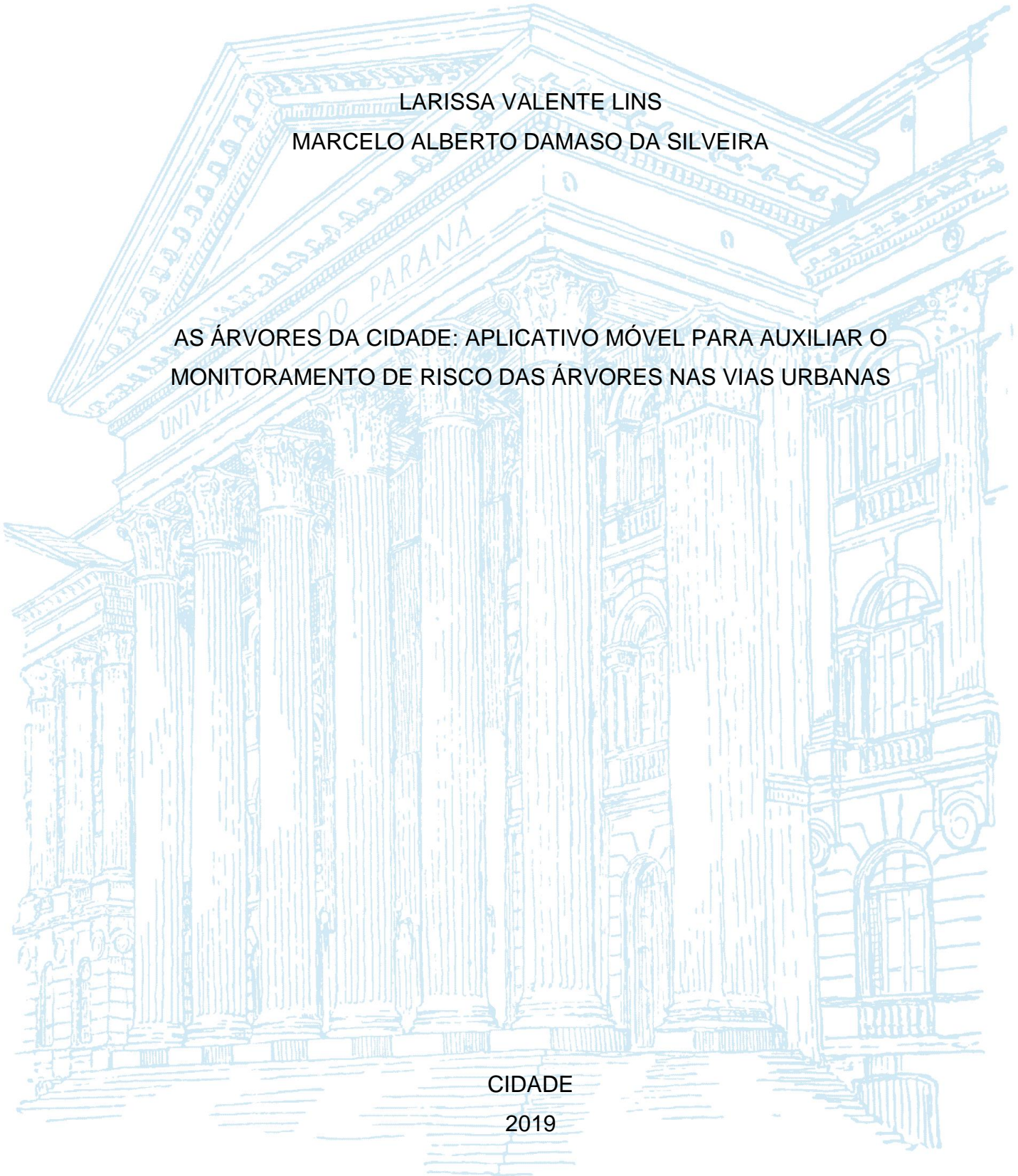


UNIVERSIDADE FEDERAL DO PARANÁ

LARISSA VALENTE LINS  
MARCELO ALBERTO DAMASO DA SILVEIRA

AS ÁRVORES DA CIDADE: APLICATIVO MÓVEL PARA AUXILIAR O  
MONITORAMENTO DE RISCO DAS ÁRVORES NAS VIAS URBANAS

CIDADE  
2019



LARISSA VALENTE LINS  
MARCELO ALBERTO DAMASO DA SILVEIRA

AS ÁRVORES DA CIDADE: APLICATIVO MÓVEL PARA AUXILIAR O MONITORAMENTO  
DE RISCO DAS ÁRVORES NAS VIAS URBANAS

Trabalho de Conclusão de Curso apresentada ao Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, Setor de Educação Profissional e Tecnológica, Universidade Federal do Paraná, como requisito parcial à obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientado: Prof. Dr. DIEVAL GUIZELINI

