

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO**

**CIDADÃO FISCAL: APLICATIVO PARA ABERTURA E  
ACOMPANHAMENTO DE PROCESSOS NO SETOR DE  
OUVIDORIA DA PREFEITURA MUNICIPAL DE BLUMENAU**

**TUANE BRÜNING WESSLER**

**BLUMENAU**  
**2014**

**2014/2-18**

**TUANE BRÜNING WESSLER**

**CIDADÃO FISCAL: APLICATIVO PARA ABERTURA E  
ACOMPANHAMENTO DE PROCESSOS NO SETOR DE  
OUVIDORIA DA PREFEITURA MUNICIPAL DE BLUMENAU**

Trabalho de Conclusão de Curso submetido à  
Universidade Regional de Blumenau para a  
obtenção dos créditos na disciplina Trabalho  
de Conclusão de Curso II do curso de Sistemas  
de Informação— Bacharelado.

Prof. Wilson Pedro Carli, Mestre - Orientador

**BLUMENAU  
2014**

**2014/2-18**

**CIDADÃO FISCAL: APLICATIVO PARA ABERTURA E  
ACOMPANHAMENTO DE PROCESSOS NA PREFEITURA  
MUNICIPAL DE BLUMENAU**

Por

**TUANE BRÜNING WESSLER**

Trabalho aprovado para obtenção dos créditos na disciplina de Trabalho de Conclusão de Curso II, pela banca examinadora formada por:

Presidente:

\_\_\_\_\_  
Prof. Wilson Pedro Carli, Mestre – Orientador, FURB

Membro:

\_\_\_\_\_  
Prof. Alexander Roberto Valdameri, Mestre – FURB

Membro:

\_\_\_\_\_  
Prof. Marcel Hugo, Mestre – FURB

Blumenau, 02 de dezembro de 2014.

Dedico este trabalho a todos que de alguma forma me incentivaram e apoiaram durante a sua elaboração, especialmente minha família e o meu namorado.

## **AGRADECIMENTOS**

A Deus, pelo seu imenso amor, graça e por sempre estar cuidando de mim.

À minha família, principalmente aos meus pais Leomar Wessler e Elsa Brüning Wessler, por me dar a vida e serem meus primeiros mestres, além do carinho, amor, apoio, incentivo e acesso a educação.

Ao meu irmão, Rodrigo Brüning Wessler pela compreensão e paciência.

Ao meu namorado, Jean Righez Bastos por sempre estar presente e me apoiando em todos os momentos, com muito amor, carinho e compreensão.

Aos meus amigos de curso Renan Moritz, Jonatan de Freitas e Daniel Ponick Botke que fizeram esses anos serem mais engraçados e divertidos.

Ao meu orientador, professor Wilson Pedro Carli, por ter acreditado na minha ideia e na conclusão deste trabalho.

A empresa FEESC pelo incentivo e pela oportunidade de realizar este trabalho.

Aos professores do Departamento de Sistemas e Computação da Universidade Regional de Blumenau por suas contribuições durante os semestres letivos.

Impossível é apenas uma grande palavra usada por gente fraca, que prefere viver no mundo como está em vez de usar o poder que tem para mudá-lo. Impossível não é um fato, é uma opinião. Impossível não é uma declaração, é um desafio. Impossível é hipotético. Impossível é temporário.

Muhammad Ali

## RESUMO

Este trabalho apresenta o aplicativo Cidadão Fiscal, que permite aos munícipes de Blumenau, uma forma rápida e desburocratizada de comunicação com a Prefeitura Municipal de Blumenau. Atualmente tem-se uma grande dificuldade enfrentada pelos cidadãos com a Prefeitura, junto a sua Ouvidoria, pois não existe um canal rápido para registrar reclamações e ou sugestões. Assim, o aplicativo desenvolvido para o sistema operacional Android permite a abertura e acompanhamento de processos junto à Ouvidoria do município. Os processos são enviados para um sistema *web*, o Cidadão Fiscal *Web*, que foi desenvolvido utilizando as tecnologias Java, ASP.NET, NHibernate e LINQ. Através desse sistema os atendentes e responsáveis pelos setores da Prefeitura conseguem identificar a reclamação ou sugestão enviada mais rapidamente. Com a disponibilização de funcionalidades que permitem o envio de fotos e da localização, tem-se um aplicativo que agiliza o registro e o acompanhamento de processos junto à Prefeitura Municipal de Blumenau, utilizando-se de *smartphones* e *tablets*.

Palavras-chave: Cidadão. Fiscalização. Prefeitura Municipal de Blumenau. Ouvidoria. Android.

## **ABSTRACT**

This work presents the Cidadão Fiscal application, that allows citizens of Blumenau, quick and non-bureaucratic form of communication with the City Hall of Blumenau. Currently it has a great difficulty faced by citizens with the City Hall, next to its ombudsman agency, as there is a quick channel for registering complaints or suggestions. Thus, the application developed for the Android operating system allows opening and monitoring processes by the municipal ombudsman agency. The cases are sent to a web service, the Cidadão Fiscal Web, which was developed using Java, ASP.NET, NHibernate, and LINQ technologies. Through this system the attendants and sectors responsible for City Hall can identify the complaint or suggestion and sent faster. With the availability of functions that allow you to send photos and location, there is an application that do the registration and monitoring processes with the Municipality of Blumenau, using smartphones and tablets.

**Keywords:** Citizen. Surveillance. City Hall of Blumenau. Ombudsman agency. Android.

## LISTA DE FIGURAS

Figura 1 - Abertura de Processos na Prefeitura Municipal de Blumenau .....	17
Figura 2 - Resumo das Rotinas da Ouvidoria.....	18
Figura 3 - Tela Fiscalize do Aplicativo Colab.....	21
Figura 4 - Tela de Reclamações do aplicativo Cidadera .....	22
Figura 5 - Tela do Mapa de todas as reclamações da cidade de Santos .....	23
Figura 6 - Tela de cadastro da localização do aplicativo Urbotip .....	24
Figura 7 - Diagrama de Casos de Uso "Aplicativo Móvel" .....	29
Figura 8 - Diagrama de Casos de Uso "Sistema Web" .....	30
Figura 9 - Diagrama de abertura de solicitação .....	31
Figura 10 - Distribuição e análise das solicitações.....	32
Figura 11 - Modelo de entidade e relacionamento do aplicativo .....	33
Figura 12 - Modelo entidade e relacionamento do sistema web.....	34
Figura 13 - Diagrama de classes do serviço wsCidadaoFiscal .....	36
Figura 14 - Diagrama de classes Web Service de recebimento de solicitações.....	37
Figura 15 - Eclipse com o <i>plugin Android Developer Tools</i> .....	38
Figura 16 - Estrutura de Arquivos do Aplicativo .....	39
Figura 17 - Diretório src com subdiretórios.....	40
Figura 18 - Biblioteca Ksoap2.....	41
Figura 19 - Diretório res .....	41
Figura 20 - Diretório assets .....	42
Figura 21 - Arquivo AndroidManifest.xml .....	42
Figura 22 - Comunicação com o Web Service - GravarUsuario.....	43
Figura 23 - Consumo do Web Service .....	44
Figura 24 - Código fonte responsável por carregar localização .....	44
Figura 25 - Plataforma de desenvolvimento do Visual Studio 2013.....	45
Figura 26 - Método responsável por retornar pessoa .....	46
Figura 27 - Estrutura de diretórios e arquivos .....	46
Figura 28 - Código fonte relação do setor com categoria.....	47
Figura 29 - Código fonte conclusão solicitação .....	48
Figura 30 - Tela inicial do aplicativo.....	49
Figura 31 - Tela de Cadastro do usuário.....	49

Figura 32 - Tela de cadastro do usuário .....	50
Figura 33 - Mensagem de confirmação de cadastro .....	51
Figura 34 - <i>Menu</i> do aplicativo.....	51
Figura 35 - Tela de cadastro da foto .....	52
Figura 36 - Tela de cadastro da solicitação .....	53
Figura 37 - Mensagem de ativação do GPS .....	53
Figura 38 - Tela de localização da solicitação.....	54
Figura 39 - Tela de cadastro contendo o endereço .....	54
Figura 40 - Mensagem de aviso antes do envio da solicitação.....	55
Figura 41 - Tela de Solicitações Pendentes .....	56
Figura 42 - Menu com opções das solicitações pendentes .....	56
Figura 43 - Tela de Consulta das Solicitações.....	57
Figura 44 - Tela de editar perfil.....	58
Figura 45 - Tela de Reenvio de Senha.....	59
Figura 46 - <i>E-mail</i> de reenvio de senha .....	59
Figura 47 - Tela de <i>Login</i> do Cidadão Fiscal <i>Web</i> .....	60
Figura 48 - Tela inicial do Atendente .....	61
Figura 49 - Tela de distribuição de solicitações .....	61
Figura 50 - Tela distribuir solicitações .....	62
Figura 51 - Tela distribuir solicitação visualizar localização .....	62
Figura 52 - Tela distribuição solicitação deferimento .....	63
Figura 53 - Tela lista de solicitações para análise .....	64
Figura 54 - Tela de Análises.....	64
Figura 55 - Tela de Análise-1 .....	65
Figura 56 - Conclusão da análise.....	66
Figura 57 - Tela de consulta de solicitações.....	66

## LISTA DE QUADROS

Quadro 1 - Requisitos funcionais .....	27
Quadro 2 - Requisitos não funcionais .....	28
Quadro 3 - Comparativo entre aplicativos correlatos .....	68
Quadro 4 - Descrição dos casos de uso .....	73
Quadro 5 - Tabela de bairros de Blumenau .....	78
Quadro 6 - Tabela de categorias das solicitações .....	79
Quadro 7 - Tabela de logradouros de Blumenau .....	79
Quadro 8 - Tabela com os dados dos munícipes .....	80
Quadro 9 - Tabela com os dados dos usuários .....	81
Quadro 10 - Tabela de solicitações .....	81
Quadro 11 - Tabela de pessoas .....	83
Quadro 12 - Tabela de bairros da cidade .....	84
Quadro 13 - Tabela de Endereço .....	84
Quadro 14 - Tabela de usuários cadastrados pelo aplicativo.....	84
Quadro 15 - Tabela de situações do processo.....	85
Quadro 16 - Tabela de tipos de protocolo .....	86
Quadro 17 - Tabela de tipos de processo.....	86
Quadro 18 - Tabela de assuntos do processo.....	86
Quadro 19 - Tabela de setores .....	87
Quadro 20 - Tabela de setores e assuntos.....	87
Quadro 21 - Tabela de processos.....	88
Quadro 22 - Tabela de arquivos .....	89
Quadro 23 - Tabela de endereço de fiscalizações.....	90
Quadro 24 - Tabela de distribuição de processos .....	91
Quadro 25 - Tabela de usuário do ERP .....	91
Quadro 26 - Tabela de análises do processo.....	92

## LISTA DE SIGLAS

ADT – *Android Developer Tools*

API – *Application Programming Interface*

FAEMA – Fundação do Meio Ambiente de Blumenau

FEESC – Fundação de Ensino e Engenharia de Santa Catarina

GRP – Sistema de Gestão Integrada

IDE – *Integrated Development Environment*

LINQ – *Language Integrated Query*

SOAP – *Simple Object Access Protocol*

UML - *Unified Modeling Language*

XML – *eXtensible Markup Language*

## SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>12</b>
1.1 OBJETIVOS DO TRABALHO .....	13
1.2 ESTRUTURA DO TRABALHO .....	13
<b>2 FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>15</b>
2.1 OUVIDORIA NO SETOR PÚBLICO BRASILEIRO .....	15
2.2 OUVIDORIA NA PREFEITURA MUNICIPAL DE BLUMENAU .....	16
2.2.1 Prefeitura Municipal de Blumenau .....	16
2.2.2 Sistema atual .....	17
2.2.3 A tecnologia na Ouvidoria .....	19
2.3 TRABALHOS CORRELATOS .....	20
<b>3 DESENVOLVIMENTO DO APLICATIVO E DO SISTEMA WEB.....</b>	<b>25</b>
3.1 LEVANTAMENTO DE INFORMAÇÕES .....	25
3.2 ESPECIFICAÇÃO .....	26
3.2.1 Requisitos do Aplicativo e do sistema <i>Web</i> .....	26
3.2.2 Diagrama de Caso de Uso .....	28
3.2.3 Diagrama de atividades do registro da solicitação .....	30
3.2.4 Diagrama de atividades da distribuição e encaminhamento das solicitações .....	31
3.2.5 Modelo de entidade e relacionamento.....	33
3.2.6 Diagrama de classes .....	35
3.3 IMPLEMENTAÇÃO .....	38
3.3.1 Técnicas e ferramentas utilizadas.....	38
3.3.2 Operacionalidade da implementação .....	48
3.4 RESULTADOS E DISCUSSÃO .....	67
<b>4 CONCLUSÕES.....</b>	<b>69</b>
4.1 EXTENSÕES .....	70
<b>REFERÊNCIAS .....</b>	<b>71</b>
<b>APÊNDICE A – Descrição dos Casos de Uso .....</b>	<b>73</b>
<b>APÊNDICE B – Descrição do Dicionário de Dados .....</b>	<b>78</b>

## 1 INTRODUÇÃO

Os aplicativos móveis tem se desenvolvido tecnologicamente cada vez mais e isso tem ajudado as pessoas com algumas tarefas simples do dia-a-dia. Segundo Monk (2013, pag. 6), a migração em grande quantidade dos usuários de computadores pessoais (PCs) para a utilização de dispositivos móveis está derrubando uma grande barreira existente na possibilidade de acessar conteúdos digitais de qualquer lugar e a qualquer hora.

Schmidt e Cohen (2013) afirmam que “todo cidadão armado com um dispositivo móvel é uma testemunha e um investigador em potencial, mais atento do que qualquer autoridade policial e pronto para documentar injustiças”. Desta forma, o cidadão tem buscado os seus direitos principalmente através das redes sociais, nas quais é possível expressar sua opinião, assim como colocar imagens e vídeos para enfatizar o que está sendo compartilhado com todos os usuários desta rede. A dificuldade é que a maioria da população acaba se comovendo e aderindo a várias correntes existentes, mas toda essa repercussão na maioria das vezes não chega aos administradores públicos que realmente podem fazer algo pela população.

Segundo Butzer (2011), um comentário negativo sobre um produto ou serviço corre como um rastilho de pólvora nas redes sociais, sendo que em instantes milhares de pessoas passam a expor sua posição sobre o comentário. No município de Blumenau, estado de Santa Catarina, os munícipes também têm aderido a essas reclamações, ao compartilhamento de informações e aos problemas da cidade pelas redes sociais. Alguns munícipes conhecem a existência da Ouvidoria na Prefeitura Municipal de Blumenau e acabam fazendo as sugestões e reclamações para a mesma. Mas a grande maioria dos munícipes trabalha e não tem tempo para entrar em contato com a ouvidoria, uma vez que a mesma só atende em horário comercial. Desta forma o presente trabalho tem como objetivo facilitar a comunicação entre os munícipes de Blumenau com a prefeitura.

A maioria dos setores da Prefeitura Municipal de Blumenau conta com a opinião do munícipe, que através da ouvidoria estabelece um canal de comunicação entre ambas as partes. A grande dificuldade enfrentada pelos cidadãos de Blumenau é a falta de um canal com horário de atendimento ampliado para a comunicação com a prefeitura. Atualmente as formas existentes de se realizar uma reclamação ou solicitação/sugestão para a ouvidoria municipal são através do telefone, *e-mail* ou ir até a Ouvidoria, sendo que por telefone ou na

Praça do Cidadão o horário de atendimento é de segunda a sexta-feira, das 8 horas às 17 horas.

Já existem nas lojas virtuais dos sistemas operacionais de dispositivos móveis alguns aplicativos propostos a ajudar o cidadão a relatar as dificuldades enfrentadas nos municípios do Brasil. A grande maioria destes aplicativos apenas acaba gerando registros para a própria aplicação. Sendo assim, os dados não são enviados a nenhum órgão público que realmente possa resolver as solicitações propostas.

Baseado nessas informações verificou-se tal dificuldade e a necessidade de melhoria em uma área tão importante. A finalidade é facilitar a abertura e o acompanhamento dos processos no setor de Ouvidoria da Prefeitura Municipal de Blumenau.

## 1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é apresentar o desenvolvimento de um aplicativo que permita aos cidadãos uma maior disponibilidade de atendimento para se comunicar com a Ouvidoria da Prefeitura Municipal de Blumenau. Como objetivos secundários tem-se:

- a) disponibilizar aplicação para abertura e acompanhamento de processos no setor de Ouvidoria;
- b) permitir o envio de uma foto para agregar ao processo;
- c) permitir a indicação da posição geográfica para agregar ao processo;
- d) permitir o acompanhamento dos processos;
- e) manter um histórico dos processos.

## 1.2 ESTRUTURA DO TRABALHO

No primeiro capítulo tem-se a introdução ao tema principal deste trabalho com a apresentação da justificativa e dos objetivos.

No segundo capítulo apresenta-se a fundamentação teórica pesquisada sobre a Ouvidoria no Setor Público Brasileiro, a Ouvidoria na Prefeitura Municipal de Blumenau, o Sistema Atual e a tecnologia na Ouvidoria e trabalhos correlatos.

O terceiro capítulo apresenta o desenvolvimento do aplicativo e do sistema *web* iniciando-se com o levantamento de informações, tendo na sequência a especificação, a implementação e por fim resultados e discussões.

No quarto capítulo tem-se as conclusões deste trabalho bem como apresentam-se sugestões para trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda assuntos a serem apresentados nas seções a seguir, tais como a Ouvidoria no Setor Público Brasileiro, a Ouvidoria na Prefeitura Municipal de Blumenau, o Sistema Atual encerrando-se com os trabalhos correlatos.

### 2.1 OUVIDORIA NO SETOR PÚBLICO BRASILEIRO

O *Ombudsman*, palavra que tem origem na Suécia no início do século XIX, é uma expressão nórdica. É a junção das palavras *ombud* que significa “representante”, “procurador” juntamente com a palavra *man* que significa homem. A figura do ouvidor já existia no Brasil Colônia, mas o mesmo não exercia uma representação do cidadão. Ao contrário, o mesmo reportava tudo o que ocorria na colônia. O padrão atual e mais clássico de ouvidor tem como função ser o representante do cidadão, buscar a defesa dos direitos. Foi primeiramente adotado no Brasil pela Prefeitura Municipal de Curitiba em 1986 (VISMONA et al., 2002, p. 24).

Segundo Acesso à Informação (2014), a Lei nº 15.527 sancionada em 2011, regulamenta o direito de acesso às informações públicas e criou mecanismos que possibilita a qualquer pessoa o recebimento de informações públicas dos órgãos e entidades. Vismona et al.(2002, p. 23) afirmam que “a Administração Pública pertence ao cidadão, pois toda a força de trabalho está voltada para promover a melhoria da qualidade de vida da comunidade”. Assim, Silva e Hüskes (2007, p.2) sustentam que quando o cidadão faz denúncias ou sugestões o mesmo participa da gestão administrativa e assim exerce seu papel de cidadão.

A difícil comunicação entre o governo, os servidores e o público é o motivo para as frequentes reclamações que surgem, sendo que o que é esperado do governo é apenas o cumprimento dos seus deveres de forma eficiente, com justiça qualidade e democracia. O principal objetivo da ouvidoria é aproximar o administrador público e o cidadão para que ambos percebam a importância da opinião e do registro de uma reivindicação. O cidadão que utiliza o serviço de ouvidoria além de exercitar a cidadania consegue vislumbrar a melhoria da qualidade dos serviços e a solução dos problemas que são executados pelo setor público (VISMONA et al., 2002, p. 24) .

Ferreira e Santos (2011) acrescentam que, “a essência do trabalho da ouvidoria é, de forma bastante resumida, ajudar a transformar em oportunidade as ameaças surgidas do relacionamento da organização com o cidadão, humanizando a relação e perseguindo o equilíbrio nesse processo”.

## 2.2 OUVIDORIA NA PREFEITURA MUNICIPAL DE BLUMENAU

A seguir é apresentado como a ouvidoria é estruturada junto a Prefeitura Municipal de Blumenau, bem como a tecnologia utilizada para simplificar o processo.

### 2.2.1 Prefeitura Municipal de Blumenau

A cidade de Blumenau foi fundada no ano de 1850, pelo filósofo alemão Hermann Bruno Otto Blumenau. A mesma é conhecida em todo o Brasil pela influência germânica na cultura e história (PREFEITURA MUNICIPAL DE BLUMENAU, 2014a).

O município de Blumenau foi criado pela Lei 860 de 4 de fevereiro de 1880 quando o governador da Província de Santa Catarina declarou extinta a Colônia Blumenau. Com esta lei o município começou a contar com um representante eleito pelo voto popular, contudo por causas das cheias ocorridas naquele ano, a posse ocorreu apenas três anos depois (PREFEITURA MUNICIPAL DE BLUMENAU, 2014a).

Segundo Prefeitura Municipal de Blumenau (2014b), a cidade de Blumenau está localizada ao nordeste de Santa Catarina e possui uma área total de 519,8 Km<sup>2</sup>. A cidade possui a área urbana dividida em 35 bairros, incluindo 2 distritos, o do Garcia e o da Vila Itoupava. Atualmente a Prefeitura Municipal de Blumenau conta com 28 secretarias que são responsáveis por executar as atividades que a elas forem responsáveis de realizar a análise e execução (PREFEITURA MUNICIPAL DE BLUMENAU, 2014c).

De acordo com o Instituto Brasileiro de Geografia e Estatística (2014), na última pesquisa realizada na cidade de Blumenau no ano de 2010 a população era de 309.011, sendo que a estimativa para o ano de 2014 é a que população chegue a 334.002.

