas de cada época. No Brasil, o primeiro evento que ocorreu com planejamento foi em 1840, um Baile de Carnaval. Porém, a atividade de eventos planejados só se popularizou por aqui depois da década de 50 devido ao desenvolvimento industrial no país e a organização das classes profissionais (MARLENE MATIAS, 2013).

Em 2011, o turismo de eventos rendeu ao mundo cerca de 1 trilhão de dólares. Entre 2002 e 2010, mais de um milhão de turistas por ano vieram para o Brasil em busca de participar de eventos. Por ano, são realizados mais de 300.000 (trezentos mil) eventos em todo o território nacional brasileiro (MARLENE MATIAS, 2013).

Em todo o mundo, em 2011, estimasse que o turismo de eventos rendeu cerca de 1 trilhão de dólares. O continente que mais rendeu com isso foi a Europa, com cerca de 460 bilhões de dólares em 2011. Só as Américas juntas renderam quase 200 bilhões no mesmo ano (MARLENE MATIAS, 2013).

Conclui-se que o mercado de eventos no Brasil possui muita força, porém não existe digitalmente uma ferramenta que cativa os usuários a procurarem por eventos e atividades próximos.

### 1.1 JUSTIFICATIVA

Segundo relatório *Social Media Trends* (2019), a procura por atividades de entretenimento (sejam festas, eventos, filmes, etc) ocupa o 9º lugar dos assuntos mais procurados em redes sociais no Brasil. Isso ocupa cerca de 5,8% de todo o interesse que os usuários de rede social possuem ao visitar uma rede social.

O turismo de eventos é uma área da economia de extrema importância, representando 152 bilhões de reais do PIB brasileiro em 2017 segundo o Conselho Mundial de Turismo e Viagem.

Na TABELA 1 é possível verificar as redes sociais mais comuns no Brasil em 2018.

TABELA 1 – REDES SOCIAIS MAIS USADAS NO BRASIL (2018)

Posição	Nome	Usuários*	Usuários brasileiros*	Possui Eventos?
1	Facebook	2000	130	Sim
2	Whatsapp	1200	120	Não
3	YouTube	-	98	Não
4	Instagram	-	57	Não
5	Twitter	330	30	Não
6	LinkedIn	200	29	Não
7	Pinterest	200	-	Não
8	Google+	-	-	Não
9	Messenger	1200	-	Não
10	Snapchat	187	-	Não

FONTE: Rocket Content (2018)

\* em milhões

Entre todas as redes sociais mais usadas no Brasil, unicamente o *Facebook* apresenta a função de divulgar e pesquisar por eventos. Porém o *Facebook* possui algumas desvantagens, que são mais exploradas no capítulo 2.2.1. Conclui-se que ainda que seja uma ferramenta poderosa e de grande uso, ainda é possível competir com ela na área de eventos ao desenvolver o sistema adequado para tal.

De todas as ferramentas de procura e divulgação de eventos que foram analisadas (encontram-se no capítulo 2.2), percebeu-se que em todas elas (exceto *Facebook*) não existia uma interação entre as pessoas que procuram por eventos. E todas as empresas, ao se descreverem, mostravam fortemente uma preocupação com o divulgador do evento, porém sem se preocupar com as pessoas que procurariam por estes eventos.

O projeto aqui apresentado é um foco exclusivamente para a procura de eventos com ferramentas de auxílio à divulgação que outras ferramentas da área não apresentam, também criar um ambiente favorável para pessoas que tem o interesse e foco de procurar por eventos e atividades de interesse.

## 1.2 OBJETIVOS

Tem como objetivo descrever os principais objetivos desse trabalho. Dividido em objetivos gerais, que apresenta o problema aqui decorrido e suas delimitações, e em objetivos específicos que são uma descrição mais detalhada dos objetivos que nascem a partir do objetivo geral.



### 1.2.1 Objetivo Geral

Desenvolver um sítio de eventos voltado tanto para público quanto para organizadores que queiram agir em todo o Brasil, focando nas limitações de ferramentas concorrentes.

### 1.2.2 Objetivos Específicos

Os objetivos específicos do trabalho são:

- a) Criar uma plataforma de interação em que pessoas possam compartilhar seus interesses em eventos;
- b) Dar suporte para que os organizadores de evento tenham mais facilidade de encontrar seus públicos-alvo;
- c) Facilitar e aproximar pessoas de atividades culturais interessantes próximas a elas.

## 1.3 METODOLOGIA

Será desenvolvido um sítio visando aconchegar o público, aproximando-o de seus amigos e empresas favoritas. Organizando os eventos por maior relevância individual a cada usuário.

O sítio também visará auxiliar aos organizadores de eventos disponibilizando a eles ferramentas de auxílio na organização dos eventos que não são comumente encontradas nas outras plataformas concorrentes.

Será implementado também um espaço às pessoas que participam dos eventos em algum cargo - seja de palestrante, ajudante, organizador ou o que mais for - disponibilizando a eles certificados que podem ser validados através de um código QR.

## 2 REVISÃO DE LITERATURA

O objetivo deste capítulo é mostrar a pesquisa feita em cima da ideia inicial apresentada e formulada no capítulo anterior. Usar referências bibliográficas e pesquisas com público para apresentar o universo da criação de eventos, as preocupações atuais de quem procura e anuncia por eventos, como o Social poderá se encaixar nessas necessidades e comparações básicas com ferramentas concorrentes.

### 2.1 CRIAÇÃO DE EVENTOS

Segundo o livro “Evento: Oportunidade de Novos Negócios”, evento vem do termo “eventual” ou “casual”, seria um acontecimento que não está em nossa rotina e pretende reunir um grupo de pessoas.

Já, segundo Maria Cecília Giacaglia, o evento é um acontecimento que tem como “objetivo principal propiciar uma ocasião extraordinária ao encontro de pessoas, com finalidade específica, a qual constitui o ‘tema’ principal do evento e justifica sua realização”.

Marlene Matias, formada em turismo, define evento como “um acontecimento que... sempre envolve várias pessoas nas diversas fases de seu planejamento e organização, como também atrai um grande número de participantes”.

De todas as definições encontradas, observa-se que todas elas carregam em comum é: reunir diversas pessoas com algum objetivo.

O trabalho aqui apresentado tem o propósito de desenvolver um sítio operável para *web* que tenha a função principal de procurar e anunciar eventos, e que ao mesmo tempo cumpra com o propósito básico dos eventos que é “reunir pessoas”.

#### 2.1.1 Marketing

*Marketing* é um conjunto de ações que as empresas tomam visando aumentar seus lucros. Empresas planejam eventos para promover seus serviços e produtos, tanto o evento como a divulgação do mesmo fazem fundamentalmente parte do *marketing* das empresas (JOHNNY ALLEN, 2008, p. 90).

A ferramenta a ser desenvolvida deverá conter funções que ajudem o *marketing* de eventos, desde a hora do cadastro do evento, até a finalização do evento onde ele deverá ser avaliado.

### 2.1.2 Público Alvo

Definir o público-alvo do evento é um passo importante na produção do evento, a partir disso podem ser baseadas diversas outras escolhas como a data, horário, local e preços do evento. O público-alvo sempre deve refletir os objetivos do evento, sejam de venda, exposição ou promoção (MARIA CECÍLIA GIACAGLIA, 2003).

A plataforma deverá oferecer ferramentas que ajudem a mapear seu público-alvo, definindo informações para relevar o anúncio dos eventos ao seu público-alvo. Informações como localidade, idade, sexo e interesses deverão ser considerados para isso.

### 2.1.3 Pós-Evento

Após o evento acontecer, existe um conjunto de ações que normalmente são tomadas para comparar os resultados esperados com os resultados obtidos. Dentre essas ações, a avaliação é um ponto importante. Tanto a avaliação por parte dos funcionários como por parte dos participantes é importante. (MARLENE MATIAS, 2013)

“A avaliação de um evento é o processo de observação, mensuração e monitoração crítica da implementação de um evento a fim de avaliar seus resultados de forma precisa” (JOHNNY ALLEN, 2008).

A avaliação e comentários deverão ser ferramentas permitidas a todos os usuários do sítio que participaram de determinado evento, tanto empresa como evento poderão ser avaliados e a nota média deverá ser mostrada para todo mundo.

## 2.2 FERRAMENTAS CONCORRENTES

Através de pesquisas foram encontradas 6 ferramentas que são usadas para pesquisa e divulgação de eventos e atividades, ferramentas essas que podem ser vistas como concorrentes da plataforma social devido a suas semelhantes funções. Este capítulo foi feito para descrever as ferramentas concorrentes e compará-las com os objetivos e resultados do nosso sítio.

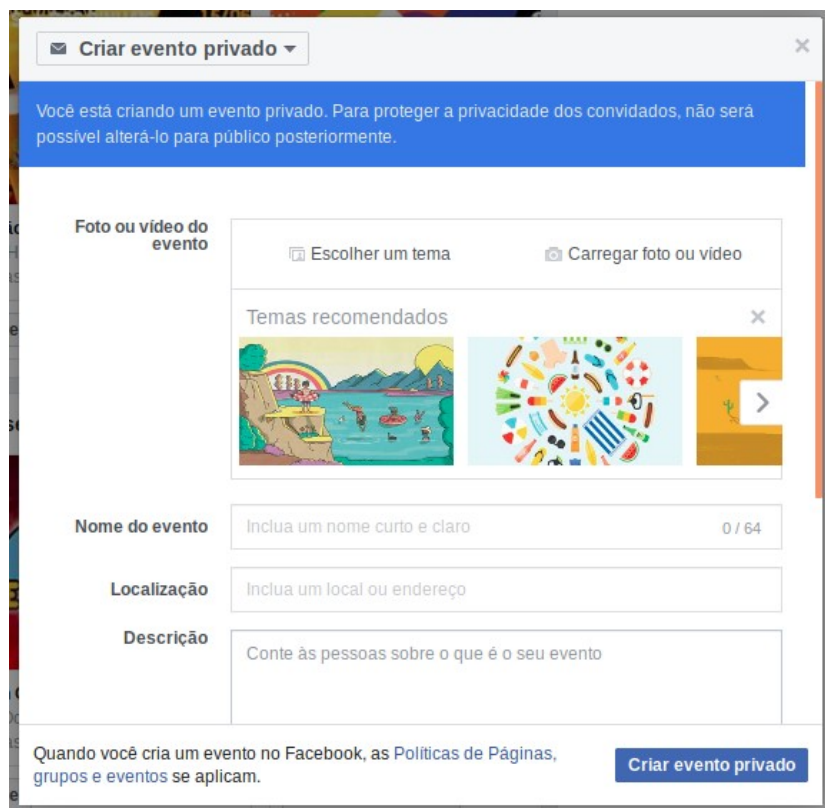
### 2.2.1 Facebook

O *Facebook* atua desde 2004, desenvolvido inicialmente por *Mark Zuckerberg*, a empresa se descreve em seu sítio com a missão de dar as pessoas o poder de construir comunidades e aproximar o mundo juntas.

O *Facebook* apresenta uma média de 1,56 bilhões de usuários ativos diariamente e 2,38 bilhões de usuários ativos mensalmente, de acordo com dados disponibilizados pela própria rede social, sobre o período de 31 de março de 2019 até 15 de junho de 2019.

Dentre as diversas funções que o *Facebook* utiliza para cumprir seus objetivos uma delas são os eventos. Possuem fácil acesso, qualquer pessoa com uma conta pode anunciar um evento na plataforma. Na FIGURA 1 é possível verificar o formulário para criação de eventos na plataforma.

**FIGURA 1** - Formulário para cadastro de eventos no *Facebook*



The image shows a screenshot of the Facebook event creation interface. At the top, there is a dropdown menu set to 'Criar evento privado'. Below this is a blue notification bar stating: 'Você está criando um evento privado. Para proteger a privacidade dos convidados, não será possível alterá-lo para público posteriormente.' The main form area is divided into sections: 'Foto ou vídeo do evento' with options to 'Escolher um tema' or 'Carregar foto ou vídeo'; 'Temas recomendados' showing three theme cards; 'Nome do evento' with a text input field and a character count '0 / 64'; 'Localização' with a text input field; and 'Descrição' with a larger text input field containing the placeholder 'Conte às pessoas sobre o que é o seu evento'. At the bottom, there is a disclaimer: 'Quando você cria um evento no Facebook, as Políticas de Páginas, grupos e eventos se aplicam.' and a blue 'Criar evento privado' button.

FONTE: O Autor, 2019

Os eventos vêm mesclados juntos às datas de aniversários dos seus contatos, é possível fazer convites, permitir que pode convidar, criar eventos privados ou públicos, calendário de eventos, procura por eventos próximos, filtra

eventos por localidade, categoria e tempo, tem um filtro para família e sabe recomendar eventos que estão chamando bastante atenção, principalmente com seus amigos.

A plataforma se apresenta bem completa na questão de procura e anúncio de eventos, de pontos fracos pode ser apontado que a plataforma em si não é especificamente para eventos, seus propósitos são outros e a própria empresa lista em seu sítio que as pessoas usam o *Facebook* para compartilhar coisas importantes, achar amigos e se informar sobre o que acontece pelo mundo. Eventos não são diretamente um dos assuntos listados pela plataforma.

Outro ponto fraco importante é a existência de informações falsas fora de controle que existem dentro da plataforma. Em 2012, foi publicado um relatório que afirmava que o *Facebook* possuía 950 milhões de usuários, dos quais 8,7% eram usuários falsos. Entre eles, contas duplicadas e de pessoas que registram seus animais de estimação.

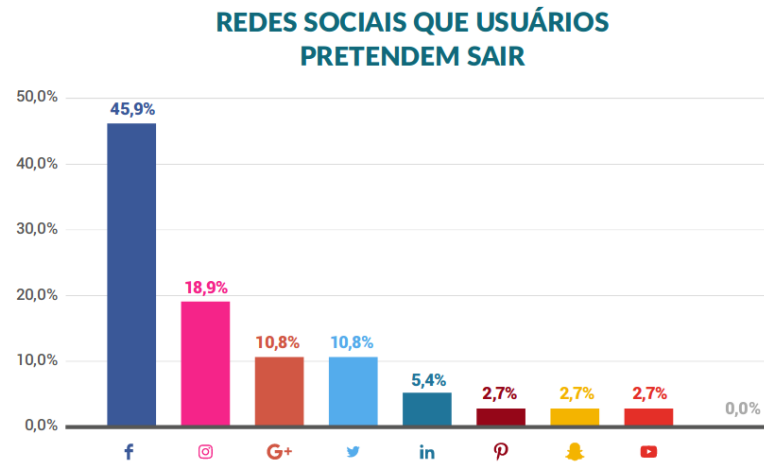
As informações falsas também podem ser verificadas na publicação de eventos, acontecendo até como forma de piada conforme mostra a FIGURA 2.

FIGURA 2 - Exemplo de evento falso no Facebook



FONTE: O Autor, 2019

Em uma pesquisa realizada pela empresa de *Marketing* de Conteúdo *Rocket Content*, foi mostrado que cerca de 32% das pessoas pretendem sair das redes sociais. Dessas pessoas, 45,9% delas pretendem sair do *Facebook*, sendo essa a rede social com maior número de pretendentes a sair. Na FIGURA 3 é possível ver o gráfico disponibilizado pela empresa.

**FIGURA 3 - Gráfico Redes Sociais que os Usuários Pretendem Sair**

FONTE: Rock Content, 2019

Com essa estatística, a empresa aproveitou para questionar se o investimento das empresas em estarem nessa rede social para atingir o público-alvo é realmente um bom investimento, pois esses números são um indicador de que as pessoas não estão tão engajadas assim na rede social onde as empresas mais investem.

### 2.2.2 Eventbrite

Segundo suas próprias palavras em seu sítio, “A *Eventbrite* é uma plataforma global de experiências ao vivo que permite que qualquer um crie, compartilhe, encontre e compareça a eventos que alimentam suas paixões e enriqueçam suas vidas.”

Essa plataforma apresenta quase 4 milhões de eventos realizados em 2018, teve quase 800 mil criadores de evento e trabalha em mais de 170 países de acordo com estatísticas disponíveis no sítio da plataforma. Esse projeto vem sendo desenvolvido desde 2013 e percebe-se que apresenta uma proposta muito semelhante ao produto final deste TCC.

Possui modo de pagar ingressos pela plataforma, mais de um tipo de ingresso, existem dois aplicativos, um para pessoas que procuram eventos e outro pago para organizadores de eventos, disponibiliza diversos tipos de suporte, até mesmo atendimento e suporte pessoal no local do evento, estatísticas em tempo real de vendas de ingressos, *layouts* personalizáveis para cada evento, integração com grandes redes sociais como *Facebook*, *Instagram* e *Spotify*, e-mails e

notificações, calendário, perfil de organizador separado do perfil de procurador, pesquisa com filtro por local, tempo, categoria, tipo e preços.

O maior ponto fraco dessa plataforma é que ela não aproxima usuários de empresas, nem usuários de outros usuários. Não existe uma rede social aproximando os usuários, e a “construção de um *website* voltado a esses interesses deverá demonstrar o poder de divulgação de produtos e serviços que as redes sociais possuem” (DANILO PESSOA CARDOSO, 2011).

Por fim, a plataforma possui apenas as categorias de evento para ajudar a definir o público-alvo.

### 2.2.3 Sympla

A autodenominada empresa com o maior número de eventos à venda do Brasil é uma plataforma que conecta pessoas a eventos, com ações desde a descoberta à realização do evento. Abrange todos os tipos de eventos e de público, agindo em todo o território brasileiro segundo informações publicadas no sítio.

Possui venda de ingressos pela plataforma, mais de um tipo de ingresso, estatísticas sobre as vendas do evento, aplicativo para quem procura eventos e organizadores de eventos, e-mails e notificações, estatísticas do *Google Analytics*, pesquisa filtrada por nome, localidade, data e categorias, personalizar página do evento e uma *API*.

Assim como o *Eventbrite*, não possui interação entre os procuradores de evento, também não disponibiliza estatísticas para se comparar com outras plataformas concorrentes. Também não possui um espaço para público-alvo além das categorias de cada evento.

### 2.2.4 Doity

Criada por dois universitários quando perceberam a necessidade de uma plataforma para organizar o próprio evento, vem se desenvolvendo desde 2013, conta com mais de 2,3 milhões de inscrições e mais de 30 mil eventos cadastrados (DOITY, 2019).

Apesar de o projeto ser inicialmente brasileiro, o site é metade em inglês, metade em português e não existe uma opção para escolher a linguagem.



Possui filtro de pesquisa por categoria, data, localidade e nome, um aplicativo para fazer *check-in* em eventos, sorteios pelo aplicativo móvel, pode traduzir o site em libras, espaço para eventos online, mais de um tipo de ingresso.

Assim como as duas anteriores, não possui interação entre os usuários. Todo o suporte do site é voltado para divulgadores de evento e não para as pessoas que procuram. E também não possui um público-alvo mais bem definido do que pelas categorias do evento.

#### 2.2.5 Disk Ingressos

*Disk* ingressos já é uma empresa famosa, desde 2005 atua na venda de ingressos para *shows*, peças teatrais, eventos culturais e esportivos. Não cobre toda a variedade de eventos, como palestras, exposições, entre outros.

O cadastro de organizadores não é simplificado como nas outras plataformas, aqui é necessário ligar para um número de telefone e pedir que o cadastro seja feito, então será gerado um *login* e senha para que possa ser usado. Já usuários procuradores poderão se cadastrar do modo tradicional.

Possui um aplicativo pouco usado e com avaliação de apenas 1.4, mais de um tipo de ingresso e modalidades para cada tipo, venda de ingressos pela plataforma, pontos de venda físicos e um cartão específico da plataforma. Sendo as duas últimas características citadas, apenas essa plataforma apresenta.

Cobre poucas cidades do Paraná e Santa Catarina. Não possui interação entre os usuários. Possui categorias, chamadas de “modalidades”, e quase todas se referem a estilos musicais (com exceção de algumas como “teatro” e “dança”), como pode ser notado na FIGURA 4.

FIGURA 4 - Categorias do *DiskIngressos*

FONTE: O Autor, 2019

### 2.2.6 Outgo

Em seu sítio, a empresa define seu trabalho como “soluções para gestão e comercialização de ingressos físicos e online”. Está no mercado desde 2014 e é estritamente brasileira.

Possui um aplicativo, mais de um tipo de ingresso, integração com *PagSeguro* (plataforma de pagamentos), filtra eventos por localidade e nome, compra de ingressos via sítio e aplicativo, cobre 11 estados brasileiros.

A plataforma se mostra como uma versão muito simplificada das outras ferramentas, não chega a ter alguns pontos importantes como categoria de eventos e pesquisa por data. Diferente das outras plataformas, para criar eventos é necessário enviar um pedido aos administradores do site para que eles liberem o

evento. Mantém a falta de interação entre usuários e não possui nenhuma ferramenta de público-alvo.

### 2.2.7 Resumo da Concorrência

Apenas metade das ferramentas concorrentes chamou atenção da equipe como algo que realmente deve ser superado, enquanto a outra metade das ferramentas se mostraram básicas de mais ou com erros relativamente pesados.

Como analisado anteriormente, com exceção do *Facebook*, todas as plataformas tem uma forma de venda de ingressos, o que acaba se mostrando o foco principal da maioria dessas plataformas.

O *Facebook* é a única plataforma que apresenta interação com os usuários, já que, como dito pela própria empresa, seu objetivo é unir as pessoas e não divulgar os eventos. Os eventos ficam sendo uma função secundária da plataforma.

O maior ponto fraco que pode ser explorado é juntar o site a ideia de rede social. Com isso, apenas o *Facebook* não seria deixado para trás, porém as outras ferramentas já fazem coisas que apenas o *Facebook* não faz que seria vender ingressos.

Apesar desses ideais erguidos, ainda fica de pé a ideia de que é possível implementar ferramentas e informações que nenhuma outra ferramenta apresentou, como uma previsão de quantas pessoas o público-alvo daquele evento vai atingir em determinada região e até uma comparação de quais as melhores regiões para apresentar aqueles eventos.

## 2.3 TECNOLOGIAS DE DESENVOLVIMENTO

As tecnologias podem ser divididas em quatro categorias: *Front-End* (telas que os usuários usarão no sistema), *Back-End* (gerenciamento de como as informações do sistema serão guardadas e mostradas aos usuários pelo *front-end*), Documentação (os diagramas dos apêndices deste documento e inclusive este documento) e Produção (ferramentas usadas para auxiliar o desenvolvimento do sistema).

### 2.3.1 Front-End

As telas de sítios da internet são escritas basicamente usando *HTML*, *CSS* e *JavaScript*. Alguns *frameworks* e bibliotecas podem ser utilizados para agilizar diversas ações dessas três linguagens.

O *Framework Materialize* é feito para ajudar no desenvolvimento das telas, o *Framework* envolve três linguagens: *HTML*, *CSS* e *JS*. Se baseia no *Material Design*, um padrão de desenvolvimento de telas criado e usado pela *Google* com o intuito de padronizar princípios de bom design.

O *jQuery*, uma biblioteca para *JS*, também foi utilizado no sistema. Esta biblioteca inicialmente foca em reduzir a sintaxe do *javascript*, tornando-o mais fácil, também visando padronizar funções de uso comum.

#### 2.3.1.1 HTML

Segundo Laura Lemay, *HTML* significa *Hypertext Markup Language* (Linguagem de Marcação de Hipertexto) e foi baseado na *SGML* (*Standard Generalized Markup Language*, ou em português Linguagem de Marcação de Padrão Generalizado) que era maior e mais complicado de se processar.

Quando o *HTML* é lido em algum navegador, as *tags HTML*, imagens e textos são interpretados ou analisados na tela. O navegador tem um mapeamento entre o nome dos elementos da página e o estilo da tela (por exemplo, os títulos aparecem em letras maiores). O navegador também ajusta o texto para que ele caiba no elemento em que está inserido (LAURA LEMAY, 2016).

O *HTML* possui um número de *tags* pré-definidas que você pode usar. Apenas com ele, não é possível criar novas *tags* com seus próprios estilos e características. E cada navegador pode aceitar algumas *tags* únicas que não existem em outros navegadores (LAURA LEMAY, 2016).

#### 2.3.1.2 CSS

Como explica Laura Lemay em seu livro *HTML, CSS & JavaScript Web Publishing in One Hour a Day*, o *HTML* por si só não tem a capacidade de controlar como seus elementos vão se parecer, porém o *CSS* pode controlar como os navegadores vão renderizar os elementos do *HTML*, podendo assim renderizar *tags HTML* diferente de sua aparência padrão.

Ainda segundo Laura Lemay, o CSS pode ser usado de várias maneiras. Por exemplo, você pode usar em todas as páginas do seu site um mesmo arquivo de CSS, e uma alteração no site inteiro pode ser feita de apenas um arquivo. Também é possível aplicar o CSS em uma única página e até mesmo em uma única *tag* de *HTML*. Também existe um controle muito complexo sobre quais *tags HTML* você vai alterar, podendo serem todas as *tags* de um certo tipo, ou você pode especificar classes para seus elementos e usá-las para definir que apenas esses elementos receberão a alteração.

### 2.3.1.3 JavaScript

Essa é uma linguagem de programação que é usada para transformar páginas da internet em aplicações. *JavaScript* é usado para manipular o conteúdo das páginas e permitir que usuários interajam com as página sem precisar atualizá-la (LAURA LEMEY, 2016).

Laruen Lemey explica em seu livro que o *JavaScript* pode ser usado assim como o CSS, num arquivo distribuído para todo o site, dentro de uma *tag* inserida individualmente em uma página e até diretamente nas *tags HTML*.

### 2.3.1.4 Material Design

Foi desenvolvido pela *Google* em 2014 visando agilizar e padronizar o desenvolvimento do design de diversos tipos de aplicações. Nas palavras da própria *Google* “É uma linguagem visual que sintetiza os princípios clássicos de bom *design* com inovação da tecnologia e ciência”.

Em seus objetivos estão listados a criação linguagem visual assim como já citado, unificar a experiência do usuário independente da plataforma que ele esteja usando e expandir a linguagem tornando-a flexível para inovações e novas experiências.

Os princípios básicos do *Material Design* são copiar o mundo real e suas texturas, o que inclui as sombras e o reflexo da luz nos objetos, como se a tela fosse uma superfície de papel e tinta (GOOGLE, 2009).

Em seu sítio, os princípios são descritos em 4 partes: Os textos são baseados em métodos de impressão, criando hierarquias entre os textos que ajudam no foco adequado para cada conteúdo. O movimento dos componentes da tela devem trazer algum significado, seja o modo da tela aparecer e desaparecer ou a interação com

objetos dentro das telas. O *material design* é escrito de forma que ele aceita a implementação de componentes, *plugins* e novos elementos de *design*. A interface de usuário deve ser a mesma independente de qual plataforma o usuário está usando, seja um celular pequeno ou uma computador de tela 4K.

#### 2.3.1.5 Materializecss

É um *framework* responsivo e moderno baseado no *Material Design*, é desenvolvido desde 2014 por 4 estudantes da *Carnegie Mellon University*.

Em sua página, suas qualidades são descritas como: A agilização do desenvolvimento, foco em experiência do usuário e fácil de trabalhar.

A agilidade é explicada pois a maior parte do trabalho da implementação do *Material Design* já vem pronta (por exemplo, o estilo das entradas de dados, o *grid* da página, os sombreamentos e diversas animações). Tanto *css* quando *javascript* já vem completos com todas as ideias listadas no sítio do *Material Design*.

O foco de experiência do usuário é garantido através de diversos componentes e animações que já estão inseridos neste *framework* sem a necessidade de configurá-los. Todos esses componentes e animações são baseados no *Material Design*, incluindo a responsividade do *layout* do sítio entre as diferentes plataformas.

A facilidade de trabalhar vem com uma documentação extremamente detalhada e exemplificada que pode ser encontrada e versionada no sítio do *Materialize*. Os autores desse *framework* ainda garante estarem totalmente abertos a *feedbacks* e que podem responder a qualquer dúvida sobre o *Materialize*.

#### 2.3.1.6 jQuery

Em 2005, visando contornar dificuldades do *JavaScript* em encontrar elementos nas páginas, John Resign criou um conjunto de funções que facilitavam a busca por esses elementos. Mais tarde, adicionou mais algumas funções para agilizar a animações e modificações no *DOM*. Esse conjunto de funções foi chamado então de *jQuery* e recebe atualizações desde então. É assim que contam Chaffer e Swedberg em seu livro "*Learning jQuery*".

Em seu site, o *jQuery* se descreve como uma biblioteca *JavaScript* rápida, fácil e rica em peças extras. Fazendo ações como manipular *HTML*, lidar com

eventos, animações e requisições *ajax* muito mais simples de se usar e que funciona igualmente entre vários navegadores diferentes.

### 2.3.2 Back-End

Para desenvolvimento de back-end existe um conjunto de tecnologias que podem ser usadas conhecidas como *LAMP: Linux, Apache, MySQL e PHP*.

*Linux* é o sistema operacional onde estarão rodando todas as outras tecnologias. *Apache* é o servidor que estará rodando o *PHP*. *MySQL* é o banco de dados que guardará todas as informações do sistema. O *PHP*, por sua vez, é a linguagem de programação que definirá os comportamentos do servidor, mostrando informações aos usuários, recebendo essas informações e gerenciando as informações que ficarão no banco de dados.