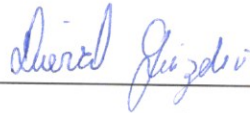
CIDADE  
2019

## TERMO DE APROVAÇÃO

Larissa Valente Lins  
Marcelo Alberto Damaso da Silveira

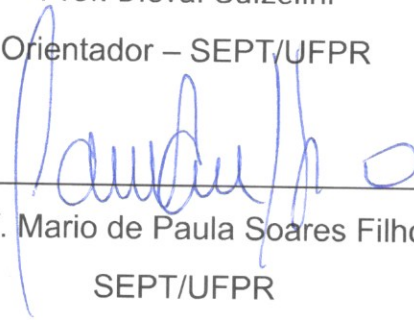
### **As Árvores das Cidades: Aplicativo Móvel para auxiliar o monitoramento de risco das árvores nas vias urbanas**

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



---

Prof. Dieval Guizelini  
Orientador – SEPT/UFPR



---

Prof. Mario de Paula Soares Filho  
SEPT/UFPR



---

Prof.º Roberto Tadeu Raittz  
SEPT/UFPR

Curitiba, 25 de Junho de 2019.

## RESUMO

As árvores no meio urbano são fundamentais para melhoria da qualidade do ar, auxílio na redução da variação de temperaturas, proteção contra incidência dos raios solares, ventanias e chuvas fortes.

Entretanto com o envelhecimento das árvores, com o plantio de espécies sem a supervisão ou planejamento urbano, com o crescimento das cidades e com a manutenção e manejo inadequados, as árvores podem causar prejuízos e riscos aos cidadãos, ao transporte e as estruturas urbanas. Como reduzir os danos e o risco?

Propomos o desenvolvimento de um aplicativo móvel, como parte de um sistema de monitoramento em tempo real, em que as pessoas possam utilizar seus celulares para enviar fotografias georeferenciadas das árvores que apresentem algum risco para um banco de imagens centralizado.

Este sistema será composto por diferentes componentes, no qual este trabalho destina-se a etapa do desenvolvimento do aplicativo para coleta dos dados. O aplicativo foi escrito em Java para Android, utilizando as APIs do Android para integração da câmera, do GPS, do Google Maps e da camada de persistência on-line e off-line. Os registros são mantidos no aplicativo até o envio para o servidor ou a sua exclusão pelo usuário. Políticas e termos de usos foram escritos para segurança jurídica da transferência de autoria e assegurar que as imagens atendem o objetivo proposto. A metodologia de desenvolvimento aplicada foi o Scrum.

O **PAC – Projeto Árvores da Cidade** é um aplicativo que permite o usuário fotografar e descrever árvores das vias urbanas, indicando a localização e informando sobre eventuais problemas observados. O aplicativo orienta a obtenção de um conjunto de fotografias que permitem a consolidação do banco de imagens segmentado.

As aplicações cliente e servidor estão prontos para iniciar os testes e possibilitar a construção da base de imagens necessárias para os trabalhos da próxima etapa do projeto, que visa o treinamento e classificação dos artefatos (fotografias), identificando grupos e análise de risco. Se o projeto for continuado e adotado pelas autoridades, a população ganhará um efetivo mecanismo de comunicação, que possibilitará uma melhor gestão da supervisão, manutenção e manejo das árvores urbanas.

Palavras-chave: Projeto Árvores da Cidades, Espécies Arbóreas, Árvores Urbanas, Georeferenciamento, fotografia, banco de imagens.

## **ABSTRACT**

This work of innovation arose in the face of a need observed with the diaval guiding professor in contributing to the department X and public ministry, through the insertion of the technology to a process that allows the counting, grouping and classification of the urban tree species present in the municipality. Providing a platform to collect information that is provide by users, thus composing a bank of records with geolocation of occurrences. A mobile application was developed on Android platform, called PA - Tree Project, which aims to unify the cataloging process, records and presentation in an easily accessible platform.

The methodology applied for the software development was Scrum, coming from the Agile Methodology. This process always aims to generate something of value to the Customer, always making "blocks" of deliveries. The PA - Project Tree is an application that records and presents all requests provided by users, shows the User this data, formatted in their records history. Through the data collected by the platform it is possible to make an analysis of the created requests and their respective types. This action results in data that may aid in decision making to act on nonconformities.

Keywords: Android, Tree Project, Arboreal Species, Agile Methodology, Scrum.