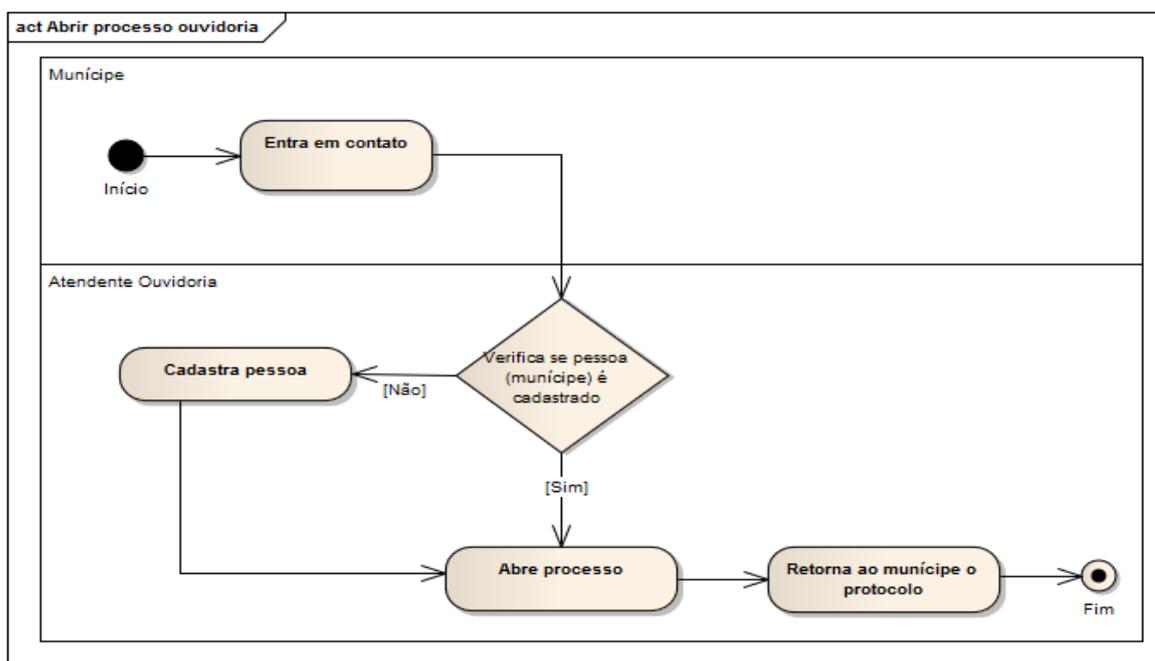
## 2.2.2 Sistema atual

Atualmente as formas existentes de se realizar uma reclamação ou sugestão para a Ouvidoria da Prefeitura Municipal de Blumenau são através do telefone, *e-mail* ou ir até a Praça do Cidadão. Após esse procedimento, o servidor da Prefeitura Municipal de Blumenau abre um processo e encaminha para o setor responsável como, por exemplo, a Fundação Municipal do Meio Ambiente de Blumenau (Faema), a Vigilância Sanitária, os Serviços Urbanos entre outros.

Spengler (2014) afirma que, todos os processos que são abertos pela ouvidoria geram protocolos através do Sistema de Gestão Integrada. De acordo com Thema Informática (2014) o Sistema de Gestão Integrada (GRP) tem como objetivo “fazer a convergência das diversas áreas existentes dentro das instituições públicas para um único ambiente de trabalho.”

De acordo com Spengler (2014), o GRP é utilizado em todos os setores da prefeitura. Quando um processo é aberto na ouvidoria através do GRP é gerado um protocolo e este protocolo é informado para o munícipe que pode então acompanhar o processo, também através do telefone, por *e-mail* ou presencialmente na Ouvidoria. Spengler (2014) afirma ainda que, as sugestões ou reclamações são tratadas como solicitações. Na Figura 1 está uma representação da abertura de processos na Ouvidoria da Prefeitura Municipal de Blumenau.

Figura 1 - Abertura de Processos na Prefeitura Municipal de Blumenau



Segundo Silva e Hüskes (2007) as principais atividades e rotinas da Ouvidoria da Prefeitura Municipal de Blumenau antes da implantação do GRP são:

- a) objetivos do setor: a satisfação, o respeito e o melhor atendimento ao cidadão, transparência do poder público, disponibilizar informações gerenciais ao Prefeito, conhecer os problemas e prioridades de todas as regiões do município, formar um banco de dados;
- b) resumo de rotinas: todos os atendimentos/chamados realizados pela Ouvidoria da Prefeitura Municipal de Blumenau são registrados no sistema disponível atualmente e no final de cada mês os dados são tabulados e apresentados ao Gabinete do Prefeito.

Na Figura 2 tem-se uma tabulação dos atendimentos dos meses de janeiro a agosto do ano de 2006 da Prefeitura Municipal de Blumenau. Os dados já estão tabulados e pode-se notar que os meses do início do ano não possuem uma grande quantidade de atendimentos, devido principalmente às férias neste período.

Figura 2 - Resumo das Rotinas da Ouvidoria

Ocorrência	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Total	%
Trâmite diário	491	462	757	609	799	495	368	616	4597	21,28
Arquivamento	290	601	509	355	633	484	159	429	3460	16,01
Proc. Novos por telefone	273	314	525	325	328	324	286	291	2666	12,34
Ligações diárias	140	194	380	290	307	190	114	192	1807	8,36
Informações de processos	205	104	341	259	243	240	164	162	1718	7,95
Informações diversas	109	33	195	167	106	171	161	138	1080	5,00
Engano	142	25	213	102	93	133	126	179	1013	4,69
Informações nº de telefone	112	24	164	122	140	103	120	136	921	4,26
Agradecimento	88	30	231	169	140	100	78	71	907	4,20
Proc. Novos pessoalmente	84	85	126	80	92	66	89	125	747	3,46
Informações pessoalmente	76	32	164	67	92	67	30	90	618	2,86
Ligações diárias não atendidas	77	59	161	166	129	96	46	115	849	3,93
Resposta de e-mail	51	16	25	121	66	33	69	64	445	2,06
Informações secretarias	39	19	72	41	29	33	41	43	317	1,47
Reencaminhamento	29	5	55	37	27	44	52	39	288	1,33
Processo p/ e-mail	9	13	18	57	29	14	18	14	172	0,80
<b>TOTAL</b>	<b>2215</b>	<b>2016</b>	<b>3936</b>	<b>2967</b>	<b>3253</b>	<b>2593</b>	<b>1921</b>	<b>2704</b>	<b>21605</b>	<b>100,00</b>

Fonte: Silva e Hüskes (2007).

Os principais pontos negativos relatados pelos funcionários da Ouvidoria da Prefeitura Municipal de Blumenau, segundo Silva e Hüskes (2007, p.15), são o espaço físico inadequado, ruídos, a falta de atendentes e de linhas telefônicas e a demora da resposta de outros setores.

### 2.2.3 A tecnologia na Ouvidoria

O avanço da informática tem permitido um grande desenvolvimento de práticas na gestão pública. Toda a informatização que existe atualmente aos serviços públicos não aumentou apenas a eficiência dos processos mais também a acessibilidade aos usuários, tornando assim as informações mais transparentes para o cidadão (CAIÇARA JUNIOR; PARIS, 2008). Já para Silva e Hüskes (2007, p.6), “as tecnologias apresentam inúmeros benefícios para que a ouvidoria seja funcional e tenha condições de oferecer informações claras, objetivas e forneça relatórios criados para os diversos tipos de controle”.

Um dos principais condicionantes para a implantação de um sistema de ouvidoria é a existência de meios de acesso para o público interessado que sejam fáceis e que possam estimular a sua participação (PINHO, 2002, p.27). As tecnologias são fundamentais para as gestões, pois as mesmas são responsáveis pelo controle e o fácil acesso a informações muito importantes, além de disponibilizar as mesmas em um tempo hábil e assim proporcionar relatórios confiáveis (SILVA; HÜSKES, 2007, p.6).

A Fundação de Ensino e Engenharia de Santa Catarina (FEESC), um dos fornecedores de sistemas a Prefeitura Municipal de Blumenau, oferece atualmente alguns sistemas para a automatização de processos em diferentes setores. Um deles, por exemplo, o Portal do Cidadão, é onde o munícipe conta com diversos serviços como, a emissão de diferentes alvarás, consulta de viabilidade, emissão da segunda via do Imposto Predial e Territorial Urbano (IPTU). Já para os setores da Fundação Municipal de Meio Ambiente (Faema) e Vigilância Sanitária são oferecidos sistemas que possibilitam a fiscalização, a cobrança de multas e a abertura de processo na Prefeitura Municipal de Blumenau.

A FEESC é uma instituição sem fins lucrativos que foi fundada em 1966 no município de Florianópolis, estado de Santa Catarina. A mesma é credenciada pelo Ministério da Educação e da Ciência e Tecnologia e Inovação como fundação de apoio a Universidade Federal de Santa Catarina (UFSC) e também ao Instituto Federal de Santa Catarina (IF-SC), tendo assim entre suas atividades o gerenciamento de projetos de ensino, pesquisa e extensão

e desenvolvimento institucional, científico e tecnológico (FUNDAÇÃO DE ENSINO E ENGENHARIA DE SANTA CATARINA, 2014).

A FEESC tem como objetivo de trabalho, aproximar o conhecimento científico e tecnológico nas diferentes realidades da sociedade, visando assim contribuir para a melhoria na qualidade de vida de todos. A fundação trabalha no apoio de diferentes projetos, principalmente os ligados a UFSC, mas a mesma possui outro portfólio que são os projetos e sistemas desenvolvidos para a Gestão Pública, entre eles o Sistema de Administração Tributária e Urbana (SIATU), a Vigilância Sanitária (VISA) e o portal da Faema. Todos estes projetos e sistemas foram desenvolvidos para a Prefeitura Municipal de Blumenau para então ajudar a informatização de alguns dos setores da mesma (FUNDAÇÃO DE ENSINO E ENGENHARIA DE SANTA CATARINA, 2014)

Através da FEESC deslumbrou-se a possibilidade de desenvolvimento deste trabalho, assim sendo, a fundação prestou apoio teórico e estrutural dos sistemas já implementados na prefeitura, assim como da estrutura do sistema GRP para o desenvolvimento deste.

### 2.3 TRABALHOS CORRELATOS

Pode-se citar como trabalhos correlatos alguns aplicativos disponíveis nas lojas *online* de distribuição de aplicativos, como o Google Play (sistema operacional Android) e na App Store (sistema operacional iOS). Entre os principais estão o Colab, Cidade Legal, Cidadera, Urbotip.

UOL TECNOLOGIA (2014) esclarece que o Colab é um aplicativo desenvolvido por brasileiros e já foi vencedor do prêmio internacional AppMyCity no ano de 2013. O objetivo deste aplicativo é abrir um espaço para o usuário propor soluções e avaliar serviços tanto públicos como privados. O Colab permite ao cidadão realizar críticas, propostas e avaliações do poder público, e auxilia as prefeituras a lidarem com as reivindicações (CORONATO, 2014).

Ainda segundo Coronato (2014), o Colab funciona como uma rede social dedicada a questões urbanas e manifestações de cidadania. Em vez de “curtir” ou “tuitar”, o usuário conta com as opções de “fiscalize”, “proponha” e “avalie”, depois de identificar a cidade a que se refere. Um administrador público pode participar como usuário comum. A prefeitura interessada pode assinar um termo de cooperação com o Colab e dispor de recurso, mais

sofisticados – ferramentas de comunicação, relatórios, estatísticas e mapas, feitos com base nos comentários. As prefeituras dos municípios de São Paulo, no estado de São Paulo e Recife, no estado de Pernambuco, assinaram esse termo.

Na Figura 3 tem-se a tela de cadastro de fiscalização do Colab, logo é possível perceber que a interface desenvolvida é prática e de fácil entendimento.

Figura 3 - Tela Fiscalize do Aplicativo Colab



Fonte: Google Play (2014).

O Cidadera tem como objetivo chamar atenção das prefeituras dos problemas enfrentados pelos cidadãos e para que busque soluções. O mesmo permite compartilhar as reclamações em redes sociais, e outros usuários das redes sociais podem apoiar a reclamação (UOL TECNOLOGIA, 2014).

Na Figura 4 tem-se a tela com todas as reclamações já publicadas por um usuário cadastrado no aplicativo Cidadera. É possível visualizar ao lado direito da imagem que o aplicativo traz um mapa sinalizando a localização das reclamações cadastradas pelo mesmo

usuário, assim como o aplicativo permite também selecionar uma das reclamações e visualizá-la no mapa.

Figura 4 - Tela de Reclamações do aplicativo Cidadera



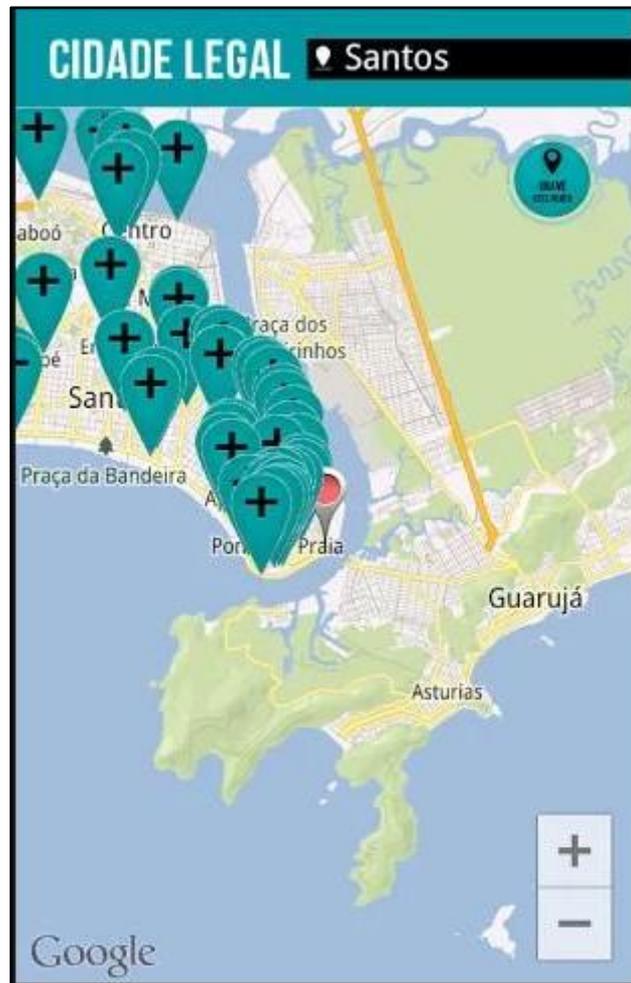
Fonte: Google Play (2014).

Outro aplicativo desenvolvido foi o Cidade Legal, que tem como objetivo registrar um problema na cidade. Tem como diferencial que todos os problemas relatados passam a fazer parte de um mapa e podem ser conferidos por quem possui o aplicativo instalado, em toda redondeza onde está sendo utilizado o mesmo e houver alguma reclamação (UOL TECNOLOGIA, 2014).

Na Figura 5 é possível visualizar que a tela apresenta um filtro por cidades e que neste caso a cidade pesquisada é o município de Santos, estado de São Paulo. O mesmo apresenta o mapa da cidade e os marcadores que representam todas as reclamações que já foram

cadastradas utilizando o aplicativo. É possível selecionar as marcações para saber alguns detalhes da mesma, como endereço e a descrição da reclamação.

Figura 5 - Tela do Mapa de todas as reclamações da cidade de Santos



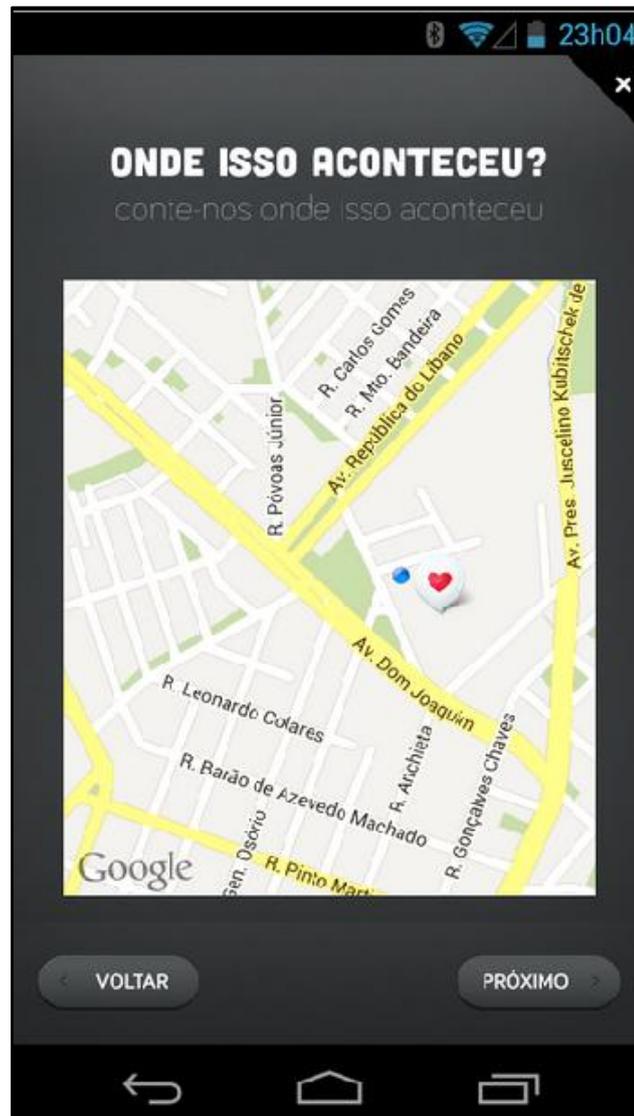
Fonte: Google Play (2014).

E por fim, o aplicativo Urbotip foi desenvolvido para que os usuários possam compartilhar questões da cidade onde vivem com o mundo inteiro, tendo como objetivo mostrar para as autoridades quais lugares precisam de uma solução imediatamente. Para utilizar o aplicativo é muito simples, basta o cidadão se deparar com algo, criar uma causa contando onde estas questões estão acontecendo e através da mídia social a reclamação tende a ser mais forte (URBOTIP, 2014).

A Figura 6 mostra a tela de cadastro da localização de uma causa. O aplicativo através da utilização do Sistema de Posicionamento Global (GPS) identifica a localização e apresenta

para o usuário da aplicação sendo que o mesmo confirma a localização apresentada pelo aplicativo.

Figura 6 - Tela de cadastro da localização do aplicativo Urbotip



Fonte: Google Play (2014).

### 3 DESENVOLVIMENTO DO APLICATIVO E DO SISTEMA WEB

Neste capítulo estão descritos particularidades técnicas do aplicativo e do sistema *web* desenvolvidos, tais como o levantamento de informações com a descrição dos mesmos, a especificação com a apresentação dos requisitos funcionais e não funcionais, os diagramas de atividades, o diagrama de casos de uso e suas descrições. Na sequência tem-se a operacionalidade do aplicativo e do sistema, bem como são apresentados os resultados e discussões.

#### 3.1 LEVANTAMENTO DE INFORMAÇÕES

O Cidadão Fiscal é um aplicativo na plataforma Android para a abertura e acompanhamento de processos no setor de Ouvidoria da Prefeitura Municipal de Blumenau, facilitando assim a comunicação entre ambas as partes.

O aplicativo registra as solicitações e as reclamações, sendo que no cadastro da mesma é informada uma descrição sobre o que está motivando esta reclamação ou a sugestão. Também é necessário selecionar uma categoria, informando em que área o problema está ocorrendo como, por exemplo, se for um problema de semáforos, deve ser encaminhado para a categoria de trânsito. Caso o usuário não souber em que categoria a solicitação se encaixa, poderá escolher a opção “Outra”, onde o atendente da prefeitura irá realizar posteriormente a correta classificação.

O aplicativo possibilita ainda salvar as coordenadas geográficas na reclamação ou sugestão, assim como a opção de agregar uma foto do que está sendo reivindicado. Sendo assim, após todos os campos serem preenchidos a reivindicação pode ser enviada à ouvidoria do município. A solicitação é enviada via *web service* para o Cidadão Fiscal Web e o mesmo realiza a integração com o sistema GRP, sendo que esta comunicação se dá pelo repasse das informações para banco de dados do sistema. Por motivos de segurança, para a realização do desenvolvimento do presente trabalho foi criada uma nova base de dados contendo apenas as tabelas que serão utilizadas no presente trabalho, mas mantendo o padrão tanto de tabelas como de nomenclaturas, assim como as ligações das tabelas, mudando apenas o nome de tabelas e campos.

Todas as sugestões ou reclamações enviadas através do aplicativo passarão pelo setor de Ouvidoria da Prefeitura Municipal de Blumenau, onde os atendentes farão uma triagem para verificação do conteúdo das solicitações, descrições, imagens, bem como correta classificação. Após essa verificação o atendente irá confirmar o envio para o setor responsável ou então poderá indeferir a solicitação (rejeitar), finalizando assim a solicitação e gerando uma notificação com o motivo do não atendimento.

Caso o atendente do setor de Ouvidoria tenha deferido a solicitação, logo a mesma será encaminhada para o setor responsável. A partir desse momento fica sob responsabilidade do coordenador do setor e/ou dos analistas verificarem novamente a solicitação e realizarem a análise da solicitação, sendo que o analista poderá realizar varias análises.

Atualmente todas as solicitações que são realizadas por telefone, *e-mail* ou atendimentos presenciais da Ouvidoria são registrados e protocolados no sistema. As solicitações enviadas através do aplicativo também receberão o protocolo, após a verificação e aceite da solicitação pela Ouvidoria. Desta forma a consulta do andamento da solicitação poderá ser verificada pelo próprio aplicativo.

## 3.2 ESPECIFICAÇÃO

Nesta seção é apresentada a especificação do aplicativo para dispositivos móveis, Cidadão Fiscal e também do sistema *web* (acessado pela *Internet*), Cidadão Fiscal *Web*. Para a modelagem utilizou-se da ferramenta Enterprise Architect. Utiliza-se ainda a notação *Unified Modeling Language* (UML) para a representação dos diagramas de casos de uso, de atividades e classes.

### 3.2.1 Requisitos do Aplicativo e do sistema *Web*

O Quadro 1 apresenta os requisitos funcionais previstos para o aplicativo (Cidadão Fiscal) e o sistema *web* (Cidadão Fiscal *Web*) e sua rastreabilidade, ou seja, vinculação com o(s) caso(s) de uso associado(s).