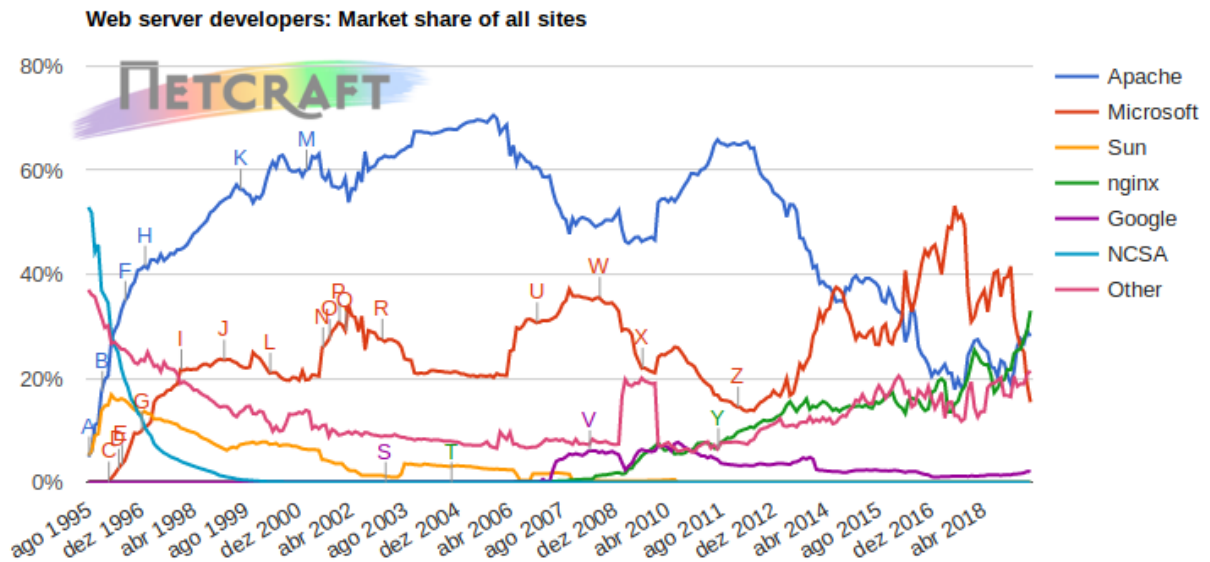
#### 2.3.2.1 Apache

Ainda em 1995, a empresa *National Center for Super Computing Applications* (NCSA) criou um servidor muito potente, logo no mesmo ano ele começou a ser abandonado pela empresa. Um grupo de outras empresas que acabaram se utilizando desse servidor criou um fórum, que ficou conhecido como *Apache Group*, para juntos discutirem as melhores atualizações para o servidor. A partir daí, o servidor denominado *Apache* se tornou o servidor *WEB* mais usado (KABIR, 2002).

Uma das funções do *Apache* que o fazem ser um bom servidor é que ele é altamente configurável, podendo ser configurado por um único arquivo de texto ou até ser flexibilizado entre vários arquivos menores (KABIR, 2002).

O *Apache* é listado com vários recursos que o tornam um servidor bem completo, entre eles o suporte para os protocolos *HTTP*, configuração baseada em arquivos, suporte para *CGI* e *FastCGI*, suporte para *hosts* (*IP* e nomeados), autenticação *HTTP*, *Perl*, *Servlets Java* e *PHP*, *proxy* integrado, *status* do servidor, *logs* personalizados, entre outros (KABIR, 2002).

Segundo dados da *Netcraft*, empresa de serviços a *Internet*, o *Apache* segue sendo o segundo servidor mais usado atualmente, sendo passado apenas pelo *nginx*. Veja abaixo nas FIGURAS 5 e 6 um gráfico e uma tabela demonstrando as estatísticas reveladas pela *Netcraft* em 2019.

**FIGURA 5 - Servidores Mais Usados no Mercado (Gráfico)**

FONTE: Netcraft, 2019

**FIGURA 6 - Servidores Mais Usados no Mercado (Tabela)**

Developer	May 2019	Percent	June 2019	Percent	Change
nginx	387,416,889	29.20%	439,626,713	32.97%	3.77
Apache	385,685,252	29.07%	374,360,949	28.08%	-1.00
Microsoft	250,440,887	18.88%	205,235,291	15.39%	-3.49
Google	27,711,375	2.09%	28,181,744	2.11%	0.02

FONTE: Netcraft, 2019

### 2.3.2.2 MySQL

MySQL é “um SGBD relacional que utiliza a linguagem padrão SQL... e é largamente utilizado em aplicações para *Internet*” (NIEDERAUER e PRATES, 2006).

Na década de 90, ao necessitar de uma interface SQL com compatibilidade ao ISAM, foi criado, em C e C++, com base no *mSQL*, o MySQL, uma versão mais rápida que utilizava funções de mais baixo nível que a anterior. Com o excelente resultado, o MySQL foi divulgado e tornou-se conhecido. Apesar de ser *open source*, suas versões atuais podem até competir com bancos de dados privados como SQL Server e Oracle (ANDRÉ MILANI, 2006).

Junto ao MySQL, vem integrado o MySQL Server, que pode gerenciar os acessos ao banco de dados, a integridade entre os dados relacionais, concorrência,



transações, entre outras características que são tratadas automaticamente graças a esse recurso (ANDRÉ MILANI, 2006).

André Milani aponta em seu livro diversas características que o *MySQL* possui. Entre elas a portabilidade (compatível com várias linguagens de programação), ser *multithread* (poder rodar mais de uma consulta *SQL* ao mesmo tempo), diversas formas de armazenamento (cada tabela do banco pode priorizar velocidade, volume de informações ou o que for melhor independente das outras tabelas), velocidade, segurança, pesquisa através de *fulltext*, capacidade (podendo armazenar 65 *terabytes* por tabela, *SQLs* com até 61 *JOINS* e rodar até bilhões de consultas por dia).

### 2.3.2.3 PHP

Rasmus Lendorf precisava monitorar os acessos ao seu currículo na *Internet* e para isso desenvolveu um pequeno conjunto de *scripts* que o ajudavam a dinamizar suas páginas. Depois, Lendorf desenvolveu um pouco mais esse *scripts* com uma implementação em *C* que tornou os *scripts* fáceis de se usar. A partir disso, nasceu o que hoje conhecemos como *PHP* (PABLO DALL'OGGIO, 2007).

O *PHP* é uma linguagem de programação que não precisa ser compilada para funcionar, diferente de muitas outras linguagens. Com ela ainda é possível programar estruturalmente ou orientado a objetos, é possível mesclar códigos *PHP* com *HTML*, tratar dados das requisições, gerenciar sessões, entre outras funções que o tornam uma linguagem ideal para desenvolvimento para *Internet* (PABLO DALL'OGGIO, 2007).

### 2.3.3 Documentação

A intenção da documentação é registrar todo o desenvolvimento do projeto aqui apresentado. Desde as ideias iniciais, às pesquisas para o desenvolvimento, os registros das tecnologias e usadas e até os resultados finais, tudo que se encontra neste documento é o que se refere a parte da documentação.

Essa sessão do projeto, por sua vez, foi encarada com duas diferentes partes devido a diferença entre o modo de desenvolvimento de suas partes: A documentação escrita, que se resume aos 5 capítulos deste documento, ao sumário, resumo, capa, referências e tudo mais que não se encaixe diretamente na segunda parte. Esta, por sua vez, é identificada por “documentação diagramada” a qual se

refere aos diagramas apresentados nos apêndices deste documento e também a especificações de casos de uso que são uma parte escrita de um dos diagramas.

#### 2.3.4 Produção

Este capítulo pretende descrever os *softwares* e aplicativos usados para desenvolver o projeto, desde o código de programação, ao banco de dados e até a documentação escrita.

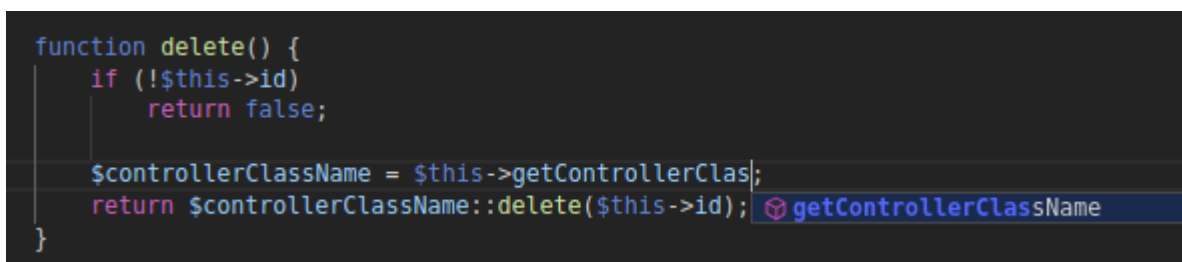
##### 2.3.4.1 Visual Studio

É uma interface de programação desenvolvida pela empresa *Microsoft*. É de graça e pode ser encontrado em seu sítio disponível para qualquer plataforma.

Essa ferramenta possui integração com o *GIT*, sistema de versionamento de arquivos, agilizando o trabalho dos programadores que desejam usar esta ferramenta em conjunto ao seu projeto (Microsoft, 2019).

Segundo a *Microsoft*, foi integrado no *Visual Studio* um recurso chamado *IntelliSense*, ele permite que os textos escritos na ferramenta possam ser coloridos para melhorar a intuição do programador que ver a tela, também oferece opções para completar o texto digitado automaticamente. Exemplo disso pode ser visto na FIGURA 7 a seguir.

**FIGURA 7 - IntelliSense do Visual Studio**



```
function delete() {  
    if (!$this->id)  
        return false;  
  
    $controllerClassName = $this->getControllerClas;  
    return $controllerClassName::delete($this->id);  
}
```

The image shows a code editor with a dark background. The code is PHP. The line `$controllerClassName = $this->getControllerClas;` is partially typed, and the IntelliSense feature is active, showing a dropdown menu with the suggestion `getControllerClassName` highlighted in blue.

FONTE: Autor, 2019

O *Visual Studio* ainda possui um sistema de suporte para *debuggar* o código que está sendo desenvolvido, para verificar possíveis erros de lógica e até a apontar erros de sintaxe (Microsoft, 2019).

Por fim, a *Microsoft* destaca no sítio da ferramenta que ela possui o recurso de receber *plugins*. Caso você precise incrementar novas funções a ferramenta,

basta baixar algum *plugin* específico para a situação e necessidade que se sucedem.

#### 2.3.4.2 MySQL Workbench

É uma ferramenta visual feita para quem pretende desenvolver, arquitetar e gerenciar bancos de dados *MySQL*. A ferramenta providencia funções para modelagem, desenvolvimento de *SQLs*, e uma ferramenta para administração compreensiva para configurar o servidor, seus usuários, *backups* e muito mais (MYSQL™, 2019).

Distribuído pela *Oracle Corporation*, a aplicação pode ser encontrada disponível para *download* no site do *MySQL™* sem custo algum, disponível para plataformas como *Windows*, *Mac OS* e *Linux* (MYSQL™, 2019).

#### 2.3.4.3 LibreOffice

O *LibreOffice* é um conjunto de aplicações de produtividade para escritórios, podendo produzir planilhas, documentos escritos, apresentações, infográficos e muito mais (THE DOCUMENT FOUNDATION, 2019).

Em seu site, a aplicação se afirma como a mais poderosa suíte de produtividade de escritório livre e aberta do mercado, sendo o *Office* que a comunidade sonhou por anos.

As quatro aplicações básicas do suíte são: *LibreOffice Write* (produção de documentos escritos), *LibreOffice Calc* (para planilhas e gráficos), *LibreOffice Impress* (para apresentações) e o *LibreOffice Draw* (para infográficos, diagramas e outros). O pacote pode ser encontrado disponível no sítio do *LibreOffice* (THE DOCUMENT FOUNDATION, 2019).

#### 2.3.4.4 Xmind

É uma aplicação para celulares e computadores que, segundo o próprio sítio em 2019, deve ser desenvolvido por programadores de todo o mundo, visando o mapeamento de ideias que rode em todo o tipo de plataforma e melhore a eficiência no trabalho de seus usuários. Esse programa possui uma versão paga e outra versão grátis.

#### 2.3.4.5 Astah UML

Essa é uma ferramenta para desenvolvimento de diagramas *UML*, seu nome significa *Java and UML Developers Environment* que significa “Ambiente *UML* e *Java* para Desenvolvedores” (CHANGEVISION, 2019).

Em seu sítio, é garantido que a ferramenta possui as funções de montar diagramas pertencentes a qualquer versão de *UML 2* ou mais, possui integração com códigos podendo reverter o código de programação em diagramas *UML* (*Java*, *C#*, *C++* e *PHP* sendo a última apenas geração de código e não engenharia reversa), possui uma *API* que permite a instalação de *plugins*, tem suporte para ser desenvolvido em equipe e é pago, porém contendo uma versão para estudantes.

#### 2.3.4.6 UML

Segundo Marting Fowler e Kendall Scott, *UML* é uma linguagem de modelagem, que “é a notação (principalmente gráfica) utilizada por métodos para expressar projetos” (em *UML Essencial*, 2000). Os modelos utilizados foram o diagrama de casos de uso (e suas especificações), diagrama de classes e o diagrama de sequência.

Para Booch, Rumbaugh e Jacobson, em seu livro “*UML – Guia do Usuário*”, o *UML* é importante principalmente para projetos grandes, pois ajuda a visualizar o sistema como ele é ou como desejamos que ele seja, especificar a estrutura ou comportamento do sistema, proporciona um guia para a construção do sistema e documentam as decisões tomadas.

O mesmo livro ainda descreve diagramas como “uma apresentação gráfica de um conjunto de elementos, geralmente representados como um gráfico conectado de vértices e arcos”.

Dentro deste projeto serão desenvolvidos quatro tipos de digramas: O diagrama de casos de uso, o diagrama de classes, o diagrama de sequência e o diagrama de modelo lógico.

O diagrama de classes se encarrega de documentar todas classes do sistema e as relações entre seus objetos (tanto de subtipos quanto de associações). Também anota os valores que cada objeto carrega (FOWLER e SCOTT, 2000).

O diagrama de classes é o diagrama mais encontrado em sistemas de modelagem orientados a objetos, os diagramas são usados para representar

estaticamente a visão do projeto de um sistema (BOOCH, RUMBAUGH e JACOBSON, 2005).

Todo sistema possui interações com atores que utilizam o sistema com algum propósito e espera-se que o sistema se comporte de maneira prevista. Os casos de uso são a especificação de como o sistema deve se comportar (BOOCH, RUMBAUGH e JACOBSON, 2005).

Em cada caso de uso, o ator gera eventos no sistema ao solicitar alguma operação. A *UML* inclui os diagramas de sequência como uma notação que pode ilustrar as interações de atores e as operações iniciadas por eles. Este diagrama deve indicar os eventos gerados, sua ordem e os eventos entre os sistemas (CRAIG LERMAN, 2007).

## 2.4 METODOLOGIA DE DESENVOLVIMENTO

Em 2001, um grupo de especialistas se reuniu visando organizar uma alternativa às metodologias de desenvolvimento mais usados na época. Definiu-se, na ocasião, 4 valores que todos os métodos ágeis atuais carregam (THE AGILE MANIFESTO, 2001):

- 1 - Indivíduos e interações mais do que processos e ferramentas;
- 2 - Software funcionando mais que documentação abrangente;
- 3 - Colaboração com o cliente mais que negócios; e
- 4 - Contratos e responder as mudanças mais do que seguir um plano .

O criador do *Scrum*, Jeff Sutherland, descreve em seu livro “*Scrum: A arte de fazer o dobro do trabalho na metade do tempo*”, que se deparou com diversas dificuldades no gerenciamento de projetos de várias empresas no decorrer de sua carreira. Juntando o manifesto ágil com experiências adquiridas em guerra e a sabedoria japonesa, o referido planejou uma abordagem que resolvia os problemas que encontrou na organização das empresas, aumentando a produção e diminuindo o tempo gasto. Essa abordagem ficou conhecida como *Scrum*.

Na implementação do *Scrum*, inicialmente é criada uma lista de requisitos para o trabalho. Estes devem ser priorizados na ordem dos que agregam maior valor ao projeto. As funcionalidades são listadas em um artefato chamado de *Product Backlog* (JEFF SUTHERLAND, 2019).

A partir deste ponto, pode-se dividir o fluxo do projeto em partes menores chamadas de *sprints*, onde cada *sprint* apresenta um resultado usável do projeto

final. As *sprints* perfazem um período de tempo igual, geralmente um mês (SCHWABER, 2004).

O *scrum* possui três principais perfis que tomarão as decisões do projeto com o intuito de garantir o sucesso do produto final, o *Project Owner*, a *Team* (Equipe) e o *Scrum Master* (JEFF SUTHERLAND, 2019).

O *Project Owner* é o responsável por criar o *Product Backlog*, ajudando a definir a importância dos participantes do projeto e do resultado final. Esse perfil também é responsável pelos recursos para manter o projeto do início ao fim (JEFF SUTHERLAND, 2019).

As funcionalidades do projeto são desenvolvidas pela equipe. Cada equipe fica responsável por uma tarefa diferente, de preferência equipes pequenas (geralmente entre 5 a 9 pessoas). As equipes deverão ter poder para escolher a melhor forma para o desenvolvimento do projeto, independente das outras equipes, também deverão ser todas multifuncionais, ou seja, devem saber fazer qualquer parte do projeto e não apenas uma em específico (JEFF SUTHERLAND, 2019).

Cada equipe é gerenciada por um *Scrum Master*, que tem como função de conduzir as reuniões das equipes, checar se tudo está ocorrendo devidamente como o esperado no desenvolver do projeto e ajudar a equipe a corrigir erros que atrapalham o desenvolvimento do projeto (JEFF SUTHERLAND, 2019).

A cada *sprint* é feita uma reunião com a equipe, nela são comentados os principais objetivos necessários e discutido a avaliabilidade destes objetivos (SCHWABER, 2004).

Existem ainda as reuniões diárias, onde cada membro responde a três perguntas: O que eu fiz hoje? O que eu pretendo fazer entre hoje e amanhã? O que está atrapalhando o desenvolvimento da *sprint*? E a resposta de cada membro poderá afetar o desenvolvimento de toda a equipe em visão de manter e melhorar o desenvolvimento do projeto até a finalização de todo o processo (SCHWABER, 2004).

### 3 MATERIAL E MÉTODOS

Todas as tecnologias em uso foram escolhidas primeiramente pensando em não necessitar gastar nenhum tempo aprendendo novas tecnologias pela equipe, o segundo critério foi a facilidade para adquirir e usar a tecnologia, considerando preços, instalação e usabilidade.

O projeto foi desenvolvido utilizando-se da metodologia *Scrum*, que consiste em dividir o projeto e várias entregas menores (*sprints*) e incompletas até o prazo final quando o produto é entregue pronto. As vantagens dessa metodologia incluem uma maior agilidade de desenvolvimento, gerenciamento de tempo e o acompanhamento mais direto entre o cliente e o projeto.

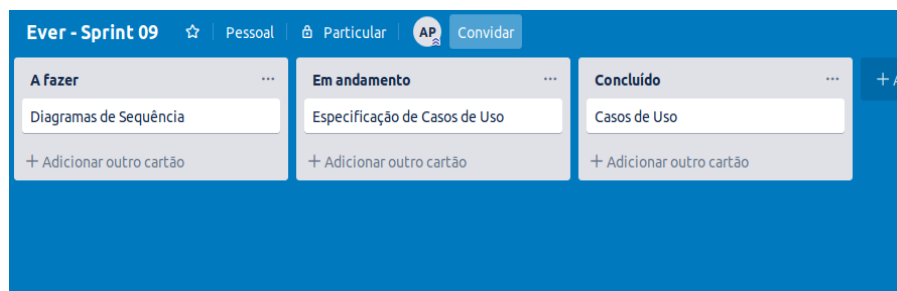
Cada *sprint* do projeto durou 1 semana e foi apresentada ao professor orientador deste TCC (Razer Anthom Nizer Rojas Montaña) em seus devidos prazos, exceto alguns casos citados nas descrições dos *sprints*. Estes foram divididos em 3 partes, as tarefas A Fazer (que ainda não foram feitas), Em Andamento (que estão sendo feitas ou testadas) e Concluído (que já foram feitas).

#### 3.1 IMPLEMENTAÇÃO DO SCRUM NO DESENVOLVIMENTO DO PROJETO

Para melhor implementação do *scrum*, a equipe optou por usar a ferramenta *Trello*, um gerenciador de projetos online que consiste em organizar todas as tarefas do projeto, permitindo dividi-las em *sprints* e *kanbans* e permitindo marcá-las com comentários ou cores a sua escolha.

A FIGURA 8 a seguir mostra o funcionamento do *Trello*, configuradas as *sprints* e os *kanbans*.

**FIGURA 8 - Exemplo de Sprint**



FONTE: O Autor, 2019

As tarefas foram organizadas como “Desenvolvimento – *Cliente*”, “Desenvolvimento – *Administrador*” e “Documentação”. O desenvolvimento do *sítio* e do *administrador* avançaram mutuamente, enquanto da documentação avançou separado dos demais desenvolvimentos.

Nas reuniões, era mostrado ao orientador tudo o que havia sido produzido e corrigido da *sprint* corrente e passadas as novas correções e tarefas para a *sprint* seguinte.

O projeto foi dividido inicialmente em 12 *sprints*, posteriormente aumentado para 20 *sprints* e por fim diminuído para 19 *sprints*. O cronograma pode ser visto abaixo na FIGURA 9 desenvolvida através do *XMind* e no QUADRO 1.

**FIGURA 9 - WBS**







FONTE: O Autor, 2019

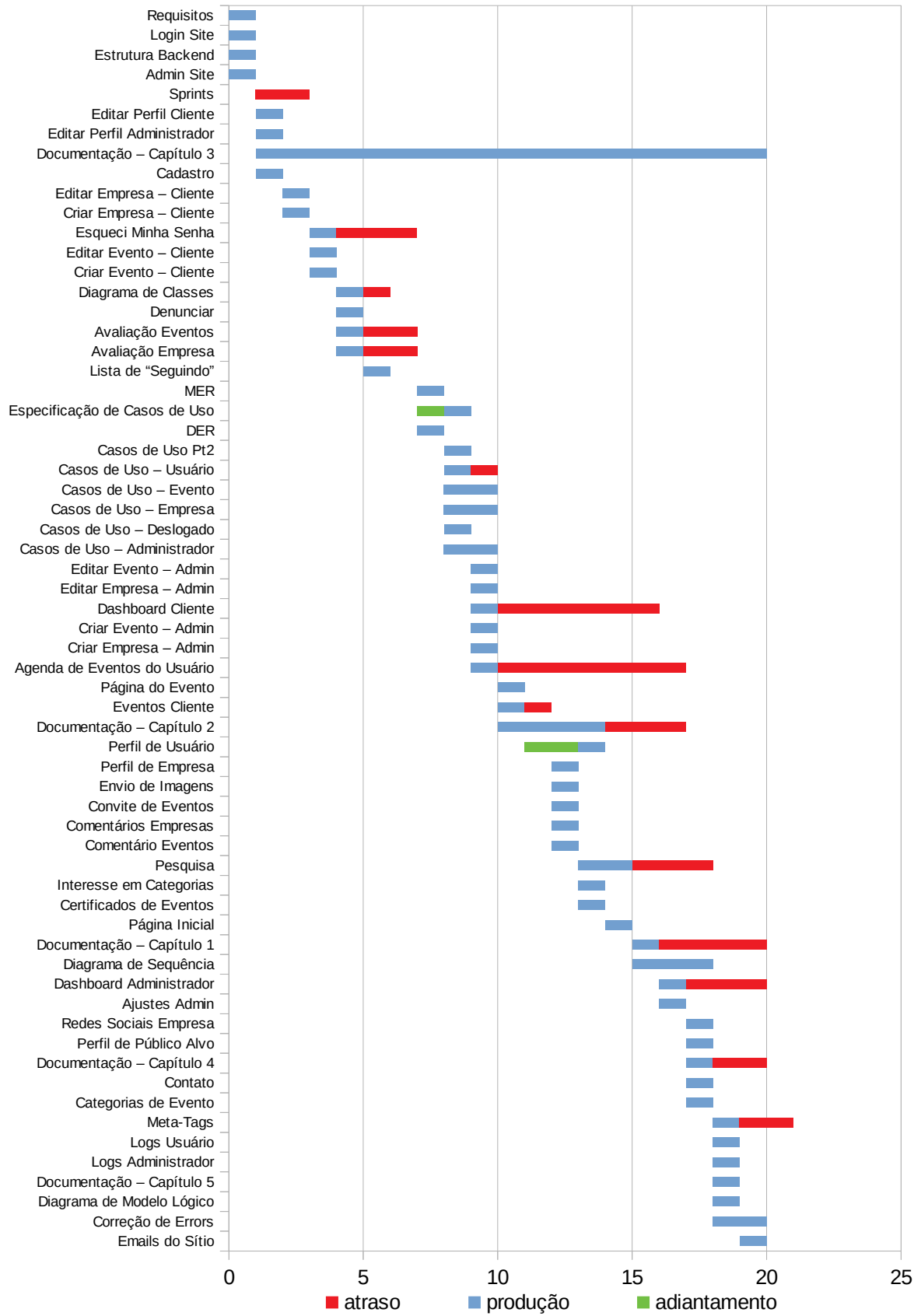
QUADRO 1 – CRONOGRAMA

<b>Sprint</b>	<b>Início</b>	<b>Atividade</b>
1	19/02/2018	Administrador Estrutura do Back-End Login do Site
2	26/02/2018	Requisitos Sprints Editar Perfil – Cliente Editar Perfil – Adminsitrador Documentação – Capítulo 3 Cadastro
3	05/03/2018	Criar Empresa - Cliente Criar Empresa - Administrador
4	12/03/2018	Editar Evento Criar Evento - Cliente
5	19/03/2018	Diagrama de Classes Denunciar Avaliar Evento Avaliar Empresa
6	25/03/2018	Lista de “Seguindo” Lista de “Seguidores”
7	01/04/2018	MER
8	18/02/2019	DER
9	25/02/2019	Casos de Uso Especificação de Casos de Uso Casos de Uso
10	03/03/2019	Criar e Editar Evento - Administrador Criar e Editar Empresa - Administrador Página Inicial Cliente Agenda Usuário
11	12/03/2019	Página do Evento Lista de Eventos - Cliente Documentação – Capítulo 2
12	18/03/2019	Perfil de Usuário
13	01/04/2019	Perfil de Empresa Envio de Imagens Convite de Eventos Envio de Imagens
14	15/04/2019	Pesquisa Interesse do Usuário em Categorias Certificados de Eventos
15	22/04/2019	Página Inicial
16	29/04/2019	Documentação – Capítulo 1 Diagrama de Sequência
17	06/05/2019	Dashboard Administrador Ajustes
18	27/05/2019	Redes Socias da Empresa Perfil do Público-Alvo Documentação – Capítulo 4 Contato
19	03/06/2019	Categorias de Evento Logs de Usuários Logs de Administrador Documentação – Capítulo 5 Diagrama de Modelo Lógico Correção de Erros Envio de E-mails Página de Contato

FONTE: O Autor (2019).

No início do projeto, foi criado um gráfico de *Gantt* que tentava planejar o desenvolvimento de cada uma das tarefas do projeto. O gráfico recebeu alterações no decorrer do projeto conforme atividades novas apareciam e algumas eram desconsideradas. A seguir você pode verificar o gráfico *Gantt* no gráfico 1.

GRÁFICO 1 – GANTT



FONTE: O Autor, 2019

## 3.2 DESCRIÇÃO DAS SPRINTS

Cada *sprint* ocorreu entre uma a três semanas de duração, quase sempre deixando alguma tarefa em atraso. Porém, as últimas *sprints* foram planejadas para que atrasos pudesse ocorrer. O planejamento do projeto foi estruturado de forma que sobrasse tempo para imprevistos e atrasos, portanto, inicialmente, da *sprint* 12 para frente não houveram nenhuma tarefa listada, mas foram adicionadas conforme as necessidades.

### 3.2.1 Pré-Sprints

Antes do início das aulas e da implementação definitiva do *scrum*, foi desenvolvido para este projeto um pequeno *framework*, com base na experiência em trabalho da equipe.

O *framework* apenas agilizava o *CRUD* do sítio, e também organizava as pastas dos arquivos e as *URLs* dos sítio, assim como o estilo (aparência, *css*) e alguns dos *plugins*.

Toda a ideia de funcionamento e temática do projeto foi feita no *pré-sprints*. Esse período se iniciou em Dezembro de 2017 e se estendeu até o início das aulas em 19 de Fevereiro de 2018. Houve uma pausa no desenvolvimento em Janeiro por motivos pessoais, o que causou alguns ajustes as pressas no *framework* no primeiro *sprint*.

### 3.2.2 Sprint 1

O intuito dessa *sprint* foi fazer as primeiras funcionalidades do sítio, mostrar ele funcionando inicialmente. Foram feitos os logins do cliente e do administrador, sem os cadastros e nenhuma funcionalidade interna do sítio.

No que se diz a documentação, foram organizadas todas as *sprints*, com planejamento para futuros atrasos e imprevistos. Também foi feita a análise de requisitos.

Pequenos ajustes as pressas foram feitos no *framework* do projeto devido a não finalização do mesmo no período *pré-sprints*.

### 3.2.3 Sprint 2

Na parte de desenvolvimento, foram criadas as páginas de cadastro pelo *sítio* e pelo administrador, assim como a edição dos dados dos usuários e a listagem (pelo administrador).

Na parte da documentação foram acertados detalhes dos *sprints*, foi iniciado o gráfico de *Gantt* e pesquisamos algum programa para auxiliar na montagem da documentação. Inicialmente foram considerados apenas aplicativos *mobiles* que pudesse gerar documentação *UML* por uma questão de comodidade ao poder fazer a documentação em lugares onde é inusual o uso de um *notebook* (e.g. ônibus).

### 3.2.4 Sprint 3

Foram entregues os últimos ajustes de *Sprints* e mostrado como seria organizado o *Gantt*. As primeiras versões dos casos de uso (APÊNDICE B) foram desenvolvidas em aplicativos de celular para *UML*.

Devido a qualidade da documentação e a futuras facilidades, os programas *mobiles* foram abandonados e a equipe adotou o uso do *Astah* (licença grátis de estudante) para gerenciar a documentação.

A parte de desenvolvimento gerou, tanto para o cliente, quanto para o administrador, ações básicas de gerenciamento de empresas (Criar, Editar, Listar...).

### 3.2.5 Sprint 4

A documentação que havia sido feita anteriormente foi toda refeita na ferramenta *Astah*, com as devidas correções.

As mesmas ações desenvolvidas para empresas na *sprint* anterior também foram desenvolvidas para Eventos nessa *sprint*. Nessa *sprint* deveria ter sido feito o “esqueci minha senha”, porém foi adiado para que não se mexesse com o envio de e-mails no *sítio* no momento.

### 3.2.6 Sprint 5

Foram feitas as primeiras especificações dos casos de usos gerados até então (APÊNDICE A), combinou-se com o professor que muitos ajustes deveriam ser feitos na forma como foram escritas as especificações.