## LISTA DE FIGURAS

FIGURA 1 – FLUXO DE PROCESSO SCRUM.....	24
FIGURA 2 – ARQUITETURA ÁGIL .....	25
FIGURA 3 – ARQUITETURA MVC .....	26
FIGURA 4 – MODELO MVC APLICADO AO ANDROID.....	27
FIGURA 5 – KANBANFLOW - DIVISÃO DE ATIVIDADES.....	33
FIGURA 6 – DIAGRAMA DE CASO DE USO (GERAL) .....	37
FIGURA 7 – DIAGRAMA DE CASO DE USO (UC01) .....	38
FIGURA 8 – DIAGRAMA DE CASO DE USO (UC02) .....	40
FIGURA 9 – T01 – TELA DE LOGIN.....	41
FIGURA 10 – DIAGRAMA DE CASO DE USO (UC03) .....	44
FIGURA 11 – T02 – TELA DE TERMOS.....	45
FIGURA 12 – T03 – TELA DE MANTER SOLICITAÇÕES .....	45
FIGURA 13 – DIAGRAMA DE CASO DE USO (UC04) .....	48
FIGURA 14 – T04 – TELA DE INSTRUÇÃO DENUNCIA.....	49
FIGURA 15 – T05 – TELA DE PREVIA DENUNCIA.....	49
FIGURA 16 – T06 – TELA DE MAIS INFORMAÇÕES.....	50
FIGURA 17 – T07 – TELA DE SUCESSO.....	50
FIGURA 18 – T08 – TELA DE FALHA .....	51
FIGURA 19 – DIAGRAMA DE CASO DE USO (UC05) .....	54
FIGURA 20 – T09 – TELA DE INSTRUÇÃO REGISTRO .....	55
FIGURA 21 – T10 – TELA DE PREVIA REGISTRO.....	55
FIGURA 22 – T11 – TELA DE MAIS INFORMAÇÕES.....	56
FIGURA 23 – T12 – TELA DE SUCESSO.....	56
FIGURA 24 – T13 – TELA DE FALHA.....	57
FIGURA 25 – T13 – TELA DE FALHA.....	61
FIGURA 26 – T13 – TELA DE FALHA.....	62
FIGURA 27– T13 – TELA DE FALHA.....	62

## **LISTA DE TABELAS**

TABELA 1 – PAPÉIS DOS INTEGRANTES DENTRO DO SCRUM.....	34
--	----

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	<b>16</b>
1.1 JUSTIFICATIVA .....	18
1.2 OBJETIVOS .....	19
1.2.1 Objetivo geral .....	19
1.2.2 Objetivos específicos.....	19
1.3 METODOLOGIA.....	20
1.3.1 Scrum .....	22
1.4 LINGUAGEM DE MODELAGEM UNIFICADA (UML).....	25
1.5 DEFINIÇÃO DE CASO DE USO .....	26
1.6 PADRÃO DE DESENVOLVIMENTO .....	26
<b>1.7 TESTES DE SOFTWARE</b> .....	<b>28</b>
<b>1.8 TECNOLOGIAS DE DESENVOLVIMENTO</b> .....	<b>28</b>
1.8.1 Linguagem Java .....	28
1.8.2 Android Studio .....	29
<b>1.9 OUTRAS FERRAMENTAS</b> .....	<b>29</b>
1.9.1 Microsoft Office .....	29
1.9.2 KanbanFlow .....	29
1.9.3 Bizagi.....	30
1.9.4 GitHub .....	30
1.9.5 QuantUX.....	30
<b>2 REVISÃO DE LITERATURA</b> .....	<b>ERRO! INDICADOR NÃO DEFINIDO.</b>
<b>3 MATERIAL E MÉTODOS</b> .....	<b>31</b>
3.1 GERENCIAMENTO DE PROJETOS.....	32
3.2 REQUISITOS FUNCIONAIS .....	34
3.3 REQUISITOS NÃO FUNCIONAIS .....	34
3.4 DIAGRAMA DE CASO DE USO .....	36
3.5 ESPECIFICAÇÃO DE CASO DE USO.....	37
3.5.1 UC01 - Verificar Autenticação. ....	37
3.5.2 UC02 - Login. ....	39
3.5.3 UC03 - Manter Solicitações.....	43
3.5.4 UC04 - Criar Denuncia. ....	47
3.5.5 UC05 - Criar Registro 2.0. ....	53



3.6 DIAGRAMA DE CLASSE .....	60
3.7 DIAGRAMA DE SEQUENCIA .....	61
<b>4 CONCLUSÃO .....</b>	<b>62</b>
<b>5 CONSIDERAÇÕES FINAIS .....</b>	<b>63</b>
5.1 RECOMENDAÇÕES PARA TRABALHOS FUTUROS .....	63
<b>REFERÊNCIAS.....</b>	<b>64</b>
<b>ANEXO 1 – TERMO DE AUTORIZAÇÃO DE USO DE IMAGENS E RESPONSABILIDADE FOTOGRÁFIC.....</b>	<b>67</b>

## 1 INTRODUÇÃO

Este trabalho de conclusão de curso surgiu frente a uma necessidade observada junto ao professor orientador Dieval Guizelini, em parceria com a Fundação de Pesquisas Florestais – FUPEF, através da inserção da tecnologia a um processo que possibilite a contagem, agrupamento e classificação das espécies arbóreas urbanas presentes no município. Disponibilizando uma plataforma para coletar informações que serão fornecidas pelos cidadãos, estes dados passam a compor um banco de registros com geolocalização das árvores fotografadas. Desenvolveu-se uma aplicação mobile em plataforma Android, denominada PA – Projeto Árvore, que visa criar um mecanismo de coleta de informações de árvores presentes nas ruas das cidades, para consolidação de banco de imagens geolocalizadas, constituindo um catálogo de imagens e um sistema de fácil acesso.