Quadro 1 - Requisitos funcionais

<b>Requisitos Funcionais</b>	<b>Caso de Uso</b>
RF01: O aplicativo <i>mobile</i> deverá permitir o cadastramento de usuários contendo nome, CPF, endereço, data de nascimento, <i>login</i> e senha.	UC01, UC02
RF02: O aplicativo <i>mobile</i> deverá permitir o usuário efetuar o <i>login</i> no aplicativo através do fornecimento de usuário e senha corretos.	UC03
RF03: O aplicativo <i>mobile</i> deverá possibilitar o registro de uma solicitação contendo descrição, categoria, foto ou imagem e posição geográfica obtida através do GPS do dispositivo móvel.	UC04
RF04: O aplicativo <i>mobile</i> deverá permitir ao usuário salvar a solicitação <i>offline</i> e enviar a mesma quando a conexão com a internet estiver disponível, sendo que as mesmas ficarão como solicitações pendentes.	UC04, UC06
RF05: O aplicativo <i>mobile</i> deverá permitir ao usuário visualizar uma solicitação pendente antes de enviá-la.	UC06
RF06: O aplicativo <i>mobile</i> deverá permitir que o usuário possa excluir as solicitações pendentes de envio.	UC07
RF07: O aplicativo <i>mobile</i> deverá permitir ao usuário a consulta das solicitações realizadas.	UC05
RF08: O sistema (módulo <i>web</i> ) deverá disponibilizar um <i>web service</i> para o recebimento das solicitações oriundas do aplicativo <i>mobile</i> .	UC08
RF09: O sistema (módulo <i>web</i> ) deverá disponibilizar uma interface para que o Atendente da Ouvidoria faça uma triagem das solicitações antes de encaminhar para o setor responsável, podendo já indeferir a solicitação.	UC08
RF10: O sistema (módulo <i>web</i> ) deverá permitir ao Atendente da Ouvidoria a inserção de um texto esclarecendo o motivo de uma rejeição.	UC09
RF11: O sistema (módulo <i>web</i> ) deverá disponibilizar uma interface para os Analistas onde os mesmos poderão realizar várias análises das solicitações e possuir um campo para descrever a aprovação ou rejeição final das solicitações.	UC09

No Quadro 2 são apresentados os requisitos não funcionais utilizados no desenvolvimento do aplicativo e o sistema *web*.

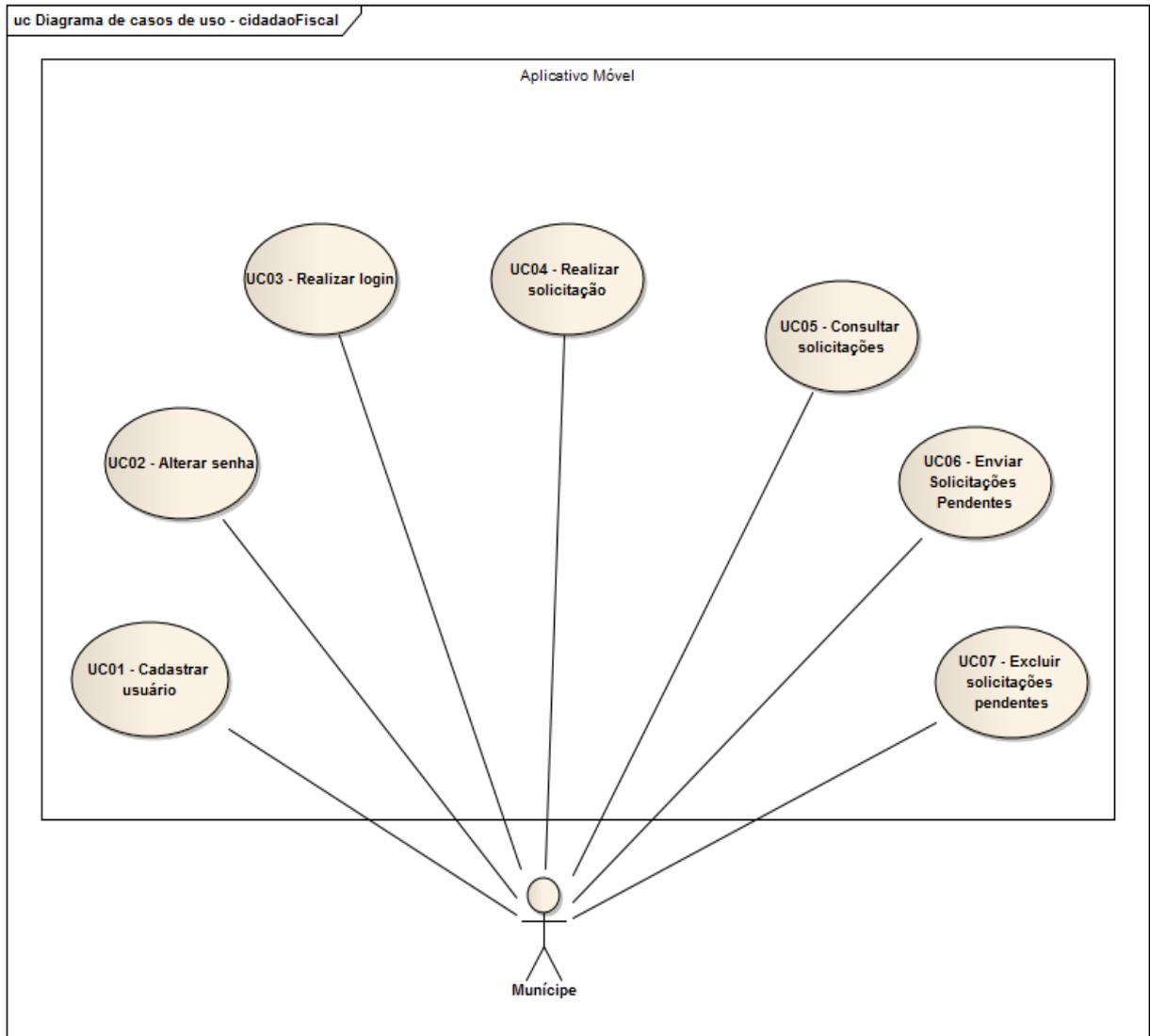
Quadro 2 - Requisitos não funcionais

<b>Requisitos Não Funcionais</b>
RNF01: O aplicativo <i>mobile</i> deverá ser desenvolvido na plataforma Android sendo compatível a versão 4.0 ou superior.
RNF02: O aplicativo <i>mobile</i> deverá utilizar o banco de dados SQLite na versão 2.4.5.
RNF03: O aplicativo <i>mobile</i> deverá utilizar a API do Google Maps versão 3.
RNF04: O aplicativo <i>mobile</i> deverá transmitir as solicitações via <i>web service</i> para o módulo <i>web</i> .
RNF05: O aplicativo <i>mobile</i> deverá ser desenvolvido na linguagem Java para a plataforma Android para os serviços <i>web</i> .
RNF06: O sistema (módulo <i>web</i> ) será desenvolvido na linguagem C# (ASP.NET).

### 3.2.2 Diagrama de Caso de Uso

Esta subseção apresenta na Figura 7 e na Figura 8, os diagramas de casos de uso do aplicativo, sendo que o detalhamento dos casos de uso encontram-se no Apêndice A. A Figura 7 representa o caso de uso do fluxo principal.

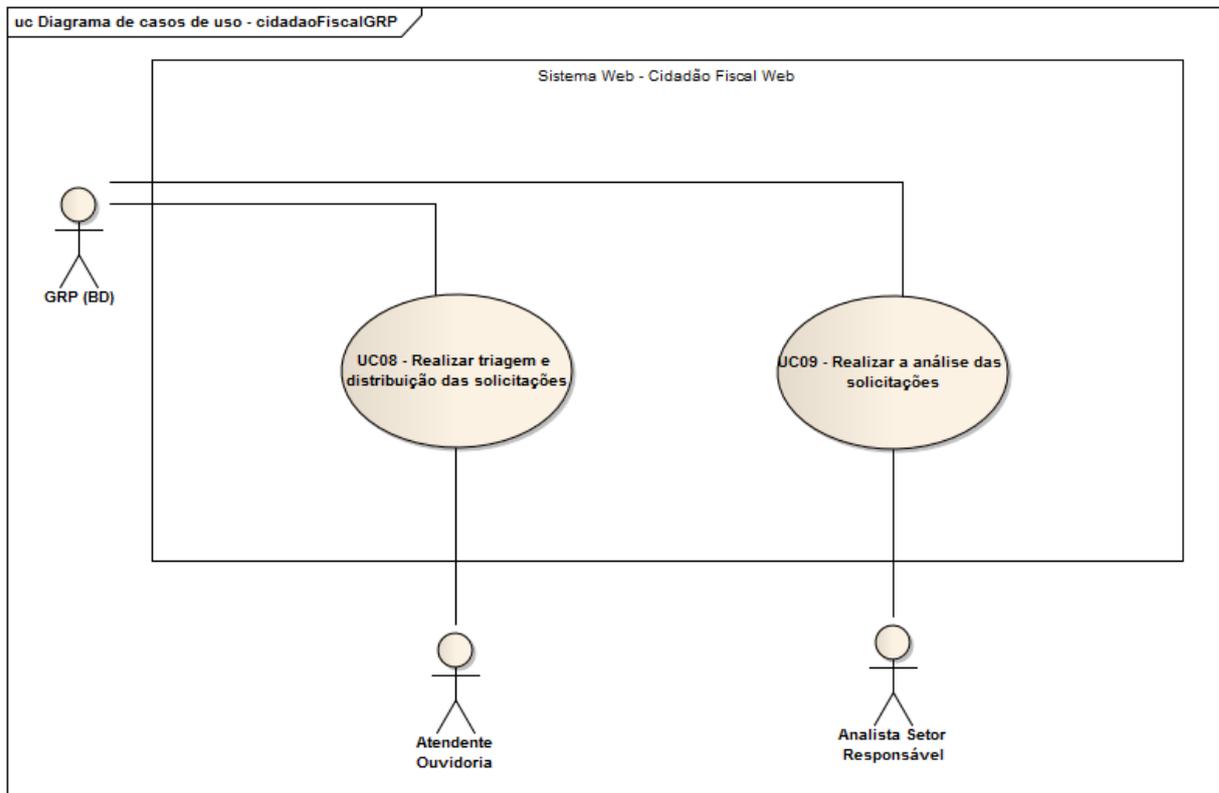
Figura 7 - Diagrama de Casos de Uso "Aplicativo Móvel"



O munícipe é responsável por realizar a fiscalização de um problema ou sugestão, sendo que após instalar o aplicativo no dispositivo móvel e realizar o cadastro, o munícipe poderá optar por realizar uma fiscalização, onde essa fiscalização se tornará uma solicitação que será enviada para a prefeitura. Poderá atualizar informações do seu perfil, consultar processos pendentes, consultar processos enviados e excluir solicitações pendentes.

A Figura 8 representa a comunicação direta do aplicativo com o sistema *Web*, onde todas as informações que foram enviadas através do aplicativo são salvas no banco de dados do GRP, realizando assim a integração com o mesmo.

Figura 8 - Diagrama de Casos de Uso "Sistema Web"



Todas as solicitações enviadas pelo munícipe ao sistema *web* são recebidas e gravadas no banco de dados, onde o Atendente da Ouvidoria é responsável por realizar a primeira triagem para verificar se os dados são coerentes e condizem com a solicitação. Após a realização da triagem se os dados estiverem incorretos a mesma é indeferida e um texto explicando o motivo é enviado ao munícipe. Se os dados estiverem corretos a solicitação é encaminhada para o setor responsável onde será analisada por um Analista do setor responsável e o mesmo fica responsável por solucionar a solicitação e a concluir escrevendo um pequeno texto sobre a solicitação.

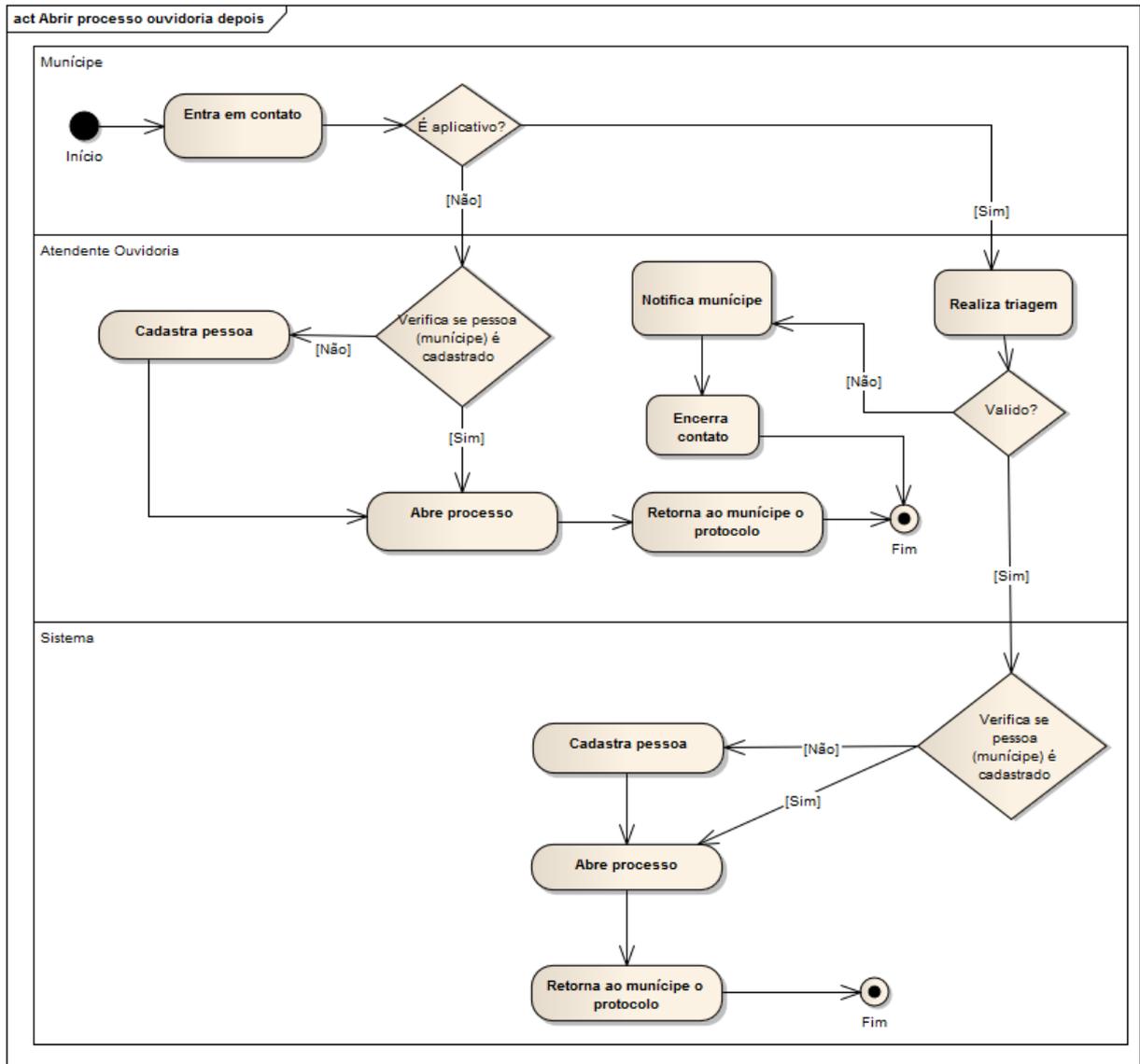
A seguir apresenta-se os diagramas de atividades necessários para um melhor entendimento do fluxo básico do funcionamento do sistema, entre elas: o registro da solicitação e a distribuição e o encaminhamento das solicitações,

### 3.2.3 Diagrama de atividades do registro da solicitação

Na Figura 9 é apresentado o diagrama de atividades do registro da solicitação. O fluxo mostra como funciona a abertura tanto diretamente das solicitações que são abertas na Praça

do Cidadão, localizada na Prefeitura, assim como as solicitações abertas através do aplicativo desenvolvido, lembrando que internamente para os funcionários da Prefeitura a solicitação é conhecida como um processo.

Figura 9 - Diagrama de abertura de solicitação

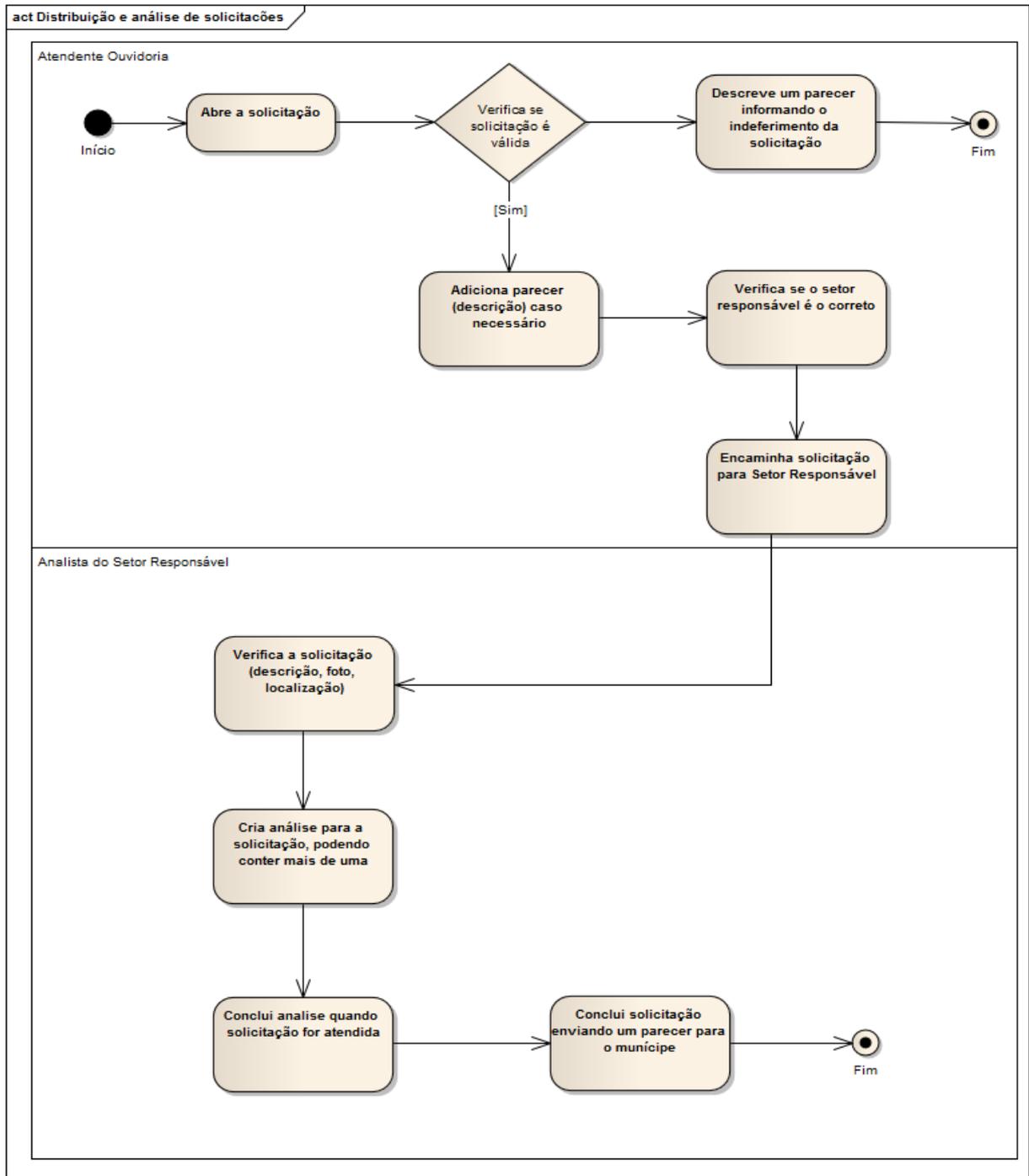


### 3.2.4 Diagrama de atividades da distribuição e encaminhamento das solicitações

Na Figura 10 é apresentado o diagrama de atividades da distribuição e encaminhamento das solicitações, onde o papel humano nesta etapa é desempenhado pelo ator Atendente da Ouvidoria que é responsável pela triagem das solicitações e a distribuição para o setor responsável, e pelo ator Analista do Setor Responsável que é responsável pela análise

das solicitações e a conclusão da análise.

Figura 10 - Distribuição e análise das solicitações



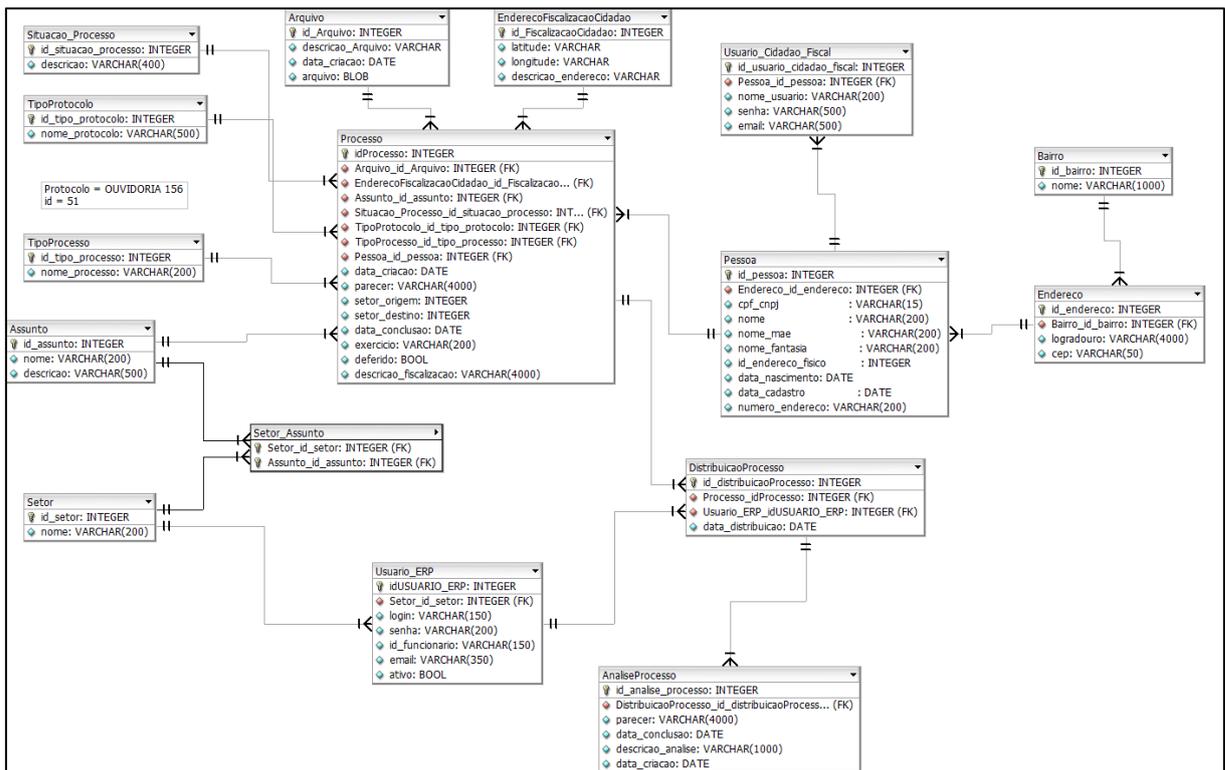
O GRP realiza a abertura e o encaminhamento das solicitações atualmente na Prefeitura Municipal de Blumenau e como neste trabalho a integração é realizada através da base de dados no sistema atual, o número dos processos será sequencial aos utilizados pelo sistema atual. O sistema *web* foi modelado desta forma imaginando-se a integração com



- c) **Município:** entidade que armazena os dados do município;
- d) **Usuario:** entidade que armazena os dados do usuário cadastrado no aplicativo;
- e) **Solicitacao:** entidade que armazena os dados das solicitações no aplicativo, esta entidade contém o campo de `endereco`, que é responsável por gravar a localização geográfica da solicitação e também o campo `fotoSolicitacao`, que é responsável por salvar a foto enviada pelo usuário do aplicativo a solicitação, entre outras;
- f) **Categoria:** entidade que armazena os dados das categorias referente a solicitação.