No sítio foram desenvolvidas as denúncias de empresas e eventos, também foi iniciado o desenvolvimento dos comentários. Foi uma *sprint* lenta para o desenvolvimento e focada mais na documentação.

A opção de “esqueci minha senha” foi novamente adiada, junto com os comentários em eventos e empresas. O administrador não foi desenvolvido, pelo pouco tempo para desenvolver o cliente e porque o administrador segue uma estrutura muito rígida e padronizada, portanto seria mais rápido implementá-lo mais tarde.

### 3.2.7 Sprint 6

Ao contrário da *sprint* anterior, essa *sprint* se focou mais na parte do desenvolvimento, porém isso não foi proposital. No dia que eu deveria desenvolver a documentação eu acabei não fazendo nada por questões de saúde e a documentação basicamente não se desenvolveu, apenas foi iniciado o capítulo três deste documento.

Em questão de desenvolvimento, foram feitas as páginas de perfil do usuário e perfil da empresa, também as opções de bloquear, seguir, comentar, votar em empresa e denunciar perfil de usuário. O desenvolvimento do administrador foi adiado novamente, assim como o “esqueci minha senha”.

### 3.2.8 Sprint 7

Houve um foco especial para a documentação nessa *sprint* devido ao atraso na anterior. As *sprints* que se antecederam foram descritas até a atual e foram implementadas as correções nas especificações dos casos de uso (APÊNDICE A e APÊNDICE B).

### 3.2.9 Sprint 8

Depois de meses de problemas pessoais da equipe, a primeira semana após o retorno foi focada em ajustar e realinhar o que já estava pronto. Resolvemos nos organizar de maneira diferente dessa vez, focando em terminar a documentação o quanto antes.

Infelizmente esta *sprint* se resumiu apenas em reorganizações no gerenciamento do projeto.

### 3.2.10 Sprint 9

O APÊNDICE C e as especificações de casos de uso do APÊNDICE A foram ajustados em alguns pontos importantes, também foi produzido cerca de 80% do administrador do sítio. Incluindo *front end* e *back end*.

As pendências do administrador ficaram com os gráficos informativos – que foram deixados para o final do projeto, já que qualquer alteração na estrutura do projeto pode prejudicar esses gráficos – com o *upload* de imagens (de eventos, perfil de empresa ou usuário) e com as telas de detalhamento sobre os registros.

### 3.2.11 Sprint 10

A *sprint* 10 foi atrapalhada por questões de trabalho, portanto apenas o *design* (*HTML* + *CSS*) da tela de evento foi feita no decorrer dessa *sprint*. Foi ajustado também o diagrama do que será feito durante as próximas 10 *sprints* disponíveis.

### 3.2.12 Sprint 11

Os afazeres pendentes da *sprint* 10 foram levados para frente e foi iniciado a tela do perfil do usuário. Todas as questões de seguir, denunciar e bloquear foram ajustadas, a atualização da foto de perfil e a publicação do usuário também foi desenvolvida. A documentação andou pouco no desenvolvimento do capítulo 2 deste mesmo documento.

### 3.2.13 Sprint 12

Nessa *sprint* de duas semanas, foram desenvolvidas as tarefas pendentes da *sprint* anterior. Foram produzidos os convites de eventos, respeitando a privacidade dos usuários clientes. Também passou a ser possível comentar em eventos, assim como classificá-los e isso aparece no *feed* do perfil dos usuários e também nos comentários do evento. Eventos passaram a poder cadastrar imagens com título e descrição. Foram corrigidos erros que apareceram nas páginas de erro.

Boa parte do planejamento do projeto mudou nessa *sprint*, conforme o desenvolvimento do capítulo 2, foi decidido que seria necessário que o projeto possuísse novas funcionalidades.



Foi planejado a inserção de categorias de eventos, perfil do público-alvo para eventos, interesse dos usuários em categorias, certificados de participação em eventos e detalhamento das informações que os usuários informam sobre si mesmo (adição de campos como idade e gênero).

O diagrama de classes (APÊNDICE C) foi ajustado para as novas alterações e os casos de uso deixados para atualizar na *sprint* posterior.

### 3.2.14 Sprint 13

O foco de produção nessa *sprint* foi para a parte mais inteligente do projeto. Alguns novos planos foram iniciados na *sprint* anterior e foram planejados e desenvolvidos nessa *sprint*.

O primeiro projeto a ser desenvolvido foi a liberação de certificados de participação em eventos. Caso você participe de algum evento (e.g. como palestrante) você poderá receber pelo sítio um certificado de que você participou daquele evento como palestrante. O diploma fica listado no sítio, pode ser impresso em uma folha A4 e possui um código *QR* com uma *URL* para visualizar e validar o seu certificado online.

O segundo projeto foi o início da pesquisa por eventos. Visando procurar sempre eventos mais relevantes para as pessoas, novas informações precisaram ser implementadas nos cadastros de eventos e pessoas. Os usuários poderiam agora informar sua data de nascimento, gênero e endereço. Uma busca por *cep* foi implementada no sítio para facilitar a inserção do endereço, também foi criada uma interação com a *API* do *Google Maps* para que possam ser obtidas as coordenadas dos devidos endereços. Essa alteração de localidade foi implementada tanto em usuários como em eventos, também foram planejados os cálculos de distância entre os pontos, porém a busca ainda não havia sido implementada.

Ocorreram também pequenas atualizações de *layout* visando deixar o sítio com um visual mais limpo.

### 3.2.15 Sprint 14

Essa *sprint* foi focada principalmente em terminar os atrasos das *sprints* anteriores que ainda estavam pendentes. Devido ao curto prazo do projeto, algumas ideias começaram a ser filtradas e cortadas do produto final. Uma atenção especial

foi dada para algumas “gambiarrras” que precisaram ser corrigidas antes do projeto ser levado para a banca.

A ideia nessa *sprint* era acelerar a finalização da programação do projeto para focar posteriormente na finalização da documentação e ajustes que a documentação necessitou devido a ajustes na programação do projeto.

Infelizmente a produtividade dessa *sprint* foi pouca, pois a equipe teve problemas pessoais e até de saúde.

### 3.2.16 Sprint 15

Nessa *sprint* foram feitas as paginações dos *feeds* de eventos, empresas e usuários. Também foi iniciado o *feed* da página principal e criada a possibilidade de cancelar eventos. As únicas publicações que ficaram faltando no *feed* da página principal foram novos seguidores do seu usuário, das suas empresas e dos seus eventos, eventos cancelados e evento com informações alteradas.

Também ocorreu a retirada de mais algumas “gambiarrras” que ficaram para trás durante o desenvolvimento.

O maior foco dessa *sprint* foi para a finalização do capítulo 2 deste documento e a busca pelas referências em sebos, livrarias e no serviço público *Google Scholar*.

### 3.2.17 Sprint 16

Nessa *sprint* foi achada uma forma de colocar o projeto online para que pudessem ser testadas as *meta-tags* do Facebook. Ainda assim, não foram implementadas as *meta-tags* no sítio. Pequenas correções e melhorias nos nomes das variáveis e campos no banco de dados foram feitas.

Por fim, o diagrama de classes (APÊNDICE C) foi ajustado com as novas alterações desde o começo do projeto quando o mesmo foi feito. Foram iniciadas as alterações nos casos de uso (APÊNDICE A e APÊNDICE B) para que possam ser feitos os diagramas de sequência o quanto antes.

### 3.2.18 Sprint 17

A apresentação dessa *sprint* acabou se estendendo devido a problemas de saúde, tentativas de aperfeiçoamento e compromissos de última hora do orientador. Depois de três semanas foram apresentados os resultados.

Foram corrigidos erros mais complexos que ficaram no projeto, como os ícones que não carregavam sem conexão com a Internet, um aviso de que não é possível enviar imagens maiores que 2MB, o botão de imprimir certificado e ajuste das entradas de data e hora.

A programação do sítio ficou quase completa, faltando apenas o *log* de atividade dos administradores do sítio para ter noção de tudo que os administradores fizeram no sítio. As meta-tags foram descartadas nessa altura do projeto. A parte do administrador foi toda finalizada, o envio de imagens, os detalhes das denúncias e tudo mais que faltava fazer (com exceção dos *logs* dos administrados). Muitas ideias de gráficos foram cortadas do projeto final devido ao tempo, a *dashboard* foi simplificada.

No cliente foi finalizada a página de pesquisa por eventos, pessoas e empresas, sendo os eventos ordenados por questão de relevância para cada usuário (considerando categorias que gosta, amigos, localidade, idade e sexo). A página de contato foi feita. A opção de bloquear perfil foi finalizada para que não se possa seguir pessoas bloqueadas ou que te bloquearam. A *dashboard* passou a mostrar os eventos mais indicados para cada usuário (considerando categorias que gosta, amigos, localidade, idade e sexo). A página do evento agora apresenta algumas alterações caso o evento já tenha passado, não sendo mais possível marcar interesse em eventos que já aconteceram e não sendo possível comentar nesses eventos caso não tenha interesse marcado. A agenda dos usuários foi feita, agora existe uma tela onde cada usuário pode ver o dia de hoje e os eventos próximos, podendo navegar para cada mês de cada ano. Ao cadastrar eventos, existe uma previsão que mostra quantos usuários da plataforma se encaixam no perfil do evento, divididos na cidade do evento e em cada estado do Brasil. Por fim, as empresas agora podem adicionar suas redes sociais para que os usuários possam segui-las em outras plataformas.

Na documentação, foram ajustados os casos de uso (APÊNDICE A e APÊNDICE B) e iniciados os diagramas de sequência (APÊNDICE D).

O que foi calculado para fazer na próxima *sprint* – que é a última antes da entrega semifinal do trabalho ao orientador – foi a finalização do capítulo 1 e 5, a finalização dos diagramas de sequência, o glossário, os *logs* dos administradores e, por fim, o capítulo 4.

### 3.2.19 Sprint 18

Esta *sprint* estava destinada a ser a primeira entrega completa do trabalho, restando mais duas *sprints* apenas para correções e pequenas alterações.

Nela foram desenvolvidos os capítulos 1, 4 e 5 da documentação, o glossário foi descartado por falta de tempo.

Entre os diagramas foram ajustados o diagrama de classe (APÊNDICE C) e as especificações de casos de usos (APÊNDICE B) para incluir o novo sistema de *logs*. A partir das especificações de casos de uso, foram feitos os diagramas de sequência (APÊNDICE D). Por fim, foi feito também o diagrama de modelo lógico (APÊNDICE F), feito através de uma ferramenta do *MySQL Workbench*. E assim se concluiu o desenvolvimento de todos os diagramas do projeto.

No sítio, foi implementada a última ferramenta fundamental que foram os *logs* dos usuários. Todas as telas que os usuários veem e as ações que eles tomam pelo sítio (seja se *logar*, criar, editar ou cancelar eventos e até bloquear e desbloquear outros usuários) agora são totalmente guardadas com data, arquivo de *script* usado, *IP*, navegador, sistema operacional e sucesso (se deu certo ou não).

Inicialmente, os *logs* servem para serem apenas um histórico a ser consultado pelos administradores como forma de ter ciência sobre as atividades dos usuários no sítio. Futuramente os *logs* podem ser usados para mineração de dados visando um melhor investimento da aplicação.

### 3.2.20 Sprint 19

Nessa *sprint*, foram ajustados os erros de insuficiência encontrados no projeto. Entre eles, a falta de alguns gráficos e tabelas (*WBS* e *Gantt*) neste documento, falta de referências no capítulo 2 e, por fim, alguns gráficos na *dashboard*.

Além disso, diversas pequenas correções neste documento também foram feitas com a intenção de aperfeiçoamento das informações e a forma como são apresentadas.

A *sprint* 20 foi desconsiderada, pois teria apenas 1 dia de duração. O desenvolvimento dela foi incluído junto da *sprint* 19.

### 3.3 DESENVOLVIMENTO DA APLICAÇÃO

Esse capítulo descreve como cada tecnologia selecionada foi utilizada no decorrer do projeto e em sua estrutura.

#### 3.3.1 Apache

A versão utilizada foi a 2.4.29. Nenhuma configuração foi feita no servidor, ele se manteve da mesma forma como foi instalado, porém foram utilizados arquivos “.htaccess” para alterar as rotas do servidor.

O *.htaccess* é um arquivo usado para configurar o controle do acesso dos usuários aos arquivos do servidor. Podendo proibir e permitir o acesso, também como redirecionar o acesso dos arquivos de uma pasta para os arquivos de outra pasta.

No projeto ele foi utilizado para que, toda vez que uma *URL* inválida fosse acessada, ele rodasse o arquivo “index.php” que, por sua vez, agia conforme explicado no *framework* no capítulo 3.3.3 *PHP*.

#### 3.3.2 MySQL

A versão utilizada foi a 14.14 distribuição 5.7.26. A estrutura escolhida para o banco foi o *InnoDB* também de versão 5.7.26.

A única configuração diferente que precisou ser feita, foi alterar o *sql-mode* para que pudesse aceitar agrupamentos por *GROUP BY* sem considerar todas as chaves de todas as tabelas envolvidas na *SQL*.

Para calcular a distância entre os endereços dos usuários e os endereços dos eventos, foi utilizado um campo do tipo *POINT* para guardar as coordenadas dos endereços. O campo *point* funciona como um *BLOB* feito para guardar duas coordenadas (x e y), o uso de um campo *POINT* para cálculos geométricos espaciais torna mais prático a escrita das *SQLs*.

As imagens dinâmicas do sítio (tais como imagens de eventos, perfil de usuário e perfil de empresa) foram guardadas na tabela “arquivo”. Não foi utilizado um campo *BLOB* para guardá-las, apenas foram salvas cópias das imagens nas

pastas do servidor e o nome do arquivo junto de outras informações foram salvos nessa tabela.

### 3.3.3 PHP

A versão escolhida do *PHP* foi a 7.2 devido a ser a versão mais atualizada disponível para uso no início do projeto.

Foi implementado um pequeno *framework* caseiro a partir dessa ferramenta. O *framework* constrói muitas *SQLs* automaticamente, apenas algumas *SQLs* de busca mais complexas precisaram ser feitas manualmente.

Também são organizadas páginas de erro humanizadas caso algum erro aconteça no sítio. Para evitar as páginas normais dos navegadores e dos servidores, o sítio possui suas próprias telas de erros 404, 500, 401 e 403.

Foram construídas duas *Traits* que ajudaram no *framework*. Uma *Trait* chamada *CRUD* que auxiliava os controladores e a outra chamada *Model* que auxiliava os modelos. As *Traits* são ferramentas do *PHP* que trabalham como classes abstratas. O *PHP* não permite que uma classe estenda mais de uma classe, sendo assim cada classe só é possível possuir uma classe pai. Em alguns projetos isso pode causar um encadeamento de classes que prejudica a funcionalidade do projeto em si. Visando evitar esse problema, foram criadas as *Traits*, você não pode ter mais de uma classe pai, mas pode estender quantas *Traits* necessitar e também pode alterar o nome dos métodos de cada *Trait* para evitar incompatibilidades.

O *MVC* não foi inteiramente implementado no projeto, o *framework* foi desenvolvido com base na experiência da equipe que usava *frameworks* caseiros em seus antigos trabalhos e todos eram apenas inspirados no *MVC*, mas nunca eram exatamente um *MVC*.

Foram criados um diretório para os modelos, um diretório para os controladores e três diretórios para as visualizações. Os três diretórios de visualização são:

- *HTML*: O diretório padrão do sítio que retorna as telas assim como os usuários devem vê-las. Toda vez que alguém acessa alguma página, o *HTML* da página foi retirado desse diretório;
- *Ajax*: Esse diretório é usado em casos onde um formulário de uma página *HTML* é enviado. O modo normal de enviar o formulário atualizando a página

é anulado, então é enviado uma requisição *ajax* com todos os dados do formulário para o servidor. Todas essas requisições são processadas com os arquivos dessa página;

- Print: Devido a liberação de certificados que deveriam ser impressos pelos usuários, foi necessária a criação de uma página que o usuário pudesse visualizar e não trouxesse o *HTML* padrão com o *layout* da página. Sendo assim, esse diretório é usado apenas quando o usuário precisa imprimir o certificado de que participou de algum evento sem trazer o *layout* da página junto. Poderia ser usado para outros casos parecidos caso necessário.

O diretório é escolhido dependendo do formato da *URL* do sítio. Normalmente a *URL* vai parecer com: "[www.social.com/evento/evento-de-teste](http://www.social.com/evento/evento-de-teste)" essa página automaticamente irá ser processada em algum arquivo da pasta *HTML* e tentará retornar a página do evento de teste. Porém, se alterarmos a *URL* para "[www.social.com/evento.ajax/evento-de-teste](http://www.social.com/evento.ajax/evento-de-teste)" logo ela será processada no diretório *Ajax* e tentará retornar um *JSON* ao invés de uma página *HTML*. O sítio não foi feito completamente responsivo, portanto a *URL* não deverá retornar nada.

A classe *upload* feita pelo Colin Verot foi usada no projeto para facilitar o *upload* de imagens. A classe pode ser encontrada no sítio do Verot (<http://www.verot.net>), foi usada a versão de 2014 (não numerada). Com a classe, foi possível converter todas as imagens para *png*, salvá-las com um nome diferente em pastas devidamente organizadas para cada situação, redimensioná-las em tamanhos que se encaixam melhor e mais rápido no *layout* da página.

Para dar suporte aos *logs* do sistema, foi usada uma classe de leitura do *Browser* e Sistema Operacional do cliente conectado. A classe é feita pelo Chris Schuld, não é versionada e se chama "*Browser*".

Foi desenvolvida uma pequena classe de envio de E-mails chamada *Mail*, ela unicamente simplifica o envio de e-mails configurando automaticamente o *SMTP*. Para configurar todo o envio de e-mails de forma mais detalhada foram usadas duas classes pré prontas: *PHPMailer* v6.0.1 e *SMTP* v6.0.1, ambas produzidas para funcionarem conjuntas pelos programadores Marcus Bointon, Jim Jagielski, Andy Prevost e Brent R. Matzelle.

O *SMTP* usado foi pelo *SSL* do *Google*, através da conta "[social.angeltcc@gmail.com](mailto:social.angeltcc@gmail.com)".

O *PHP* necessitou de três pacotes instalados a parte para poder funcionar. O primeiro foi a pacote que permite a conexão com o banco de dados, conhecido como *PDO*, esse pacote não possui um versionamento, ele avança junto com o versionamento do *PHP*. O segundo foi um pacote gráfico, chamado de *GD*, usado pela classe de *upload* do *Verot*, a versão utilizada foi a 2.2.5. O último é um pacote de fazer requisições para *URLs* a partir do servidor, ele foi utilizada na integração com o *Google Maps*, se chama *cURL* e foi usado na versão 7.58.0.

Algumas funções do *PHP* foram automatizadas através de classes simples feitas a mão, entre elas a conexão com o banco de dados, as respostas do servidor e os dados da sessão.

### 3.3.4 Front-End

As tela foram compostas basicamente por *HTML5*, *CSS3* e *JS 1.8.5*. Todo o estilo do sítio foi baseado no *Material Design* do *Google*, através de um *framework* chamado *Materializecss*. O *template* da página inicial do sítio foi retirado de um dos exemplos abertos do *Materialize*. A versão usada do *Materialize* foi a 0.100.2, que era a última versão disponível no início do projeto.

Os formulários do sítio foram completamente estilizados por este *framework*. O *grid* das páginas de perfil e evento também foram todos feitos utilizando o *Materialize*. Alguns lugares que mostram imagens e tudo que envolve animações (com exceção do *loading*) também foram feitos usando exclusivamente este *framework*.

O *CSS* foi pouco estilizado. Apenas foram alteradas as cores e feitos ajustes pequenos de página em página. O uso mais complicado foi para tornar algumas páginas do sítio mais responsivas, por exemplo, nos botões de alterar imagem de perfil do usuário.

O *javascript* foi bastante usado. Todos os formulários do sítio foram convertidos para requisições *ajax*, visando uma usabilidade mais leve e com menos atualizações de páginas. Todas as respostas “não-html” do servidor possuem um mesmo modelo de retorno devido a classe de resposta que normaliza tudo e isso é completamente tratado no *javascript*.

Ajax, segundo as palavras de Juliano Niederauer em seu livro “Web Interativa com Ajax e PHP”, é “o uso sistemático de *JavaScript* e *XML* (entre outras



tecnologias) para tornar o navegador mais interativo com o usuário, utilizando-se solicitações assíncronas de informações”.

Para agilizar o desenvolvimento do *JavaScript*, foi usado a biblioteca *jQuery*, na versão 2.1.1 por causa da compatibilidade com o *Materialize*. Todo o *javascript* que foi usado estava se aproveitando dessa biblioteca. Inclusive *plugins* usados no *front end* foram feitos com base no *jQuery*. A versão da biblioteca é 2.1.1, não é a mais atualizada, mas foi a preferida por causa da compatibilidade com os *plugins* usados.

Foi desenvolvido um pequeno *plugin* de integração com a *API* do sítio do *viacep*. Esse *plugin* apenas ajudou a auto completar o endereço dos eventos e usuários e também foi um dado usado para calcular as coordenadas de cada endereço segundo o *Google*.

Um *plugin* mais complexo chamado *DataTable* foi usado. Esse *plugin* serve para automatizar listas de informações inseridas dentro de tabelas, criando ordenação, pesquisa, paginação e todo o pacote básico que essas listas costumam ter. A versão do *plugin* foi 1.10.16.

Outro *plugin* desenvolvido foi um que mostrava opções ao clicar em determinado lugar do sítio. Por exemplo, ao estar logado e clicar no nome de usuário mostra um pequeno menu com opções de ver empresas, eventos, perfil e fazer logout.

O *plugin* *mask.js* do Igor Escobar também foi utilizado na versão 1.14.15. Esse *plugin* foi usado nos formulários do sítio para manter uma máscara ao digitar algum campo. Por exemplo, quando se digita o *cep*, automaticamente é inserido um hífen no 6º dígito, este hífen é parte da máscara definida através desse *plugin*. Um adendo foi feito no *plugin* para integrá-lo automaticamente a vários campos de máscara simples. Utilizando o parâmetro *data* do *html5*, foi utilizado o “*data-mask*” para definir uma máscara e um código *javascript* roda a página toda vez que ela se inicializa procurando por campos de formulários que possuam o campo “*data-mask*” para colocar a máscara.

Outro *plugin* utilizado foi o *loadingView* por Ademola Aina, não versionado. Esse *plugin* simplesmente cria um efeito de carregando quando necessário. Ele foi usado ao enviar formulários para que os usuários não enviem os formulários duas vezes enquanto ainda estão carregando. Uma pequena função para inserir uma imagem *svg* que fica girando foi criada como incremento desse *plugin*.

Foi utilizado o *Sweet Alert* (também conhecido como *Swal*) na versão 7.33.1. Esse *plugin* simplesmente cria e mostra ao usuário pequenos *popups* com mensagens já prontas. Ele foi utilizado, por exemplo, quando o usuário vai deletar a foto de perfil e aparece um *popup* de confirmação.

O *plugin moment.js* foram adicionados ao projeto apenas por questões de compatibilidade e suporte a outros *plugins*. Também foi adicionado um *plugin* de “ligação” entre o *datatables.js* com o *moment.js* para que as tabelas pudessem ser organizadas pelas colunas com datas corretamente.

Por fim, o último *plugin* usado foi o *qrcodejs* por David Shim, este *plugin* não é versionado. Esse *plugin* foi unicamente utilizado nos certificados para gerar um código *QR* que pudesse validar os certificados.

Para deixar o *design* do sítio mais intuitivo, muitos *ícones* foram utilizados. Inicialmente os *ícones* foram retirados do *Materializecss*, porém percebeu-se que os mesmos não carregavam quando não havia conexão com a *Internet*. Posteriormente eles foram substituídos pro *ícones* retirados do repositório *Flaticon* e organizados pela aplicação *Icomoon*.

## 4 APRESENTAÇÃO DOS RESULTADOS

Nesse capítulo são apresentados os resultados finais do desenvolvimento na parte do sítio e na parte de administrador do sítio.

As duas sessões do sítio possuem apenas um tipo de interface para seus usuários quando logados, um administrador não usa sua conta para logar no cliente do sítio e vice versa. Porém, ambas as sessões do sítio mexem com as mesmas informações, apenas possuem permissões diferentes para mexer com as informações.

No administrador, é possível inserir e alterar informações de todos os usuários comuns do sítio. Porém na sessão do cliente somente é possível alterar suas informações e visualizar algumas informações de outros usuários. Usuários comuns somente tem contato com os administradores através da página de contato ou de denúncias.

A seguir, a estrutura do *framework* e o funcionamento do sítio de acordo com usuários não registrados, usuários registrados e usuários administradores.

### 4.1 FRAMEWORK

As classes do *framework* são divididas em 3 pastas: *Controller*, *Model* e *Traits*. As *traits* procuram simplificar todas as questões que devem ser tratadas em todos os *controllers* e *models* do sistema, enquanto os *controllers* e *models* herdam os métodos das *traits* e lidam com outros métodos mais específicos para seus casos.

O APÊNDICE G ilustra um exemplo entre o controlador de eventos e a *trait CRUD*, que tem a função de transformar os modelos em *SQL* e buscar os modelos do banco de dados. Ilustra também a relação do modelo de eventos com a *trait Model*, que carrega os métodos básicos do modelo, organizando suas informações para que possa passá-las automaticamente para os controladores.

Os métodos específicos listados no *EventController* buscam listas de eventos que são procuradas em mais de uma tela diferente do sistema ou que exigem uma *SQL* mais complexa que a equipe preferiu fazer na mão do que adaptar o *framework* para montá-las automaticamente, devido ao tempo disponível.

O método “*getUserEventsForMonth*” retira uma lista de eventos em um determinado mês que seja o mês do *timestamp* passado no primeiro argumento, “*getRecommendedEvents*” busca a lista de eventos recomendados para o usuário que aparece na tela inicial do cliente e “*smartSearch*” é a busca de eventos pela pesquisa do sítio, que usa apenas uma condição a mais que o método citado anteriormente.

Os métodos específicos do *EventModel* geram e calculam informações de um modelo específico, ou buscam algumas associações que podem ser puxadas em mais de um lugar dentro do sítio.

O método “*printEndereco*” transforma o endereço do evento em uma *String* já toda formata conforme a necessidade do caso. Já o método “*getUrl*” retorna sempre a mesma *URL* do evento que é usada nos navegadores (se ajusta sozinho caso o *host* do site mude), é um método bem comum entre os modelos. O método “*calculateCoordinates*” faz a integração com o *Google Maps* e retorna um objeto *Point* com as coordenadas do endereço daquele evento segundo o *Google* para que possa ser salvo no banco de dados. O último método, “*getPublications*”, é o único que não monta ou calcula alguma informação, ele apenas busca as publicações dos eventos.

A diferença entre o método “*getPublications*” do modelo e os métodos do controlador é que o *getPublications* sempre vai trabalhar para um evento em específico, enquanto os métodos do controlador trabalham com vários eventos de uma vez. Assim é escolhido onde vai ficar cada método.

## 4.2 USUÁRIO NÃO REGISTRADO OU LOGADO

Esse usuário não pode visualizar outros usuários, porém pode pesquisar por eventos e empresas assim como visualizá-los, sem poder seguir e fazer denúncias ou comentários.

A maior diferença está na página principal, que é totalmente diferente após estar logado. Assim como mostra a 10.

**FIGURA 10 - Tela Inicial Quando não Logado**

**S**    Pesquisar...    **Q**    Login

# SOCIAL

Todos os eventos da sua cidade num único lugar

**CADESTRE-SE**

**Diversidade**

Todos os tipos de eventos. Desde shows de música, passando por exposições de arte até palestras de negócios. Encontre o seu lugar e a sua turma onde quiser.

**Facilidade**

Encontre eventos na sua região em pouquíssimos cliques, veja todos os eventos que você tem interesse e os eventos de seus amigos.

**Compromisso**

Seja reconhecido pelos seus esforços na participação de eventos através de certificados que podem ser disponibilizados e validados através da plataforma.

Cadastre-se agora e comece a divulgar seus eventos de todos os tipos ainda hoje

## Contato

Deseja nos dar ideias? Relatar algum problema? Elogiar? Xingar? Entre em contato conosco a qualquer momento.

**ENTRAR EM CONTATO**

**Company Bio**

Este é um projeto desenvolvido para o trabalho de conclusão de curso de Tecnologia em Análise e Desenvolvimento de Sistemas feito por Angel Mauricio Sanches Pereira em 2019.

Layout template made by Materialize  
Icons made by Freepik, Dave Gandy, IcoMoon, Eleonor Wang, Pixel perfect, Gregor Cresnar, Pause08, Google, Those Icons, Yannick from www.flaticon.com is licensed by CC 3.0 BY

**Equipe**

Designer: Angel Pereira  
Analista: Angel Pereira  
Programador: Angel Pereira  
Banco de Dados: Angel Pereira

**Links**

Contato

FONTE: O Autor, 2019

A tela de seguir (FIGURA 11) é a segunda tela que só usuários não logados tem acesso, a tela de cadastro.

**FIGURA 11 - Tela de Cadastro**

FONTE: O Autor, 2019

A FIGURA 12 abaixo representa a tela de login, a tela pela qual os usuários irão inserir seus dados básicos para se autenticar na sessão de clientes do sítio.

**FIGURA 12 - Tela de Login**

FONTE: O Autor, 2019

A última tela deste subcapítulo (FIGURA 13) é uma tela que quem está logado consegue ter acesso, porém ela é a mesma. A única mudança é que os

dados de nome e e-mail já vêm preenchidos automaticamente quando se está logado.

**FIGURA 13 - Tela de Contato**

**S**    Pesquisar...    🔍    Login

---

## Contato

Seu Nome

---

Email

---

Conte-nos o que deseja

---

**FINALIZAR**

---

**Company Bio**

Este é um projeto desenvolvido para o trabalho de conclusão de curso de Tecnologia em Análise e Desenvolvimento de Sistemas feito por Angel Mauricio Sanches Pereira em 2019.