O objetivo deste aplicativo é auxiliar os profissionais, órgãos fiscalizadores e prefeituras na atividade de monitoramento das condições das árvores, especialmente nas vias urbanas. Atualmente, para avaliar estes possíveis riscos há apenas uma ouvidoria, que recebe registros. Quando a reincidência de um problema em uma determinada localização é maior, uma equipe é disponibilizada para análise direta do risco. Sendo assim, quando ocorre baixa taxa de denúncias/reclamações, o problema não é repassado para as equipes e conseqüentemente, o risco persiste.

Como envolvem custo e os recursos são escassos, foi proposta a criação de um aplicativo mobile, que facilite a obtenção de mais informações, com maior precisão na localização das árvores, com um conjunto de imagens, técnicos podem avaliar os riscos e priorizar o envio das equipes especializadas.

Com os dados coletados pelo aplicativo será possível que a equipe de avaliadores verifique os registros por suas imagens no banco de dados da aplicação. Diminuindo assim a necessidade do avaliador estar presencialmente na localização para a verificar a queixa. Desta forma, a demanda de tempo para a conclusão de uma avaliação e elaboração de um plano de ação pelo órgão responsável será menos trabalhosa e mais rápida. Como é um processo bastante demorado, foi proposta a criação de um aplicativo mobile que facilite a manipulação de relatórios de abordagens realizadas através de uma interface de fácil visualização e manuseio, tornando a avaliação mais prática.

O aplicativo para dispositivo móvel visa auxiliar os usuários. Através de uma interface simples, facilitando o registro de denúncias, fornecendo informações importantes e necessárias para as equipes de avaliação.

Para viabilizar o desenvolvimento da aplicação para dispositivos móveis, foi utilizada a IDE (Ambiente de Desenvolvimento Integrado) Android Studio (versão 2.1), uma plataforma open source criada pelo Google, que dispõe de um ambiente de desenvolvimento, depuração e testes.

A metodologia Ágil (Scrum) foi adotada, para o acompanhamento e desenvolvimento do sistema, permitindo que os requisitos fossem sendo definidos e refinados ao longo do processo de desenvolvimento.

O aplicativo desenvolvido nesse trabalho é um protótipo, importante de um processo em construção, que precisa amadurecer e ser adotado pela comunidade, pelos órgãos governamentais e pelas empresas, no intuito de sistematizar uma rede de colaboração, onde o cidadão fornece imagens e dados das árvores das cidade para um sistema que organiza, cataloga e classifica essas imagens, agrupando e ranqueando os indivíduos com maior risco, possibilitando a priorização de intervenção das equipes técnicas da Prefeitura. Nesse momento, o projeto é uma ideia, que foi submetido a Editais de financiamento público, buscando parcerias para integrar os quatro instituições, cada uma, com interesse específico sobre o tema, são eles: Prefeitura, responsável pela manutenção de parte das árvores em nossas ruas, da COPEL, responsável das árvores que estão no mesmo lado da rua em que passa a fiação elétrica, Ministério Público que tem a responsabilidade de fiscalizar e da Fundação de Pesquisas Florestais da UFPR (FUPEF) que desenvolve pesquisas na área e que procurou o Prof. Dr. Dieval Guizelini, especialista em bioinformática, para desenvolver propostas para o problema. Ao procurarmos o professor, em busca de um tema para o presente TCC, o orientador apresentou a ideia, o projeto e fez a proposta de desenvolvermos o primeiro protótipo, com a finalidade de subsidiar a constituição de um banco de fotografias georeferenciadas, etapa essa essencial, para iniciar as pesquisas em reconhecimento de imagens, agrupamento e identificação das espécies, para em uma nova etapa do projeto de pesquisa, o sistema poder prever e ranquear os riscos.

## 1.1 JUSTIFICATIVA

A necessidade de um processo de monitoramento das espécies urbanas, que potencialize a identificação de riscos e a prevenção de acidentes. Somente em Curitiba são mais de 300.000 árvores nas ruas e 1/3 delas apresentam potencial risco (RPC, 2019). Em maio de 2019, fortes chuvas provocaram a queda de galhos e árvores, derrubando fiação elétrica, interditando ruas, danificando carros e casas. Quantos desses prejuízos poderiam ser evitados com a manutenção adequada?

Com o aplicativo proposto nesse trabalho, os cidadãos podem ser os olhos da Prefeitura, no que se refere a ação de observar todas as árvores presentes na cidade e com baixo custo, identificar através de fotografias, as árvores que demandam manutenção ou manejo, de forma preventiva, evitando acidentes.