Na Figura 12 é exibido o modelo de entidade e relacionamento (MER) do sistema *web* desenvolvido. Para o MER utilizou-se também a notação *cross foot* e a ferramenta DB DesignerFork. O dicionário de dados também é apresentado no Apêndice B.

Figura 12 - Modelo entidade e relacionamento do sistema *web*



Na sequência é apresentada uma breve descrição das entidades criadas (baseadas no banco de dados do GRP) para o desenvolvimento do sistema *web*:

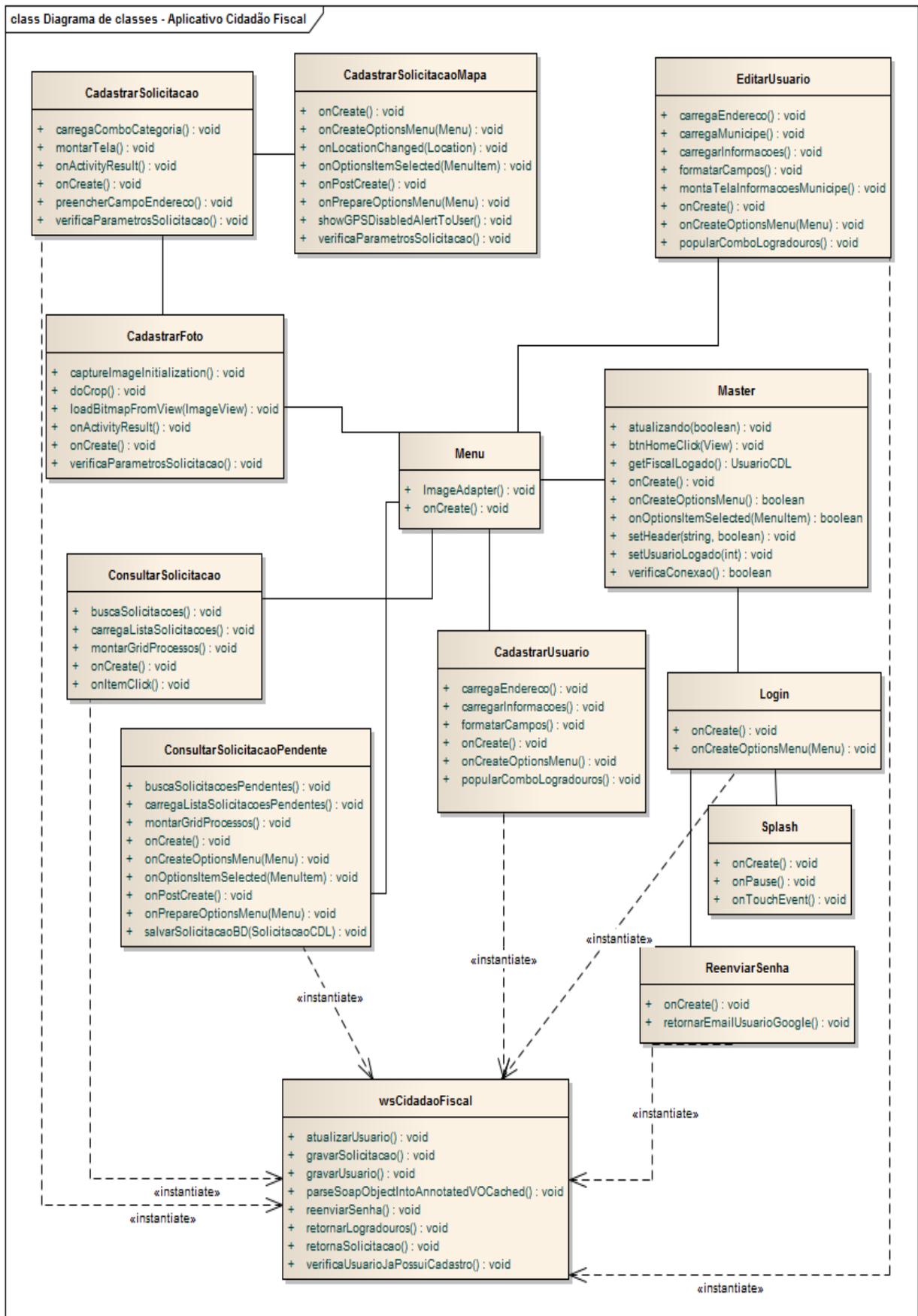
- a) **SituacaoProcesso:** entidade que armazena os dados referente a situação do processo;
- b) **TipoProcotolo:** entidade que armazena os dados dos tipos de protocolos;

- c) `TipoProcesso`: entidade que armazena os dados dos tipos de processos;
- d) `Assunto`: entidade que armazena os dados dos assuntos referentes ao processo;
- e) `Setor`: entidade que armazena os dados dos setores;
- f) `SetorAssunto`: entidade que armazena os dados agrupados do setor e do assunto;
- g) `UsuarioERP`: entidade que armazena os dados dos usuários do ERP;
- h) `Pessoa`: entidade que armazena os dados das pessoas cadastradas;
- i) `Arquivo`: entidade que armazena os dados dos arquivos enviados;
- j) `EnderecoFiscalizacaoCidadao`: entidade que armazena os dados enviados pelo aplicativo referente a localização da solicitação;
- k) `UsuarioCidadaoFiscal`: entidade que armazena os dados contendo as informações de *login* do aplicativo Cidadão Fiscal;
- l) `Bairro`: entidade que armazena os dados dos bairros da cidade de Blumenau;
- m) `Endereco`: entidade que armazena os dados das ruas da cidade de Blumenau;
- n) `Processo`: entidade que armazena os dados dos processos;
- o) `DistribuicaoProcesso`: entidade que armazena os dados da distribuição dos processos;
- p) `AnaliseProcesso`: entidade que armazena os dados das análises de processos.

### 3.2.6 Diagrama de classes

O diagrama de classes auxilia na visualização de como as classes estão estruturadas e quais são as relações entre as mesmas. Na Figura 13 é apresentado através do diagrama de classes a comunicação das telas do aplicativo com a classe que realiza o envio das solicitações para o sistema *web* através do *web service*, que é implementado através da classe `wsCidadaoFiscal`.

Figura 13 - Diagrama de classes do serviço wsCidadaoFiscal



Na Figura 14 é apresentado como é realizado o recebimento das solicitações oriundas do aplicativo no sistema *web*. As solicitações chegam ao sistema *web* através do *web service*, que é implementado pela classe `CidadaoFiscalServico`; essa por sua vez instância as classes de acesso a dados que são: `AnaliseProcessoDao`, `ProcessoDao` e `DistribuicaoProcessoDao`, que são responsáveis pela realização de consultas no banco de dados, inserção e atualização. Essas classes especializam a classe abstrata `AbstractNHibernateDao`, que disponibiliza operações comuns de manipulação das informações junto ao banco de dados.

Figura 14 - Diagrama de classes *Web Service* de recebimento de solicitações



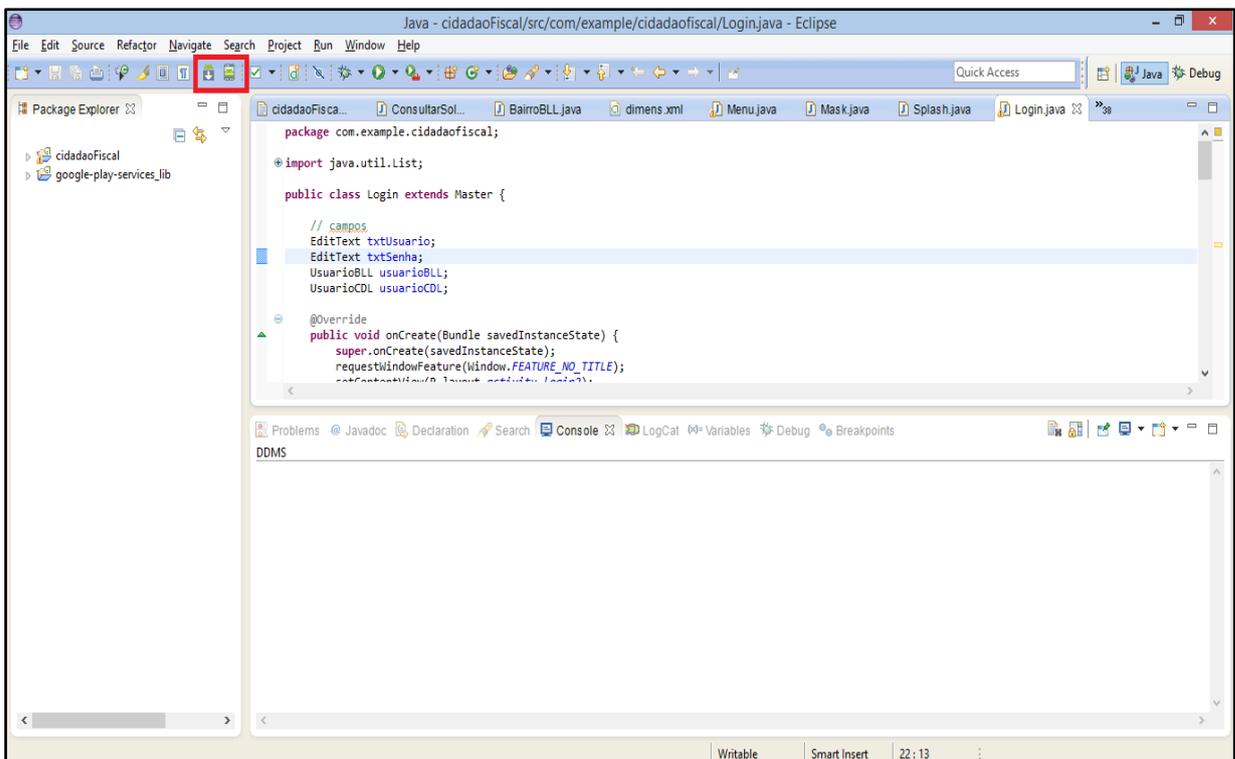
### 3.3 IMPLEMENTAÇÃO

A seguir são mostradas as técnicas e ferramentas utilizadas e a operacionalidade da implementação do aplicativo e do sistema *web*.

#### 3.3.1 Técnicas e ferramentas utilizadas

Para o desenvolvimento do aplicativo foi utilizada a *Integrated Development Environment* (IDE) Eclipse (versão Kepler) com o *plugin Android Developer Tools* (ADT). Esta ferramenta proporciona as funcionalidades padrões da IDE Eclipse e contam também com uma ampliação das ferramentas e recursos para o desenvolvimento de aplicativos para a plataforma Android. Logo a linguagem utilizada para o desenvolvendo do aplicativo é o Java. Na Figura 15 é exibida a tela inicial, o destaque mostra a integração da IDE do Eclipse com o *plugin* ADT.

Figura 15 - Eclipse com o *plugin Android Developer Tools*

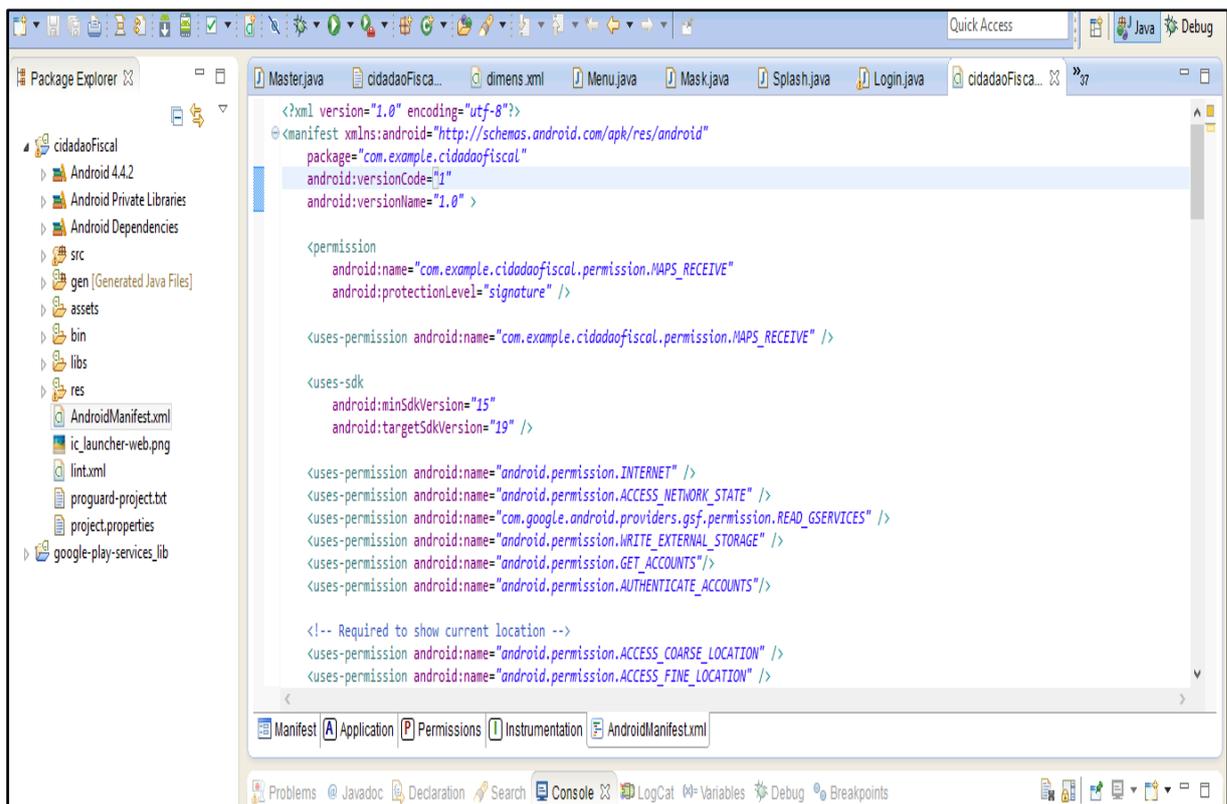


Para o desenvolvimento do aplicativo Cidadão Fiscal foi necessário realizar o *download* das *Applications Programming Interface* (APIs) de números 15,16,17,18 e 19 através da ferramenta *Android SDK Manager*, que pode ser encontrada no próprio Eclipse, para ser compatível com a versão 4.0.3 do Android ou superior.

A execução e depuração da aplicação Android foi realizada utilizando o *smartphone* LG L5 cujo sistema operacional está na versão 4.0.3, o *smartphone* LG G2 cujo sistema operacional está na versão 4.4, e também o *tablet* Xoom 2 da Motorola cujo sistema operacional está na versão 4.0.4, além do emulador virtual do Android existente no Eclipse.

A Figura 16 mostra a estrutura de arquivos criada no desenvolvimento do aplicativo. A correta criação da estrutura do projeto faz com que o seu funcionamento ocorra da forma esperada pelo sistema operacional Android.

Figura 16 - Estrutura de Arquivos do Aplicativo

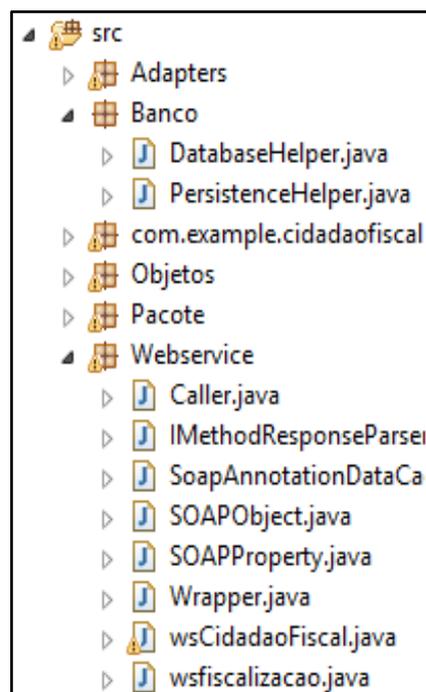


Os diretórios e arquivos se dividem da seguinte maneira:

- a) *src*: este diretório contém o código fonte do aplicativo. Está dividido em subdiretórios como pode-se perceber na Figura 17. Os *Adapters* são subdiretórios que possuem tarefas específicas. Este diretório é responsável por conter as classes

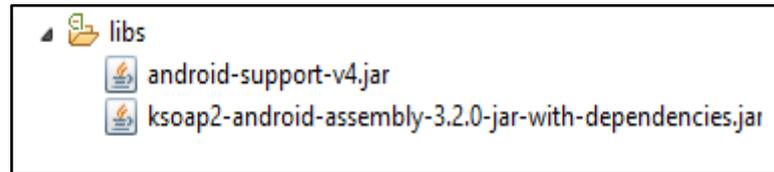
que possuem a função ler dados de uma fonte, como por exemplo, de arquivos *eXtensible Markup Language* (XML) ou de banco de dados e fornecer estes dados para uma *View* ou *List View* (se trata se um componente que pode ser adicionado a tela), sendo que o mesmo é responsável por criar cada item de lista. O `Banco` é um subdiretório que é responsável por criar o banco de dados assim como todas as operacionalidades necessárias como inserir dados, alterar e remover. O diretório `com.example.cidadaofiscal` possui as classes que são extensões das telas desenvolvidas. O `Objetos` é um diretório que contém todas as classes necessárias para realizar a comunicação da tela com o banco de dados. O `Pacote` apresenta uma classe contendo métodos para utilidade geral como, por exemplo, formatação de dados e Cadastro de Pessoa Física (CPF). O `Web Service` é o diretório que contém várias classes para realizar a comunicação com o *web services*;

Figura 17 - Diretório `src` com subdiretórios



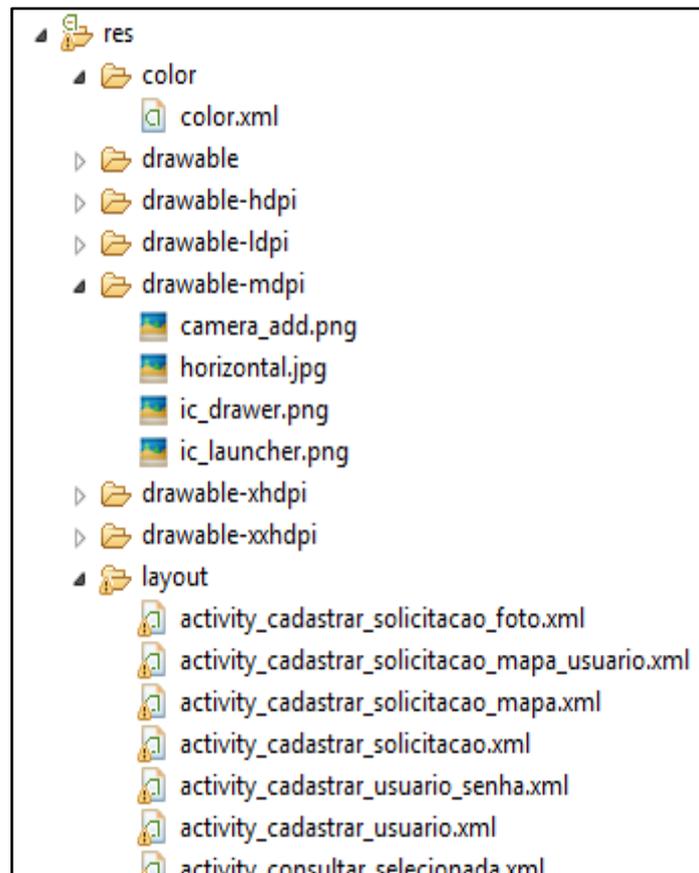
b) `libs`: diretório em que as bibliotecas externas são adicionadas. Como por exemplo, a biblioteca `KSOAP2`, que é responsável por permitir o consumo de *Web Services*, como mostra a Figura 18;

Figura 18 - Biblioteca Ksoap2



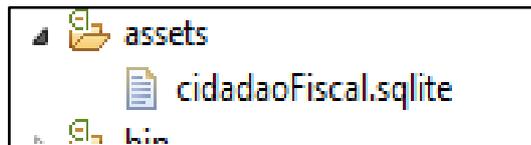
- c) `res`: neste diretório se encontram todos os arquivos de imagens ( que são divididos em subdiretórios conforme a densidade), assim como os *layouts* das telas (que são em formato XML), e também valores chaves que são aqueles utilizados ao longo da aplicação como mensagens exibidas na aplicação. A Figura 19 mostra o diretório com alguns dos arquivos citados;

Figura 19 - Diretório res



- d) `assets`: contém o diretório que é responsável por criar as tabelas de banco de dados necessárias na primeira instalação do aplicativo no dispositivo. A Figura 20 mostra o arquivo com extensão `sqlite` (banco utilizado no desenvolvimento deste aplicativo);

Figura 20 - Diretório assets



- e) `AndroidManifest.xml`: este arquivo pode ser considerado o mais importante da aplicação pois nele existe a declaração de todas as classes que herdam de *Activity*. Possui também todas as informações de permissões como por exemplo, de internet, mapas, GPS, entre outras. A Figura 21 mostra alguns destes campos;

Figura 21 - Arquivo AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.cidadaoFiscal"
    android:versionCode="1"
    android:versionName="1.0" >

    <permission
        android:name="com.example.cidadaoFiscal.permission.MAPS_RECEIVE"
        android:protectionLevel="signature" />

    <uses-permission android:name="com.example.cidadaoFiscal.permission.MAPS_RECEIVE" />

    <uses-sdk
        android:minSdkVersion="15"
        android:targetSdkVersion="19" />

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="com.google.android.providers.gsf.permission.READ_GSERVICES" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.GET_ACCOUNTS"/>
    <uses-permission android:name="android.permission.AUTHENTICATE_ACCOUNTS"/>