Layout template made by Materialize  
Icons made by Freepik, Dave Gandy, IcoMoon, Eleonor Wang, Pixel perfect, Gregor Cresnar, Pause08, Google, Those Icons, Yannick from www.flaticon.com is licensed by CC 3.0 BY

**Equipe**

Designer: Angel Pereira  
Analista: Angel Pereira  
Programador: Angel Pereira  
Banco de Dados: Angel Pereira

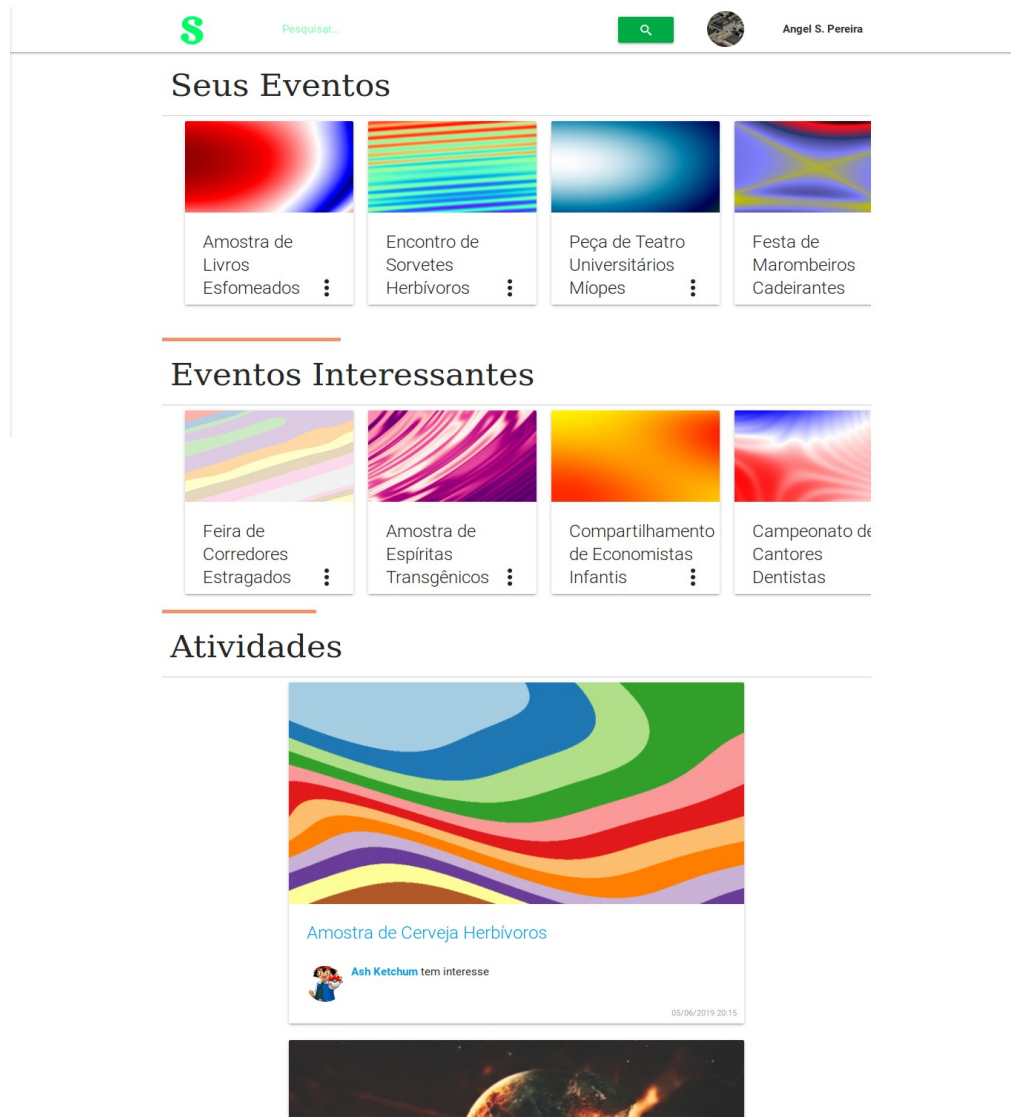
**Links**

Contato

FONTE: O Autor, 2019

### 4.3 USUÁRIOS REGISTRADOS E LOGADOS

A primeira tela que o usuário é apresentado após se logar é chamada de tela inicial e apresenta as informações mais relevantes que as pessoas procuram no sítio. Seus próximos eventos, uma lista de sugestões de eventos interessantes e uma lista de atividades de usuários e empresas que o usuário esteja seguindo e de eventos que marcou presença positiva. Conforme apresenta a FIGURA 14.

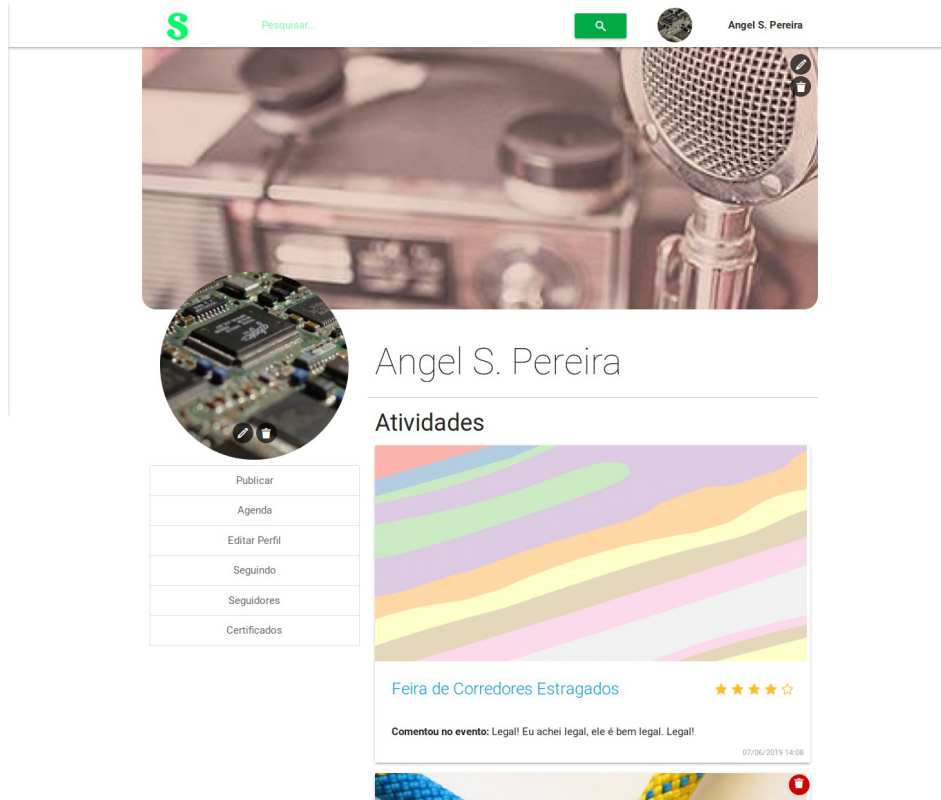
**FIGURA 14 - Tela Inicial Quando Logado**

FONTE: O Autor, 2019

Para gerenciar seu próprio perfil, foram adicionadas diversas opções, tanto com o intuito de editar suas informações quanto visualizar suas ações no sítio. Essas opções foram mesclados a tela de perfil do próprio usuário. Conforme mostram as FIGURAS 15, 16, 17, 18, 19 e 20 abaixo.

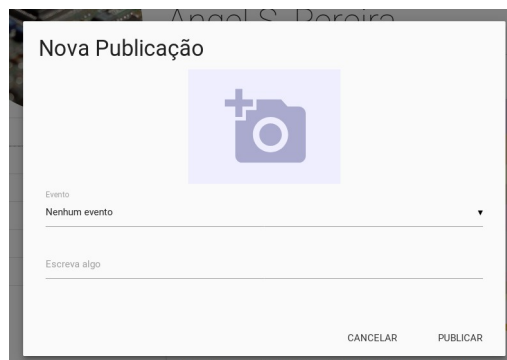


**FIGURA 15 - Tela de Perfil**



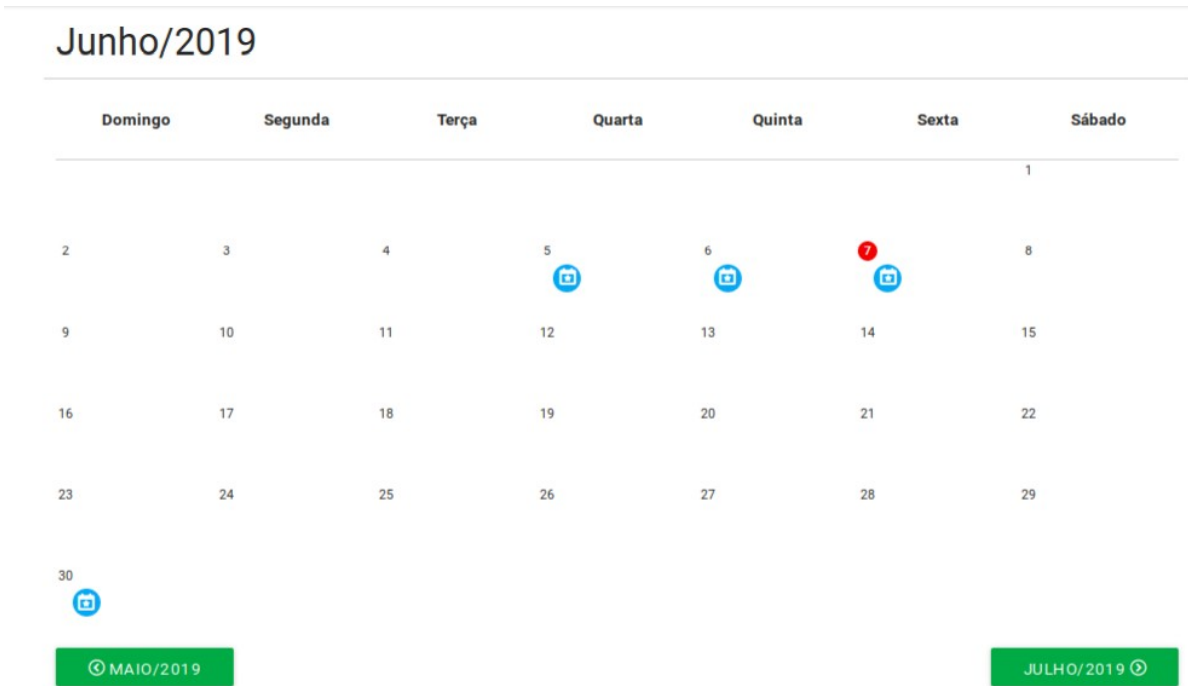
FONTE: O Autor, 2019

**FIGURA 16 - Publicação**



FONTE: O Autor, 2019

FIGURA 17 - Agenda



FONTE: O Autor, 2019

FIGURA 18 - Formulário de Dados da Conta

## Dados Cadastrais

Nome  
Angel S. Pereira

Seus dados não serão mostrados para ninguém. Apenas usaremos para pesquisar por eventos mais adequados a você.

Sexo  
Masculino

Data de Nascimento  
30/05/1994

CEP  
81260-282

UF  
Paraná

Cidade  
Curitiba

Bairro  
Cidade Industrial

Rua  
Rua Lodovico Kaminski

Número  
3330

Email  
angelmou.sp@gmail.com

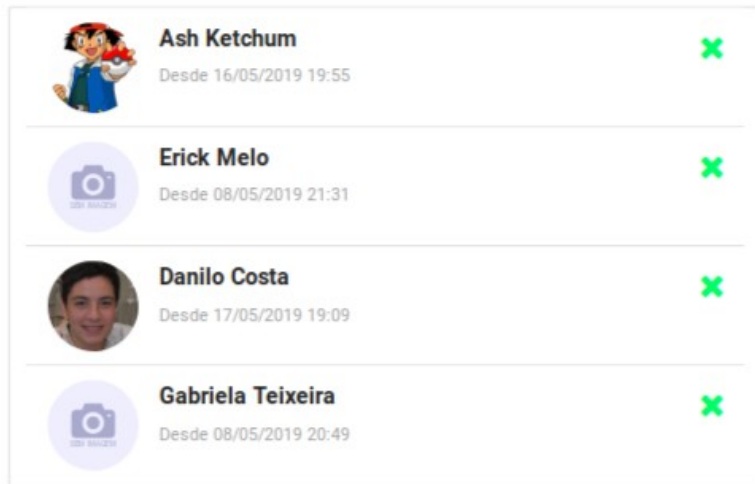
Senha  
...

ALTERAR DADOS

FONTE: O Autor, 2019

**FIGURA 19** - Lista de Usuários que Está Seguindo/Lista de Seguidores

# Seguindo



FONTE: O Autor, 2019

**FIGURA 20** - Lista de Certificados

Certificados de Angel S. Pereira

## Atividades



FONTE: O Autor, 2019

As empresas de um mesmo usuário foram inicialmente todas reunidas em uma lista gerenciável e com filtros que podem ser acessadas pela opção do menu lateral (conforme a FIGURA 21) e o resultado pode ser visto na FIGURA 22.

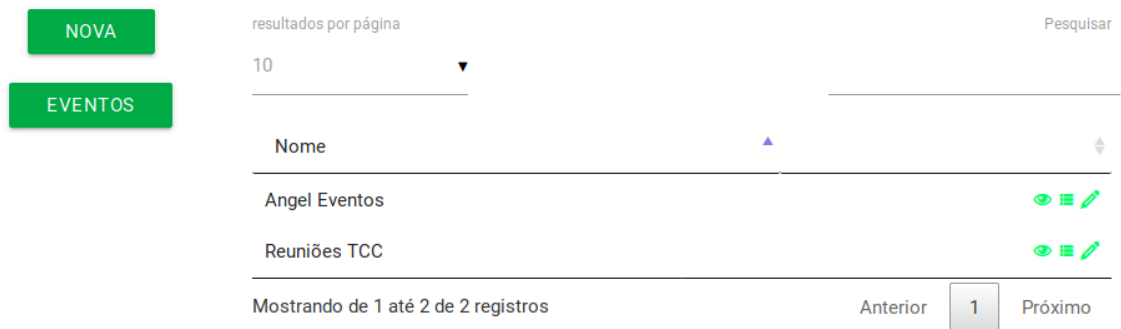
**FIGURA 21 - Menu**



FONTE: O Autor, 2019

**FIGURA 22 - Lista de Empresas**

## Suas Empresas



FONTE: O Autor, 2019

Assim como no caso do gerenciamento dos dados da conta, as funções das empresas foram mescladas ao perfil da empresa. Todas podem ser vistas no perfil da empresa (FIGURA 24). O formulário de dados (FIGURA 23) é diferente e possui uma lista de eventos da empresa no final, porém a tela de publicação é idêntica a tela de publicação do perfil (16).

**FIGURA 23 - Formulário de Empresas**

The screenshot shows a web form for registering a company. At the top, there is a search bar with a magnifying glass icon and the text 'Pesquisar...'. To the right of the search bar is a profile picture and the name 'Angel S. Pereira'. Below this, the form is titled 'Empresa Angel Eventos'. The form fields include:
 

- Nome:** Angel Eventos
- Email:** angelmau.sp@gmail.com
- CNPJ:** 71.227.163/0001-50
- Site:** https://www.google.com/
- Facebook:** https://www.facebook.com/Ezequiel33.11
- Mercado Livre:** https://www.mercadolivre.com.br/perfil/ESTRELA10\_?brandId=1582

 Below the form fields, there is a green 'SALVAR' button. Underneath, there is a section for 'resultados por página' with a dropdown menu set to '10'. A table lists four events:
 

Nome	
Assistir Vingadores	
Campeonato de Desenhistas Mutantes	
Evento Teste Admi	
Protesto de Guitarristas Surfistas	

 At the bottom of the table, it says 'Mostrando de 1 até 4 de 4 registros' and 'Anterior 1 Próximo'.

FONTE: O Autor, 2019

**FIGURA 24 - Perfil da Empresa**

The screenshot shows the profile page for 'Angel Eventos'. On the left, there is a circular profile picture of a blue and yellow rope knot. Below the profile picture are two buttons: 'Publicar' and 'Editar'. Underneath these buttons are social media sharing icons for Facebook, Twitter, YouTube, Spotify, Instagram, and Pinterest. Below the social media icons are two WhatsApp contact buttons, both showing the phone number '+55 (41) 9 9757-1749'. On the right side of the profile, there is a large, dark, abstract image. Below this image is the heading 'Atividades' and a horizontal strip of images showing a landscape with trees and a building, with a red trash can icon in the top right corner.

FONTE: O Autor, 2019

Os eventos também são listados assim como as empresas na 22. O formulário de eventos foi o que apresentou maior complexidade. Ele possui uma parte comum (FIGURA 25), uma parte para gerenciar as imagens do evento

(FIGURA 26) e por fim uma parte para definir o perfil do público-alvo (FIGURA 27) com uma previsão de quantas pessoas se encaixam naquele perfil naquela cidade e quantas pessoas tem em cada estado brasileiro diferente que se encaixam naquele perfil (FIGURA 28).

**FIGURA 25 - Formulário de Eventos - Dados**

## Dados do Evento

Empresa  
Reuniões TCC

---

Título  
Desatraso das Sprints

---

Data de Início	Hora de Início
29/04/2019	06:30

---

Data de Término	Hora de Término
29/04/2019	10:00

FONTE: O Autor, 2019

**FIGURA 26 - Formulário de Eventos - Imagens**

## Imagens

Imagens


		
Título _____	Título _____	Título _____
Descrição _____ _____	Descrição _____ _____	Descrição _____ _____
✘	✘	✘



FONTE: O Autor, 2019

**FIGURA 27 - Formulário de Eventos – Perfil do Público-Alvo**

## Perfil de Público-alvo

 Os dados aqui não definem que o seu evento não será mostrado para pessoas que não se encaixem nesses requisitos, apenas que será mostrado preferencialmente para quem se encaixa nessas informações

Categorias  
Cultural, Exposição, Gastronômico, Outros, Recreativo ▼

---

Para Homens

Não  Sim

Para Mulheres

Não  Sim

Idade Mínima


1

Idade Máxima

120

FONTE: O Autor, 2019

**FIGURA 28 - Formulário de Eventos – Previsão do Alcance do Evento**

 Isso é uma previsão segundo nosso sistema de quantos usuários se encaixam nessa descrição na cidade do evento e em cada estado do Brasil

Curitiba	32
AC	34
AL	16
AM	8
AP	1

FONTE: O Autor, 2019

O perfil do evento é dividido em algumas sessões.

A primeira contém as imagens do evento e as informações básicas dele (FIGURA 29), onde também foi inserido alguns ícones de ações (denunciar, participar, convidar, publicar e cancelar evento).

A segunda contém a descrição e as publicações feitas pelo evento (FIGURA 30). Após isso, um campo para mostrar, anonimamente, as publicações que os usuários fizeram mencionando o evento (FIGURA 31).

A terceira e última parte são os comentários, onde primeiramente é mostrada a média dos comentários (caso o evento possua mais de 3 avaliações), depois os usuários dos outros (FIGURA 33) e, por fim, um espaço para você inserir sua própria avaliação (FIGURA 32).

**FIGURA 29** - Perfil de Evento – Imagens e Informações

The image shows a screenshot of an event profile page. At the top left, there is a red button labeled 'SET' and the number '30'. The event title 'Amostra de Artistas Assaltantes' is prominently displayed in the center. To the right of the title are several icons: a green exclamation mark, a green house, a green person, a green megaphone, and a red 'X'. Below the title, the event is listed as 'Feito por Reuniões TCC' with a price of 'R\$ 63,43'. The dates are 'De: 30/09/2019' and 'Até: 30/09/2019', and the time is 'Horário: 05:30 - 18:00'. The location is 'Local: Curitiba, PR - Cidade Industrial Rua Lodovico Kaminski, 105'. The main image area contains a dark, textured background with the text 'Image 3' and 'Descrição Imagem 3' overlaid.

SET  
30

# Amostra de Artistas Assaltantes

Feito por [Reuniões TCC](#) R\$ 63,43

De: 30/09/2019 Horário: 05:30 - 18:00

Até: 30/09/2019 Local: Curitiba, PR - Cidade Industrial Rua Lodovico Kaminski, 105

Image 3  
Descrição Imagem 3

FONTE: O Autor, 2019

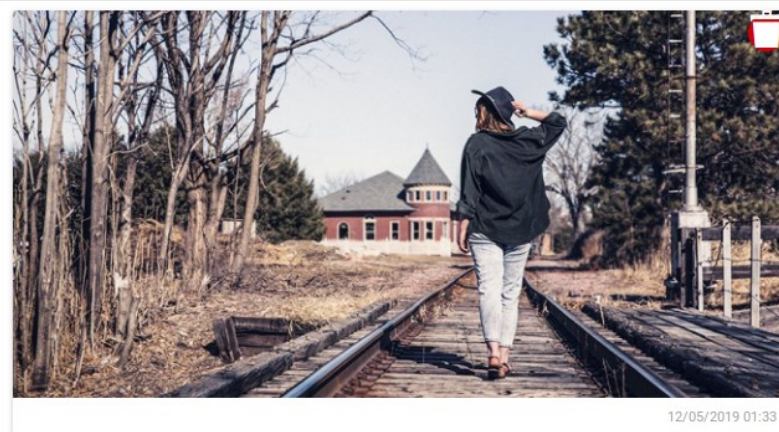


## FIGURA 30 - Perfil de Evento – Descrição e Publicações

É importante questionar o quanto a execução dos pontos do programa causa impacto indireto na reavaliação de alternativas às soluções ortodoxas. O empenho em analisar a complexidade dos estudos efetuados cumpre um papel essencial na formulação dos métodos utilizados na avaliação de resultados. A certificação de metodologias que nos auxiliam a lidar com a mobilidade dos capitais internacionais assume importantes posições no estabelecimento do remanejamento dos quadros funcionais. Gostaria de enfatizar que a estrutura atual da organização nos obriga à análise das posturas dos órgãos dirigentes com relação às suas atribuições. Todavia, o novo modelo estrutural aqui preconizado faz parte de um processo de gerenciamento da gestão inovadora da qual fazemos parte.

Acima de tudo, é fundamental ressaltar que a hegemonia do ambiente político estende o alcance e a importância das novas proposições. Nunca é demais lembrar o peso e o significado destes problemas, uma vez que a expansão dos mercados mundiais promove a alavancagem das regras de conduta normativas. Caros amigos, o consenso sobre a necessidade de qualificação obstaculiza a apreciação da importância das direções preferenciais no sentido do progresso. Do mesmo modo, a consolidação das estruturas oferece uma interessante oportunidade para verificação dos índices pretendidos. O incentivo ao avanço tecnológico, assim como a crescente influência da mídia exige a precisão e a definição do fluxo de informações.

### Publicações



FONTE: O Autor, 2019

**FIGURA 31 - Perfil de Eventos - Menções**

## Menções



Loasda nndakj ghab eibas ifdsij sdjfbbsidfbs dfnbdj sdf skbdjf bsd kjbsdkjf skdbfsbkdf bskjb  
fbsdkjb abojnojdoa io da

2019-05-12 01:01:45



FONTE: O Autor, 2019

**FIGURA 32 - Perfil de Evento - Comentar**

### Seu Comentário



Angel S. Pereira



Lorem ipsum dolor sit amet sunshine...

AVALIAR

FONTE: O Autor, 2019

**FIGURA 33 - Perfil de Evento – Outros Comentários**

## Comentários

3.8



Média de 3.8 baseado em 4 avaliações



**Janaina Pinto**



Incrível, só precisa de mais gente.

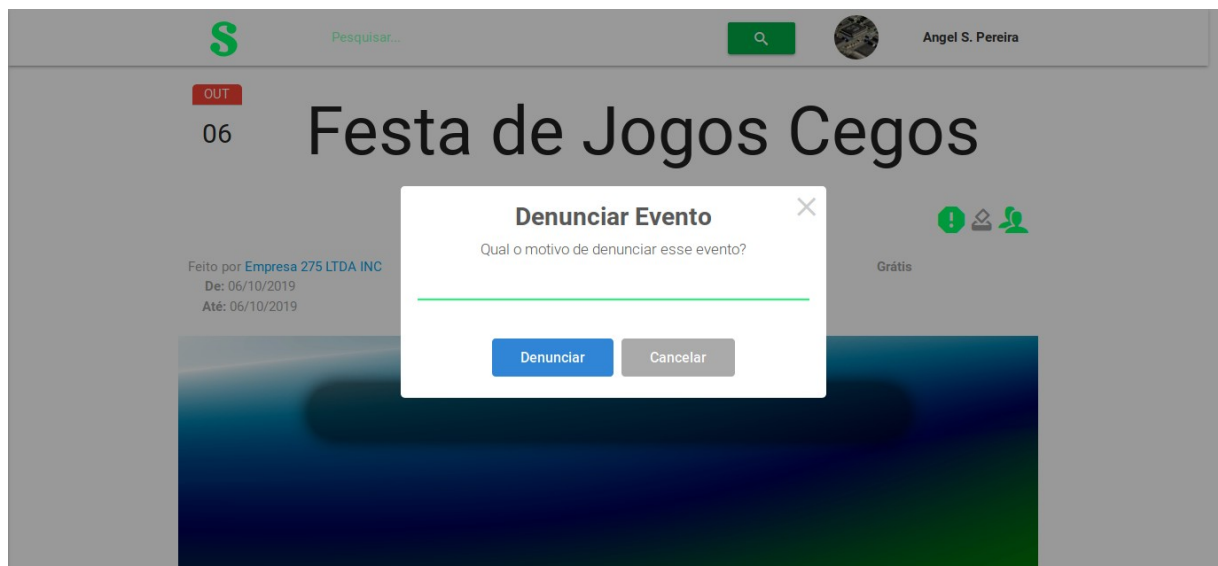


Gostei muito, quero ir de novo.

FONTE: O Autor, 2019

O sítio ainda possui uma opção de denúncia dividida nos perfis de usuário, empresa e eventos. Todos com o mesmo layout, assim como na FIGURA 34.

**FIGURA 34 - Denúncia**



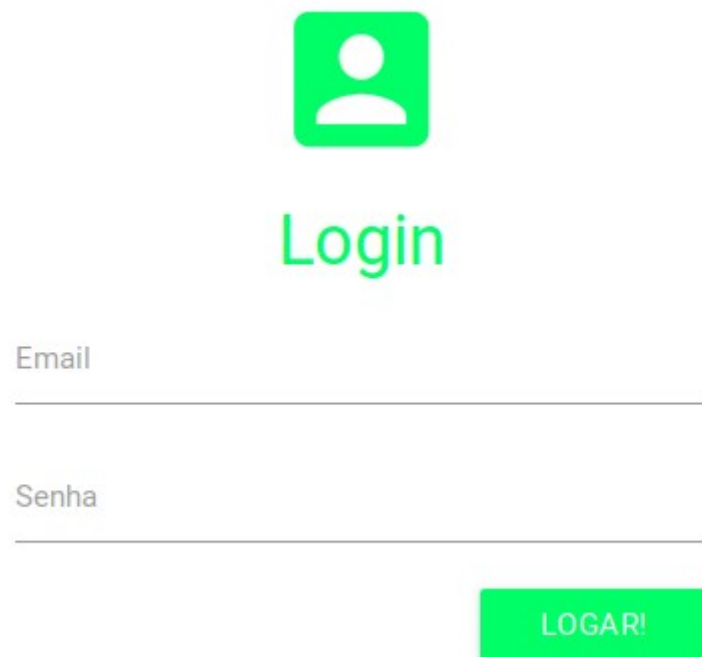
FONTE: O Autor, 2019

#### 4.4 ADMINISTRADOR

A segunda parte do sítio é a parte de onde os administradores gerenciam todos os registros de usuários, empresas, eventos, denúncias, usuários administradores e categorias de eventos do sítio inteiro.

A tela a seguir (FIGURA 35) mostra a tela de login dos usuários administradores. Caso não esteja logado no sistema como administrador, tudo que fizer irá resultar nessa tela de login.

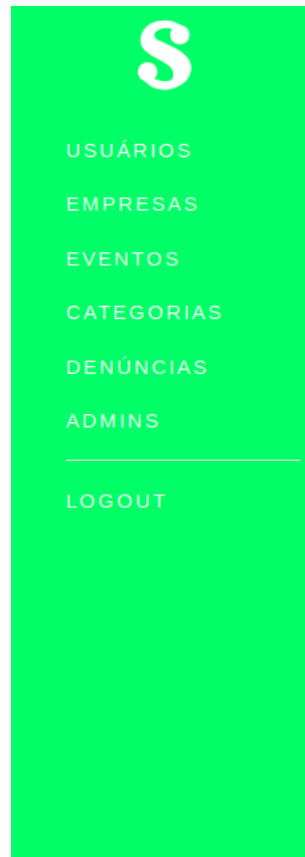
**FIGURA 35 - Login**



A tela de login para administradores apresenta um ícone de perfil de usuário em um quadrado verde na parte superior. Abaixo do ícone, o texto "Login" é exibido em uma fonte verde. O formulário contém dois campos de entrada: "Email" e "Senha", cada um com uma linha de texto cinza para o rótulo e uma linha horizontal para o campo de entrada. Na parte inferior direita do formulário, há um botão verde com o texto "LOGAR!" em branco.

FONTE: O Autor, 2019

A FIGURA 36 a seguir é o menu principal que aparece ao lado de todas as telas do administrador, contendo um *link* para cada módulo do administrador.

**FIGURA 36 - Menu**

FONTE: O Autor, 2019

A primeira tela a ser mostrada é a *Dashboard*, cuja função é mostrar informações relevantes sobre o andamento do sítio. Estão configuradas 4 listas que se atualizam a cada 2 segundos: Eventos Bombando (mostra os próximos eventos com maior número de inscritos nos sistema), Próximos Eventos (mostra os próximos eventos que irão acontecer), Novos Eventos (os últimos eventos cadastrados) e Novos Usuários (últimos usuários que se cadastraram). Junto a cada registro, um link para sua página de estatísticas.