## 1.2 OBJETIVOS

### 1.2.1 Objetivo geral

O objetivo geral deste trabalho é o desenvolvimento de um aplicativo móvel para a sistematização do processo de registro realizado pelos cidadãos, fornecendo um aplicativo mobile que permita aos usuários catalogar árvores ou denunciar possíveis riscos nas árvores presentes nas vias urbanas.

### 1.2.2 Objetivos específicos

Levando em consideração o objetivo geral, este trabalho propõe como objetivos específicos:

- Orientar o registro fotográfico, indicando à obtenção da imagem do perfil, das folhas, das sementes, das flores e dos artefatos que indiquem risco a saúde da árvore;
- Criar um registro do conjunto de imagens, da geolocalização e das informações prestadas pelo usuário - possibilitando o registro ou denuncia de problemas com as espécies ou outras inadequações levantadas nas árvores presentes das vias urbanas.
- Preparar uma estrutura para formar um banco de imagens com base nos registros. Permitindo assim que uma equipe avalie as espécies e riscos por meio dos registros fotográficos anexados nas queixas.
- Permitir comentários e registros de informações complementares; Visto que o processo de Ouvidoria por contato telefônico traz informações superficiais que não auxiliam na localização e diagnóstico específico.
- Permitir o armazenamento do histórico dos registros. Possibilitar o arquivamento dos registros realizados pelo usuário junto com suas informações e registros fotográficos.
- Permitir as operações em modo desconectado (off-line) e sincronização com o servidor, quando a conexão estiver disponível.
- Avaliar a anotação e o padrão das informações de geolocalização e o posicionamento dos objetos no google maps.

### 1.3 METODOLOGIA

A Metodologia Ágil é um conjunto de práticas que visa melhores maneiras de desenvolvimento de *software*. Esta metodologia é amplamente utilizada a fim de melhorar a produtividade e agregar valor ao cliente, com as constantes entregas do produto. Observa-se alguns itens relevantes das metodologias ágeis, tais como: o foco na comunicação contínua com o cliente, as entregas constantes e ainda, uma equipe de desenvolvimento auto gerenciável.

O Ágil proporciona que o cliente tenha uma oportunidade de estar envolvido durante todo o projeto, o que resulta de uma alta colaboração entre o cliente e o time de desenvolvimento do projeto, fornecendo mais oportunidades para os integrantes entenderem realmente a visão do cliente. Mostrando o desenvolvimento do *software* a cada ciclo de trabalho (chamado de *Sprint*), faz com que o cliente aumente a confiança nas habilidades da equipe, valorizando assim os seguintes itens, segundo Kent Beck e cols. destacam no Manifesto Ágil (2001):

1. **Indivíduos e interação entre eles** mais que processos e ferramentas
2. **Software em funcionamento** mais que documentação abrangente
3. **Colaboração com o cliente** mais que negociação de contratos
4. **Responder a mudanças** mais que seguir um plano.

Portanto, os autores ressaltam que mesmo existindo valor em processos e ferramentas, documentação, negociação de contratos e avançar de acordo com um plano, os itens mais valorizados ainda serão o *software* funcionando, colaboração com Cliente, acompanhamento das mudanças e Indivíduos e sua interação.

Kent Beck e cols. (2001) ainda evidenciam os princípios por eles seguidos, que estão por trás do Manifesto Ágil.

- Nossa maior prioridade é satisfazer o cliente, através da entrega adiantada e contínua de *software* de valor. Aceitar mudanças de requisitos, mesmo no fim do desenvolvimento. Processos ágeis se adequam a mudanças, para que o cliente possa tirar vantagens competitivas."
- Entregar *software* funcionando com frequência, na escala de semanas até meses, com preferência aos períodos mais curtos.

- Pessoas relacionadas à negócios e desenvolvedores devem trabalhar em conjunto e diariamente, durante todo o curso do projeto.
- Construir projetos ao redor de indivíduos motivados. Dando a eles o ambiente e suporte necessário, e confiar que farão seu trabalho.
- O método mais eficiente e eficaz de transmitir informações para, e por dentro de um time de desenvolvimento, é através de uma conversa cara a cara.
- *Software* funcional é a medida primária de progresso.
- Processos ágeis promovem um ambiente sustentável. Os patrocinadores, desenvolvedores e usuários, devem ser capazes de manter indefinidamente, passos constantes.
- Contínua atenção à excelência técnica e bom design, aumenta a agilidade.
- Simplicidade: a arte de maximizar a quantidade de trabalho que não precisou ser feito.
- As melhores arquiteturas, requisitos e designs emergem de times auto organizáveis.
- Em intervalos regulares, o time reflete em como ficar mais efetivo, então, se ajustam e otimizam seu comportamento de acordo.