    <!-- Required to show current location -->
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />

```

As principais interações realizadas com o usuário pelo aplicativo são através das *Activities*, que são as classes Java que são estendidas da classe *Activity*, responsável por ligar o *layout* aos métodos necessários para o correto funcionamento de todas as ações ligadas à interface. Segundo Android Developers (2013) uma *Activity* possui quatro principais estados: ativa ou em execução, que ocorre quando a *activity* está em primeiro plano na interface; a outra é o estado pausado, que ocorre quando a mesma está aberta, mas perdeu o foco; outro estado é o parado, quando a *Activity* é sobreposta por outra e finalmente se uma *Activity* for pausada ou parada o sistema operacional poderá eliminá-la, mas quando retornada deve ser restaurada com o seu estado anterior.

Na *Activity* `CadastrarUsuarioSenha` do aplicativo, que responsável por gravar os

dados do usuário, é implementada a classe `GravarUsuarioWS()` que utiliza a extensão do `AsyncTask` que é responsável por realizar todo o processamento pesado, que neste caso é a comunicação com o *Web Service*, onde é executado em uma *thread* distinta como é apresentado na Figura 22.

Figura 22 - Comunicação com o Web Service - GravarUsuario

```

class GravarUsuarioWS extends AsyncTask<MunicipeCDL, UsuarioCDL, String> {

    protected ProgressDialog progressDialog;

    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressDialog = ProgressDialog.show(CadastrarUsuarioSenha.this,
            "Aguarde...", "Cadastrando usuário", true, false);
    }

    @Override
    protected String doInBackground(MunicipeCDL... params) {
        String retorno = wsCidadaoFiscal.GravarUsuario(municipeCDL,
            usuarioCDL);
        retornoGravarUsuario = retorno;
        return retornoGravarUsuario;
    }

    @Override
    protected void onPostExecute(String result) {
        super.onPostExecute(result);
        progressDialog.dismiss();
    }

}

```

Para a comunicação com o *Web Services* é fundamental que os dados que estão no dispositivo móvel sejam encaminhados corretamente da *Activity* para os métodos que irão executar a chamada do serviço. Logo após, o objeto que contém as informações da solicitação, estiver preenchido, o *request* é realizado utilizando um *SoapObject*, onde cada parâmetro da solicitação é uma propriedade do *SoapObject*. A chamada do servidor de aplicações é realizada utilizando um *HttpTransportSE* onde é passado um envelope SOAP com a *request* gerada. Após a execução do serviço o envelope irá conter um retorno do que foi executado, sendo que dependendo do retorno o tipo primitivo pode mudar como é o caso do retorno de um número neste caso o objeto é serializado como *SoapPrimitive*, mas caso o objeto fosse do tipo complexo o retorno seria um *SoapObject*.

A Figura 23 mostra como a estrutura foi montada para realizar o consumo do *Web Service*, através do método que grava a solicitação.

Figura 23 - Consumo do *Web Service*

```

SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(
    SoapEnvelope.V11);
new MarshalBase64().register(envelope);
envelope.dotNet = true;
envelope.implicitTypes = true;
envelope.setOutputSoapObject(request);

try {

    HttpTransportSE androidHttpTransport = new HttpTransportSE(URL,
        30000);
    androidHttpTransport.debug = true;
    String SOAP_ACTION = "http://tempuri.org/GravarSolicitacao";

    androidHttpTransport.call(SOAP_ACTION, envelope);

    SoapPrimitive response = (SoapPrimitive) envelope.getResponse();

    if (response.getValue() != null) {
        return response.getValue().toString();
    } else {
        return "Erro";
    }

} catch (Exception e) {
    e.printStackTrace();
}
return "Erro, tente novamente";

```

Ao abrir a tela da localização geográfica que está presente na classe *CadastrarSolicitacaoMapa*, o código fonte apresentado na Figura 24 é responsável por realizar a busca da localização geográfica do munícipe, através da utilização do GPS do dispositivo.

Figura 24 - Código fonte responsável por carregar localização

```

map = ((MapFragment) getFragmentManager().findFragmentById(
    R.id.map_solicitacao)).getMap();
// Adiciona um ponto no mapa no local pré determinado
@SuppressWarnings("unused")
Marker frameworkSystem = map.addMarker(new MarkerOptions().position(
    frameworkSystemLocation).title("Meu Local"));
// Move a câmera para Framework System com zoom 15.
map.moveCamera(CameraUpdateFactory.newLatLngZoom(
    frameworkSystemLocation, 15));
map.animateCamera(CameraUpdateFactory.zoomTo(17), 2000, null);
// Marca sua posição no mapa

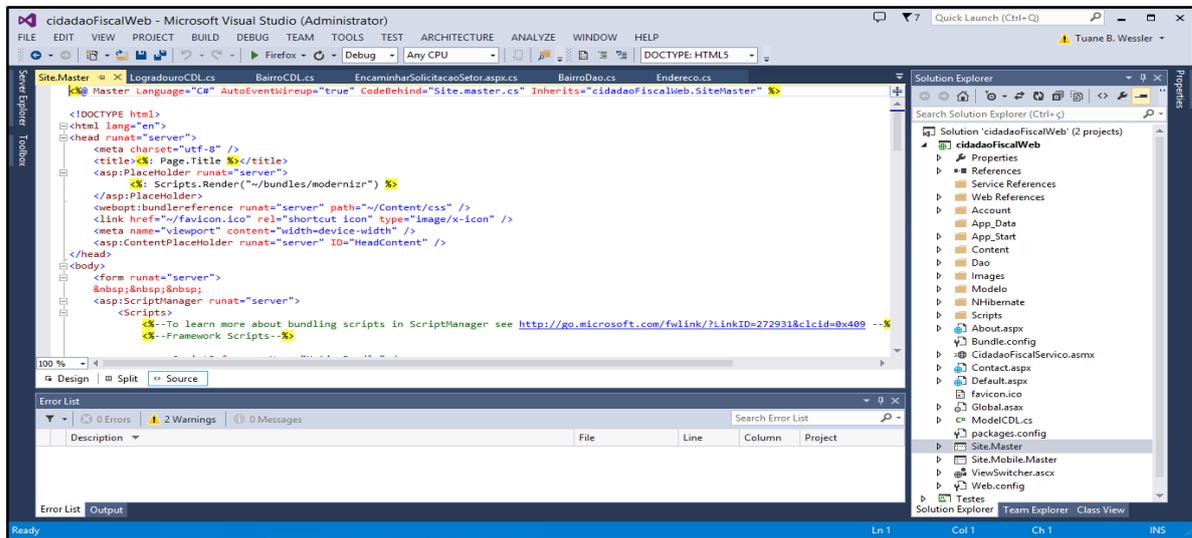
map.getUiSettings().setMyLocationButtonEnabled(true);
map.getUiSettings().setCompassEnabled(true);
map.getUiSettings().setZoomControlsEnabled(true);
map.setMyLocationEnabled(true);

```

Todas as informações enviadas e recebidas no aplicativo através da comunicação com o *Web Services* são realizadas no sistema *web*. O sistema *web*, o *Cidadão Fiscal Web*, foi desenvolvido na linguagem C# / ASP.NET sobre a plataforma Microsoft .NET (versão 4.5).

Para realizar a codificação foi utilizado o ambiente Microsoft Visual Studio 2013, a Figura 25 mostra o ambiente de desenvolvimento.

Figura 25 - Plataforma de desenvolvimento do Visual Studio 2013



A fim de agilizar o desenvolvimento e obter um melhor resultado foi utilizado a biblioteca do NHibernate (versão 4.0) para persistir os objetos para a base de dados. Esta biblioteca facilita o desenvolvimento, pois ao invés de escrever códigos SQL dentro e fora da base de dados, a biblioteca gera o código SQL necessário, garantindo que todos os tipos e valores serão criados corretamente.

Outra tecnologia utilizada no desenvolvimento foi o *Language Integrated Query* (LINQ). Segundo Microsoft Corporation (2014) o LINQ é uma inovação apresentada no Visual Studio 2008 com o *framework* versão 3.5, que engloba consulta, atribuição e operações de transformação de dados em praticamente todas as fontes de dados. Com uma sintaxe semelhante ao SQL é possível realizar pesquisas em blocos de informações, filtrar os dados, quando necessário através da associação de operadores.

No Cidadão Fiscal Web essa tecnologia foi utilizada em todo o sistema para a manipulação das informações no banco de dados e coleções em memória. A Figura 26 mostra o método `ObterPessoaID` que realiza uma consulta no banco para buscar a pessoa que contém a chave primária informada.

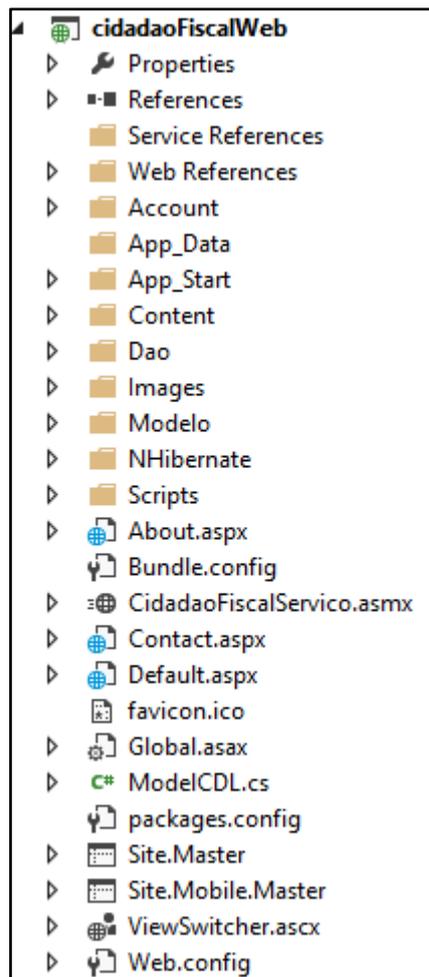
Figura 26 - Método responsável por retornar pessoa

```
1 reference
internal Modelo.Pessoa ObterPessoaID(int idPessoaID)
{
    var consulta = from pessoa in Query()
                   where pessoa.IdPessoa == idPessoaID
                   select pessoa;

    return consulta.FirstOrDefault();
}
```

A Figura 27 mostra a estrutura de arquivos criada no desenvolvimento do sistema *web*, os principais diretórios e arquivos serão explicados mais adiante.

Figura 27 - Estrutura de diretórios e arquivos



O arquivo `Web.config` é um dos arquivos mais importantes do projeto, pois nele estão todas as configurações do banco de dados. Outro arquivo importante é o `Site.Master`, esta página é responsável pelo padrão de *layout* em todas as telas do sistemas. Entre os principais diretórios está o `Modelo` que é responsável por encapsular os dados juntamente com a sua lógica de processamento. O diretório `Dao` é responsável por abstrair e encapsular os mecanismos de acesso a dados ocultando detalhes da execução da origem dos dados. Outro diretório fundamental para o projeto é o `Account` que contém todas as telas do sistema.

A Figura 28 apresenta o código fonte responsável por realizar a verificação de qual setor pertence a categoria selecionada pelo munícipe no aplicativo, pois todas as categorias cadastradas no aplicativo Cidadão Fiscal estão relacionadas com algum setor existente na tabela `Setor` que pertence ao banco de dados do GRP.

Figura 28 - Código fonte relação do setor com categoria

```
// seleciona setor destino conforme a categoria selecionada pelo usuário no aplicativo
// Ex: categoria trânsito = Setor Serterb.
if(processoRetornado.SetorDestino != null){
    Setor setor = setorDao.ObterSetor(Convert.ToInt32(processoRetornado.SetorDestino));
    cbSetores.Items.FindByValue(setor.IdSetor.ToString()).Selected = true;
}
```

A Figura 29 mostra o código fonte responsável por realizar o registro do parecer final da solicitação e a conclusão da solicitação, sendo que se esta solicitação for consultada pelo munícipe no aplicativo após a conclusão será informado ao mesmo que a solicitação foi finalizada e também o último parecer contendo as considerações finais do analista responsável pela execução da mesma.

Figura 29 - Código fonte conclusão solicitação

```

try
{
    using (ITransaction transacao = NHibernateSessionManager.Instance.GetSession().BeginTransaction())
    {
        processo.Parecer = txtParecer.Text;
        processo.DataConclusao = DateTime.Now;

        SituacaoProcessoDao situacaoProcessoDao = new SituacaoProcessoDao();
        SituacaoProcesso situacao = situacaoProcessoDao.ObterSituacaoProcesso(SituacaoProcesso.Finalizado);
        processo.SituacaoProcesso = situacao;

        processoDao.SaveOrUpdate(processo);

        // commit
        transacao.Commit();

        btnCriarNovaAnalise.Enabled = false;
        chkConcluirAnalise.Enabled = false;
        txtParecer.Enabled = false;
        ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mesagem1", "alert('Processo concluído com su
    }
}