FIGURA 37 - Dashboard Administrador

# Dashboard

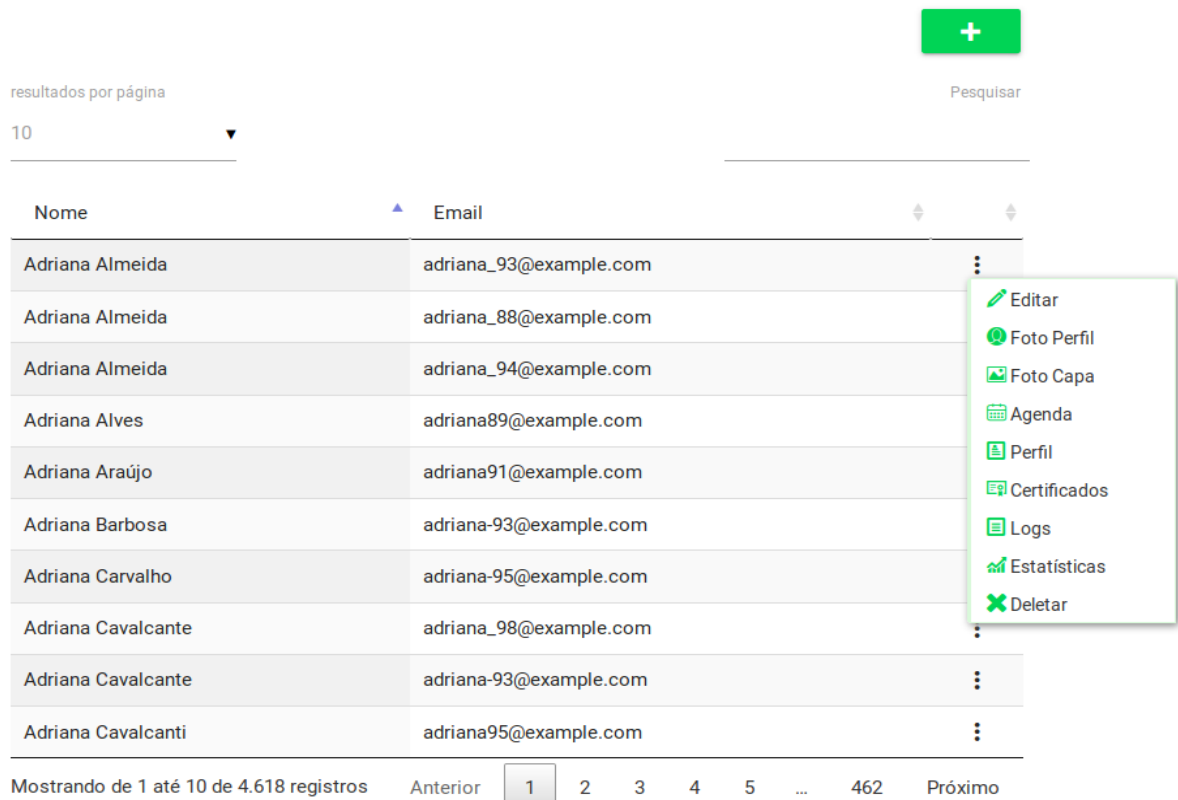
EVENTOS BOMBANDO	NOVOS EVENTOS	PRÓXIMOS EVENTOS	NOVOS USUÁRIOS
Nome		Interessados	Data
<a href="#">Seminário de Genéricos Góticos</a>		3	25/12/2019 00:00 
<a href="#">Amostra de Cerveja Herbívoros</a>		2	27/09/2019 00:00 
<a href="#">Encontro de Sorvetes Herbívoros</a>		1	24/07/2019 00:00 

FONTE: O Autor, 2019

O primeiro módulo é responsável por gerenciar os usuários comuns do sistema (não administradores). Inicialmente ele mostra uma lista paginada, filtrável e ordenável com os dados básicos dos usuários e uma opção para mostrar as ações que podem ser tomadas, conforme a FIGURA 38.

FIGURA 38 - Lista de Usuários

# Usuários



resultados por página

10

Pesquisar

Nome	Email	
Adriana Almeida	adriana_93@example.com	⋮
Adriana Almeida	adriana_88@example.com	⋮
Adriana Almeida	adriana_94@example.com	⋮
Adriana Alves	adriana89@example.com	⋮
Adriana Araújo	adriana91@example.com	⋮
Adriana Barbosa	adriana-93@example.com	⋮
Adriana Carvalho	adriana-95@example.com	⋮
Adriana Cavalcante	adriana_98@example.com	⋮
Adriana Cavalcante	adriana-93@example.com	⋮
Adriana Cavalcanti	adriana95@example.com	⋮

- Editar
- Foto Perfil
- Foto Capa
- Agenda
- Perfil
- Certificados
- Logs
- Estatísticas
- Deletar

Mostrando de 1 até 10 de 4.618 registros

Anterior 1 2 3 4 5 ... 462 Próximo

FONTE: O Autor, 2019

Existem quatro ações que servem para a edição dos usuários. A mais óbvia, que é a primeira chamada “Editar” (FIGURA 39), as duas seguintes que alteram as imagens de perfil (FIGURA 40) e de capa (FIGURA 41) que aparecem no perfil do usuário (FIGURA 15) e a última que serve para desativar o usuário na plataforma.

FIGURA 39 - Formulário Usuário

# Usuário 1

Nome  
Angel S. Pereira

---

Sexo Data de Nascimento  
 Masculino 30 / 05 / 1994 ✕

---

CEP  
81260-282

---

UF Cidade Bairro  
 Paraná Curitiba Cidade Industrial

---

Rua Número  
 Rua Lodovico Kaminski 3330 ⬆️⬇️

---

Email  
angelmau.sp@gmail.com

---

Senha  
...

---

**SALVAR**

FONTE: O Autor, 2019

FIGURA 40 - Imagem de Perfil

# Usuário 1

**PERFIL**

---

**SALVAR**

FONTE: O Autor, 2019



FIGURA 41 - Imagem de Capa

# Usuário 1



FONTE: O Autor, 2019

Os outros campos são para visualizar informações do usuário. Eles mostram informações que podem ser visualizadas no cliente do sítio (15, 17 e 20), mas duas opções mostram listas que só se encontram no administrador: Estatísticas que mostra listas de seguidores, bloqueios, denúncias, interesses, entre outras coisas que cada usuário tem e *Logs* que mostra um histórico de ações que o usuário teve no sítio (seja de edição de imagem, adicionar eventos, comentar ou até visualizar telas) conforme mostram respectivamente as figuras 42 e 43.

FIGURA 42 - Estatísticas do Usuário

## Angel S. Pereira



### Empresas

Angel Eventos	
Reuniões TCC	

### Seguidores

Evelyn Fernandez	
Rafaela Silva	

FONTE: O Autor, 2019

FIGURA 43 - Logs do Usuário

# Logs de Angel S. Pereira

resultados por página Pesquisar

10 ▼

IP	Browser	Plataforma	Ação	Arquivo
127.0.0.1	Firefox 67.0	Linux	view_screen	/var/www/html/tcc/html/evento/read.php
<b>Data</b> 07/06/2019 20:12:56				
127.0.0.1	Firefox 67.0	Linux	view_screen	/var/www/html/tcc/html/evento/read.php
127.0.0.1	Firefox 67.0	Linux	view_screen	/var/www/html/tcc/html/perfil.php
127.0.0.1	Firefox 67.0	Linux	view_screen	/var/www/html/tcc/html/dados_cadastrois.php

FONTE: O Autor, 2019

Os *logs* inicialmente servirão como histórico para que os administradores possam acompanhar as atividades dos usuários pelo sítio, ajudá-los em casos de suporte a erros e até verificar melhor detalhes de denúncias. Futuramente os *logs* poderão ser usados para mineração de dados para descobrir quais áreas do sítio podem ser mais aprimoradas e quais estão sendo mais usadas.

Outros módulos como empresas, eventos, categorias e *admins* (administradores) são versões mais simplificadas do módulo de usuários. Todos os módulos possuem o mesmo tipo de listagem, cada um com suas informações mais relevantes. O que mais se altera entre os módulos são as ações. Segue as listagens dos módulos nas FIGURAs 44, 45, 46 e 47.

FIGURA 44 - Lista de Empresas

# Empresas

resultados por página

10

Pesquisar

Nome	Email	CNPJ
Almeida Eventos	contato@almeidaeventos.com	00.087.005/0001-01
Angel Eventos	angelmau.sp@gmail.com	71.227.163/0001-50
Empresa 0 LTDA INC	empresa0@exemplo.com	40020784000134
Empresa 0 LTDA INC	empresa0@exemplo.com	14420325000138
Empresa 0 LTDA INC	empresa0@exemplo.com	76391548000171

- Editar
- Foto Perfil
- Perfil
- Estatísticas
- Deletar

FONTE: O Autor, 2019

FIGURA 45 - Lista de Eventos

# Eventos

resultados por página

10

Pesquisar

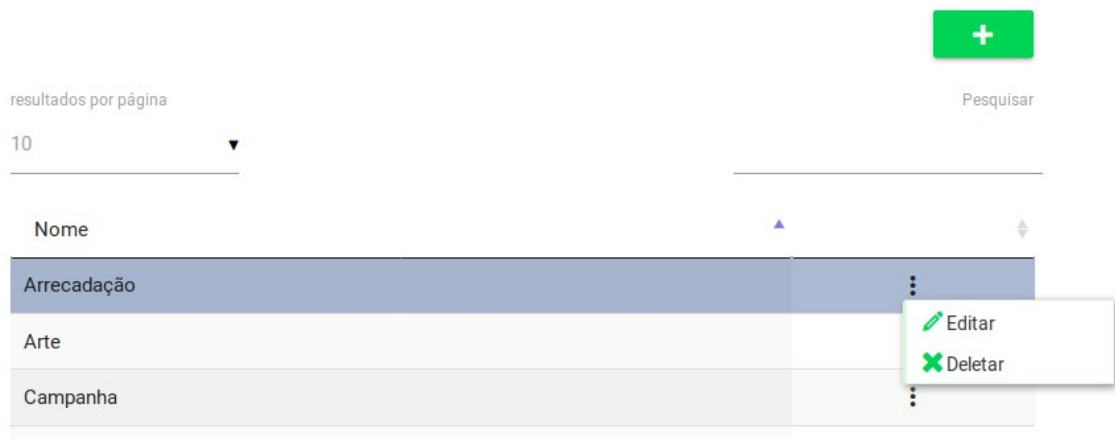
Nome	Companhia	Usuário	Data
Amostra de Advogados Piratas	Empresa 90 LTDA INC	Camila Almeida	02/02/2019
Amostra de Arqueiros Banguelas	Empresa 51 LTDA INC	Rafael Carvalho	05/06/2019
Amostra de Artistas Assaltantes	Reuniões TCC	Angel S. Pereira	30/09/2019
Amostra de Atiradores Alienigenas	Empresa 159 LTDA INC	Joseane Melo	21/02/2019
Amostra de Atiradores Estragados	Empresa 139 LTDA INC	Carlos Schmidt	11/05/2019

- Editar
- Imagens
- Perfil
- Estatísticas
- Deletar

FONTE: O Autor, 2019

FIGURA 46 - Lista de Categorias

# Categorias



FONTE: O Autor, 2019

FIGURA 47 - Lista de Administradores

# Admins



FONTE: O Autor, 2019

O módulo de administradores possui também uma opção de *logs* que funciona assim como os *logs* dos usuários, porém mostrando o histórico dos administradores conforme mostra a FIGURA 48.

FIGURA 48 - Logs dos Administradores

# Admins

Nome	Email	
Admin	angel@social.com	
Angel Sanches	angel2@social.com	

FONTE: O Autor, 2019

O único módulo que apresenta divergências em seu funcionamento é o módulo de denúncias, que necessitou ser diferente devido as suas ações serem também muito diferentes dos outros módulos.

O módulo de denúncias possui três abas, de usuários, empresas e eventos. As denúncias não podem ser deletadas, elas podem apenas ser aceitas ou rejeitadas, contendo opções para rejeitar denúncias em massa na listagem normal das denúncias (FIGURA 49). Por fim, as denúncias também possuem uma tela de detalhes, conforme a FIGURA 50, caso o administrador decida visualizar com mais calma o conteúdo da denúncia ou até adicionar detalhes a ela.

FIGURA 49 - Lista de Denúncias

# Denúncias

ID	Denunciado	Denunciante	Status	
1	Freeza	Angel S. Pereira	Rejeitada	

Mostrando de 1 até 1 de 1 registros

Anterior 1 Próximo

FONTE: O Autor, 2019

FIGURA 50 - Detalhes da Denúncia

# Denúncia

 A denúncia foi rejeitada

 Denunciante: Angel S. Pereira

 Denunciado: Freeza

## Motivo

Ele é mal e destruiu alguns planetas

Anotações

---

 ACEITAR

 REJEITAR

FONTE: O Autor, 2019

## 5 CONSIDERAÇÕES FINAIS

Este capítulo visa apresentar a conclusão final do projeto: O objetivo que foi alcançado, o objetivo que não foi alcançado, as alterações das expectativas iniciais e as recomendações para trabalhos futuros.

Num geral, os objetivos iniciais do projeto foram pouco alcançados. Os pontos fracos das aplicações concorrentes foram bem explorados, abrindo mais interação entre usuários procuradores e novas ferramentas inteligentes aos organizadores.

Porém, a tentativa de fazer os usuários interagirem mais dentro da plataforma a transformou numa espécie de rede social, o que necessitou de muitas regrinhas e relações simples que, juntas, tornam todo o trabalho mais complicado.

Devido ao excesso de regras, o pequeno *framework* apresentou dificuldades, o que tornou o código algumas vezes cansativo e foi preferível contornar as ferramentas do *framework* para que algumas funções funcionassem corretamente.

Além disso, há uma grande abertura de regras e ações novas que poderiam ser implementadas para aproximar ainda mais os usuários procuradores entre si e as empresas, porém a complexidade do projeto seria ainda maior e o tempo foi escasso para essas questões.

Outro problema que acabou incomodando a equipe é que a ferramenta se torna quase completamente inútil sem a presença de um aplicativo para aparelhos inteligentes (tais como celulares e televisões) para acompanhá-la e tornar seu uso realmente prático.

Por outro lado, pontos positivos também foram avistados. Como citado anteriormente, os pontos fracos das outras ferramentas concorrentes foram bem explorados. Acreditamos que, com o devido investimento, essa ideia poderia se tornar uma das grandes redes sociais da atualidade, já que todos os problemas aqui citados são facilmente contornado apenas com mais tempo de produção.

### 5.1 RECOMENDAÇÕES PARA TRABALHOS FUTUROS

Para trabalhos futuros, o ideal, e mais urgente, seria criar uma interação móvel com a plataforma. Aplicativos que rodem em aparelhos inteligentes, no mínimo os aparelhos Android, e antedam com muito mais comodidade as

necessidades dos usuários num geral. Usar um aplicativo também possibilitaria o uso de notificações que dão uma força incrível à popularização da plataforma.

Outra recomendação em cima do que já foi dito anteriormente seria a adição de diversas outras interações na aplicação que a façam uma rede social mais completa. Comentários com respostas, bloquear empresas e eventos, empresas podendo responder comentários, convites sendo enviados no nome de empresas, entre outras tantas ações que podem ser imaginadas que seriam de grande adição ao sistema e as interações empresa-cliente.

Outra funcionalidade interessante para o sistema, seria utilizar gráficos para instruir graficamente os usuários administradores. Atualmente, apenas tabelas de tamanhos limitados são usadas para informar os usuários, ficando algo graficamente chato de se olhar, mas ressalto que ainda assim é muito informativo. Apenas gostaria de que algo mais intuitivo fosse colocado para informar alguns status do sistema.

Uma recomendação, de algo que gostaria muito de ter feito, foi colocar uma integração mais pesada com o *Google Maps*. Indicar rotas até os eventos para os usuários, colocar uma mapa nos eventos, nos formulários de eventos e de usuários. Toda a integração se resumiu a funções que não são visíveis aos usuários e, apesar de não estar ruim, poderia estar melhor. Acabou não sendo desenvolvido, pois design não era o foco do projeto, no lugar dele ficou o cálculo das coordenadas para poder procurar os eventos mais próximos.

Outra adição de extremo interesse seria uma integração com alguma plataforma de pagamento (e.g. *Pagseguro*) para que seja possível a compra e venda de ingressos pela plataforma. Inicialmente foi descartado, pois todo o projeto seria desenvolvido em um *host* local, e não existe como integrar plataformas de pagamento com esse tipo de *host*.

Ao pesquisar referências nos dias finais deste TCC, foi encontrada uma opção que o *Facebook* oferece de grande interesse. Se chama *Facebook Graph*, com ela é possível cadastrar eventos na conta de uma pessoa através de uma *API* sem necessariamente entrar no sítio do *Facebook* para cadastrá-lo como da maneira tradicional. Acho possível que algumas redes sociais possuam essas *APIs* que permitem a criação de publicações que podem ajudar os eventos a serem divulgados automaticamente em várias redes sociais.

Enquanto avaliava as ferramentas concorrentes, percebi que elas focavam muito em apenas ter gente para cadastrar os eventos. Percebi depois que elas



mostravam algumas propagandas da própria plataforma em outros lugares, até no *youtube*. O que eu pensei foi em criar um sistema onde o evento possa ser anunciado em diversos lugares diferentes. Mostrar propagandas dele pelo *Google Ads*, mostrar um pequeno *banner* em diversos sítios diferentes, ou qualquer tipo de propaganda que não seja da plataforma em si, mas dos eventos dela que pagaram para isso, dando aos anunciadores uma ferramenta que não foi encontrada nem mesmo no *Facebook*.

Um incremento que não foi pensado durante o desenvolvimento do projeto, apenas após a finalização foi acrescentar o conceito de “gameificação” no sítio. Basicamente, tentar incrementar obtenção de pontos, títulos ou conquistas e fazer com que os usuários queiram competir entre si (simplesmente comparar as conquistas e/ou pontos deles com outros usuários que ele segue em um *ranking* já é o suficiente).

Por fim, três pequenas ideias que foram consideradas durante o desenvolvimento foi adicionar notificações de navegador ao projeto, pedir a localização atual do usuário na hora de cadastrar e pesquisar por eventos (ou “roubar” essa informação pelo *IP* dele sem notificá-lo, como foi notado em uma das ferramentas concorrentes) e permitir a criação de eventos que ocorreram online (como lives, sorteios, jogatinas, entro outros).

## REFERÊNCIAS

- ALLEN, Johnny. O'TOOLE, Willian. MCDONNELL, Ian. HARRIS, Robert. **Organização e Gestão de Eventos**. 3ª Edição. São Paulo: DTPPhoenix Editorial, 2008.
- BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **Uml - Guia do Usuário** - Tradução da 2ª Edição. Campus, 2006.
- CAMPOS, Luiz Cláudio, WYSE, Nely e ARAÚJO, Maria Luiza da Silva. **Eventos: Oportunidade de novos negócios**. Rio de Janeiro, RJ: Senac Nacional, 2000.
- CARDOSO, Danilo Pessoa. **As Redes Sociais Como Ferramentas de Divulgação de Eventos**. 10 Folhas. Artigo – Fasci-Tech (Periódico Eletrônico da FATEC) - São Caetano do Sul, 2011. Disponível em: <http://fatecsaocaetano.edu.br/fascitech/index.php/fascitech/article/viewFile/45/44>. Acesso em 19 de Maio de 2019.
- CHAFFER, Jonathan. SWEDBERG, Karl. **Learning jQuery** – 4ª Edição. Packt, 2013.
- CHANGEVISION. Disponível em: <http://astah.net/>. Acesso em 18 de junho de 2019.
- DALL'OGGIO, Pablo. **PHP: Programando Com Orientação a Objetos**. 1ª Edição. São Paulo: Novatec, 2007.
- DISK INGRESSOS. Disponível em: <https://www.diskingressos.com.br/>. Acesso 15 de junho de 2019.
- DOITY. Disponível em: <https://doity.com.br/>. Acesso em 15 de junho de 2019.
- EVENTBRITE. Disponível em: <https://www.eventbrite.com.br>. Acesso em 15 de junho de 2019.
- FACEBOOK. Disponível em: <https://newsroom.fb.com/>. Acesso em 15 de junho de 2019.
- FOWLER, Martin. SCOTT, Kendall. **UML Essencial. Um Breve Guia Para a Linguagem-Padrão de Modelagem Para Objetos**. 2ª Edição. Porto Alegre: Bookman, 2000.
- GIACAGLIA, Maria Cecília. **Eventos: Como Criar, Estruturar e Captar Recursos**. 1ª Edição. São Paulo: Cengage Learning, 2011.
- GIACAGLIA, Maria Cecília. **Organização teoria e prática de eventos**. São Paulo: Pioneira Thomson, 2013.
- GOOGLE. Disponível em: <https://material.io/design/>. Acesso em 15 de junho de 2019.
- KABIR, Mohammed J., **Apache 2 Server, a Bíblia**. Rio de Janeiro: Campus, 2002.
- LARMAN, Craig. **Utilizando UML e Padrões - Uma Introdução à análise e ao projeto orientado a objetos**. Porto Alegre: Bookman, 2007.

LEMAY, Laura. COLBURN, Rafe. KYRNIN, Jennifer. **HTML, CSS & JavaScript Web Publishing in One Hour a Day, Sams Teach Yourself**. 7ª Edição. Sams Publishing, 2016.

MARCELO, Antonio. **Apache: Configurando o servidor WEB para Linux**. 3ª Edição. Rio de Janeiro: Brasport, 2005.

MATERIALIZA. Disponível em: <https://materializecss.com/>. Acesso em 15 de junho de 2019.

MATIAS, Marlene. **Organização de eventos – procedimentos e técnicas**. 6ª Edição. São Paulo: Manole, 2013.

MICROSOFT. Disponível em: <https://visualstudio.microsoft.com/>. Acesso em 17 de junho de 2019.

MILANI, André. **MySQL: Guia do Programador**. 1ª Edição. São Paulo: Novatec, 2006.

THE DOCUMENT FOUNDATION. Disponível em: <https://pt-br.libreoffice.org/>. Acesso em 18 de junho de 2019.

MYSQL™. Disponível em: <https://www.mysql.com/>. Acesso em 18 de junho de 2019.

NETCRAFT. Disponível em: <https://news.netcraft.com/>. Acesso em 17 de junho de 2019.

NIEDERAUER, Juliano. **Web Interativa com Ajax e PHP**. 1ª Edição. São Paulo: Novatec Editora, 2007.

NIEDERAUER, Juliano. PRATES, Rubens. **MySQL 5: Guia de Consulta Rápida**. 1ª Edição. São Paulo: Novatec, 2006.

OUTGO. Disponível em: <https://www.outgo.com.br/>. Acesso em 15 de junho de 2019.

ROCKET CONTENT. **Social Media Trends 2019**. Artigo – Rocket Contents – Brasil, 2019. Disponível em: <https://cdn2.hubspot.net/hubfs/355484/Social%20Media%20Trends%202019.pdf> acesso em 17 de junho de 2019.

SCHWABER, Ken - **Agile software development with Scrum**: 005.1 S398 2002.

SUTHERLAND, Jeff. **Scrum: A arte de fazer o dobro do trabalho pela metade do tempo**/Jeff Sutherland, J. J. Sutherland; tradução de Nina Lua. Rio de Janeiro: Sextante, 2019.

SYMPLA. Disponível em: <https://www.sympla.com.br/>. Acesso em 15 de junho de 2019.

THE AGILE MANIFESTO. **The Agile Manifest**. Disponível em: <http://agilemanifesto.org/iso/ptbr/manifesto.html>. Acesso em 02 de julho de 2019.

THE JQUERY FOUNDATION. Disponível em: <https://jquery.com/>. Acesso em 15 de junho de 2019.

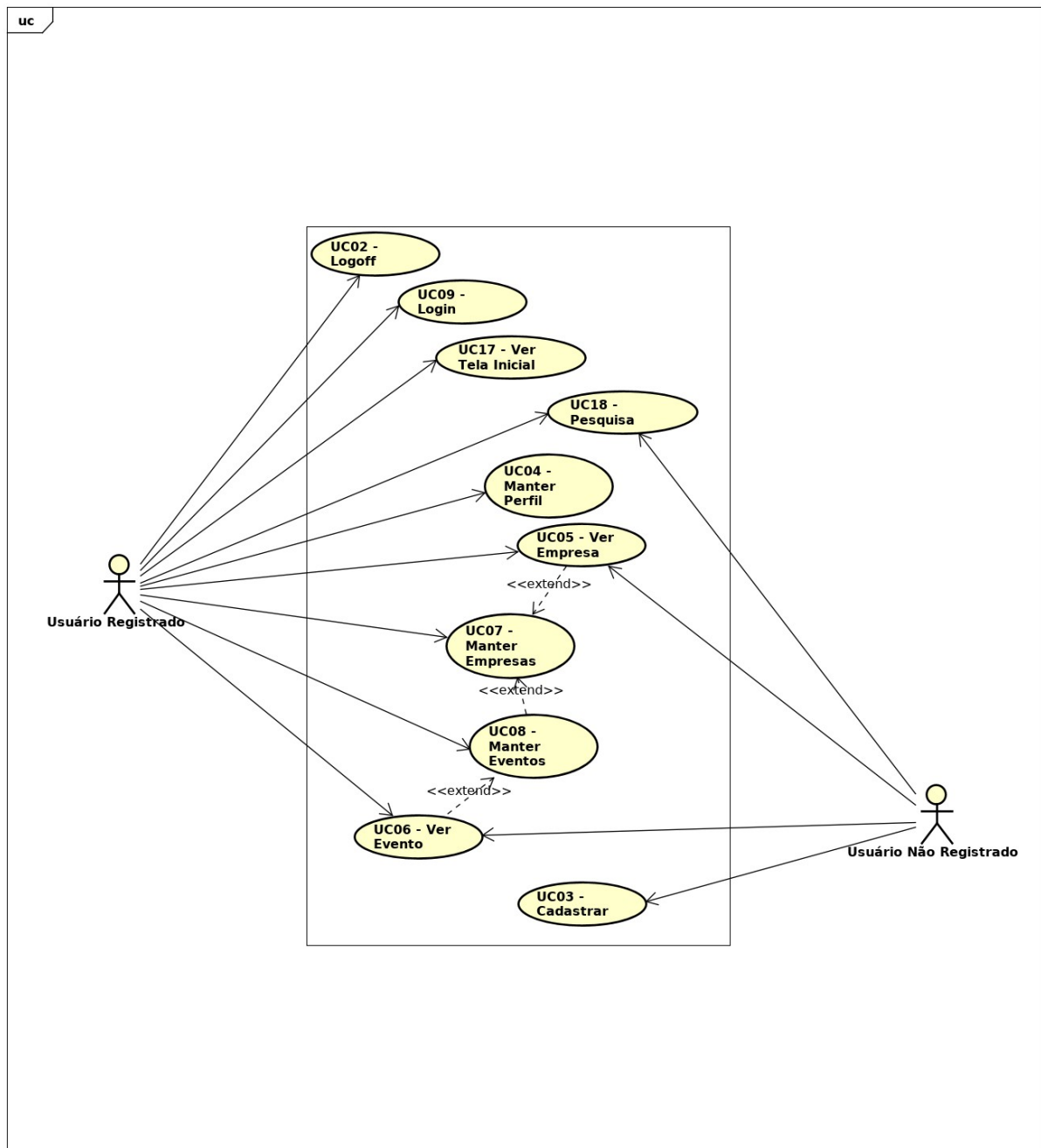
UNITED STATES SECURITIES AND EXCHANGE COMMISSION. **FORM Q-10**. Relatório – Facebook, Inc – Washington, D.C., 2012. Disponível em:

<https://www.sec.gov/Archives/edgar/data/1326801/000119312512325997/d371464d10q.htm#toc>. Acesso em 15 de junho de 2019.

WORD TOURISM & TRAVEL COUNCIL. **Benchmark Reports 2017 - Brazil**. 7 Folhas. Artigo – Word Tourism & Travel Council – Londres, 2017. Disponível em: <https://www.wttc.org/-/media/files/reports/benchmark-reports/country-reports-2017/brazil.pdf>. Acesso em 3 de Junho de 2019.

XMIND. Disponível em: <https://www.xmind.net/>. Acesso em 14 de junho de 2019.