Em sua origem, a Metodologia Ágil tinha por objetivo superar as insuficiências reais observadas dentro da engenharia de software convencional. Em discordância, Pressman (2011) comenta que é possível obter resultados benéficos com o desenvolvimento ágil, entretanto ele não se aplica a todos os projetos, produtos, pessoas e situações. Menciona ainda, que o desenvolvimento ágil não está em desacordo com o exercício de Engenharia de *Software* tradicional, e que a prática da mesma, atua como uma filosofia prevalecte a todo o trabalho de *software*.

No que tange as características do processo ágil de *software*, apresentam-se três princípios básicos, segundo Pressman (2011):

1. Impossibilidade da antecipação dos requisitos, se estes serão alterados posteriormente, bem como as prioridades do cliente.
2. Impraticabilidade em mensurar a demanda de trabalho do projeto, sem ter uma implementação.
3. Dificuldade em prever o quanto de análise, projeto, desenvolvimento e testes serão necessários, até a conclusão do projeto.

Conforme já enfatizado, um processo ágil será adaptado e incrementado constantemente. Para o sucesso neste processo, a colaboração do Cliente através dos *feedbacks* em protótipos ou partes operacionais do sistema é fundamental.

Na modelagem ágil cita-se como primordial a construção de uma equipe de trabalho competente, onde devem-se convocar as pessoas certas. Segundo Cockburn e Highsmith (2001) “O desenvolvimento ágil enfoca os talentos e as competências dos indivíduos, moldando o processo a pessoas e equipes específicas”.

Pressman (2011) afirma que o processo de desenvolvimento de *software* é ajustado às necessidades dos indivíduos e da equipe de desenvolvimento, e não o contrário. De acordo com Ambler (2004), para que uma pessoa seja produtiva em um determinado cargo, ele deve ser razoavelmente apropriado às suas características, mas, mesmo que ela não possua todas as competências necessárias para assumi-lo, deverá ter condições de adquiri-las.

Ambos os autores concordam que determinadas características são necessárias para que hajam modeladores eficientes na equipe de desenvolvimento:

- **Competência** – habilidades específicas em software, talento inato e conhecimento do processo. Conhecimento este que poderá ser ensinado aos demais integrantes da equipe.
- **Disciplina e Organização** – Não agir inapropriadamente achando que sabe mais que o cliente, ou ainda que acrescentar coisas não fará mal.
- **Colaboração e Comunicação** – Conseguem escrever quando necessário e possuem boa capacidade de comunicação. Esta comunicação leva a colaboração dentro da equipe para o entendimento mutuo e do Cliente.
- **Coragem** – São capazes de propor uma ideia e executá-la, mas caso ela não seja apropriada a descartam, e iniciam outra, se necessário do zero.

### 1.3.1 Scrum

O *Scrum* é um modelo de processo ágil elaborado por Jeff Sutherland em 1993 em colaboração com Ken Schwaber. É um *framework* para desenvolvimento de novos produtos, geralmente usado para o desenvolvimento de *softwares*. Os



elementos que compõe esse modelo são: *roles*, artefatos e reuniões e serão detalhados ao longo deste tópico.

Os princípios do *Scrum* (1993) acompanham o Manifesto Ágil (2001):

- **Pequenas Equipes:** A fim de aumentar a comunicação e compartilhamento de ideias e conhecimentos;
- **Processo Adaptado:** Tanto em negócio quanto em parte técnica, com o intuito de produzir o melhor produto.
- **Incrementos ao Software:** Que poderão ser documentados, melhorados, adaptados de acordo com a necessidade do Cliente.
- **Distribuição de Tarefas e Equipe:** O trabalho e a equipe é dividida de forma clara, em partições ou pacotes.
- **Testes e Documentação:** São realizados continuamente à medida que o produto é desenvolvido.
- **Declaração de Finalização do Produto:** No Scrum isso pode ser realizado de acordo com os interesses das partes, seja por entregas da concorrência ou por cronograma de entrega, ou ainda, porque o usuário necessita das funções do produto.

Segundo Pressman (2011) os conceitos *Scrum* norteiam as atividades de desenvolvimento de acordo com os itens: requisitos, análise, projeto, evolução e entrega. O autor comenta ainda que o *Scrum* salienta a prática de um padrão de processos de software, os quais, têm se mostrado satisfatório aos projetos com prazos extremamente curtos, requisitos em constante edição e criticidade do negócio.

O *Scrum* não define uma técnica específica para o desenvolvimento de *software* durante a etapa de implementação, ele se concentra em descrever como os membros da equipe devem trabalhar para produzir um sistema flexível, num ambiente de mudanças constantes. (FADEL; SILVEIRA, 2010, p. 13).

É constatado na literatura que alguns autores concordam que com um ambiente de instabilidade e criticidade de determinados projetos, o *Scrum* é uma alternativa flexível às equipes de desenvolvimento.

Na figura 1 é possível ter uma visão mais ampla sobre o fluxo de processo *Scrum*.