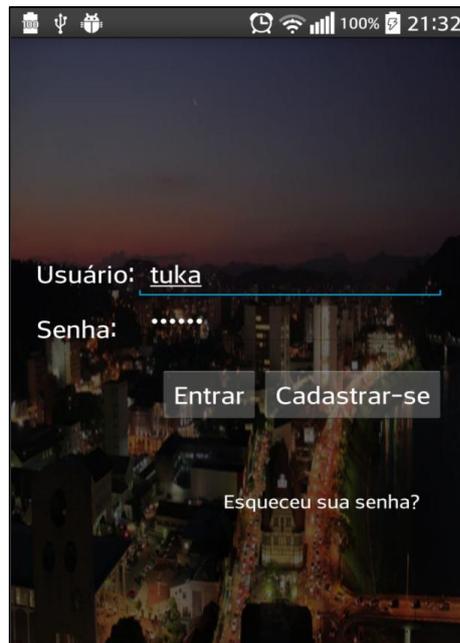
```

### 3.3.2 Operacionalidade da implementação

Nesta subseção será apresentada a operacionalidade do sistema sob a perspectiva do munícipe (Cidadão Fiscal), e dos funcionários da Prefeitura Municipal de Blumenau (PMB) (Cidadão Fiscal *Web*).

Sob as perspectiva do munícipe, o mesmo efetuará o acesso ao aplicativo Cidadão Fiscal através do respectivo ícone no dispositivo móvel. A tela inicial do aplicativo permite ao usuário efetuar o *Login*. Essa funcionalidade atende respectivamente ao requisito funcional RF02. A Figura 30 apresenta a tela inicial.

Figura 30 - Tela inicial do aplicativo



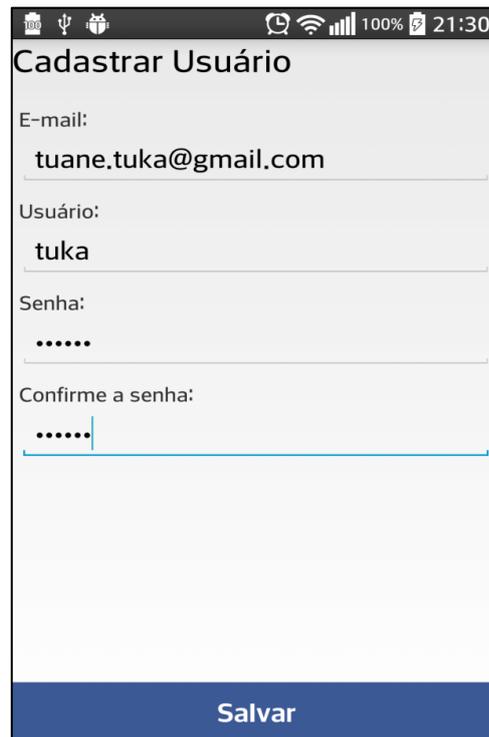
Caso o munícipe não possua uma conta para acessar o aplicativo, o mesmo poderá se cadastrar utilizando o botão “Cadastrar-se”, que está disponível na mesma tela de *Login* e deverá estar conectado a *Internet*, conforme pode ser visto anteriormente. Ao acessar a tela de cadastro primeiramente o munícipe deverá informar o nome, a data de nascimento, o CPF, o bairro, o logradouro, o número e Código de Endereçamento Postal (CEP). Após todos os campos serem preenchidos o munícipe deverá selecionar o botão “Continuar”. A Figura 31 apresenta a tela de cadastro do usuário no aplicativo.

Figura 31 - Tela de Cadastro do usuário

A captura de tela mostra a tela de cadastro do usuário. No topo, há uma barra de status com ícones de vibração, Wi-Fi, 100% de bateria e o horário 21:24. O formulário contém os seguintes campos preenchidos: 'Nome Completo:' com 'Tuane B Wessler', 'Data de Nascimento' com '17/09/1993', 'CPF:' com '999.999.999-99', 'Bairro:' com 'AGUA VERDE', 'Logradouro:' com um ícone de seta para cima, 'Número:' com '372' e 'CEP:' com '89023660'. Na base do formulário, há um botão azul com o texto 'Continuar'.

Após o munícipe selecionar o botão “Continuar” o aplicativo irá apresentar a próxima tela de cadastro do usuário, onde o mesmo deverá informar o *e-mail* (caso o munícipe esteja utilizando o dispositivo Android com uma conta Google, ao carregar a tela o campo de *e-mail* vai estar preenchido com o *e-mail* da conta conectada. Caso o munícipe deseje trocar poderá realizar esta operação), usuário, e senha (a senha deve conter mais de 5 caracteres), como mostra a Figura 32.

Figura 32 - Tela de cadastro do usuário



A imagem mostra a interface de usuário para o cadastro de um novo usuário. O título da tela é "Cadastrar Usuário". O formulário contém os seguintes campos:

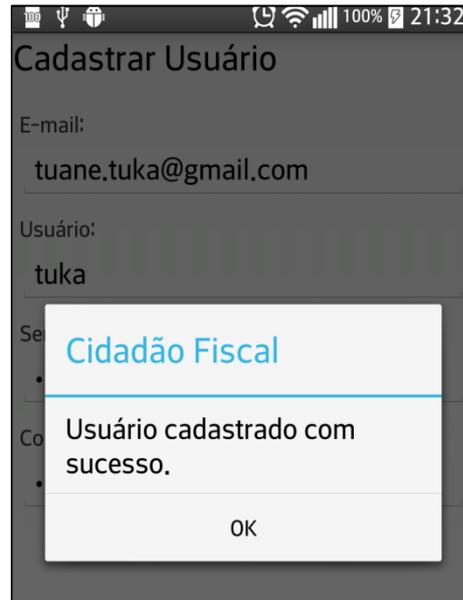
- E-mail:** tuane.tuka@gmail.com
- Usuário:** tuka
- Senha:** ..... (ocultada por pontos)
- Confirme a senha:** ..... (ocultada por pontos)

Um botão azul com o texto "Salvar" está localizado na parte inferior da tela.

Com todos os campos preenchidos o usuário deverá selecionar o botão “Salvar”. Depois de selecionado o botão o aplicativo irá realizar as validações como, por exemplo, se o usuário que se pretende cadastrar já não possui um usuário cadastrado para a utilização do aplicativo, quantidade de caracteres da senha entre outras validações.

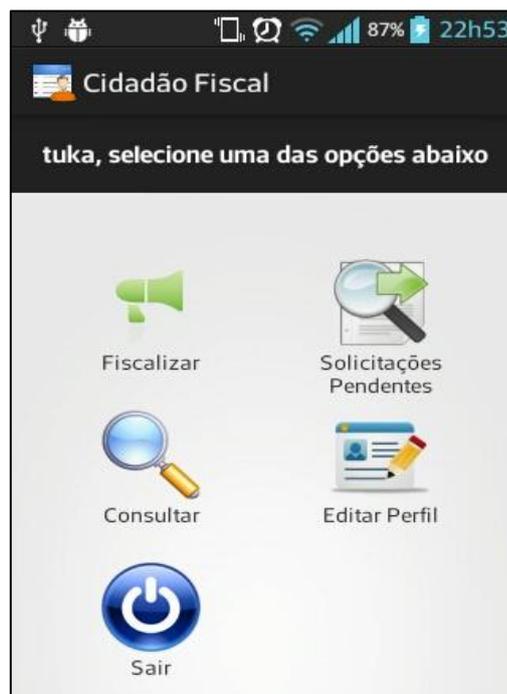
Se todos os campos estiverem preenchidos corretamente e o usuário for cadastrado corretamente no banco de dados através do sistema *web* a mensagem a seguir irá ser apresentada como mostra Figura 33, caso contrário será mostrado uma mensagem de erro.

Figura 33 - Mensagem de confirmação de cadastro



Após realizar o cadastro no aplicativo o munícipe poderá acessar o aplicativo informando o usuário e senha cadastrados. Como mostra a Figura 34 o aplicativo apresentará a tela contendo o *menu* com as seguintes opções: Fiscalizar, Solicitações Pendentes, Consultar, Editar Perfil e Sair.

Figura 34 - Menu do aplicativo



Para realizar uma fiscalização o usuário deverá selecionar a opção “Fiscalizar”, após selecionar esta opção o aplicativo irá apresentar a primeira tela referente a solicitação como mostra a Figura 35, que é a de cadastro de uma foto ou imagem. Não é obrigatório ao usuário adicionar uma foto ou imagem para enviar a solicitação (caso a mesma não seja adicionada o sistema apresenta uma imagem padrão, que não contém foto), mas quanto mais informações e detalhes da reclamação ou sugestão mais fácil para ser analisada pelos funcionários da Prefeitura. Esta funcionalidade atende ao requisito funcional RF03.

Figura 35 - Tela de cadastro da foto



Para dar continuidade a fiscalização o usuário deverá selecionar o botão “Continuar”, onde a próxima tela a ser apresentada é a de cadastro da solicitação, onde devem ser informados: a categoria, a descrição da solicitação e o endereço da solicitação. Caso o usuário não consiga definir em qual categoria a solicitação se encaixa o mesmo deverá selecionar a categoria “Outros”. O endereço da solicitação poderá ser escrito diretamente no campo que é apresentado na Figura 36, ou caso queira adicionar a localização diretamente do Google Maps o usuário deverá selecionar a opção “Selecione aqui para adicionar a localização”, onde será apresentado o mapa e o mesmo buscará a localização atual do usuário. Estas funcionalidades atendem ao Requisito Funcional RF03.

Figura 36 - Tela de cadastro da solicitação



Cidadão Fiscal

### Cadastre sua Solicitação

Categoria:  
Poluição

Descrição:  
Muita fumaça da empresa X.

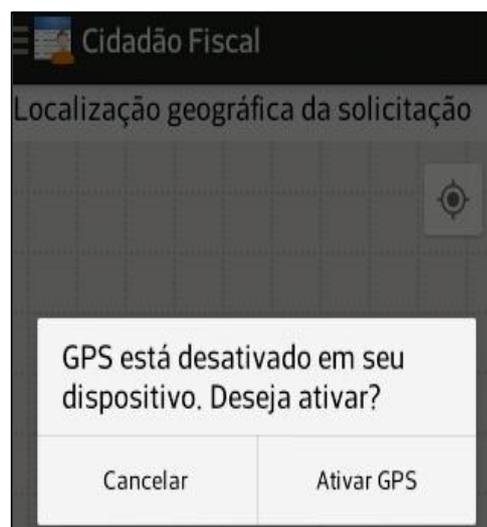
Endereço:  
R. Manoel Salvador, 408 -  
Valparaíso, 89023-660 - Santa  
Catarina/Blumenau - SC

Selecione aqui para adicionar a localização

**Enviar Solicitação**

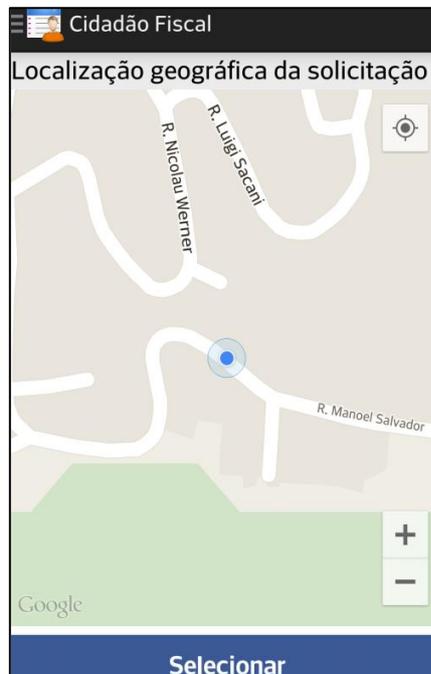
Como mencionado anteriormente, se o usuário estiver no local em que está realizando a reclamação ou sugestão o mesmo poderá selecionar a opção para adicionar a localização atual, o que facilita o trabalho tanto do atendente da ouvidoria que realiza a triagem das solicitações como para os analistas que são responsáveis por resolver a solicitação para conseguirem localizar a solicitação exata através da longitude e a latitude. Ao acessar a tela o aplicativo verifica se o GPS está ativo, caso não esteja o mesmo pergunta se deseja ativar, como mostra a Figura 37.

Figura 37 - Mensagem de ativação do GPS



A Figura 38 mostra o mapa após o GPS identificar a localização do usuário. Para dar continuidade a solicitação, basta o usuário selecionar o botão “Selecionar”.

Figura 38 - Tela de localização da solicitação

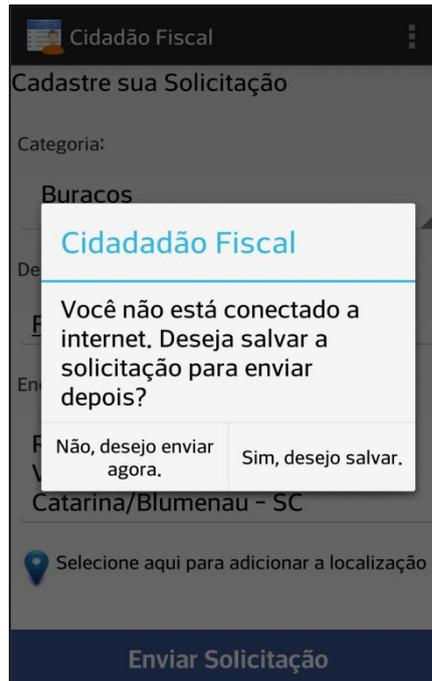


Após selecionar o botão “Selecionar”, o aplicativo irá retornar a tela anterior com o campo endereço preenchido com a localização do usuário como mostra a Figura 39.

Figura 39 - Tela de cadastro contendo o endereço

Para realizar o envio da solicitação o usuário deverá selecionar o botão “Enviar Solicitação”. A Figura 40 mostra a mensagem de aviso, que é apresentada se o usuário não estiver conectado a internet e tentar enviar a solicitação.

Figura 40 - Mensagem de aviso antes do envio da solicitação



Caso o usuário selecione a opção “Não, desejo enviar agora”, o mesmo deverá conectar seu dispositivo móvel a internet e assim conseguirá realizar o envio da solicitação, mas caso o usuário selecione a opção “Sim, desejo salvar”, a solicitação será salva e ficará como pendente para o envio mais tarde, quando o usuário possuir uma conexão com a *Internet*. Esta funcionalidade atende ao Requisito Funcional RF04. Em ambos os casos ao final do processo o aplicativo vai apresentar uma mensagem informando ou que a solicitação foi enviada com sucesso e agora o usuário deverá acompanhar a mesma na tela de “Consultar”, ou apresentará a mensagem que a solicitação foi salva e que para realizar o envio da mesma o usuário deverá acessar a tela de “Solicitações Pendentes”.

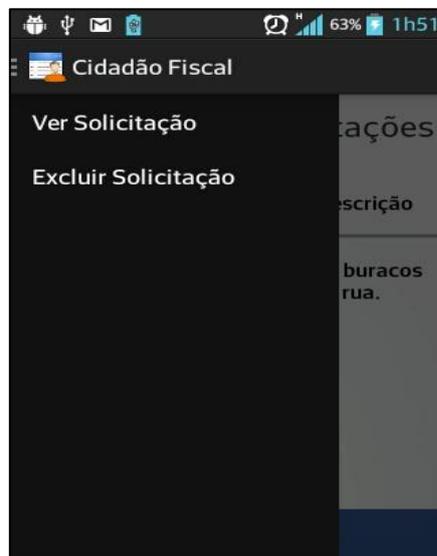
A opção do *menu* principal do aplicativo “Solicitações Pendentes” apresenta uma lista contendo todas as solicitações que estão aguardando o envio para a Ouvidoria da Prefeitura (Cidadão Fiscal *Web*), como é apresentado na Figura 41.

Figura 41 - Tela de Solicitações Pendentes



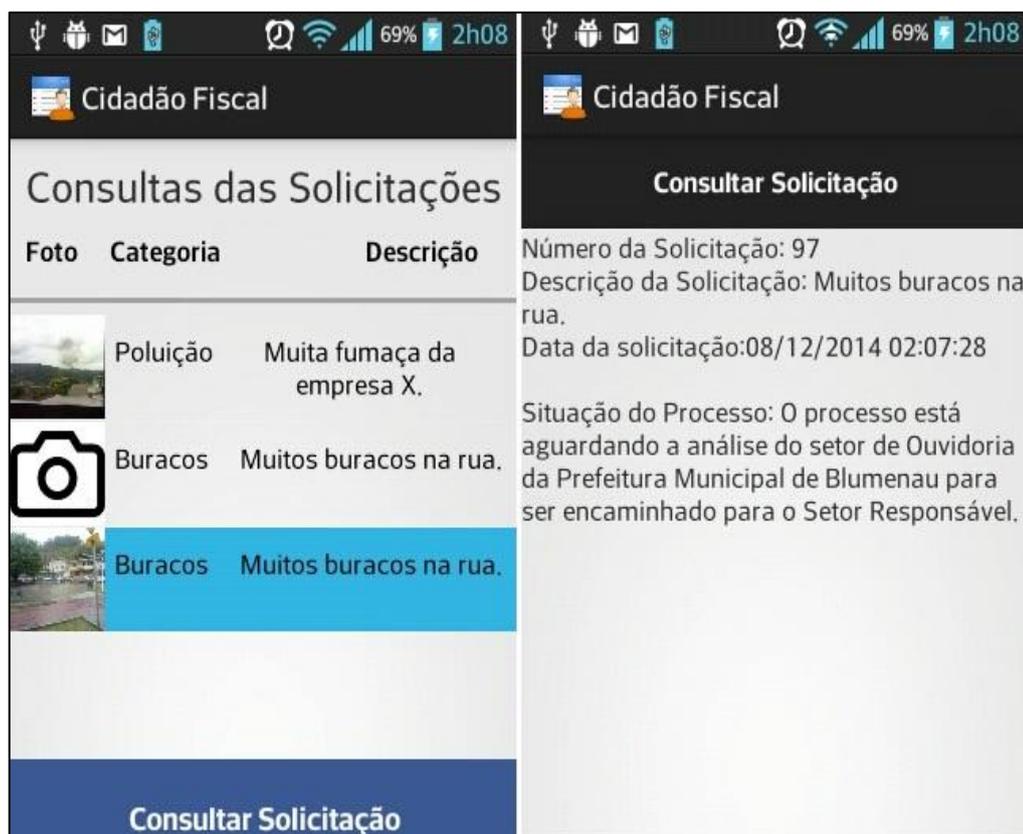
Antes de realizar o envio da solicitação o usuário pode visualizar as solicitações, mas não poderá modificar a mesma, sendo que esta funcionalidade corresponde ao Requisito Funcional RF05. Outra opção disponível é a possibilidade de excluir as solicitações que estão pendentes (esta funcionalidade corresponde ao Requisito Funcional RF06), pois em alguns casos pode acontecer que a sugestão ou problema seja resolvido no tempo em que o usuário permaneceu com a solicitação pendente, como é o caso da reclamação de lixos, sendo que os mesmos podem ser removidos. A Figura 42 mostra o menu interno da tela com as opções para visualizar e excluir as solicitações.

Figura 42 - Menu com opções das solicitações pendentes



Outra opção presente no menu principal da aplicação é a de “Consultar”. Nesta tela serão apresentadas todas as solicitações enviadas pelo usuário a Ouvidoria. Para saber todas as atualizações das mesmas basta o usuário selecionar a desejada, logo a aplicação apresenta uma tela contendo as informações que foram disponibilizadas na solicitação. Está tela busca atualizações cada vez que a mesma é acessada, sendo que para isto o dispositivo deverá conter uma conexão com a internet, caso contrário o mesmo irá retornar a última atualização baixada. A Figura 43 apresenta à direita, a tela de consulta de solicitações e a esquerda a tela que é carregada após o usuário selecionar uma solicitação e selecionar o botão “Consultar Solicitação” que possui todas as atualizações vindas da Prefeitura. Esta funcionalidade corresponde ao Requisito Funcional RF07.

Figura 43 - Tela de Consulta das Solicitações



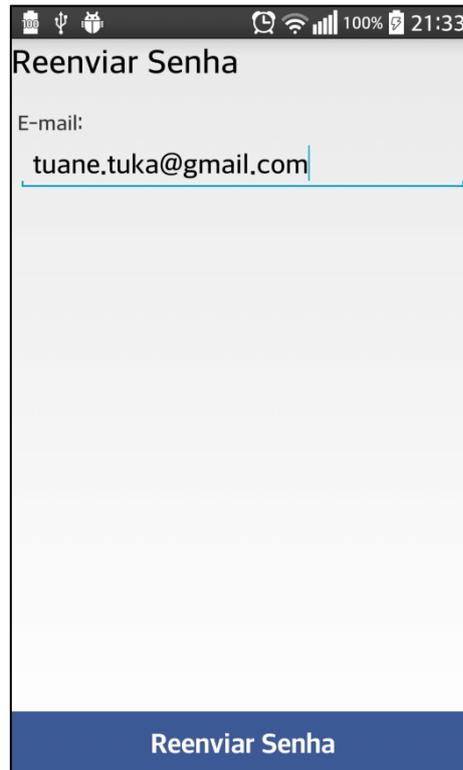
O aplicativo conta ainda com uma opção no menu principal que é “Editar Perfil”, onde após selecionar a opção o aplicativo apresentará a mesma tela de cadastro do usuário mas contendo as informações preenchidas do usuário. O usuário apenas não poderá modificar o nome de usuário. Na Figura 44 pode-se analisar as informações citadas acima.

Figura 44 - Tela de editar perfil

The image shows two side-by-side screenshots of a mobile application interface. The left screenshot, titled 'Dados Gerais', displays a form with the following fields: 'Nome Completo:' with the value 'Tuane B Wessler'; 'Data de Nascimento' with '17/09/1993'; 'CPF:' with '999.999.999-99'; 'Bairro:' with 'GARCIA'; 'Logradouro:' with '7 DE SETEMBRO'; and 'Número:'. The right screenshot, titled 'Cadastrar Usuário', displays a form with the following fields: 'E-mail:' with 'tuane.tuka@gmail.com'; 'Usuário:' with 'tuka'; 'Senha:' with masked characters '.....'; and 'Confirme a senha:' with masked characters '.....'. A blue 'Salvar' button is located at the bottom right of the second screenshot. Both screenshots show a status bar at the top with icons for signal, Wi-Fi, battery (100%), and time (0h09).

A última opção presente no menu principal da aplicação é o item “Sair”. Após selecionar o mesmo o aplicativo realiza o *logout* do usuário e retorna para a tela de *Login*. Na tela de *Login* do aplicativo existe a possibilidade do usuário recuperar a senha de acesso caso tenha esquecido. Para isto, conforme mostra a Figura 45 após o usuário selecionar “Esqueceu sua senha?”, será apresentada uma tela contendo o campo de *e-mail* que deverá ser preenchido com o *e-mail* cadastrado no aplicativo, para realizar o envio da senha para o mesmo. Caso o usuário esteja utilizando uma conta Google o *e-mail* será preenchido ao carregar a tela utilizando o da conta. Caso seja necessário modificar o *e-mail*, basta apagá-lo. Para realizar o reenvio da senha o usuário deverá selecionar o botão “Reenviar Senha”.

Figura 45 - Tela de Reenvio de Senha



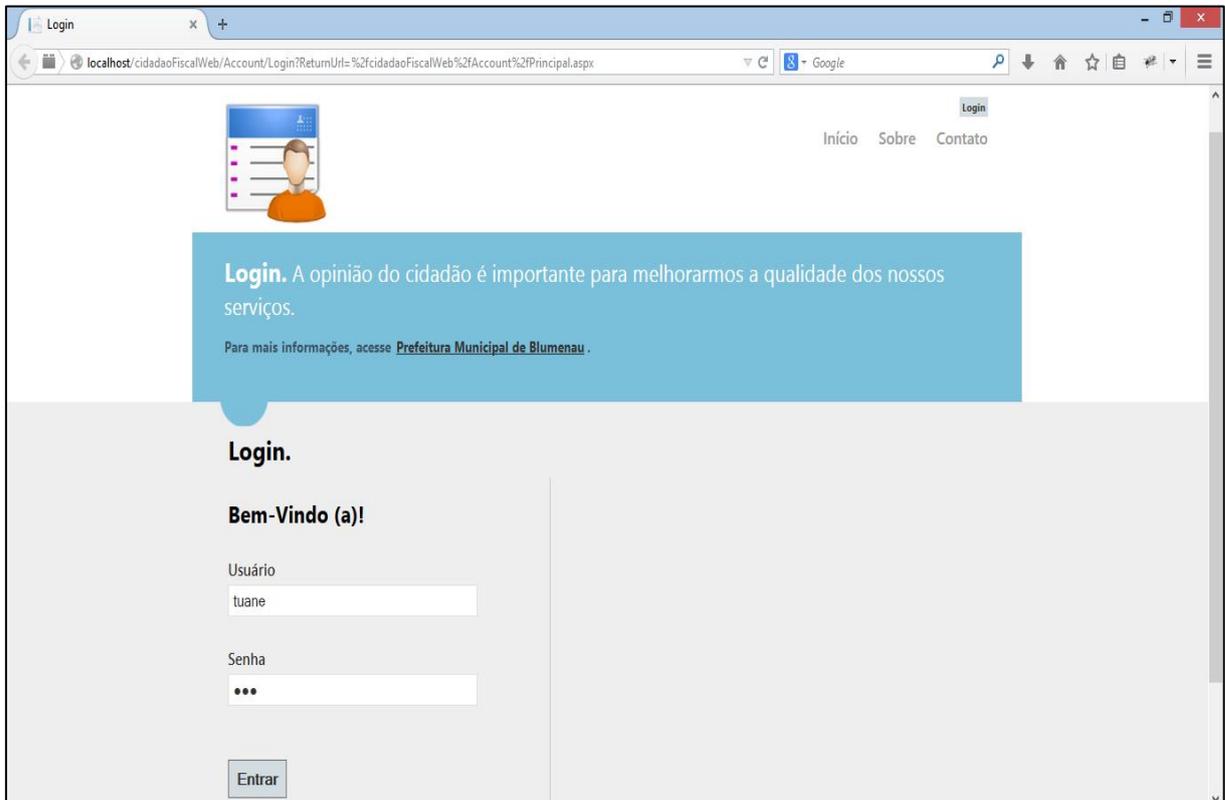
Ao selecionar o botão “Reenviar Senha”, um *e-mail* será enviado informando o usuário e senha de acesso ao aplicativo como mostra a Figura 46.

Figura 46 - *E-mail* de reenvio de senha

Sob a perspectiva dos funcionários da PMB, que neste caso são os Atendentes da

Ouvidoria e os Analistas de cada setor da Prefeitura, há o acesso ao Cidadão Fiscal *Web* (sistema *web*) a partir da tela de *Login* do sistema como é apresentado na Figura 47.

Figura 47 - Tela de *Login* do Cidadão Fiscal *Web*



O usuário e a senha utilizados neste sistema são os mesmos que os funcionários da prefeitura já utilizam no GRP, facilitando assim o acesso ao sistema. Caso o usuário que realizou o *login* seja ligado ao setor de Ouvidoria da Prefeitura, ele é conhecido como Atendente da Ouvidoria e é responsável por realizar a triagem nas solicitações antes de encaminhá-las para o setor responsável, sendo que o menu apresentado para este ator possui apenas a opção de distribuição de solicitações.

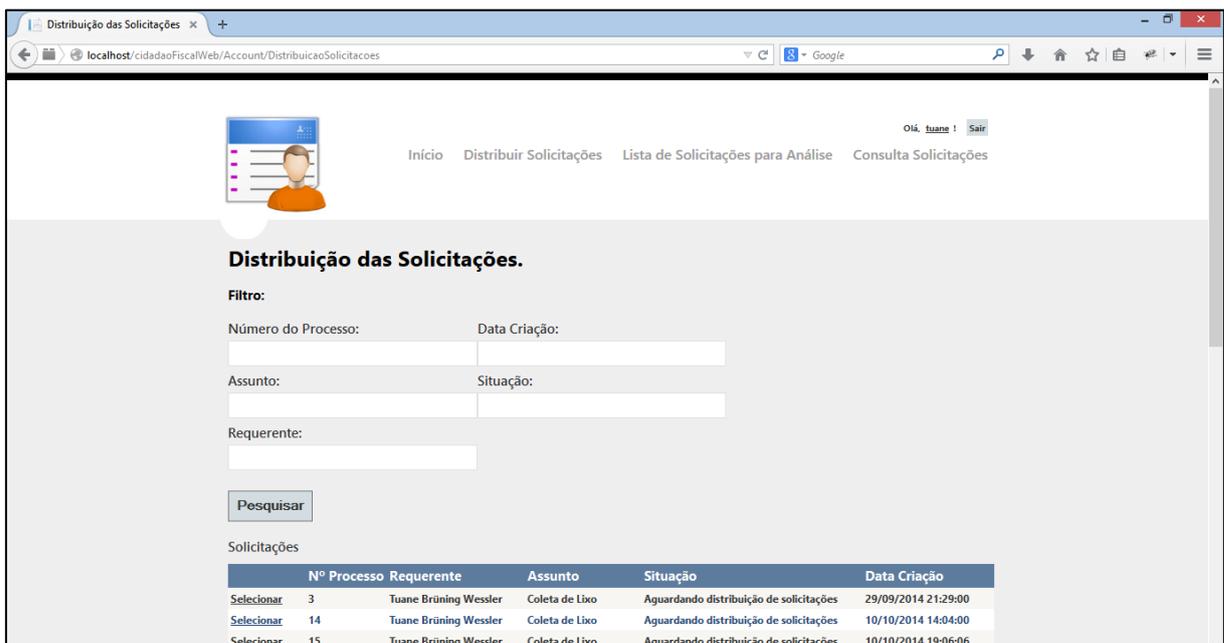
O atendente da Ouvidoria ao acessar o sistema *web* visualiza na tela inicial a informação contendo o número de solicitações que estão aguardando a triagem e distribuição para o setor responsável, essas são as solicitações que chegaram ao Cidadão Fiscal *Web* através do *Web Service*, como mostra a Figura 48. Sendo que esta funcionalidade comprova o atendimento ao Requisito Funcional RF08.

Figura 48 - Tela inicial do Atendente



Para realizar a triagem o atendente da Ouvidoria deverá selecionar a opção “Distribuir Solicitações” do menu. Esta tela contém na parte superior da tela um filtro, caso seja necessário a busca de alguma solicitações para ser encaminhada. A Figura 49 apresenta uma lista de solicitações onde o Atendente deverá selecionar a opção “Selecionar” do *grid* e logo após deverá selecionar o botão “Distribuir a Solicitação”.

Figura 49 - Tela de distribuição de solicitações



Após selecionar o botão para “Distribuir a Solicitação” será apresentada uma tela

conforme a Figura 50 onde o atendente da Ouvidoria irá analisar as informações enviadas pelo munícipe, como a descrição da solicitação, a foto e a localização geográfica.

Figura 50 - Tela distribuir solicitações

**Encaminhar Solicitação para Setor.**

Número do Processo:40

Requerente:Tuane Brining Wessler

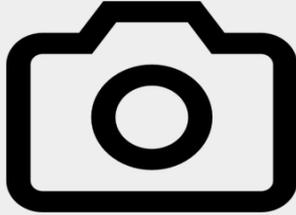
Data de Criação:02/11/2014 21:40:42

Assunto:Coleta de Lixo

Situação:Aguardando distribuição de solicitações

Descrição do Requerente:Muita fumaça da empresa X.

**Foto da Solicitação:**



Na mesma tela o atendente irá analisar a localização geográfica da solicitação, caso a latitude e a longitude tenham sido enviadas; caso contrário aparecerá apenas a descrição do endereço, como pode ser visto na análise visualizada na Figura 51.

Figura 51 - Tela distribuir solicitação visualizar localização

**Endereço da Solicitação:**

Endereço:R. Manoel Salvador, 408 - Valparaíso, 89023-660 - Santa Catarina/Blumenau - SC



**Deferimento da Solicitação:**

Deseja deferir a solicitação?

Deferido
  Indeferido

Por último, após o atendente verificar todas as informações o mesmo deverá deferir ou indeferir a solicitação. Caso a solicitação seja indeferida o mesmo deverá informar o motivo

do indeferimento e caso a solicitação seja deferida o atendente deverá confirmar se o setor responsável que já foi previamente carregado levando em consideração a categoria selecionada no aplicativo, pois todas as categorias estão ligadas a um setor responsável. A Figura 52 apresenta o que foi citado acima, sendo que esta funcionalidade comprova o Requisito Funcional RF09.

Figura 52 - Tela distribuição solicitação deferimento

**Deferimento da Solicitação:**

Deseja deferir a solicitação?

Deferido  Indeferido

Setor Responsável: SAMAЕ

Parecer:

Encaminhar

Copyright © 2014|Todos os direitos reservados

As solicitações encaminhadas pelos atendentes da Ouvidoria para os setores responsáveis só podem ser visualizadas pelos analistas responsáveis do mesmo setor, como por exemplo, se uma solicitação foi encaminhada para o setor da SAMAЕ, apenas os analistas do setor da SAMAЕ poderão analisar estas solicitações, lembrando que os atendentes possuem acesso apenas a opção contendo a lista de solicitações no menu principal. A Figura 53 mostra a tela contendo a lista de solicitações que estão aguardando a análise ou que ainda estão em análise. A tela possui um filtro para facilitar a busca por solicitações e para realizar a análise da solicitação o analista deverá selecionar uma das solicitações e selecionar o botão “Analisar Solicitação”.

Figura 53 - Tela lista de solicitações para análise

**Lista de Solicitações para Análise.**

**Análise de Solicitações:**

Número do Processo:  Data de Criação:

Assunto:  Situação:

Requerente:

Solicitações

	Nº Processo	Requerente	Assunto	Situação	Data Criação
<a href="#">Selecionar</a>	2	Tuane Brüning Wessler	Coleta de Lixo	Solicitação enviada para o Setor Responsável	24/09/2014 00:00:00
<a href="#">Selecionar</a>	6	Tuane Brüning Wessler	Coleta de Lixo	Em análise	05/10/2014 18:05:00
<a href="#">Selecionar</a>	10	Tuane Brüning Wessler	Coleta de Lixo	Solicitação enviada para o Setor Responsável	08/10/2014 05:00:00
<a href="#">Selecionar</a>	11	Tuane Brüning Wessler	Coleta de Lixo	Solicitação enviada para o Setor Responsável	08/10/2014 09:08:00
<a href="#">Selecionar</a>	12	Tuane Brüning Wessler	Coleta de Lixo	Solicitação enviada para o Setor Responsável	08/10/2014 08:05:59
<a href="#">Selecionar</a>	13	Tuane Brüning Wessler	Coleta de Lixo	Em análise	08/10/2014 20:56:58
<a href="#">Selecionar</a>	21	Tuane B Wessler	Lixo	Solicitação enviada para o Setor Responsável	21/10/2014 23:48:54
<a href="#">Selecionar</a>	25	Tuane B Wessler	Buracos	Solicitação enviada para o Setor Responsável	22/10/2014 00:19:09
<a href="#">Selecionar</a>	36	Tuane B Wessler	Coleta de Lixo	Solicitação enviada para o Setor Responsável	23/10/2014 11:44:34

Ao abrir a análise pela primeira vez, a primeira análise será criada automaticamente como mostra a Figura 54.

Figura 54 - Tela de Análises

**Informações da Análise**

Atendente da Ouvidoria: tuane

Data de distribuição: 08/10/2014 19:42:45

---

**Análises**

Selecione a Análise	Data de Criação	Data de Conclusão
<a href="#">Análise - 1</a>	13/10/2014 16:15:05	

---

**Conclusão do Processo**

Deseja concluir o processo?

Concluir o Processo

Copyright © 2014 Todos os direitos reservados

Para preencher as informações da análise o analista deverá selecionar o *link* que está no *grid* de análises (neste caso, Análise – 1). Logo após esse procedimento o sistema irá apresentar uma tela com o resumo da solicitação e um campo para preencher a análise. Caso o analista queira apenas escrever o que está acontecendo ou atualizar algo sem enviar para o município, o mesmo deverá preencher o campo e apenas selecionar o botão “Salvar Análise”. A próxima vez que acessar a tela, a informação será retornada e assim que o analista desejar concluir a análise, o mesmo deverá selecionar o campo “Concluir a Análise” e então selecionar o botão “Salvar Análise”, esta análise não poderá mais ser modificada. A tela referente a este procedimento é apresentada na Figura 55.