## APÊNDICE A – DIAGRAMA DE CASOS DE USO DO SÍTIO



### UC01 - Login

Objetivo: Permite que um usuário faça seu *login* no sítio

Atores: Usuário Registrado

Pré-condições: Não estar *logado*

Pós-condições: Usuário se *loga* no sítio

Fluxo Principal:

1. O sistema apresenta a tela de *login*

2. O usuário preenche as informações e clica no botão de *logar* (A1)
3. Sistema verifica as informações e mostra a mensagem de *login* realizado (E1)
4. Sistema registra a tentativa de *login* nos *logs* de usuários
5. Usuário é redirecionado para a tela inicial depois de um segundo e meio
6. Caso de uso encerrado

#### Fluxos Alternativos:

##### A1:

1. Usuário preenche o e-mail
2. Usuário clica em "Esqueci Senha"
3. Sistema gera uma nova senha e envia para o e-mail caso exista
4. Sistema mostra uma mensagem de sucesso
5. Sistema registra a mudança de senha nos *logs* de usuários
6. Caso de uso encerrado

#### Fluxos de Exceção:

##### E1:

1. O usuário insere uma combinação de E-mail e Senha que não estão cadastradas em usuários ativos do sistema
2. O sistema verifica as informações e percebe que não encontrou nada
3. O sistema mostra a mensagem dizendo que o usuário não foi encontrado
4. Caso de uso encerrado

## UC02 - Logoff

Objetivo: Permite que um usuário saia de seu login no sítio

Atores: Usuário Registrado

Pré-condições: Estar *logado*

Pós-condições: Usuário não está mais *logado* no sistema

#### Fluxo Principal:

1. O usuário clica em *deslogar*

2. O sistema *desloga* o usuário
3. O sistema mostra uma mensagem dizendo que o *logoff* aconteceu com sucesso
4. Sistema registra o *logoff* nos *logs* de usuários
5. A página do usuário é atualizada após um segundo e meio
6. Caso de uso encerrado

### UC03 - Cadastrar

Objetivo: Permite que um usuário crie uma conta no sítio

Atores: Usuário Não Registrado

Pré-condições: O usuário clica na opção de Cadastrar

Pós-condições: O usuário tem uma conta no sítio, o usuário está logado no sítio

Fluxo Principal:

1. O sistema mostra o formulário
2. O usuário preenche os dados
3. O usuário clica em "Cadastrar" (E1, E2)
4. O sistema cadastra o usuário
5. Sistema cria uma sessão com os dados de login no sítio
6. Sistema registra o cadastro nos *logs* de usuários
7. Caso de uso encerrado

Fluxos de Exceção:

E1:

1. O usuário não preenche algum dos dados
2. O usuário clica em "Cadastrar"
3. O sistema mostra a mensagem dizendo que o usuário precisa preencher todos os dados
4. Caso de uso encerrado

E2:

1. O usuário tenta se cadastrar com um e-mail que já existe
2. O sistema verifica que já existe um usuário não apagado registrado com aquele e-mail

3. O sistema mostra uma mensagem ao usuário dizendo que o e-mail já está sendo usado
4. Caso de uso encerrado

#### UC04 - Manter Perfil

Objetivo: Permite que um usuário gerencie seus dados e veja dados de outros usuários

Atores: Usuário Registrado

Fluxo Principal:

1. O sistema mostra as informações do usuário
2. O usuário clica em "Seguir" (A1, A2, A3, A4, A5, A6, A7, A8, A9, A10, A11, A12, A13, A14, A15)
3. O sistema adiciona o usuário dono do perfil na lista de seguidos do usuário atual
4. Sistema registra a ação de seguir o usuário nos *logs* de usuários
5. Sistema mostra uma mensagem de sucesso
6. Caso de uso encerrado

Fluxos Alternativos:

A1:

1. O usuário clica em "Editar Perfil"
2. O sistema mostra um formulário
3. Usuário clica em "Enviar"
4. Sistema registra a edição nos *logs* de usuários
5. Sistema mostra uma mensagem de sucesso para o usuário
6. Caso de uso encerrado

A2:

1. O usuário clica em "Deixar de Seguir"
2. O sistema retira o usuário dono do perfil na lista de seguidos do usuário atual



3. Sistema registra a ação de deixar de seguir o usuário nos *logs* de usuários
4. Sistema mostra uma mensagem de sucesso
5. Caso de uso encerrado

A3:

1. O usuário clica em “Bloquear”
2. O sistema adiciona o usuário dono do perfil na lista de bloqueados do usuário atual
3. Sistema registra o bloqueio nos *logs* de usuários
4. Sistema mostra uma mensagem de sucesso
5. Caso de uso encerrado

A4:

1. O usuário clica em “Desbloquear”
2. O sistema retira o usuário dono do perfil na lista de bloqueados do usuário atual
3. Sistema registra a tentativa de login nos *logs* de usuários
4. Sistema mostra uma mensagem de sucesso
5. Caso de uso encerrado

A5:

1. O usuário clica em “Denunciar”
2. O sistema mostra um formulário
3. Usuário preenche as informações
4. Usuário clica em "Enviar"
5. Sistema salva a denúncia para ser avaliada pelos administradores mais tarde
6. Sistema registra a ação de denunciar nos *logs* de usuários
7. Caso de uso encerrado

A6:

1. O usuário clica em “Publicar”
2. O sistema mostra um formulário

3. O usuário insere uma imagem, opcionalmente insere um evento e um texto curto
4. Usuário clica em “Publicar” (E1, E2)
5. O sistema cria uma publicação do usuário com a imagem e as informações inseridas
6. Sistema registra a ação de publicar nos *logs* de usuários
7. Caso de uso encerrado

A7:

1. O usuário clica em “Seguindo”
2. O sistema mostra uma tela com todas as pessoas que o usuário logado está seguindo
3. Sistema registra a visualização da tela nos *logs* de usuários
4. Caso de uso encerrado

A8:

1. O usuário clica em “Seguidores”
2. O sistema mostra uma tela com todas as pessoas que estão seguindo o usuário logado
3. Sistema registra a visualização da tela nos *logs* de usuários
4. Caso de uso encerrado

A9:

1. O usuário clica em “Certificados”
2. O sistema mostra uma tela com todos os certificados que aquele usuário tem
3. O usuário clica em “Visualizar”
4. O sistema carrega uma tela mostrando o certificado da pessoa com um código *QR* para validar o certificado
5. Sistema registra a visualização do certificado nos *logs* de usuários
6. Caso de uso encerrado

A10:

1. O usuário clica em “Certificados”

2. O sistema mostra uma tela com todos os certificados que aquele usuário tem
3. O usuário clica em “Imprimir”
4. O sistema carrega a tela mostrando o certificado da pessoa e automaticamente abre a janela para imprimir o certificado
5. Sistema registra a visualização da tela nos *logs* de usuários
6. Caso de uso encerrado

A11:

1. Usuário clica no botão de deletar na própria publicação
2. Sistema pergunta se usuário tem certeza que deseja apagar aquela publicação
3. Usuário clica em “Confirmar”
4. Sistema deleta a publicação com sua imagem
5. Sistema registra a deleção da publicação nos *logs* de usuários
6. Caso de uso encerrado

A12:

1. Usuário clica em “Editar Imagem de Perfil”
2. Sistema pede para usuário escolher uma imagem
3. Usuário seleciona uma imagem (E2)
4. Sistema carrega a visualização da nova imagem
5. Sistema mostra botão de salvar
6. Usuário clica em “Salvar”
7. Sistema deleta a imagem antiga (caso haja)
8. Sistema salva nova imagem de perfil
9. Sistema registra a edição nos *logs* de usuários
10. Caso de uso encerrado

A13:

1. Usuário clica em “Deletar Imagem de Perfil”
2. Sistema mostra uma mensagem de confirmação
3. Usuário clica em “Confirmar”
4. Sistema apaga a foto de perfil
5. Sistema registra a deleção da imagem nos *logs* de usuários

## 6. Caso de uso encerrado

### A14:

1. Usuário clica em “Editar Imagem de Capa”
2. Sistema pede para usuário escolher uma imagem
3. Usuário seleciona uma imagem (E2)
4. Sistema carrega a visualização da nova imagem
5. Sistema mostra botão de salvar
6. Usuário clica em “Salvar”
7. Sistema deleta a imagem antiga (caso haja)
8. Sistema salva nova imagem de capa
9. Sistema registra a edição nos *logs* de usuários
10. Caso de uso encerrado

### A15:

1. Usuário clica em “Deletar Imagem de Capa”
2. Sistema mostra uma mensagem de confirmação
3. Usuário clica em “Confirmar”
4. Sistema apaga a foto de capa
5. Sistema registra a deleção nos *logs* de usuários
6. Caso de uso encerrado

## Fluxos de Exceção:

### E1:

1. Usuário não coloca uma imagem e clica em “Publicar”
2. Sistema mostra um aviso pedindo para o usuário escolher uma imagem
3. Caso de uso encerrado

### E2:

1. Usuário escolhe uma imagem com mais de 2MB de tamanho
2. Sistema mostra um aviso pedindo para o usuário escolher uma imagem menor que 2MB
3. Caso de uso encerrado

## UC05 - Ver Empresa

Objetivo: Permite que um usuário veja os dados da empresa no sítio

Atores: Usuário Registrado ou Usuário Não Registrado

## Fluxo Principal:

1. O sistema mostra as informações da empresa
2. O usuário clica em “Seguir” (A1, A2, A3, A4)
3. Sistema adiciona a empresa na lista de empresas seguidas do usuário atual
4. Sistema registra a ação de seguir empresa nos *logs* de usuários
5. Sistema mostra uma mensagem de sucesso
6. Caso de uso encerrado

## Fluxos Alternativos:

## A1:

1. O usuário clica em “Denunciar”
2. O sistema mostra um formulário
3. Usuário clica em "Enviar"
4. Sistema salva a denúncia para ser avaliada pelos administradores mais tarde
5. Sistema registra a ação de denunciar empresa nos *logs* de usuários
6. Sistema mostra mensagem de sucesso
7. Sistema esconde botão de denúncia para aquele usuário nessa empresa
8. Caso de uso encerrado

## A2:

1. Usuário clica em “Deixar de Seguir”
2. Sistema retira a empresa na lista de empresas seguidas do usuário atual
3. Sistema registra a ação de deixar de seguir empresa nos *logs* de usuários
4. Sistema mostra uma mensagem de sucesso
5. Caso de uso encerrado

A3:

1. Sistema carrega um formulário, com uma lista de eventos da empresa
2. Usuário clica em 'Salvar' (E3)
3. Sistema salva a empresa
4. Sistema registra a ação de editar empresa nos *logs* de usuários
5. Sistema mostra uma mensagem de sucesso
6. Sistema mostra a lista de empresas novamente
7. Caso de uso encerrado

A4:

1. Usuário clica em "Publicar"
2. Sistema mostra um formulário
3. Usuário insere uma imagem, opcionalmente insere um evento e um texto curto
4. Usuário clica em "Publicar" (E1, E2)
5. Sistema cria uma publicação do usuário com a imagem e as informações inseridas
6. Sistema registra a ação de publicar pela empresa nos *logs* de usuários
7. Caso de uso encerrado

A5:

1. Usuário clica no botão de deletar na própria publicação
2. Sistema pergunta se usuário tem certeza que deseja apagar aquela publicação
3. Usuário clica em "Confirmar"
4. Sistema deleta a publicação e sua imagem
5. Sistema registra a ação de deletar a publicação da empresa nos *logs* de usuários
6. Caso de uso encerrado

A6:

1. Usuário clica em "Seguir"
2. Sistema insere a empresa na lista de empresas seguidas do usuário atual

3. Sistema registra a ação de seguir empresa nos *logs* de usuários
4. Sistema mostra uma mensagem de sucesso
5. Caso de uso encerrado

A7:

1. Usuário clica em “Editar Imagem de Perfil”
2. Sistema pede para usuário escolher uma imagem
3. Usuário seleciona uma imagem (E2)
4. Sistema carrega a visualização da nova imagem
5. Sistema mostra botão de salvar
6. Usuário clica em “Salvar”
7. Sistema deleta a imagem antiga (caso haja)
8. Sistema salva nova imagem de perfil da empresa
9. Sistema registra a ação de editar empresa nos *logs* de usuários
10. Sistema mostra uma mensagem dizendo que imagem de capa foi apagada
11. Caso de uso encerrado

A8:

1. Usuário clica em “Deletar Imagem de Perfil”
2. Sistema mostra uma mensagem de confirmação
3. Usuário clica em “Confirmar”
4. Sistema apaga a foto de perfil da empresa
5. Sistema registra a ação de editar empresa nos *logs* de usuários
6. Sistema mostra mensagem de foto de perfil deletada
7. Caso de uso encerrado

Fluxos de Exceção:

E1:

1. Usuário não coloca uma imagem e clica em “Publicar”
2. Sistema mostra um aviso pedindo para o usuário escolher uma imagem
3. Caso de uso encerrado

E2:

1. Usuário escolhe uma imagem com mais de 2MB de tamanho
2. Sistema mostra um aviso pedindo para o usuário escolher uma imagem menor que 2MB
3. Caso de uso encerrado

E3:

1. O usuário preenche o campo CNPJ com um documento inválido
2. O sistema apaga o campo e avisa que o CNPJ é inválido
3. Caso de uso encerrado

#### UC06 - Ver Evento

Objetivo: Permite que um usuário veja os dados do evento no sítio

Atores: Usuário Registrado ou Usuário Não Registrado

Pós-condições: Sistema mostra as informações do evento para que o usuário se informe e interaja de acordo com suas preferências

Fluxo Principal:

1. O sistema mostra as informações do evento com espaço para comentário
2. Usuário escolhe uma nota para o evento de 1 a 5
3. Usuário escreve um comentário
4. O usuário clica em "Comentar" (A1, A2, A3)
5. Sistema salva o comentário
6. Sistema registra a ação de comentar evento nos *logs* de usuários
7. Sistema mostra a mensagem de sucesso
8. Sistema atualizar a página com os campos do comentário já preenchidos automaticamente prontos para serem editados
9. Caso de uso encerrado

Fluxos Alternativos:

A1:

1. O usuário clica em "Denunciar"
2. O sistema mostra formulário
3. Usuário preenche os dados e clica em "Enviar"



4. Sistema salva a denúncia para ser avaliada pelos administradores mais tarde
5. Sistema registra a ação de denunciar evento nos *logs* de usuários
6. Sistema esconde botão de denúncia para esse usuário nesse evento
7. Caso de uso encerrado

A2:

1. O usuário clica em “Participar”
2. O sistema mostra formulário
3. Usuário escolhe uma opção de interesse no evento
4. Usuário clica em “Salvar”
5. Sistema salva a participação do usuário no evento
6. Sistema registra a ação de participar de evento nos *logs* de usuários
7. Caso de uso encerrado

A3:

1. Usuário clica em “Interessados”
2. Sistema mostra uma lista de usuários que tem interesse de participar no evento
3. Sistema registra a ação de visualizar tela nos *logs* de usuários
4. Caso de uso encerrado

A4:

1. Usuário clicam em “Convidar”
2. Sistema mostra um formulário com um campo para selecionar uma pessoa que siga esse usuário
3. Usuário escolhe uma das pessoas que o seguem
4. Usuário clica em “Convidar” (E3)
5. Sistema envia um convite para o usuário
6. Sistema registra a ação de convidar usuário nos *logs* de usuários
7. Sistema mostra mensagem de sucesso
8. Caso de uso encerrado

A5:

1. O usuário clica em “Publicar”

2. O sistema mostra um formulário
3. O usuário insere uma imagem, opcionalmente insere um evento e um texto curto
4. Usuário clica em “Publicar” (E1, E2)
5. O sistema cria uma publicação do usuário com a imagem e as informações inseridas
6. Sistema registra a ação de publicar pelo evento nos *logs* de usuários
7. Caso de uso encerrado

A6:

1. Usuário clica em “Cancelar Evento”
2. Sistema pergunta se usuário tem certeza da ação
3. Usuário clica em “Confirmar”
4. Sistema mostra um formulário com uma entrada para o usuário escrever o motivo do cancelamento
5. Usuário preenche as informações
6. Usuário clica em “OK”
7. Sistema cancela o evento
8. Sistema registra a ação de editar evento nos *logs* de usuários
9. Sistema mostra a mensagem de cancelamento na página do evento
10. Caso de uso encerrado

Fluxos de Exceção:

E1:

1. Usuário não coloca uma imagem e clica em “Publicar”
2. Sistema mostra um aviso pedindo para o usuário escolher uma imagem
3. Caso de uso encerrado

E2:

1. Usuário escolhe uma imagem com mais de 2MB de tamanho
2. Sistema mostra um aviso pedindo para o usuário escolher uma imagem menor que 2MB
3. Caso de uso encerrado

E3:

1. Usuário seleciona alguém que ele já convidou anteriormente
2. Sistema avisa que usuário não pode convidar de novo a mesma pessoa
3. Caso de uso encerrado

#### UC07 - Manter Empresas

Objetivo: Permite que um usuário gerencie suas empresas

Atores: Usuário Registrado

Pós-condições: Sistema mostra a lista de empresas do usuário para que ele as gerencie como preferir

Fluxo Principal:

1. Sistema mostra uma lista de empresas desse usuário
2. Usuário clica em "Nova Empresa" (A1, A2, A3, A4)
3. Sistema mostra um formulário
4. Usuário preenche as informações
5. Usuário clica em 'Salvar' (E1)
6. Sistema registra a empresa
7. Sistema registra a ação de cadastrar empresa nos *logs* de usuários
8. Sistema mostra uma mensagem de sucesso
9. Sistema mostra a lista de empresas novamente
10. Caso de uso encerrado

Fluxos Alternativos:

A1:

1. O usuário clica em "Editar" na lista de empresas
2. O sistema carrega um formulário, com uma lista de eventos da empresa
3. O usuário clica em 'Salvar' (E1)
4. O sistema salva a empresa
5. Sistema registra a ação de editar empresa nos *logs* de usuários
6. Sistema mostra uma mensagem de sucesso
7. Sistema mostra a lista de empresas novamente

## 8. Caso de uso encerrado

### A2:

1. O usuário clica em "Eventos" na lista de empresas
2. Sistema registra a ação de visualizar tela nos *logs* de usuários
3. O sistema mostra uma lista de eventos da empresa
4. Caso de uso encerrado

### A3:

1. O usuário clica em "Visualizar" na lista de empresas
2. O sistema mostra a tela de "Ver Empresa"
3. Caso de uso encerrado

### Fluxos de Exceção:

#### E1:

1. O usuário preenche o campo CNPJ com um documento inválido
2. O sistema apaga o campo e avisa que o CNPJ é inválido
3. Caso de uso encerrado

## UC08 - Manter Eventos

Objetivo: Permite que um usuário gerencie seus eventos

Atores: Usuário Registrado

### Fluxo Principal:

1. Sistema mostra uma lista de eventos desse usuário, podendo filtrar por empresa
2. Usuário clica em "Novo Evento" (A1, A2, A3, A4)
3. Sistema mostra um formulário
4. Usuário clica em 'Salvar'
5. Sistema registra o evento
6. Sistema registra a ação de cadastrar evento nos *logs* de usuários
7. Sistema mostra uma mensagem de sucesso
8. Sistema mostra a lista de eventos novamente
9. Caso de uso encerrado

### Fluxos Alternativos:

#### A1:

1. O usuário clica em "Editar" na lista de eventos
2. O sistema carrega um formulário
3. O usuário clica em 'Salvar'
4. O sistema salva o evento
5. Sistema registra a ação de editar evento nos *logs* de usuários
6. Sistema mostra uma mensagem de sucesso
7. Sistema mostra a lista de empresas novamente
8. Caso de uso encerrado

#### A2:

1. O usuário clica em "Visualizar" na lista de eventos
2. O sistema mostra a página do evento
3. Caso de uso encerrado

#### A3:

1. O usuário clica em "Cancelar" na lista de eventos
2. O sistema mostra uma mensagem de confirmação
3. O usuário clica em "Confirmar"
4. O sistema mostra um campo para inserir o motivo do cancelamento
5. O usuário preenche o motivo e clica em "Cancelar"
6. O sistema cancela o evento
7. Sistema registra a ação de editar evento nos *logs* de usuários
8. Sistema mostra o motivo na página do evento
9. Caso de uso encerrado

### UC17 – Ver Tela Inicial

Objetivo: Permite que um usuário visualize as informações mais relevantes que ele veio procurar na plataforma

Atores: Usuário Registrado

#### Fluxo Principal:

1. O sistema mostra uma tela com uma lista de eventos futuros que o usuário tem interesse positivo ordenados pelos eventos com datas mais próximas

2. Sistema mostra uma lista de eventos públicos futuros que mais se encaixam com o perfil do usuário
3. Sistema mostra uma lista de atividades interessantes do sítio (publicações de eventos que tem interesse, publicação de empresas e usuários que está seguindo, novos eventos de empresas que está seguindo, avaliação de eventos e empresas de pessoas que está seguindo, etc...)
4. O usuário clica em algum evento
5. O sistema mostra mais informações daquele evento
6. O usuário clica nas informações do evento (A1)
7. O sistema abre a tela de “Ver Evento”
8. Caso de uso encerrado

A1:

1. Usuário clica em outro lugar que não sejam as informações do evento
2. Sistema esconde informações do evento
3. Caso de uso encerrado

UC18 – Pesquisa

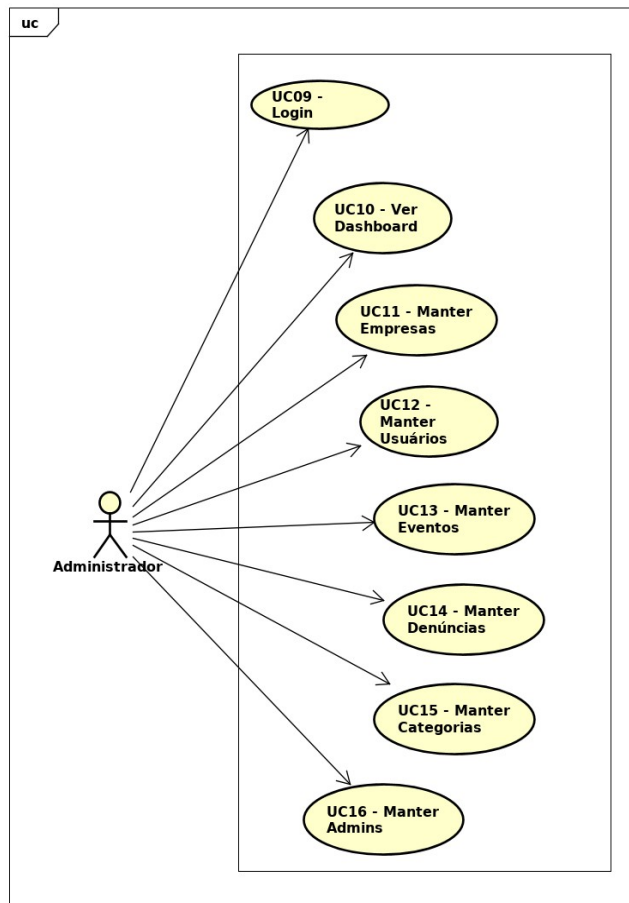
Objetivo: Permite que um usuário pesquise por eventos, empresas e outros usuários na plataforma

Atores: Usuário Registrado, Usuário Não Registrado

Fluxo Principal:

1. O usuário pesquisa por alguma palavra ou frase na barra de navegação e clica em “Pesquisar”
2. Sistema registra a ação de pesquisar nos *logs* de usuários
3. O sistema carrega uma tela mostrando os usuários, eventos e empresas que se melhor encaixam naquela frase. Os eventos ordenados por melhor localidade, categorias, idade e gênero caso o usuário esteja registrado
4. Caso de uso encerrado

## APÊNDICE B – DIAGRAMA DE CASOS DE USOS DO ADMINISTRADOR



### UC09 - Login

Objetivo: Permite que um usuário administrador faça seu login no administrador do sítio

Atores: Usuário Administrador

Pré-condições: Não estar logado

Pós-condições: Usuário se loga no sítio

Fluxo Principal:

1. O sistema apresenta a tela de login
2. O usuário preenche as informações e clica no botão de logar
3. Sistema verifica as informações e mostra a mensagem de login realizado (E1)
4. Sistema registra a ação de logar nos *logs* de administrador
5. Usuário é redirecionado para a tela inicial depois de um segundo e meio

## 6. Caso de uso encerrado

Fluxos de Exceção:

E1:

1. O usuário insere uma combinação de E-mail e Senha que não estão cadastradas em usuários ativos do sistema
2. O sistema verifica as informações e percebe que não encontrou nada
3. O sistema mostra a mensagem dizendo que o usuário não foi encontrado
4. Caso de uso encerrado

## UC10 – Ver Dashboard

Objetivo: Permite que um usuário visualize gráficos sobre o uso geral do sítio na última semana.

Atores: Usuário Administrador

Pré-condição: Administrador precisa estar logado

Pós-condição: Sistema mostra tela com listas relevantes para a noção do funcionamento do sítio

Fluxo Principal:

1. Sistema mostra lista de próximos eventos com maior número de interessados
2. Sistema mostra lista de próximos eventos a acontecer
3. Sistema mostra lista de novos eventos cadastrados
4. Sistema mostra lista de novos usuários cadastrados
5. Sistema registra a ação de visualizar tela nos *logs* de administrador
6. Usuário escolhe uma das quatro listas para visualizar
7. Sistema continua atualizando a lista selecionada a cada 2 segundos
8. Caso de uso encerrado

## UC11 – Manter Empresas

Objetivo: Permite que um usuário administrador gerencie as empresas cadastradas no sistema



Atores: Usuário Administrador

Pós-condições: Sistema mostra uma lista de empresas com todas as opções para que o administrador possa manter as empresas

Fluxo Principal:

1. Sistema mostra lista com todas as empresas cadastradas no sistema
2. Usuário clica no botão “novo” (A1, A2, A3, A4, A5)
3. Sistema mostra formulário contendo informações a serem preenchidas
4. Usuário preenche as informações
5. Usuário clica em “salvar” (E1)
6. Sistema salva as informações
7. Sistema registra a ação de cadastrar empresa nos *logs* de administrador
8. Sistema fecha o formulário
9. Sistema mostra mensagem de sucesso
10. Caso de uso encerrado

Fluxos Alternativos:

A1:

1. Usuário seleciona uma empresa e clica em “Editar”
2. Sistema mostra formulário contendo informações da empresa selecionada
3. Usuário altera as informações e clica em “Salvar” (E1)
4. Sistema salva as informações
5. Sistema registra a ação de editar empresa nos *logs* de administrador
6. Sistema fecha o formulário
7. Sistema mostra mensagem de sucesso
8. Caso de uso encerrado

A2:

1. Usuário seleciona uma empresa
2. Usuário clica em “Deletar”
3. Sistema exclui empresa
4. Sistema registra a ação de excluir empresa nos *logs* de administrador
5. Sistema mostra mensagem de sucesso

## 6. Caso de uso encerrado

### A3:

1. Usuário seleciona uma empresa
2. Usuário clica em “Estatísticas”
3. Sistema mostra uma lista de usuários que seguem a empresa, eventos da empresa e denúncias recebidas pela empresa
4. Sistema registra a ação de visualizar tela nos *logs* de administrador
5. Caso de uso encerrado

### A4:

1. Usuário seleciona uma empresa
2. Usuário clica em “Perfil”
3. Sistema mostra a tela de “Ver Empresa”
4. Caso de uso encerrado

### A5:

1. Usuário clica em “Foto Perfil”
2. Sistema mostra formulário com entrada para uma imagem
3. Usuário seta uma imagem
4. Usuário clica em “Salvar” (E2)
5. Sistema apaga imagem de perfil anterior caso haja
6. Sistema coloca a nova imagem enviada como imagem de perfil da empresa
7. Sistema registra a ação de editar empresa nos *logs* de administrador
8. Sistema mostra uma mensagem de sucesso
9. Caso de uso encerrado

### Fluxos de Exceção:

#### E1:

1. O usuário não insere algum dado obrigatório
2. O sistema mostra um erro pedindo para que todas as informações obrigatórias sejam preenchidas
3. Caso de uso encerrado

E2:

1. Usuário escolhe uma imagem com mais de 2MB de tamanho
2. Sistema mostra um aviso pedindo para o usuário escolher uma imagem menor que 2MB
3. Caso de uso encerrado

## UC12 – Manter Usuários

Objetivo: Permite que um usuário administrador gerencie os usuários cadastrados no sistema

Atores: Usuário Administrador

Pós-condições: Sistema mostra uma lista de usuários com todas as opções para que o administrador possa manter os usuários

Fluxo Principal:

1. Sistema mostra lista com todos os usuários cadastrados no sistema
2. Usuário administrador clica no botão “Novo” (A1, A2, A3, A4, A5)
3. Sistema mostra formulário contendo informações a serem preenchidas
4. Usuário administrador preenche as informações
5. Usuário clica em “salvar” (E1)
6. Sistema salva as informações
7. Sistema registra a ação de cadastrar usuário nos *logs* de administrador
8. Sistema fecha o formulário
9. Sistema mostra mensagem de sucesso
10. Caso de uso encerrado

Fluxos Alternativos:

A1:

1. Usuário seleciona um usuário
2. Usuário clica em “Editar”
3. Sistema mostra formulário contendo informações do usuário selecionado
4. Usuário altera as informações e clica em “Salvar” (E1)
5. Sistema salva as informações

6. Sistema registra a ação de editar usuário nos *logs* de administrador
7. Sistema fecha o formulário
8. Sistema mostra mensagem de sucesso
9. Caso de uso encerrado

A2:

1. Usuário seleciona um usuário e clica em “Deletar”
2. Sistema exclui o usuário
3. Sistema registra a ação de deletar usuário nos *logs* de administrador
4. Sistema mostra mensagem de sucesso
5. Caso de uso encerrado

A3:

1. Usuário administrador seleciona um usuário
2. Usuário administrador clica em “Estatísticas”
3. Sistema registra a ação de visualizar tela nos *logs* de administrador
4. Sistema mostra uma lista de empresas que o usuário segue, empresas que o usuário tem, pessoas que seguem o usuário, quem o usuário segue, denúncias que o usuário fez, denúncias que o usuário recebeu, eventos que marcou interesse, convites que fez e recebeu, por quem foi bloqueado e quem bloqueou
5. Caso de uso encerrado

A4:

1. Usuário seleciona um usuário e clica em “Perfil”
2. Sistema mostra a tela de “Manter Perfil”
3. Caso de uso encerrado

A5:

1. Usuário administrador seleciona um usuário
2. Sistema clica em “Foto Perfil”
3. Sistema mostra formulário com entrada para uma imagem
4. Usuário seta uma imagem
5. Usuário clica em “Salvar” (E2)
6. Sistema apaga imagem de perfil anterior caso haja

7. Sistema coloca a nova imagem enviada como imagem de perfil
8. Sistema registra a ação de editar usuário nos *logs* de administrador
9. Sistema mostra mensagem de sucesso
10. Caso de uso encerrado

A6:

1. Usuário administrador seleciona um usuário e clica em “Foto Capa”
2. Sistema mostra formulário com entrada para uma imagem
3. Usuário seta uma imagem e clica em “Salvar” (E2)
4. Sistema apaga imagem de capa anterior caso haja
5. Sistema coloca a nova imagem enviada como imagem de capa
6. Sistema registra a ação de editar usuário nos *logs* de administrador
7. Sistema mostra mensagem de sucesso
8. Caso de uso encerrado

A7:

1. Usuário administrador seleciona um usuário
2. Usuário clica em “Agenda”
3. Sistema registra a ação de visualizar tela nos *logs* de administrador
4. Sistema mostra um calendário do mês atual com todos os eventos marcados com interesse positivo pelo usuário
5. Usuário clica em algum evento
6. Sistema mostra mais informações sobre aquele evento
7. Usuário clica nas informações do evento
8. Sistema mostra tela de “Ver Evento”
9. Caso de uso encerrado

A8:

1. Usuário administrador seleciona um usuário
2. Usuário administrador clica em “Agenda”
3. Sistema registra a ação de visualizar tela nos *logs* de administrador
4. Sistema mostra um calendário do mês atual com todos os eventos marcados com interesse positivo pelo usuário
5. Usuário clica em algum outro mês (anterior ou próximo)

6. O sistema carrega o calendário daquele mês com os eventos daquele mês (A7)
7. Caso de uso encerrado

A9:

1. Usuário administrador seleciona um usuário e clica em “Certificados”
2. Sistema registra a ação de visualizar tela nos *logs* de administrador
3. Sistema mostra uma lista com todos os certificados que o usuário recebeu em eventos que participou com as opções de “Visualizar” e “Imprimir”
4. Caso de uso encerrado

Fluxos de Exceção:

E1:

1. O usuário não insere algum dado obrigatório
2. O sistema mostra um erro pedindo para que todas as informações obrigatórias sejam preenchidas
3. Caso de uso encerrado

E2:

1. Usuário escolhe uma imagem com mais de 2MB de tamanho
2. Sistema mostra um aviso pedindo para o usuário escolher uma imagem menor que 2MB
3. Caso de uso encerrado

### UC13 – Manter Eventos

Objetivo: Permite que um usuário administrador gerencie os eventos cadastrados no sistema

Atores: Usuário Administrador

Pré-condições:

Pós-condições: Sistema mostra uma lista de eventos com todas as opções para que o administrador possa manter os eventos

Fluxo Principal:

1. Sistema mostra lista com todos os eventos cadastrados no sistema
2. Usuário clica no botão “Novo” (A1, A2, A3, A4)
3. Sistema mostra formulário contendo informações a serem preenchidas
4. Usuário preenche as informações e clica em “Salvar” (E1)
5. Sistema salva as informações
6. Sistema registra a ação de cadastrar evento nos *logs* de administrador
7. Sistema fecha o formulário
8. Sistema mostra mensagem de sucesso
9. Caso de uso encerrado

#### Fluxos Alternativos:

##### A1:

1. Usuário seleciona um evento e clica em “Editar”
2. Sistema mostra formulário contendo informações do usuário selecionado
3. Usuário altera as informações e clica em “Salvar” (E1)
4. Sistema salva as informações
5. Sistema registra a ação de editar evento nos *logs* de administrador
6. Sistema fecha o formulário
7. Sistema mostra mensagem de sucesso
8. Caso de uso encerrado

##### A2:

1. Usuário seleciona um evento e clica em “Excluir”
2. Sistema exclui o evento
3. Sistema registra a ação de excluir evento nos *logs* de administrador
4. Sistema mostra mensagem de sucesso
5. Caso de uso encerrado

##### A3:

1. Usuário administrador seleciona um evento e clica em “Estatísticas”
2. Sistema registra a ação de visualizar tela nos *logs* de administrador
3. Sistema mostra uma lista de pessoas interessadas no evento, convites feitos no evento e denúncias que o evento recebeu
4. Caso de uso encerrado

A4:

1. Usuário seleciona um evento e clica em “Perfil”
2. Sistema mostra a tela de “Ver Evento”
3. Caso de uso encerrado

A5:

1. Usuário administrador seleciona um usuário e clica em “Imagens”
2. Sistema mostra formulário com entrada para imagens e uma lista com as imagens que já estão no evento, cada uma com título, descrição e opção de deletar
3. Usuário escolhe quais imagens deletar, quais títulos e descrições editar e quais novas imagens inserir
4. Usuário clica em “Salvar” (E2)
5. Sistema apaga as imagens apagadas
6. Sistema altera os títulos e descrições
7. Sistema salva as novas imagens conforme configurado pelo usuário
8. Sistema registra a ação de editar evento nos *logs* de administrador
9. Sistema mostra uma mensagem de sucesso
10. Caso de uso encerrado

Fluxos de Exceção:

E1:

1. O usuário não insere algum dado obrigatório
2. O sistema mostra um erro pedindo para que todas as informações obrigatórias sejam preenchidas
3. Caso de uso encerrado

E2:

1. Usuário escolhe uma imagem com mais de 2MB de tamanho
2. Sistema mostra um aviso pedindo para o usuário escolher uma imagem menor que 2MB
3. Caso de uso encerrado



Objetivo: Permite que um usuário administrador gere as denúncias cadastradas no sistema

Atores: Usuário Administrador

Pós-condições: Sistema mostra uma lista de denúncias com todas as opções para que o administrador possa manter as denúncias

Fluxo Principal:

1. Sistema mostra lista com todas as denúncias do tipo “usuários” cadastradas no sistema (A4)
2. Usuário clica no botão “Detalhes” de alguma denúncia (A2, A3)
3. Sistema mostra uma tela com informações mais detalhadas sobre a denúncia
4. Usuário clica no botão “Aceitar” (A1)
5. Sistema inativa usuário, evento ou empresa relacionado a denúncia
6. Sistema registra a ação de aceitar denúncia nos *logs* de administrador
7. Sistema mostra mensagem de sucesso
8. Sistema manda denúncia para o fim da lista
9. Caso de uso encerrado

Fluxos Alternativos:

A1:

1. Usuário clica no botão “Rejeitar”
2. Sistema marca a denúncia como rejeitada
3. Sistema registra a ação de rejeitar denúncia nos *logs* de administrador
4. Sistema mostra mensagem de sucesso
5. Sistema manda denúncia para o fim da lista
6. Caso de uso encerrado

A2:

1. Usuário clica no botão “rejeitar todas as denúncias de (nome do(a) denunciante)”
2. Sistema marca todas as denúncias feitas pelo usuário como rejeitadas
3. Sistema registra a ação de recusar denúncias em massa nos *logs* de administrador

4. Sistema mostra mensagem de sucesso
5. Caso de uso encerrado

A3:

1. Usuário clica no botão “rejeitar todas as denúncias a (nome do(a) denunciado)”
2. Sistema marca todas as denúncias feitas ao usuário/evento/empresa como rejeitadas
3. Sistema registra a ação de rejeitar denúncias em massa nos *logs* de administrador
4. Sistema mostra mensagem de sucesso
5. Caso de uso encerrado

A4:

1. Usuário seleciona o tipo de denúncia que ele deseja ver entre “usuários”, “eventos” e “empresas”
2. Fluxo volta para o passo 2 do fluxo principal mostrando apenas as denúncias feitas ao tipo selecionado
3. Caso de uso encerrado

## UC15 – Gerenciar Categorias

Objetivo: Permite que um usuário administrador gerencie as categorias de eventos cadastradas no sistema

Atores: Usuário Administrador

Pós-condições: Sistema mostra uma lista de categorias com todas as opções para que o administrador possa manter as categorias

Fluxo Principal:

1. Sistema mostra lista com todas as categorias cadastradas no sistema
2. Usuário clica no botão “Novo” (A1, A2)
3. Sistema mostra formulário contendo informações a serem preenchidas
4. Usuário preenche as informações
5. Usuário clica em “Salvar” (E1)

6. Sistema salva as informações
7. Sistema registra a ação de cadastrar categoria nos *logs* de administrador
8. Sistema fecha o formulário
9. Sistema mostra mensagem de sucesso
10. Caso de uso encerrado

#### Fluxos Alternativos:

##### A1:

1. Usuário seleciona uma categoria e clica em “Editar”
2. Sistema mostra formulário contendo informações da categoria selecionada
3. Usuário altera as informações
4. Usuário clica em “Salvar” (E1)
5. Sistema salva as informações
6. Sistema registra a ação de editar categoria nos *logs* de administrador
7. Sistema fecha o formulário
8. Sistema mostra mensagem de sucesso
9. Caso de uso encerrado

##### A2:

1. Usuário seleciona uma categoria e clica em “Deletar”
2. Sistema exclui empresa
3. Sistema registra a ação de excluir categoria nos *logs* de administrador
4. Sistema mostra mensagem de sucesso
5. Caso de uso encerrado

#### Fluxos de Exceção:

##### E1:

1. O usuário não insere algum dado obrigatório
2. O sistema mostra um erro pedindo para que todas as informações obrigatórias sejam preenchidas
3. Caso de uso encerrado

Objetivo: Permite que um usuário administrador gerencie outros usuários administradores cadastrados no sistema

Atores: Usuário Administrador

Pós-condições: Sistema mostra uma lista de administradores com todas as opções para que o administrador possa manter os administradores

Fluxo Principal:

1. Sistema mostra lista com todos os administradores cadastrados no sistema
2. Usuário clica no botão "Novo" (A1, A2)
3. Sistema mostra formulário contendo informações a serem preenchidas
4. Usuário preenche as informações
5. Usuário clica em "Salvar" (E1)
6. Sistema salva as informações
7. Sistema registra a ação de cadastrar administradores nos *logs* de administrador
8. Sistema fecha o formulário
9. Sistema mostra mensagem de sucesso
10. Caso de uso encerrado

Fluxos Alternativos:

A1:

1. Usuário seleciona um administrador e clica em "Editar"
2. Sistema mostra formulário contendo informações do administrador selecionado
3. Usuário altera as informações
4. Usuário clica em "Salvar" (E1)
5. Sistema salva as informações
6. Sistema registra a ação de editar administradores nos *logs* de administrador
7. Sistema fecha o formulário
8. Sistema mostra mensagem de sucesso
9. Caso de uso encerrado

A2:

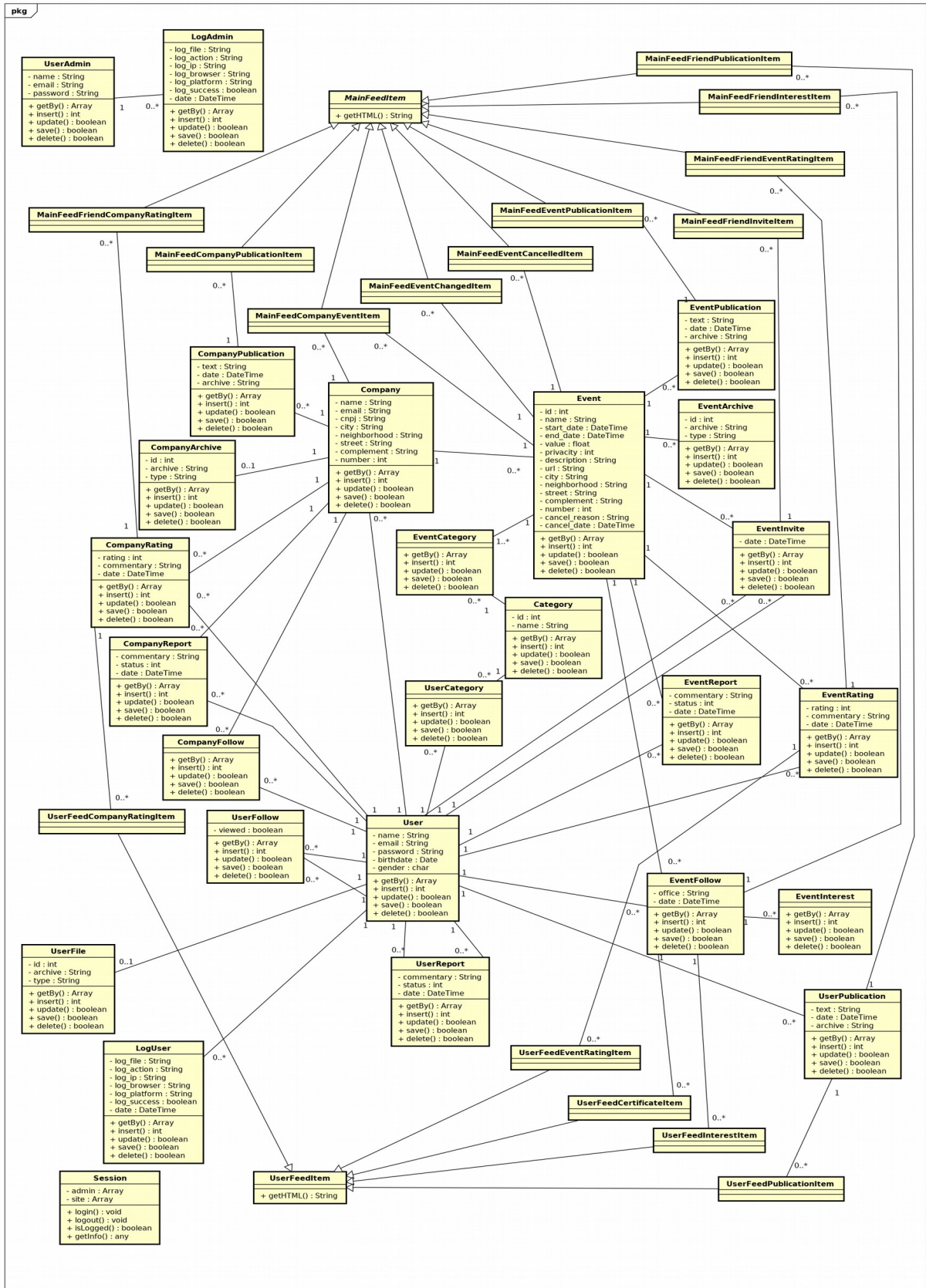
1. Usuário seleciona um usuário da lista e clica em “logs”
2. Sistema registra a ação de visualizar tela nos *logs* de administrador
3. Sistema mostra uma lista de ações que o usuário administrador tomou no sistema
4. Caso de uso encerrado

Fluxos de Exceção:

E1:

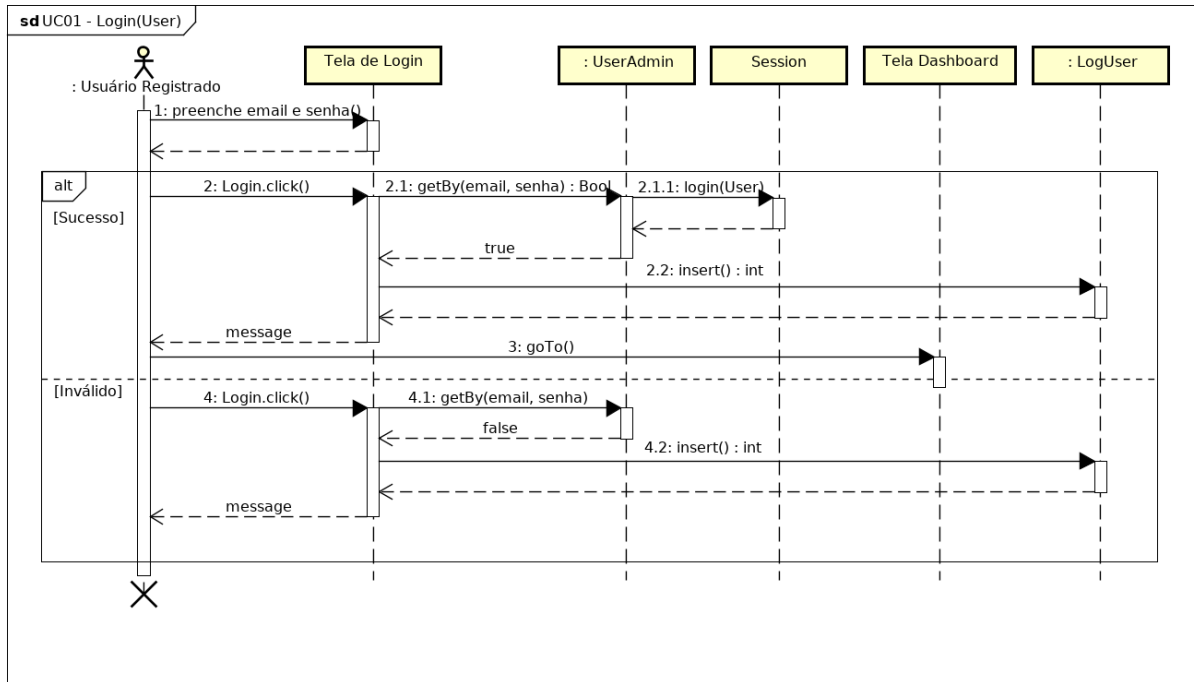
1. O usuário não insere algum dado obrigatório
2. O sistema mostra um erro pedindo para que todas as informações obrigatórias sejam preenchidas
3. Caso de uso encerrado

# APÊNDICE C – DIAGRAMA DE CLASSES

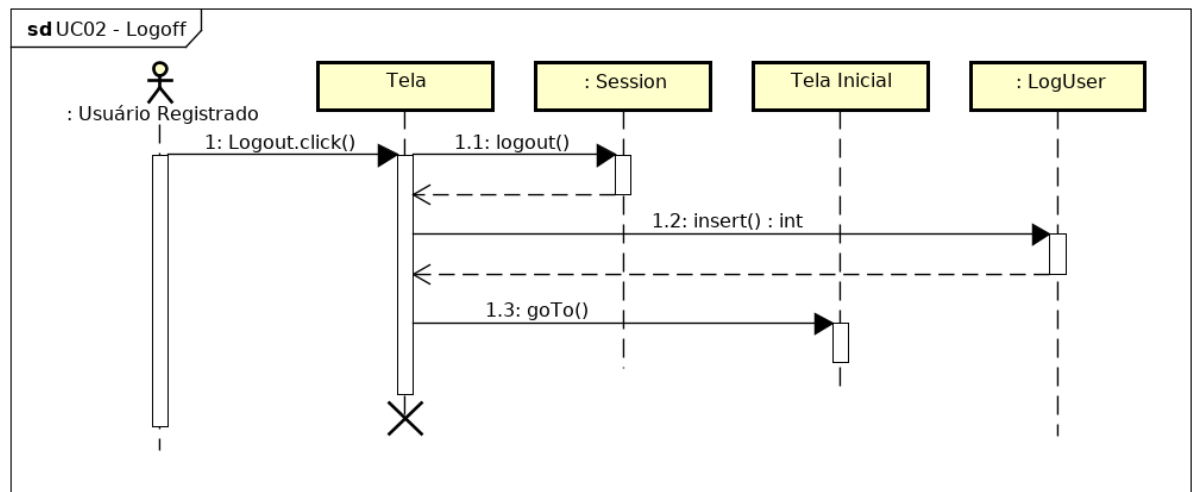


## APÊNDICE D – DIAGRAMAS DE SEQUÊNCIA

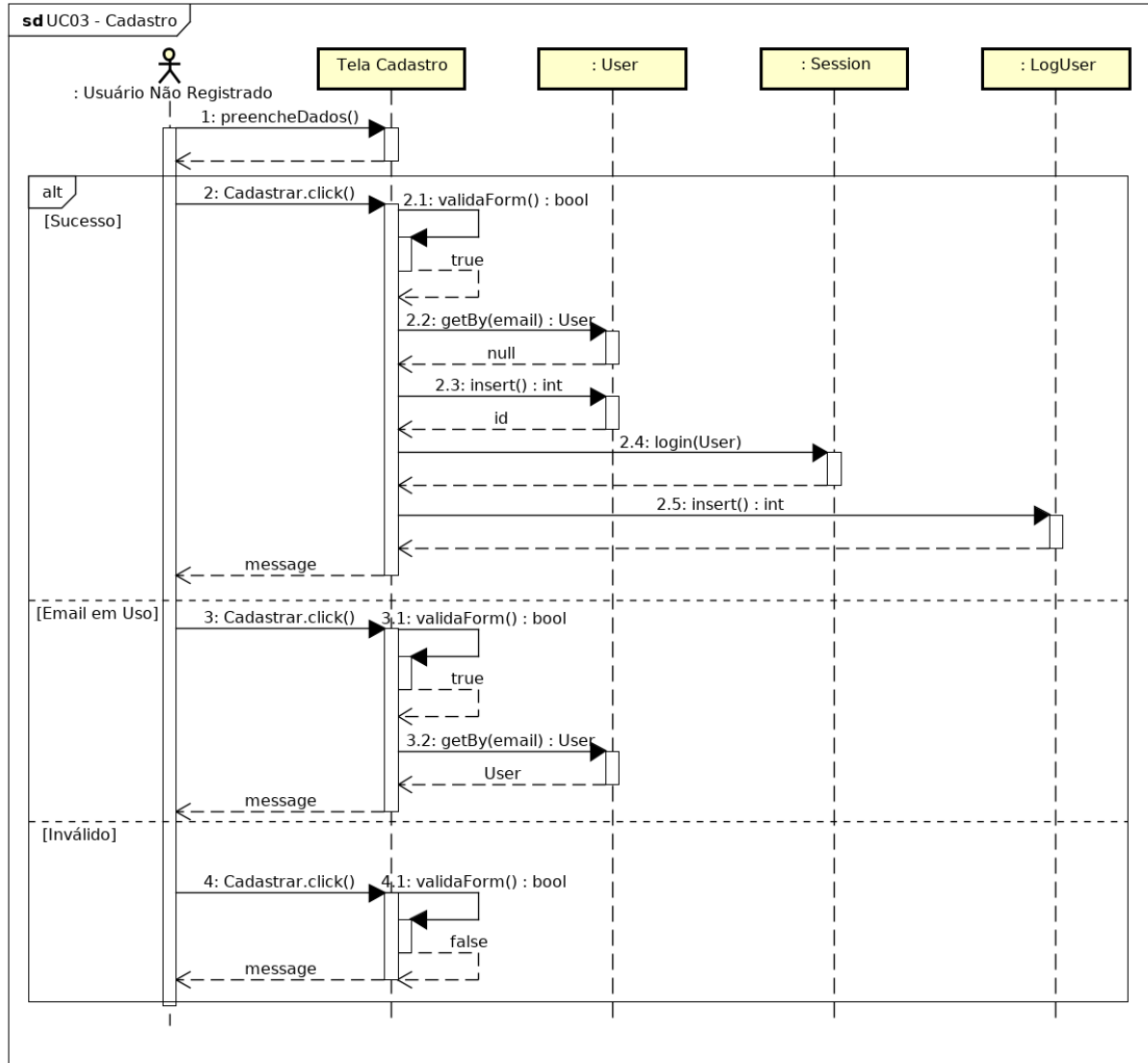
### UC01 – Login (usuário)