FIGURA 1 – FLUXO DE PROCESSO SCRUM



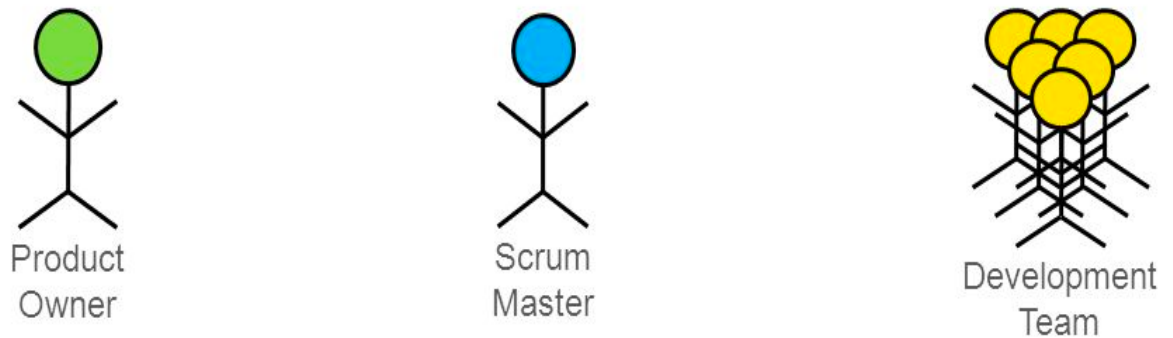
Fonte: FONTE:<http://wiki.uzed.com.br/index.php?title=Scrum>

Sobre os artefatos que existem dentro do *Scrum*, como comenta Cruz (2013):

- *Product Backlog* – Funciona como uma lista de requisitos, funcionalidades, tecnologias melhorias e funções do produto que será entregue.
- *Sprint Backlog*- São as tarefas que serão transferidas do *Product Backlog* para serem construídas pela equipe de desenvolvimento durante aquele próximo ciclo. Este ciclo pode ter duração de 2 à 4 semanas, ou mesmo 30 dias.
- *Daily Scrum* – É uma breve reunião feita pela equipe a cada dia de *Sprint*. Tem por objetivo, levar ao conhecimento da equipe o que foi realizado, levantar impedimentos e priorizar atividades.
- *Sprint Review Meeting* – Reunião realizada ao final de uma *Sprint*, para a apresentação dos resultados alcançados.
- *Sprint Retrospective* – Verifica-se itens já construídos e se planeja novos itens que entrarão na próxima *Sprint*.

No *Scrum* existem três papéis: *Scrum Master*, *Product Owner* e *Scrum Development Team*, como demonstra a figura 2.

FIGURA 2 – ARQUITETURA ÁGIL



FONTE: Benjamin Day Brookline, <http://slideplayer.com/slide/6604918/> - Presentation on theme: "Coaching Skills for Scrum Masters & The Self- Organizing Team." - Coaching Skills for Scrum Masters & The Self- Organizing Team

Abaixo será abordada a função de cada um destes papéis, conforme ÁGIL (2015):

- *Product Owner (PO)* – Dono do Produto. Responsável por gerenciar o *backlog* do produto, tornando-o acessível a todos. Atribui valor ao trabalho do time.
- *Scrum Master* – Responsável por remover quaisquer impedimentos que possam comprometer o *Scrum Development Team*, auxilia na aderência da equipe ao *Scrum* e no autogerenciamento da equipe.
- *Scrum Development Team* – Auto gerenciáveis e responsáveis por compor o produto, isto é, transformar o *backlog* em um bloco de sistema entregue ao cliente.

#### 1.4 LINGUAGEM DE MODELAGEM UNIFICADA (UML)

A UML (*Unified Modeling Language*), linguagem unificada de modelagem, é uma linguagem gráfica que permite a visualização, especificação, construção e documentação de artefatos de sistemas de *software*. Ela padroniza a preparação de estratégias de arquitetura de projetos de sistemas, no que se refere a parte conceitual, como processos de negócios e funções do sistema, além de itens como as classes escritas em determinada linguagem de programação, esquemas de bancos de dados e componentes de *software* reutilizáveis. (BOOCH e cols. 2006).

## 1.5 DEFINIÇÃO DE CASO DE USO

Conceitua-se por Caso de Uso a sequência de descrições e ações realizadas pelo sistema, que proporcionam resultados de agregado valor para um determinado ator. É utilizado para estruturar o comportamento de itens em um modelo. BOOCH e cols. (2006).