Figura 55 - Tela de Análise-1

Descrição do Requerente: Lixo na calçada, dificulta a passagem dos pedestres

---

**Análise - 1**

Parecer da Análise:

A solicitação foi encaminhada para os funcionários realizarem a limpeza da rua.

Deseja concluir a análise?

Concluir a Análise

Copyright © 2014[Todos os direitos reservados]

Poderão ser realizadas quantas análises forem necessárias, enquanto a solicitação não for resolvida o analista poderá criar novas análises e para a conclusão da solicitação o analista deverá possuir todas as análises concluídas. Para concluir a solicitação o analista deverá selecionar o campo “Concluir processo” e logo após deverá preencher o parecer geral do processo, informando a conclusão final do processo, tanto informando uma aprovação, como uma rejeição da solicitação e então selecionar o botão “Concluir Solicitação”, como mostra a Figura 56. Estas funcionalidades comprovam o atendimento do Requisito Funcional RF10 e o Requisito Funcional RF11.

Figura 56 - Conclusão da análise

Selecione a Análise	Data de Criação	Data de Conclusão
Análise - 1	13/10/2014 16:15:05	10/11/2014 01:39:39

[Criar Nova Análise](#)

---

**Conclusão do Processo**

Deseja concluir o processo?  
 Concluir o Processo

Parecer Geral do Processo:

A solicitação foi resolvida, todo o lixo foi retirado da rua.

[Concluir Processo](#)

A tela de consulta de solicitações pode ser visualizada tanto pelos atendentes da ouvidoria como pelos analistas, (coordenadores ou gerentes que possuem acesso a todos os itens do menu). Esta tela apresenta uma opção de filtro como mostra a Figura 57.

Figura 57 - Tela de consulta de solicitações

**Consultar Solicitação.**

**Consultar Solicitação:**

Número do Processo:  Exercício:

Assunto:  Situação:

Requerente:

[Pesquisar](#)

Solicitações

	Nº Processo	Requerente	Assunto	Situação	Data Criação
<a href="#">Selecionar</a>	1	Tuane Brüning Wessler	Coleta de Lixo	Finalizado	21/09/2014 00:00:00
<a href="#">Selecionar</a>	2	Tuane Brüning Wessler	Coleta de Lixo	Solicitação enviada para o Setor Responsável	24/09/2014 00:00:00
<a href="#">Selecionar</a>	3	Tuane Brüning Wessler	Coleta de Lixo	Aguardando distribuição de solicitações	29/09/2014 21:29:00
<a href="#">Selecionar</a>	4	Tuane Brüning Wessler	Coleta de Lixo	Enviado para o Setor de Ouvidoria	30/09/2014 00:34:00
<a href="#">Selecionar</a>	5	Tuane Brüning Wessler	Coleta de Lixo	Enviado para o Setor de Ouvidoria	30/09/2014 00:39:06
<a href="#">Selecionar</a>	6	Tuane Brüning Wessler	Coleta de Lixo	Em análise	05/10/2014 18:05:00

### 3.4 RESULTADOS E DISCUSSÃO

O aplicativo desenvolvido atendeu as expectativas e requisitos propostos, propiciando uma maior comodidade para os munícipes utilizarem o Cidadão Fiscal através de seus *smartphones* ou *tablets*. A utilização de uma foto para registrar a solicitação ajuda tanto os munícipes a apresentar provas do que está sendo solicitado, como também para os atendentes da Ouvidoria, os coordenadores e analistas da Prefeitura Municipal de Blumenau a verificarem com mais eficácia as solicitações encaminhadas.

A utilização das categorias foi extremamente importante para facilitar a identificação do que está sendo encaminhado para a ouvidoria, pois quando o munícipe seleciona em qual categoria está a solicitação, o aplicativo já retorna qual o setor é responsável pela mesma. O uso do Google Maps ajudou muito, pois para aqueles que queiram enviar a localização na solicitação sabem que a equipe da prefeitura será encaminhada exatamente para aquele lugar, sem contar que muitas solicitações podem ser denúncias o que facilita ainda mais a localização da mesma.

Outro item muito importante na utilização do aplicativo é a possibilidade de salvar a solicitação, se não houver uma conexão com a internet no momento. Esta opção é muito importante atualmente, pois não são todas as pessoas que possuem um dispositivo móvel com acesso a rede de internet móvel, podendo ocorrer também a falta de sinal de internet no local da solicitação.

Já no Cidadão Fiscal *Web*, entende-se que o resultado atendeu as necessidades, principalmente pela integração com aplicativo onde todas as solicitações são registradas e podem ser acompanhadas facilitando assim a comunicação entre o munícipe e a Prefeitura. Além de possibilitar a pré análise da solicitação pelos atendentes da Ouvidoria, facilitando assim a triagem das informações junto ao deferimento ou indeferimento inicial, possibilita aos coordenadores e analistas realizar várias análises da mesma solicitação.

Através dos trabalhos correlatos citados é possível perceber que todos foram desenvolvidos para ajudar os cidadãos a mostrarem os problemas que estão ocorrendo nas cidades brasileiras. O trabalho correlato mais próximo ao presente trabalho é o Colab que, diferente dos demais, é o único que apresenta integração com algumas prefeituras, como a do município de Curitiba, no estado do Paraná, e o município de Rondonópolis, estado do Mato Grosso. O Colab pode ser utilizado pelos munícipes de Blumenau, mas todas as fiscalizações

realizadas atualmente ficam apenas compartilhadas no aplicativo, sendo que outros usuários podem visualizar as mesmas, mas as informações não são encaminhadas para a prefeitura.

O Quadro 3 apresenta um comparativo entre todos os trabalhos correlatos citados juntamente com uma projeção do que foi desenvolvido no aplicativo Cidadão Fiscal.

Quadro 3 - Comparativo entre aplicativos correlatos

	Permite ao usuário detalhar textualmente o problema?	Utiliza mapas para a localização geográfica?	Utiliza uma classificação por categorização?	Possui integração com a prefeitura?
Colab	Sim	Não	Sim	Sim
Cidade Legal	Não	Sim	Sim	Não
Cidadera	Sim	Sim	Sim	Não
Urbotip	Não	Sim	Sim	Não
Cidadão Fiscal	Sim	Sim	Sim	Sim

O aplicativo Cidadão Fiscal engloba as principais funcionalidades para a realização de um bom cadastro das reclamações e sugestões dos munícipes, contando com várias informações para tornar mais fácil a identificação do problema, assim como a solução.

## 4 CONCLUSÕES

Ao término do desenvolvimento do aplicativo e do sistema *web*, todos os objetivos foram alcançados. Isso significa que o aplicativo Cidadão Fiscal atende as necessidades dos munícipes em realizar reclamações ou sugestões que serão encaminhadas para a Prefeitura Municipal de Blumenau de uma forma prática e rápida, que são através dos dispositivos móveis.

O grande diferencial deste trabalho é a questão que as solicitações não permanecem apenas no dispositivo móvel ou para os outros usuários e sim realiza o trâmite das informações para o Cidadão Fiscal *Web*, que é responsável por manter essas informações na mesma base das solicitações que são aberta na Praça do Cidadão (sendo que para o presente trabalho foi utilizada um base de dados fictícia, por motivos de segurança) onde tanto para o atendente da Ouvidoria como para os analistas de cada setor da PMB, a identificação das solicitações é mais precisa e correta, pois conta com recursos como, a foto e a localização geográfica.

Os munícipes contam principalmente com a comodidade de receber as atualizações e a conclusão da solicitação diretamente no dispositivo móvel, sendo que para o perfeito funcionamento de todos esses itens, foi fundamental a correta integração com as ferramentas e tecnologias. Um exemplo é o envio e o recebimento das informações do aplicativo para o sistema *web* através do *Web Service*, onde ambos necessitam estar configurados corretamente para a realização do consumo do mesmo.

O aplicativo foi desenvolvido para o sistema operacional Android utilizando a linguagem de programação Java, juntamente com as bibliotecas para auxiliar o desenvolvimento, como a biblioteca do Google Maps por exemplo. Para o desenvolvimento do sistema *web*, que foi realizado na linguagem de programação C# (ASP.NET) e o banco de dados Oracle, também foram utilizadas técnicas e tecnologias atuais como por exemplo, o LINQ e o NHibernate.

Por fim, a autora deste trabalho desde que iniciou a busca de uma ideia para a realização do mesmo, tinha como objetivo o desenvolvimento de um aplicativo que ajudasse a população de alguma forma. Depois de muitas ideias, surgiu o Cidadão Fiscal que tem como maiores beneficiários os munícipes de Blumenau.

#### 4.1 EXTENSÕES

Para trabalhos futuros, surge a oportunidade de complementar e melhorar o aplicativo e o sistema *web*. Como sugestões pode-se implementar o envio da solicitação para a Prefeitura como solicitação anônima, pois em alguns casos como denúncia, o munícipe não queira que seu nome fique vinculado ao mesmo. Outra sugestão seria adicionar o mapa da cidade de Blumenau no aplicativo e mostrar o índice de solicitações por categoria.

Realizar a criação de um novo *Web Service* para integração com outros sistemas de outras prefeituras, permitindo o uso da aplicação para qualquer município que ache vantajoso a sua adoção, dessa forma a integração entre os sistemas não seria realizada diretamente via banco de dados e sim abstraindo os detalhes de cada sistema através da exposição de um serviço.

Seria interessante também no Cidadão Fiscal *Web* adicionar uma opção no menu que apenas os coordenadores e ou gerentes possam acessar. Para os mesmos será possível visualizar alguns relatórios gerenciais, como por exemplo, bairros que possuem o maior número de solicitações enviadas, categorias mais enviadas, quantidade de processos deferidos e indeferidos, assim como o tempo levado para concluir um processo, entre outros. Esta funcionalidade é importante pois se o aplicativo for utilizado por outras prefeituras as mesmas necessitam possuir uma ferramenta para visualizar os relatórios gerados com os dados do aplicativo.

Por fim para atender os munícipes em geral, seria interessante desenvolver o aplicativo para as plataformas do *Windows Phone* e também do *iOS*, realizando então a integração com o Cidadão Fiscal *Web* através do mesmo *Web Service* já existente.

## REFERÊNCIAS

ACESSO À INFORMAÇÃO. **Homepage**. Brasília, 2014. Disponível em: <<http://www.acaoainformacao.gov.br/>>. Acesso em: 01 ago. 2014.

ANDROID DEVELOPERS. **Android, the world's most popular mobile platform**. Mountain View, 2013. Disponível em: <<http://developer.android.com/about/index.html>>. Acesso em: 29 out. 2014.

BUTZER, Renato. **O uso das redes sociais e o mercado consumidor**. Blumenau, 2011. Disponível em: <<http://www.cdiblumenau.com.br/artigos/index.php?s=56&codigo=87>>. Acesso em: 05 ago. 2014.

CAIÇARA JUNIOR, Cícero; PARIS, Wanderson Stael. **Informática. Internet e Aplicativos**. 1. ed. São Paulo, 2008.

CORONATO, Marcos. **Um clique contra o buraco na rua**. Revista Época. Edição 818. São Paulo, 2014.

FERREIRA, Hélio José; SANTOS, Hilma Araújo dos. **Afinal, o que é essa tal de ouvidoria?** São Paulo, 2011. Disponível em: <<http://www.abonacional.org.br/artigo.php?codigo=36>>. Acesso em: 05 ago. 2014.

FUNDAÇÃO DE ENSINO E ENGENHARIA DE SANTA CATARINA. **Homepage**. Florianópolis, 2014. Disponível em: <<http://www.feesc.org.br/site/>>. Acesso em: 21 ago. 2014.

MICROSOFT CORPORATION. **LINQ**. [S.l.], 2014. Disponível em: <<http://msdn.microsoft.com/pt-br/library/bb397906.aspx>>. Acesso em: 01 out. 2014.

GOOGLE PLAY. **Homepage**. São Paulo, 2014. Disponível em: <<https://play.google.com/store>>. Acesso em: 21 ago. 2014.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Informação sobre os municípios**. [S.l.], 2014. Disponível em: <<http://www.cidades.ibge.gov.br/xtras/perfil.php?lang=&codmun=420240>>. Acesso em: 21 nov. 2014.

MONK, Simon. **Tecnologia da Informação para a Gestão: Em busca de um melhor desempenho estratégico e operacional**. 8ª ed. Porto Alegre: Bookman Editora, 2013.

PINHO, Roberto. **Integração dos processos de ouvidoria de uma agência reguladora com os agentes regulados** – o caso da agência nacional de energia elétrica – ANEEL. 2002. Mestrado em regulação da indústria de energia - Universidade Salvador, Salvador, 2002. Disponível em: <[http://www.energia.unifacs.br/dissertacoes/roberto\\_pinho.pdf](http://www.energia.unifacs.br/dissertacoes/roberto_pinho.pdf)>. Acesso em: 15 ago. 2014.

PREFEITURA MUNICIPAL DE BLUMENAU. **Portal da Prefeitura de Blumenau**. Blumenau, 2014a. Disponível em: <<http://www.blumenau.sc.gov.br/>>. Acesso em: 21 ago. 2014.

PREFEITURA MUNICIPAL DE BLUMENAU. **Dados Geográficos**. Blumenau, 2014b. Disponível em: <<http://www.blumenau.sc.gov.br/secretarias/secretaria-de-planejamento/pagina/historia-sobre-municipio//divisa-administrativa-bairros>>. Acesso em: 20 nov. 2014.

PREFEITURA MUNICIPAL DE BLUMENAU. **Bairros – Divisão Administrativa**. Blumenau, 2014c. Disponível em: <<http://www.blumenau.sc.gov.br/blumenau/as5d1a5sd4a4sd>>. Acesso em: 20 nov. 2014.

SCHMIDT, Eric; COHEN, Jared. **A nova era digital: Como será o futuro das pessoas, das nações e dos negócios**. ed.1. Rio de Janeiro: Editora Intrínseca, 2013.

SILVA, Raquel da; HÜSKES, Monika Ilse. **Ouvidoria: Um estudo de caso na Prefeitura Municipal de Blumenau**. Blumenau, 2007. Disponível em: <<http://rica.unibes.com.br/index.php/rica/article/viewFile/51/47>>. Acesso em: 03 ago. 2014.

SPLENGER, Eliana Lizete. **Ouvidoria PMB**. - TCC [mensagem pessoal]. Mensagem recebida por <[tuane@fesc.org.br](mailto:tuane@fesc.org.br)>. Acesso em: 28 mar. 2014.

THEMA INFORMÁTICA. **Homepage**. Porto Alegre, 2014. Disponível em: <<https://www.thema.inf.br>>. Acesso em: 05 ago. 2014.

UOL TECNOLOGIA. **Dicas: Aplicativos: Aplicativos transformam smartphones em ferramentas da cidadania**. São Paulo, 2014. Disponível em: <<http://tecnologia.uol.com.br/noticias/redacao/2013/12/16/aplicativos-transformam-smartphone-em-ferramenta-de-cidadania.htm>>. Acesso em: 16 ago. 2014.

URBOTIP. **Urbotip**. Pelotas, 2014. Disponível em: <<http://www.urbotip.com>>. Acesso em: 15 ago. 2014.

VISMONA, Edson L. et al. **A ouvidoria no Brasil**. 1. ed. São Paulo : Imprensa Oficial do Estado de São Paulo, 2002.

## APÊNDICE A – Descrição dos Casos de Uso

Este Apêndice apresenta a descrição dos principais casos de uso descritos na seção de especificação deste trabalho, conforme o Quadro 4.

Quadro 4 - Descrição dos casos de uso

### **UC01 Cadastrar Usuário**

Permite ao munícipe informar os dados para criação de um usuário para ter acesso às funcionalidades do aplicativo como registrar solicitações, consultar as solicitações, bem como alterar e excluir informações.

#### **Constraints**

*Pré condição:* O munícipe deve possuir o aplicativo instalado em seu dispositivo móvel.

*Pré condição:* O munícipe deve possuir acesso à Internet.

*Pós condição:* O munícipe deverá estar devidamente cadastrado.

#### **Cenários**

##### **Cadastrar Usuário {Principal}**

1. O usuário acessa o aplicativo em seu dispositivo móvel.
2. O aplicativo solicita a inserção dos dados para realizar o *login*.
3. O usuário seleciona a opção Cadastrar-se.
4. O aplicativo apresenta a tela de cadastro onde devem ser preenchidos os campos: Nome, CPF, Data de nascimento, Bairro, Endereço, Número e o CEP.
5. O usuário preenche todos os campos e seleciona o botão “Continuar”
6. O aplicativo apresenta a tela de cadastro dos dados de acesso ao aplicativo através dos campos de E-mail, Usuário, Senha, Confirmação da senha.
7. O usuário seleciona o botão “Salvar”.
8. O aplicativo verifica os dados informados para o cadastro. (Fluxo de Exceção Verificação de dados preenchidos)
9. O usuário pode utilizar o *login* e senha cadastrados para acessar o aplicativo.

##### **Verificação de dados preenchidos {Exceção}**

No passo 8, caso o usuário não preencha algum dos campos ou o usuário já exista ou ainda a senha possua menos de 5 caracteres o aplicativo irá mostrar uma mensagem informando o erro.

### **UC02 Alterar Senha**

Permite ao usuário do aplicativo a alteração da senha usada para acessar o aplicativo.

### **UC03 Realizar Login**

Permite ao munícipe através da identificação de usuário e senha conectar-se ao aplicativo.

### **UC04 Registrar Solicitação**

Permite ao munícipe registrar as solicitações (reclamações ou sugestões). Serão cadastrados os seguintes dados: uma foto que representa o que está acontecendo, uma categoria que represente a solicitação, a descrição e a localização geográfica da solicitação.

**Constraints**

*Pré-condição* . O munícipe deve possuir um dispositivo móvel com acesso a internet.

*Pré-condição* . O munícipe deve possuir um usuário cadastrado no aplicativo.

*Pós-condição* . A(s) solicitação(ões) foi(ram) enviadas (as) para a triagem.

**Cenários****Registrar Solicitação {Principal}.**

1. O caso de uso inicia quando o munícipe seleciona a opção “Nova Solicitação”.
2. O aplicativo apresenta a tela de solicitação o existe campo para inserir a foto.
3. O usuário adiciona a foto (não é obrigatória) e seleciona o botão “Continuar”. (Fluxo alternativo Adicionar Foto)
4. O aplicativo mostra a tela onde existem os seguintes campos para serem preenchidos: Categoria, Descrição e Localização geográfica.
5. O munícipe seleciona uma categoria, adiciona a descrição da solicitação, descreve ou adiciona a localização geográfica e seleciona a opção “Enviar Solicitação”. (Fluxo alternativo Gravar localização GPS).
6. O aplicativo valida as informações digitadas pelo munícipe e a conexão com a Internet. (Fluxo de Exceção Campo não preenchido), (Fluxo de alternativo Serviço de Internet Indisponível)
7. O aplicativo realiza o envio das informações para o módulo *web*, salvando os dados no GRP (banco de dados).
8. Término do caso de uso.

**Adicionar Foto {Alternativo}.**

- 3.1. O fluxo alternativo inicia quando o aplicativo apresenta uma tela para o munícipe tirar uma foto do local. (Fluxo alternativo Adicionar foto existente no celular)
- 3.2. O munícipe confirma a foto. O caso de uso retorna ao passo 4 do fluxo principal.

**Adicionar foto existente no celular {Alternativo}.**

No passo 1, caso o munícipe deseje adicionar uma foto existente no celular.

- 3.1.1. O aplicativo apresenta uma tela com as pastas que contém imagens do dispositivo móvel.
- 3.1.2. O munícipe seleciona a foto e selecionar a opção “Salvar”.
- 3.1.3. O aplicativo valida o arquivo selecionado.
- 3.1.4. O caso retorna ao passo 3.2 do fluxo alternativo Adicionar Foto.

**Gravar localização GPS {Alternativo}.**

- 5.1. O fluxo alternativo inicia quando o aplicativo apresenta uma tela de busca da localização geográfica. (Fluxo alternativo Sistema de Posicionamento Global (GPS) desativado)
- 5.2. O munícipe confirma a localização e selecionar o botão “Selecionar”.
- 5.3. O caso de uso retorna ao passo 6 do fluxo principal.

**Sistema de Posicionamento Global (GPS) desativado {Alternativo}.**

No passo 5.1, caso o munícipe não esteja com o Sistema de Posicionamento Global (GPS) ativado no dispositivo móvel.

- 5.1.1. O aplicativo apresenta uma tela com uma mensagem informando ao munícipe que o recurso precisa ser ativado.