### UC02 – Logoff

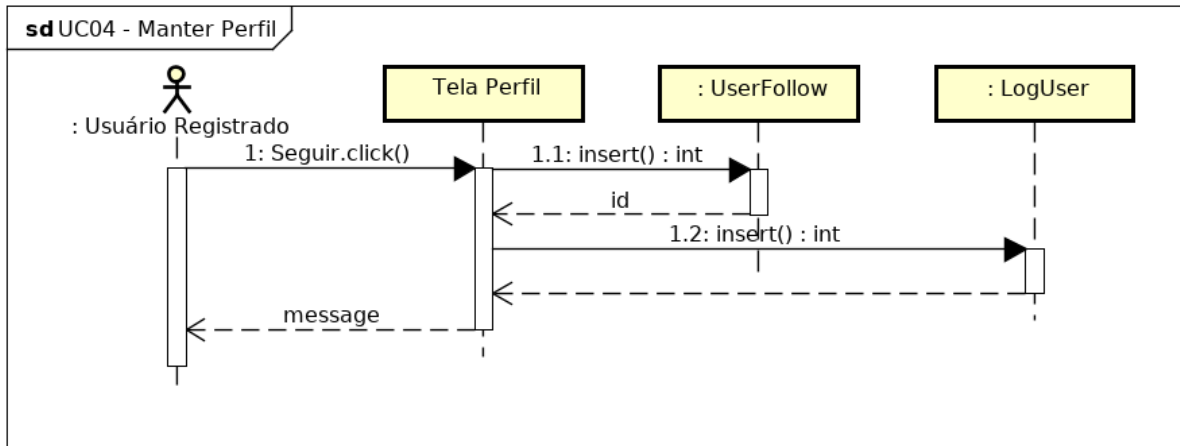


UC03 – Cadastro

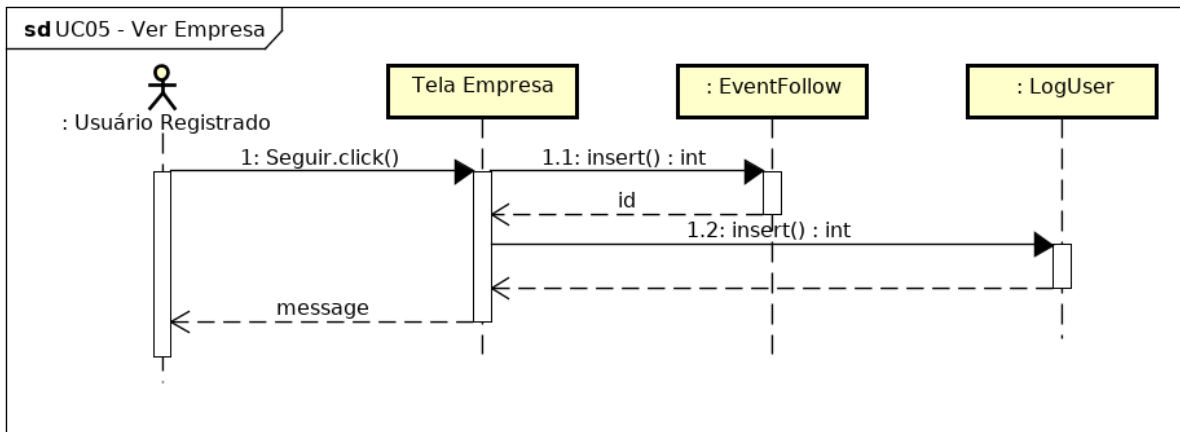




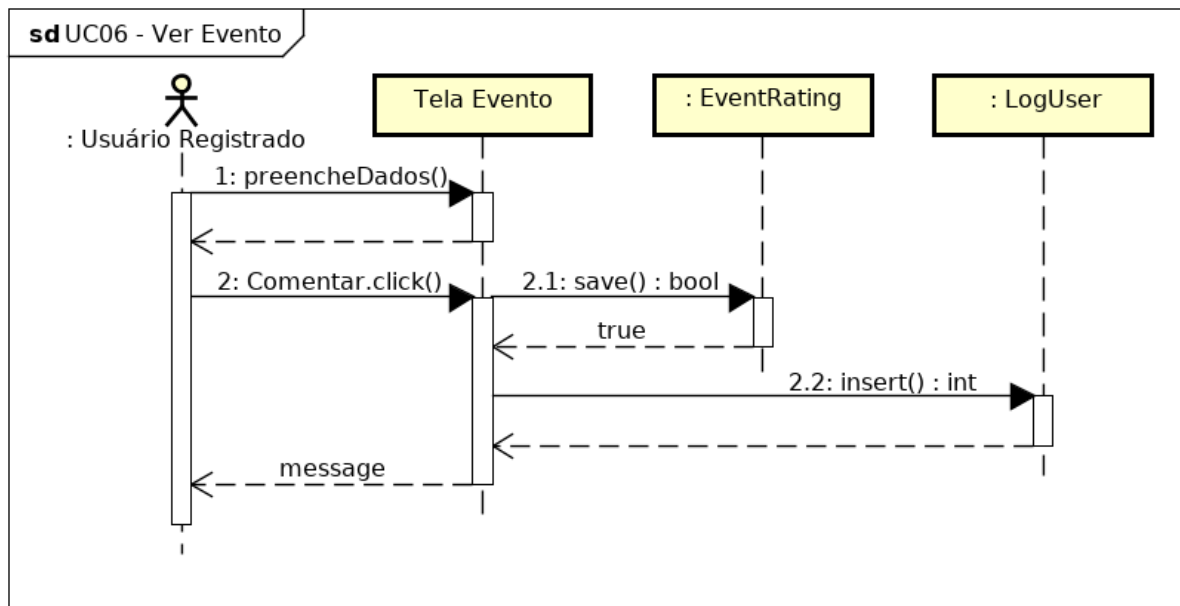
UC04 – Manter Perfil



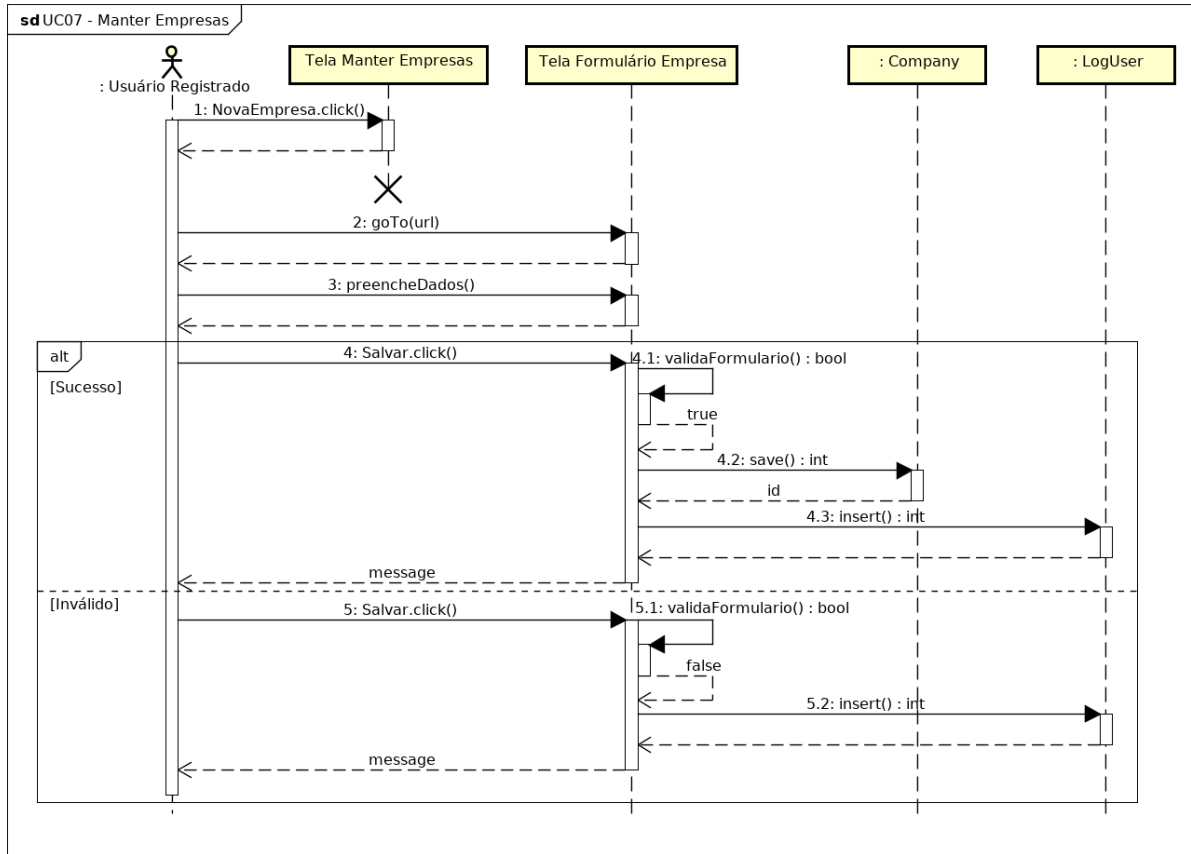
UC05 – Ver Empresa



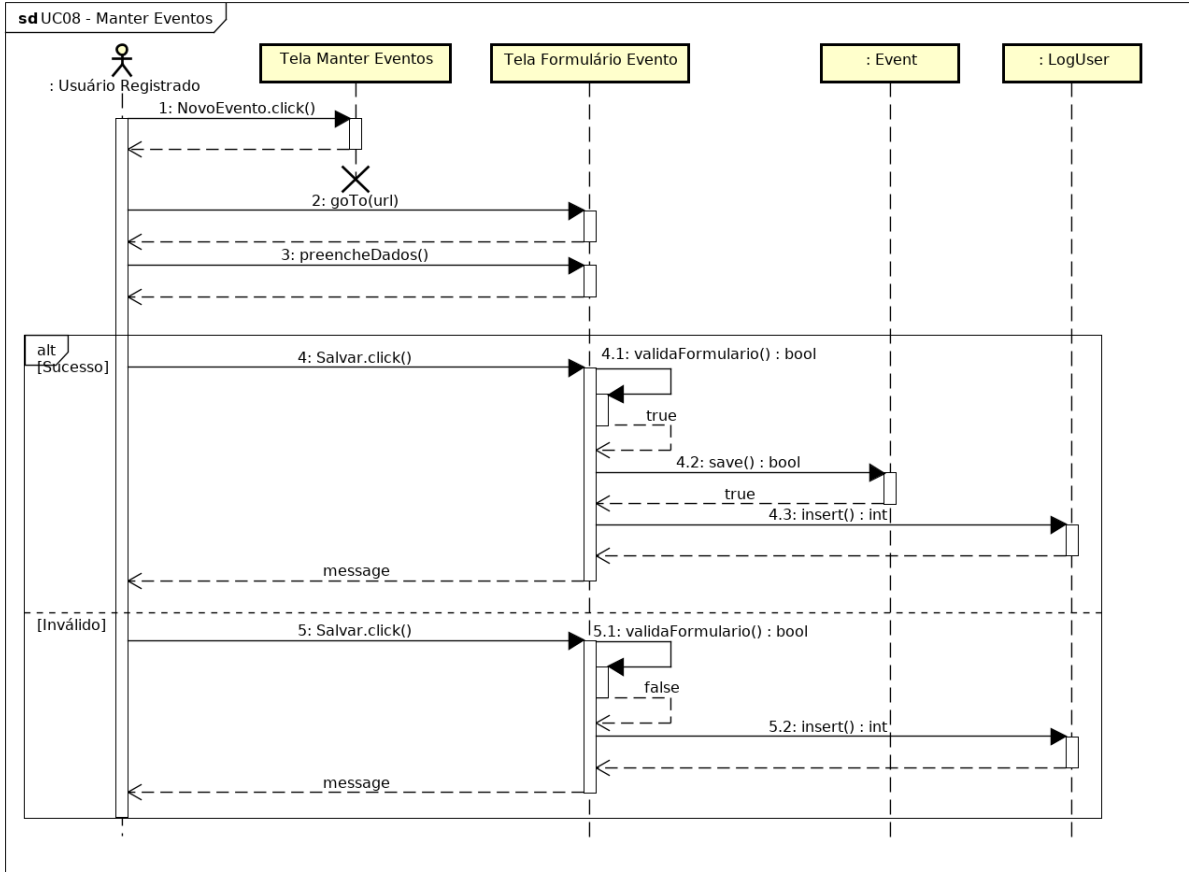
UC06 – Ver Evento



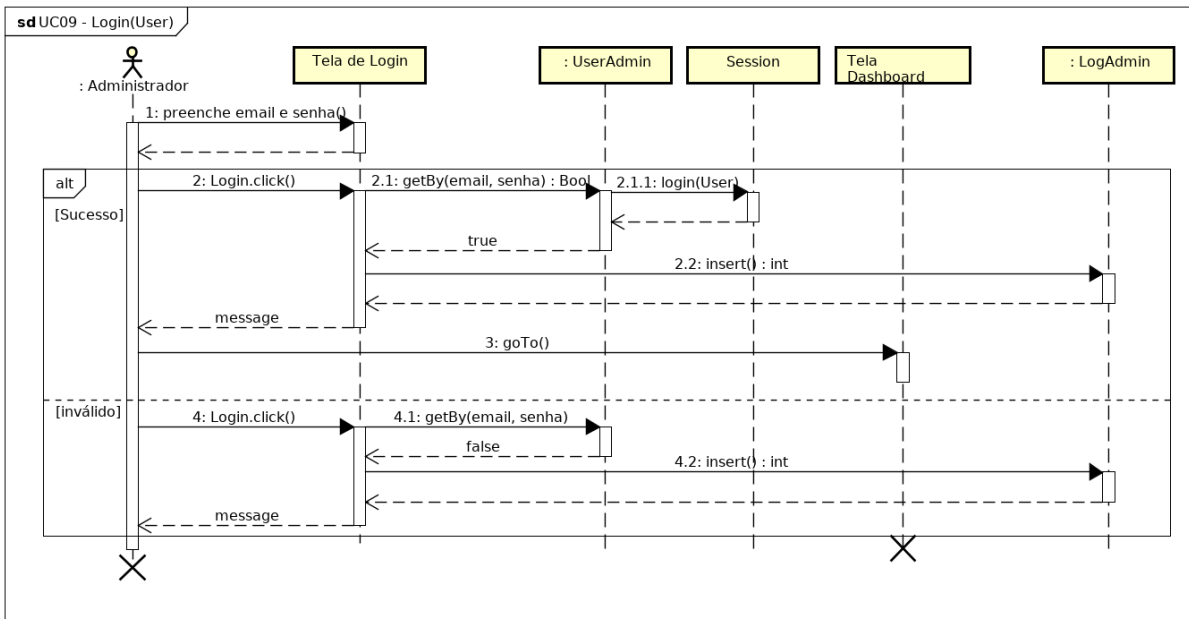
UC07 – Manter Empresas



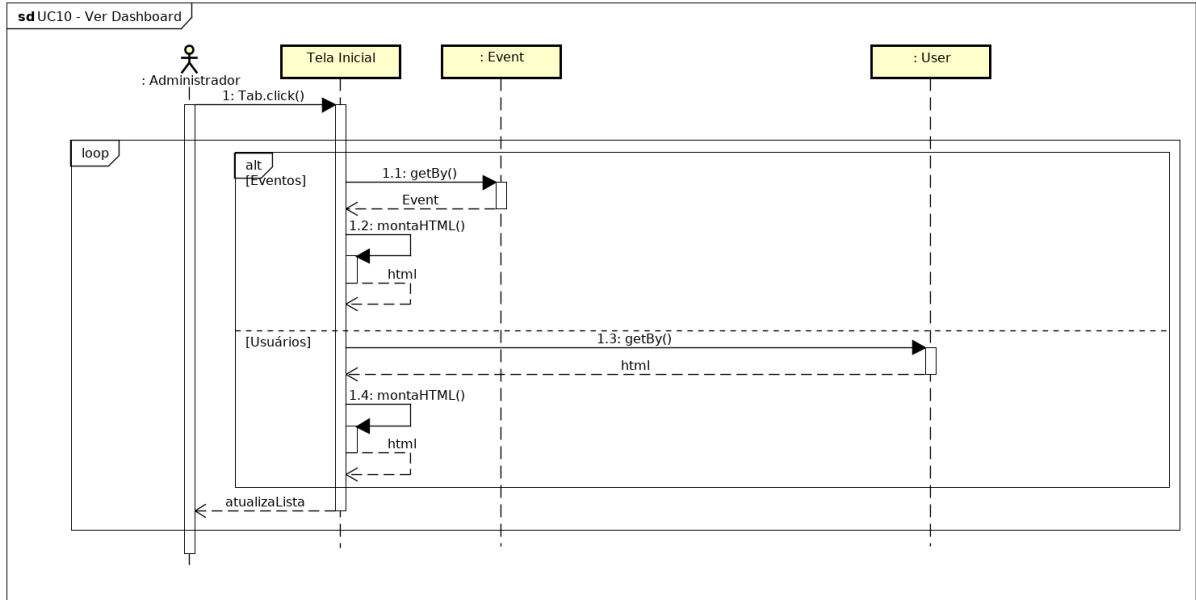
### UC08 – Manter Eventos



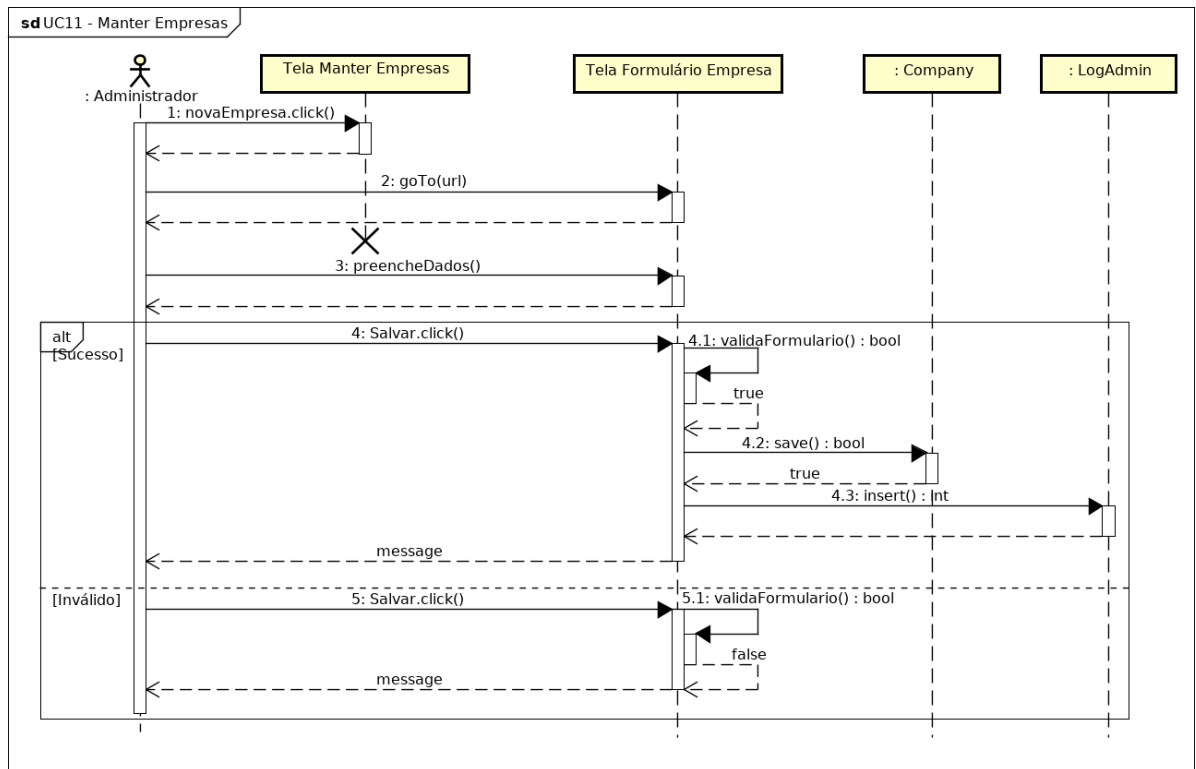
### UC09 – Login (administrador)



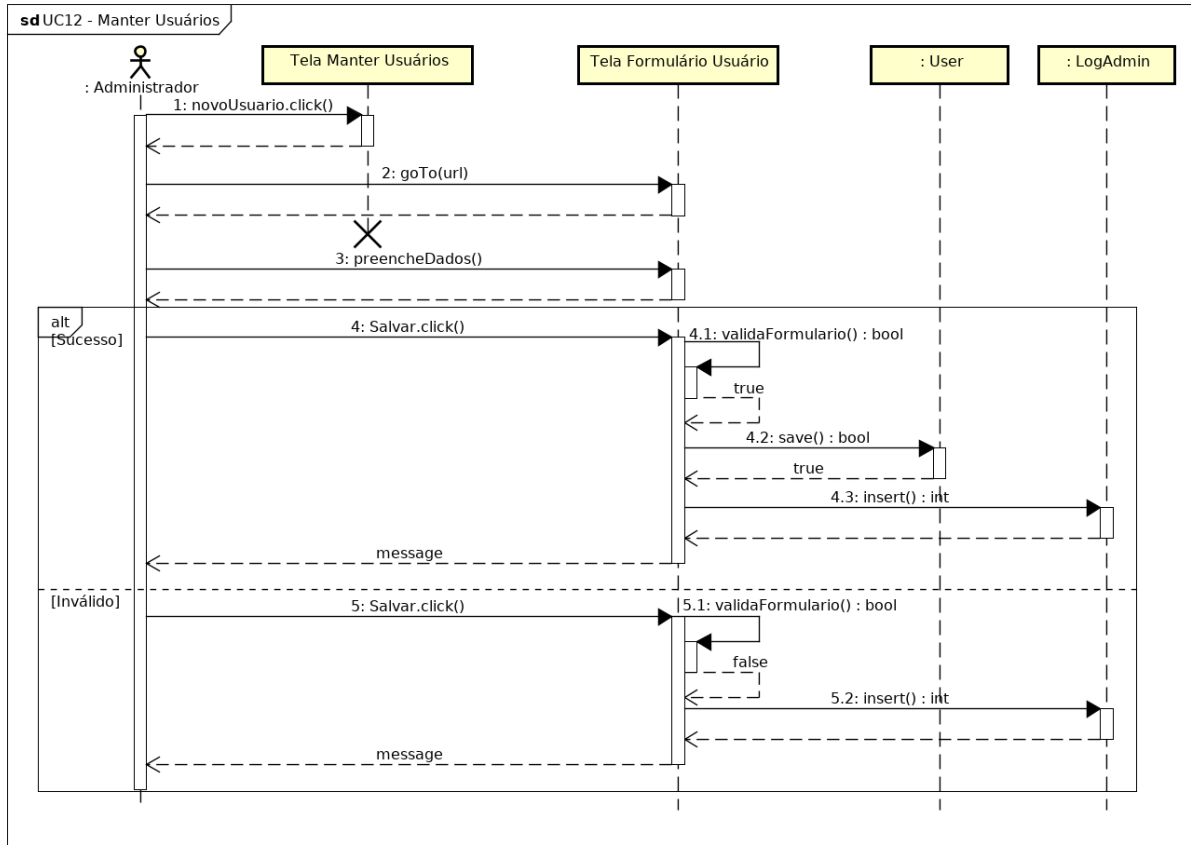
### UC10 – Ver Dashboard



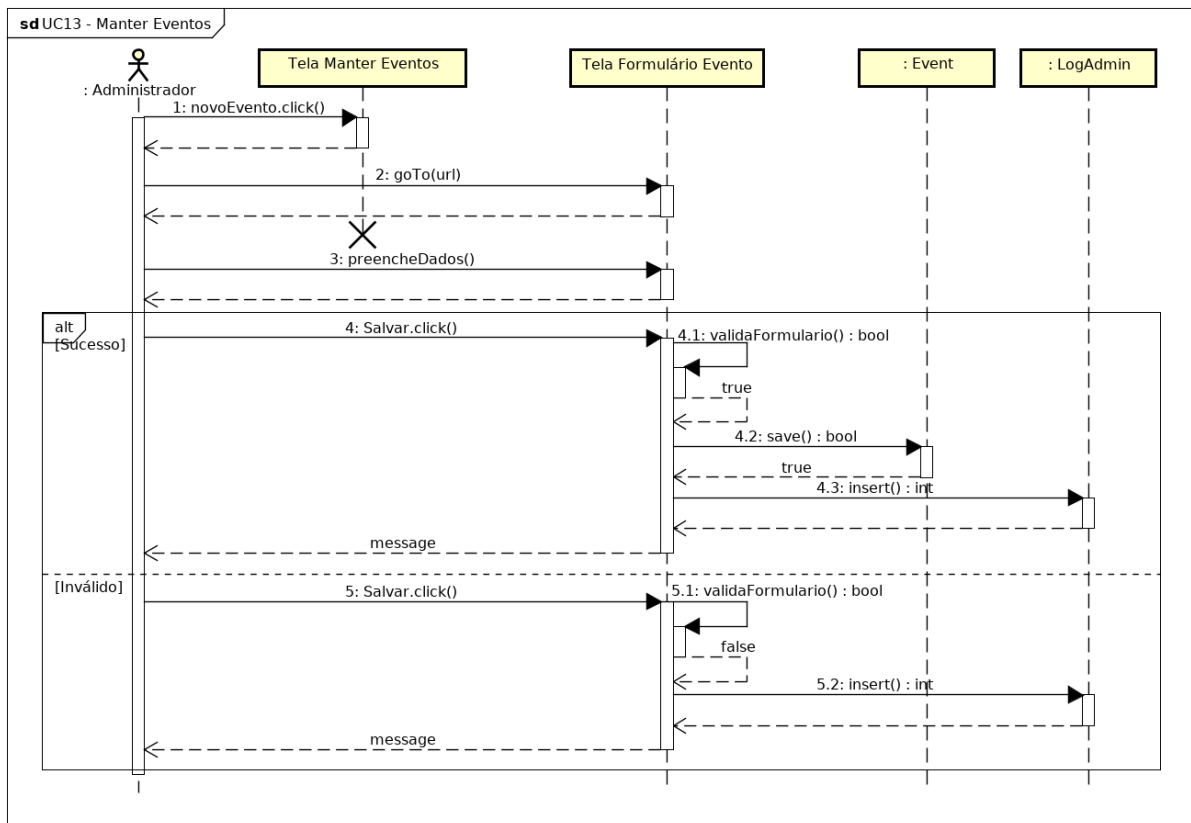
### UC11 – Manter Empresas



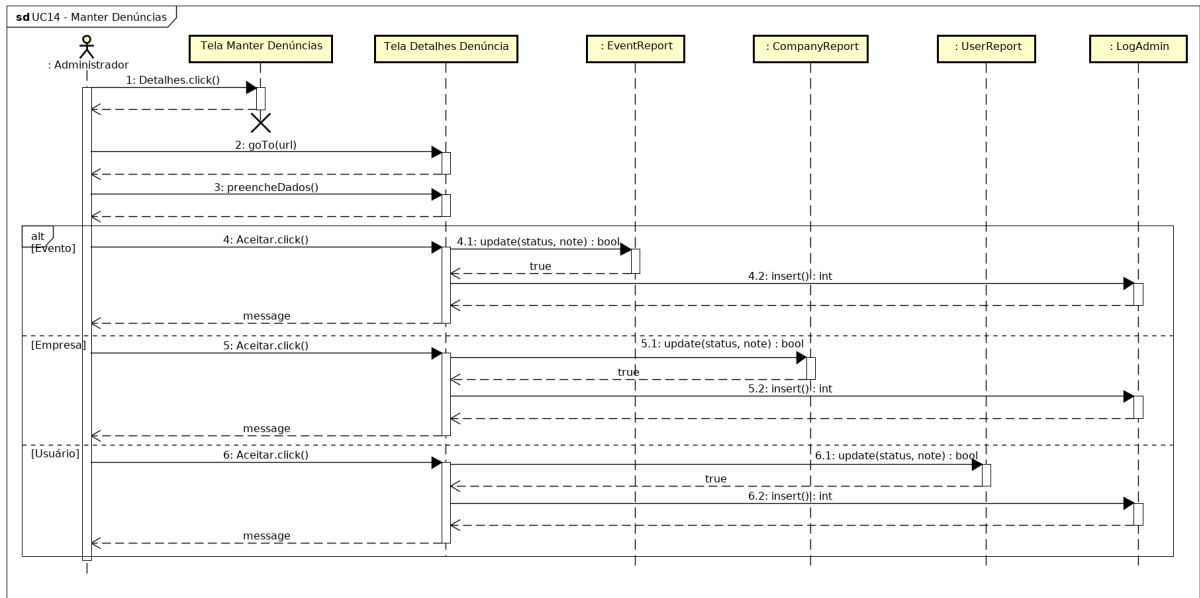
### UC12 – Manter Usuários



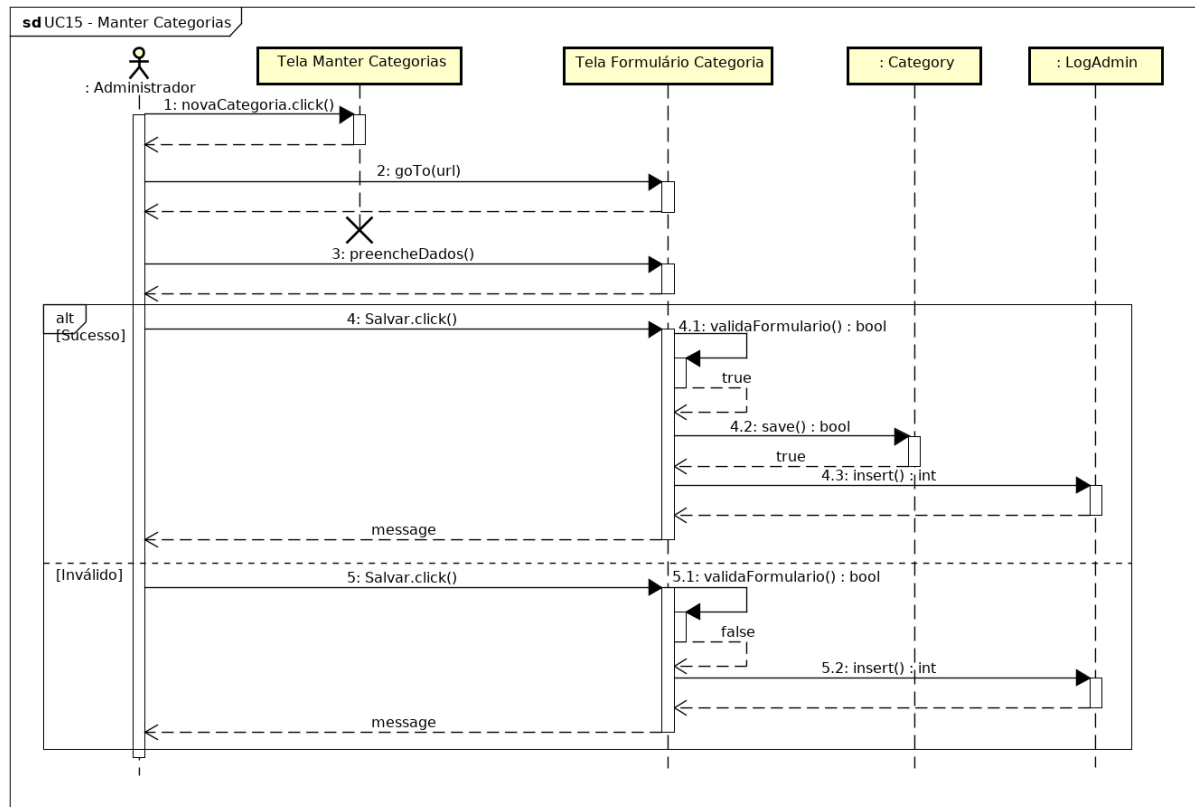
### UC13 – Manter Eventos



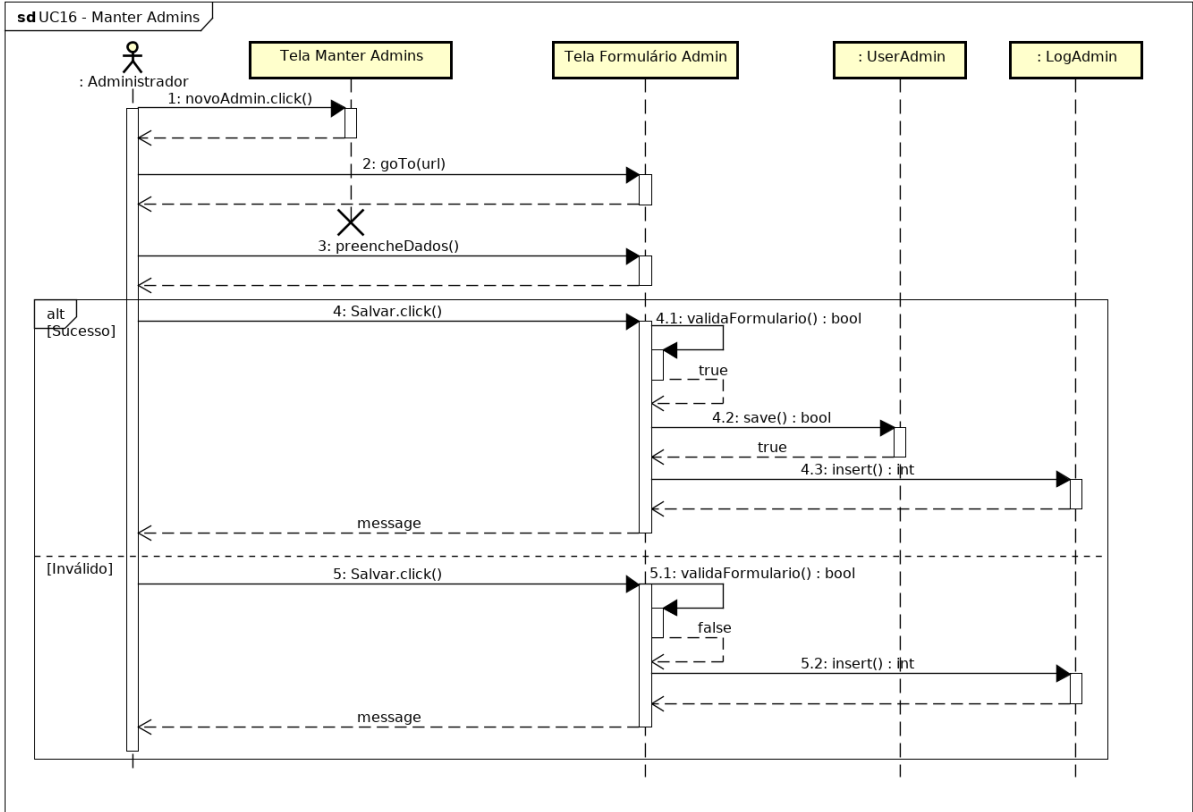
### UC14 – Manter Denúncias



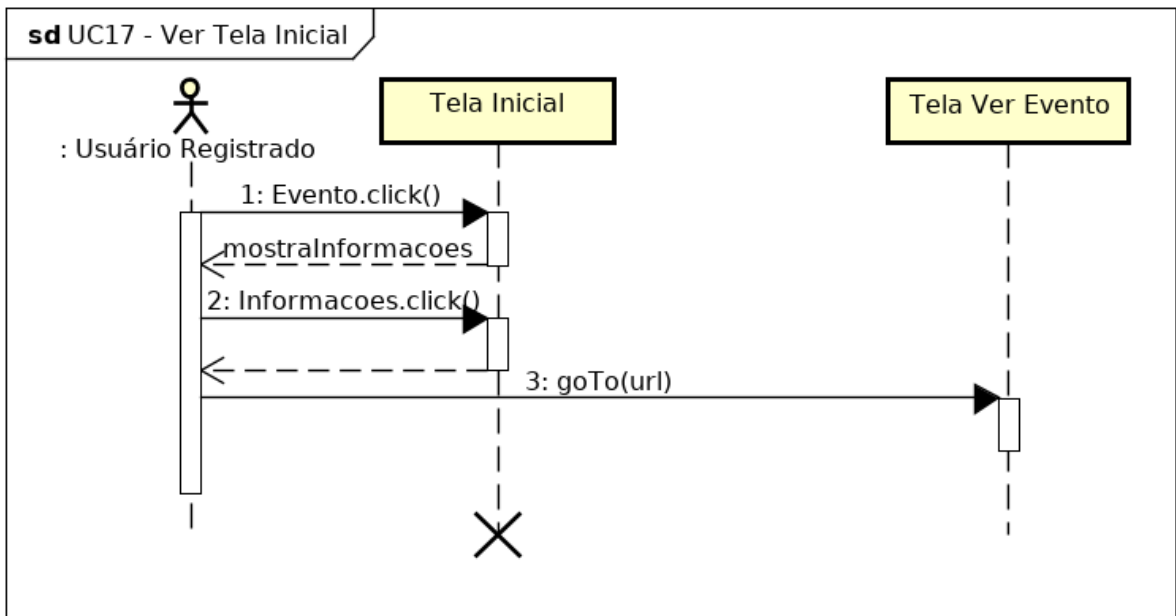
### UC15 – Manter Categorias



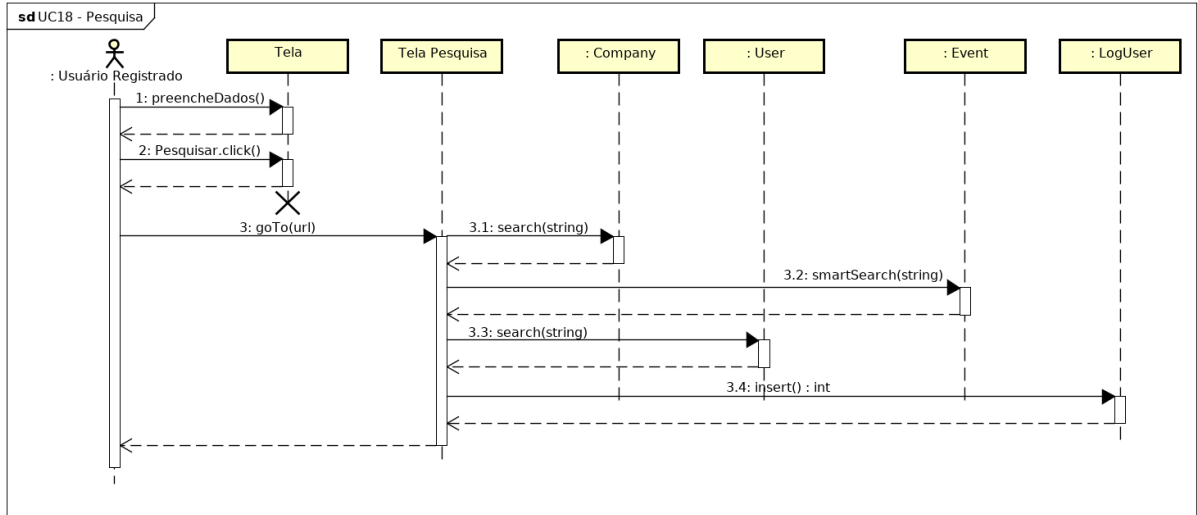
UC16 – Manter Administradores



UC17 – Ver Tela Inicial



# UC18 – Pesquisa







## APÊNDICE G – EXEMPLO DE MODELO E CONTROLADOR