- 5.1.2. O munícipe seleciona a opção “Ativar GPS”.
- 5.1.3. O caso de uso retorna ao passo 5.2 do fluxo alternativo Gravar localização GPS.

**Campo não preenchido {Exceção}**

No passo 6, caso o munícipe não informe os campos de categoria ou descrição, o aplicativo apresenta mensagem explicando qual campo deverá ser preenchido.

**Serviço de Internet Indisponível {Alternativo}.**

No passo 6, caso o munícipe tente enviar a solicitação e o Serviço de Internet não esteja disponível.

6.1. O aplicativo apresenta uma tela com uma mensagem informando ao munícipe que o dispositivo não possui conexão com uma rede de Internet e pergunta se o usuário deseja enviar a solicitação mais tarde.

6.2 O munícipe seleciona a opção que deseja enviar agora. (Fluxo alternativo Salvar Solicitação)

6.3 O munícipe deve conectar-se a uma rede de Internet.

6.4 O caso de uso retorna ao passo 7 do fluxo principal.

**Salvar Solicitação {Alternativo}.**

No passo 6.2, o munícipe selecionou a opção para salvar a solicitação e enviar mais tarde.

6.2.1 O aplicativo salva a solicitação como pendente de envio.

6.2.2 O caso de uso retornar ao passo 8 do fluxo principal.

**UC05 Consultar Solicitações**

Permite ao munícipe consultar todas as solicitações enviadas e receber as atualizações da solicitação.

**UC06 Enviar Solicitações Pendentes**

Permite ao munícipe enviar as solicitações pendentes, sendo que esta opção se encontra no *menu* do lado esquerdo da tela de Solicitações Pendentes e é possível visualizar a solicitações antes de enviá-la através do botão “Enviar Solicitação”.

**UC07 Excluir Solicitações Pendentes**

Permite ao munícipe através da tela de Solicitações Pendentes através do *menu* que se localiza a esquerda da tela excluir a solicitação.

**UC08 Realizar triagem e distribuição das solicitações**

Permite ao atendente da Ouvidoria da Prefeitura Municipal de Blumenau realizar uma triagem de todas as solicitações enviadas pelo aplicativo, para identificar se a descrição da reclamação ou sugestão é coerente assim como a foto e a localização geográfica (quando existirem). Se todos os dados estiverem de acordo o atendente encaminha para o setor responsável.

**Constraints**

*Pré-condição.* Existirem solicitações em situação igual a “Aguardando distribuição de solicitações”.

*Pós-condição.* A(s) solicitação(ões) foi(ram) encaminhada(s) para o setor responsável para a análise de um analista do mesmo.

**Cenários****Realizar triagem e distribuição das solicitações {Principal}.**

1. O caso de uso inicia quando o atendente seleciona a opção do *menu* “Distribuir

Solicitações”.

2. O sistema apresenta uma lista com todas as solicitações enviadas com a situação “Aguardando distribuição de solicitações”.
3. O atendente da Prefeitura seleciona uma solicitação da lista para realizar distribuição e triagem e então selecionar o botão “Distribuir a Solicitação”.
4. O sistema apresenta os dados da solicitação (processo) selecionado, contendo o número do processo, o nome do requerente, a data de criação, o assunto, a situação, a descrição da solicitação, a foto e a localização geográfica (se possuir).
5. O atendente da Prefeitura verifica se todos os dados são coerentes e seleciona a opção “Deferido”. (Fluxo alternativo Indeferido (Conteúdo não suficiente ou inadequado))
6. O sistema apresenta campo para o atendente escrever um parecer sobre o deferimento ou indeferimento, sendo que é obrigatório o preenchimento se o atendente optar pela opção indeferido.
7. O atendente da Prefeitura clica na opção “Encaminhar”.
8. O sistema realiza a gravação da solicitação na base de dados do GRP.
9. Término do caso de uso.

**Indeferido (Conteúdo não suficiente ou inadequado) {Alternativo}.**

No passo 5, o atendente verifica se todos os dados são coerentes.

- 5.1. Se o atendente verificar que os dados contidos na solicitação não são suficientes ou são inadequados o mesmo seleciona a opção “Indeferido”.
- 5.2. O sistema apresenta um campo para inserir uma descrição do motivo da recusa da solicitação.
- 5.3. O atendente preenche a descrição e seleciona a opção “Encaminhar”.
- 5.4. O caso de uso retorna ao passo 9 do fluxo principal.

**UC09 Realizar a análise das solicitações**

Permite ao analista responsável por determinado setor realizar a análise das solicitações e então conseguir solucionar a solicitação do munícipe.

**Constraints**

*Pré-condição* . O analista deve possuir solicitações com a situação “Solicitação enviada para o Setor Responsável” ou “Em análise”.

*Pós-condição* . A(s) solicitação(ões) foi(ram) resolvidas.

**Cenários**

**Realizar a análise das solicitações {Principal}.**

1. O caso de uso inicia quando o atendente seleciona a opção do *menu* “Lista de Solicitações para a Análise”.
2. O sistema apresenta uma lista com todas as solicitações que estão aguardando a análise ou que já estão em análise.
3. O analista seleciona uma solicitação na lista e então seleciona a opção “Analisar Processo”
4. O sistema apresenta uma nova tela contendo os dados da solicitação e a opção de criar nova análise ou concluir o processo (solicitação), a primeira vez que a solicitação for aberta será criada a primeira análise.
5. O analista seleciona a análise desejada através da opção “Selecione a Análise”.
6. O sistema apresenta um resumo da solicitação e um campo para descrever o parecer da análise. Caso na primeira análise a solicitação tenha se resolvido o analista seleciona a opção “Concluir a análise” e então “Salvar”, caso contrário basta selecionar a opção “Salvar” que a próxima vez em que acessar a análise os dados que foram preenchidos serão carregados pelo

sistema.

7. O analista realiza o preenchimento necessário na análise e seleciona o botão “Salvar”.
8. O sistema retorna a tela principal da solicitação em análise, onde o analista poderá realizar quantas análises forem necessárias.
9. O analista após realizar todas as análises necessárias seleciona o botão “Concluir Processo”.
10. O sistema apresenta um campo para o preenchimento do parecer geral do processo (solicitação).
11. O analista deverá preencher o campo do parecer geral do processo e então selecionar o botão “Concluir Processo”.
12. Término do caso de uso.

## APÊNDICE B – Descrição do Dicionário de Dados

Este Apêndice apresenta a descrição das tabelas do banco de dados apresentadas na seção de especificação deste trabalho. Os tipos de dados utilizados nos atributos pelas tabelas do banco de dados do Cidadão Fiscal são:

- a) *integer*: para variáveis numéricas inteiras;
- b) *bool*: para variáveis booleanas (verdadeiro e falso);
- c) *text*: para variáveis textuais/descritivas;
- d) *datetime*: para variáveis do tipo data e hora;
- e) *blob*: para variáveis do tipo *array* de *bytes*.

Todas as tabelas possuem ao menos um atributo identificado como chave primária, ou *Primary Key* (PK). As tabelas envolvidas em relacionamentos possuem chaves estrangeiras, ou *Foreign Key* (FK). Nos Quadros de 5 a 9 apresenta-se o dicionário de dados das tabelas do aplicativo Cidadão Fiscal.

Quadro 5 - Tabela de bairros de Blumenau

<b>Bairro</b>					
Armazena os dados dos bairros da cidade de Blumenau					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
idBairro	Número sequencial do bairro	<i>integer</i>		Sim	Não
Id	Id do bairro na tabela da prefeitura	<i>integer</i>		Não	Não
descricao	Nome do bairro	<i>varchar</i>	max	Não	Não

Quadro 6 - Tabela de categorias das solicitações

<b>Categoria</b>					
Armazena os dados das categorias referente a solicitação					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
idCategoria	Número sequencial da categoria	<i>integer</i>		Sim	Não
descricaoCategoria	Nome da categoria	<i>varchar</i>	max	Não	Não

Quadro 7 - Tabela de logradouros de Blumenau

<b>Logradouro</b>					
Armazena os dados dos logradouros da cidade de Blumenau					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
idLogradouros	Número sequencial do logradouro	<i>integer</i>		Sim	Não
id	Id do logradouro na tabela da prefeitura	<i>integer</i>		Não	Não
descricao	Nome do logradouro	<i>varchar</i>	Max	Não	Não
idBairro	Campo de ligação com a tabela Bairro	<i>integer</i>		Não	Sim

Quadro 8 - Tabela com os dados dos municípios

<b>Município</b>					
Armazena os dados do município					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
idMunicípio	Número sequencial do logradouro	<i>integer</i>		Sim	Não
nome	Nome do município	<i>varchar</i>	Max	Não	Não
idLogradouro	Campo de ligação com a tabela Logradouro	<i>integer</i>		Não	Sim
numeroEndereco	Número do endereço	<i>varchar</i>	max	Não	Não
idBairro	Campo de ligação com a tabela Bairro	<i>integer</i>		Não	Sim
cep	CEP do município	<i>integer</i>		Não	Não
CPF	Número do CPF do município	<i>integer</i>		Não	Não
dataNascimento	Data de nascimento do município	<i>DateTime</i>		Não	Não
nomeMae	Nome da mãe do município	<i>varchar</i>	max	Não	Não

Quadro 9 - Tabela com os dados dos usuários

<b>Usuario</b>					
Armazena os dados do usuário cadastrado no aplicativo					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
idUsuario	Número sequencial do logradouro	<i>integer</i>		Sim	Não
idMunícipe	Campo de ligação com a tabela Municípe	<i>integer</i>		Não	Sim
loginUsuario	Nome do usuário do aplicativo	<i>varchar</i>	Max	Não	Não
senha	Senha para acesso ao aplicativo	<i>varchar</i>	Max	Não	Não
email	<i>E-mail</i> do município	<i>varchar</i>	Max	Não	Não

Quadro 10 - Tabela de solicitações

<b>Solicitacao</b>					
Armazena os dados das Solicitações no Aplicativo					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
idSolicitacao	Número sequencial da Solicitação	<i>integer</i>		Sim	Não
idUsuario	Campo de ligação com a tabela Usuario	<i>integer</i>		Não	Sim

descrição	Descrição da solicitação	<i>varchar</i>	max	Não	Não
idCategoria	Campo de ligação com a tabela Categoria	<i>integer</i>		Sim	Sim
coordenadas	Coordenadas Geográficas da solicitação	<i>Varchar</i>	max	Não	Não
dataSolicitacao	Data da solicitação	<i>dateTime</i>		Não	Não
fotoSolicitacao	Foto enviada da Solicitação	<i>BLOB</i>		Não	Não
endereço	Descrição do Endereço da solicitação	<i>varchar</i>	max	Não	Não
idProcesso	Após enviar a solicitação a chave do processo é retornada	<i>integer</i>		Não	Não
retornoSolicitacao	As consultas são gravadas neste campo	<i>varchar</i>	max	Não	Não

Os tipos de dados utilizados nos atributos pelas tabelas do banco de dados do Cidadão Fiscal Web são:

- a) *integer*: para variáveis numéricas inteiras;
- b) *varchar*: para variáveis textuais/descritivas;
- c) *date*: para variáveis do tipo data e hora;
- d) *blob*: para variáveis do tipo *array* de *bytes*;
- e) *boolean*: para variáveis booleanas (verdadeiro e falso).

Todas as tabelas possuem ao menos um atributo identificado como chave primária, ou *Primary Key* (PK). As tabelas envolvidas em relacionamentos possuem chaves estrangeiras, ou *Foreign Key* (FK). Nos Quadros de 11 a 25 estão o dicionário de dados das tabelas do sistema *web*, Cidadão Fiscal *Web*.

Quadro 11 - Tabela de pessoas

<b>Pessoa</b>					
Armazena os dados das pessoas cadastradas					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_pessoa	Número sequencial da pessoa	<i>integer</i>		Sim	Não
endereco_id_Endereco	Campo de ligação com a tabela Endereco	<i>integer</i>		Não	Sim
cpf_cnpj	Número do CPF/CNPJ da pessoa	<i>varchar</i>	15	Não	Não
nome	Nome da pessoa	<i>varchar</i>	200	Não	Não
nome_mae	Nome da mãe da pessoa	<i>varchar</i>	200	Não	Não
nome_fantasia	Nome fantasia, caso for um CNPJ	<i>varchar</i>	200	Não	Não
data_nascimento	Data de nascimento da pessoa	<i>date</i>		Não	Não
data_cadastro	Data de cadastro da pessoa	<i>date</i>		Não	Não
numero_endereco	Número do endereço da pessoa	<i>varchar</i>	200	Não	Não

Quadro 12 - Tabela de bairros da cidade

<b>Bairro</b>					
Armazena os dados dos Bairros da Cidade de Blumenau					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_bairro	Número sequencial do bairro	<i>integer</i>		Sim	Não
nome	Nome do bairro	<i>varchar</i>	1000	Não	Não

Quadro 13 - Tabela de Endereco

<b>Endereco</b>					
Armazena os dados das ruas da Cidade de Blumenau					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_endereco	Número sequencial do endereço	<i>integer</i>		Sim	Não
bairro_id_Bairro	Campo de ligação com a tabela Bairro	<i>integer</i>		Não	Sim
logradouro	Nome da rua	<i>varchar</i>	4000	Não	Não
cep	CEP do endereço	<i>varchar</i>	50	Não	Não

Quadro 14 - Tabela de usuários cadastrados pelo aplicativo

<b>UsuarioCidadaoFiscal</b>					
Armazena os dados contendo as informações de <i>login</i> do aplicativo Cidadão Fiscal					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_usuario_cidadao_fiscal	Número	<i>integer</i>		Sim	Não

	sequencial do Usuário Cidadão Fiscal				
peessoa_id_Pessoa	Campo de ligação com a tabela Bairro	<i>integer</i>		Não	Sim
nome_usuario	Nome do Usuario do aplicativo	<i>integer</i>	200	Não	Não
senha	Senha do usuário do aplicativo	<i>varchar</i>	500	Não	Não
email	<i>E-mail</i> do usuário do aplicativo	<i>varchar</i>	500	Não	Não

Quadro 15 - Tabela de situações do processo

<b>SituacaoProcesso</b>					
Armazena os dados referente a situação do processo					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_situacao_processo	Número sequencial da Situação do Processo	<i>integer</i>		Sim	Não
descricao	Descrição da Situação do Processo	<i>varchar</i>	400	Não	Não

Quadro 16 - Tabela de tipos de protocolo

<b>TipoProtocolo</b>					
Armazena os dados dos tipos de protocolos					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_tipo_protocolo	Número sequencial do Tipo de Protocolo	<i>integer</i>		Sim	Não
nome_protocolo	Nome do protocolo	<i>varchar</i>	500	Não	Não

Quadro 17 - Tabela de tipos de processo

<b>TipoProcesso</b>					
Armazena os dados dos tipos de processos					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_tipo_processo	Número sequencial do Tipo de Processo	<i>integer</i>		Sim	Não
nome_processo	Nome do processo	<i>varchar</i>	200	Não	Não

Quadro 18 - Tabela de assuntos do processo

<b>Assunto</b>					
Armazena os dados dos assuntos referentes ao processo					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_assunto	Número sequencial do	<i>integer</i>		Sim	Não

	Assunto				
nome	Nome do Assunto	<i>varchar</i>	200	Não	Não
descricao	Descrição do Assunto	<i>varchar</i>	500	Não	Não

Quadro 19 - Tabela de setores

<b>Setor</b>					
Armazena os dados dos setores					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_setor	Número sequencial do Setor	<i>integer</i>		Sim	Não
nome	Nome do Setor	<i>varchar</i>	200	Não	Não

Quadro 20 - Tabela de setores e assuntos

<b>SetorAssunto</b>					
Armazena os dados agrupados do Setor e do Assunto					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
setor_id_setor	Campo de ligação com a tabela Setor	<i>integer</i>		Não	Sim
assunto_id_assunto	Campo de ligação com a tabela Assunto	<i>integer</i>		Não	Sim

Quadro 21 - Tabela de processos

<b>Processo</b>					
Armazena os dados dos processos					
Campo	Descrição	Tipo	Tam anho	Chave primária	Chave estrangei ra
id_processo	Número sequencial do Processo	<i>integer</i>		Sim	Não
arquivo_id_Arquivo	Campo de ligação com a tabela Arquivo	<i>integer</i>		Não	Sim
enderecoFiscalizacaoCidadao_id_Fiscalizacao	Campo de ligação com a tabela EnderecoFiscalizacaoCidadao	<i>integer</i>		Não	Sim
assunto_id_Assunto	Campo de ligação com a tabela Assunto	<i>integer</i>		Não	Sim
situacao_Processo_id_Situacao_Processo	Campo de ligação com a tabela SituacaoProcesso	<i>integer</i>		Não	Sim
tipoProtocolo_id_tipo_protocolo	Campo de ligação com a tabela Tipo Protocolo	<i>integer</i>		Não	Sim

pessoa_id_Pessoa	Campo de ligação com a tabela Pessoa	<i>integer</i>		Não	Sim
data_criacao	Data de criação do processo	<i>Date</i>		Não	Não
parecer	Parecer do processo	<i>varchar</i>	max	Não	Não
setor_origem	Setor de origem do processo	<i>integer</i>		Não	Não
setor_destino	Setor de destino do processo	<i>integer</i>		Não	Não
data_conclusao	Data de conclusão do processo	<i>Date</i>		Não	Não
exercício	Exercício do processo	<i>varchar</i>	200	Não	Não
deferido	Deferimento do processo	<i>bool</i>		Não	Não
descrição_fiscalizacao	Descrição da fiscalização vinda do aplicativo	<i>varchar</i>	max	Não	Não

Quadro 22 - Tabela de arquivos

<b>Arquivo</b>					
Armazena os dados dos arquivos enviados					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_Arquivo	Número sequencial do Arquivo	<i>integer</i>		Sim	Não

descricao	Descrição do Arquivo	<i>varchar</i>	1000	Não	Não
data_criacao	Data de criação do arquivo	<i>date</i>		Não	Não
arquivo	Arquivo	<i>blob</i>		Não	Não

Quadro 23 - Tabela de endereço de fiscalizações

<b>EnderecoFiscalizacaoCidadao</b>					
Armazena os dados enviados pelo aplicativo referente a localização da solicitação					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_EnderecoFiscalizacaoCidadao	Número sequencial do Endereço da Fiscalizacao do Cidadao	<i>integer</i>		Sim	Não
latitude	Latitude do Endereço da Fiscalização	<i>varchar</i>	500	Não	Não
longitude	Longitude do Endereço da Fiscalização	<i>varchar</i>	500	Não	Não
descricao_endereço	Descrição do endereço da fiscalização	<i>varchar</i>	4000	Não	Não

Quadro 24 - Tabela de distribuição de processos

<b>DistribuicaoProcesso</b>					
Armazena os dados da distribuição dos processos					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_distribuicaoProcesso	Número sequencial da Distribuição de Processos	<i>integer</i>		Sim	Não
processo_idProcesso	Campo de ligação com a tabela Processo	<i>integer</i>		Não	Sim
usuario_ERP_idUsuario_ERP	Campo de ligação com a tabela UsuarioERP	<i>integer</i>		Não	Sim
data_distribuicao	Data em que foi realiza a distribuição para o Setor Responsável	<i>date</i>		Não	Não

Quadro 25 - Tabela de usuário do ERP

<b>UsuarioERP</b>					
Armazena os dados dos usuários do ERP					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_usuario_ERP	Número sequencial do Usuário do ERP	<i>integer</i>		Sim	Não

login	Login do usuário	<i>varchar</i>	150	Não	Não
senha	Senha do Usuário	<i>varchar</i>	max	Não	Não
id_funcionario	Número do funcionário	<i>integer</i>		Não	Não
email	E-mail do funcionário	<i>varchar</i>	max	Não	Não
ativo	Campo se funcionário ainda possui cadastro ativo	<i>Bool</i>		Não	Não
setor_id_setor	Campo de ligação com a tabela Setor	<i>integer</i>		Não	Sim

Quadro 26 - Tabela de análises do processo

<b>AnaliseProcesso</b>					
Armazena os dados das análises de processos					
Campo	Descrição	Tipo	Tamanho	Chave primária	Chave estrangeira
id_analise_processo	Número sequencial da análise do processo	<i>integer</i>		Sim	Não
distribuicao_id_DistribuicaoProcesso	Campo de ligação com a tabela Distribuição Processo	<i>integer</i>		Não	Sim
parecer	Descrição do parecer na análise	<i>varchar</i>	max	Não	Não

data_conclusao	Data de conclusão da análise	<i>date</i>		Não	Não
descrição_analise	Descrição da análise	<i>varchar</i>	max	Não	Não
data_criacao	Data de criação da análise	<i>date</i>		Não	Não