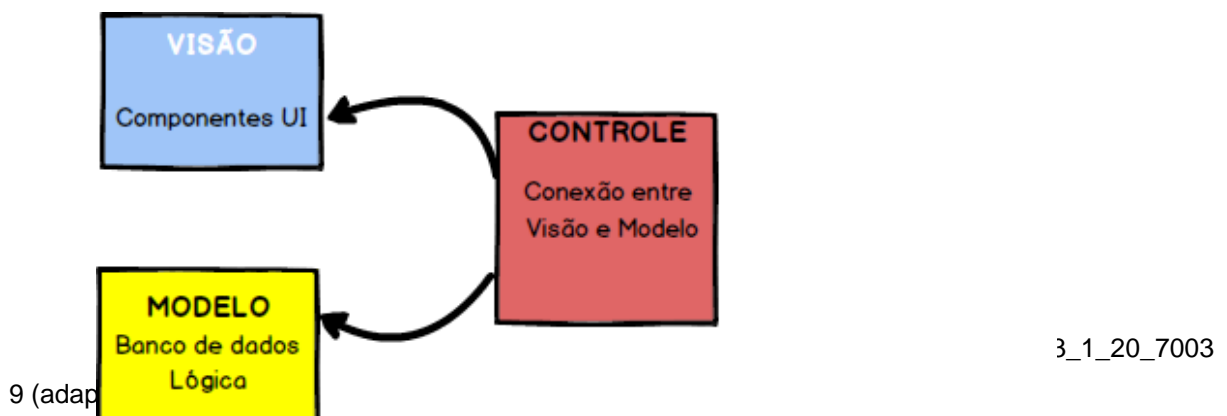
## 1.6 PADRÃO DE DESENVOLVIMENTO

Segundo Sommerville (2011), com a utilização do padrão de desenvolvimento é possível apresentar, compartilhar e reusar o conhecimento sobre sistemas de *software*. Um padrão de desenvolvimento deve descrever uma organização de sistema bem-sucedido em sistemas e ambientes anteriores.

O padrão de projeto comumente utilizado para aplicações *web* é o chamado de Padrão MVC (*model-view-controller*), ou na tradução para o português, modelo-visão-controle.

Baptistella (2016) afirma que o “MVC” ajuda na tarefa de separar as responsabilidades promovendo um baixo acoplamento e alta coesão, tornando o sistema escalável.”

FIGURA 3 – ARQUITETURA MVC



Seguindo este modelo de arquitetura, foi desenvolvido um aplicativo *mobile* preocupando-se com a separação do sistema em camadas. É possível encontrar na literatura várias demonstrações estruturais desse padrão e na figura 3 encontra-se

uma adaptação do modelo MVC. É possível correlacionar o comportamento em camadas (Baptistella, 2016) com a descrição abaixo:

**Camada *View*:** Responsável por receber informações das interações com o usuário, apresentando dados de resposta fornecidos pela Camada *Controller*.

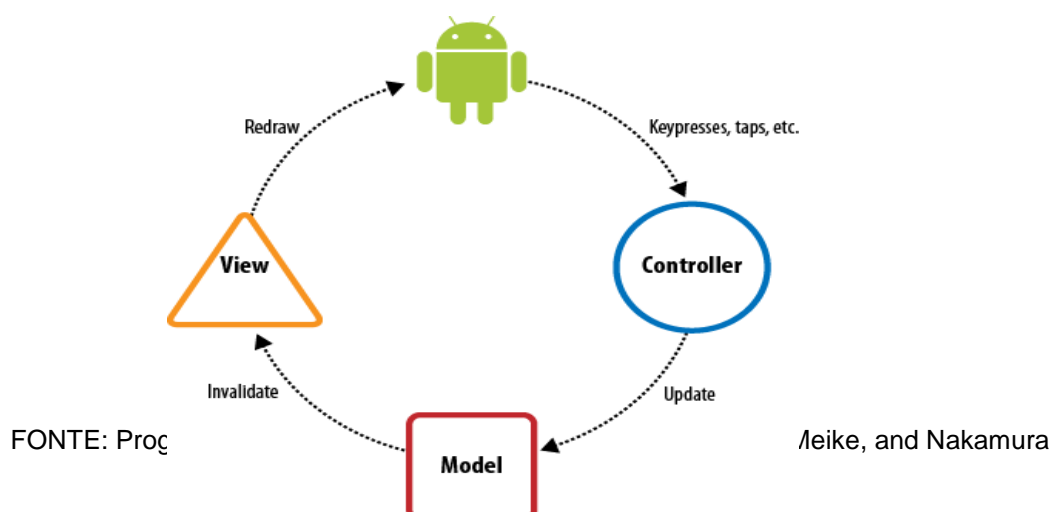
**Camada *Controller*:** Tem por responsabilidade mapear ações do usuário, e enviar o resultado para a Camada *Model* processar. Como retorno deste processamento ela seleciona qual *View* exibirá a resposta da interação.

**Camada *Model*:** Se responsabiliza por ações do sistema, lógica de negócio e comunicação com o banco de dados. O núcleo de toda ação, manipulação e persistência de dados estão nessa camada.

O MVC tem como principal objetivo: separar dados ou lógicos de negócios (*Model*) da interface do usuário (*View*) e o fluxo da aplicação (*Controller*), a idéia é permitir que uma mensagem da lógica de negócios possa ser acessada e visualizada através de várias *interfaces*. Na arquitetura MVC, a lógica de negócios, ou seja, nosso *Model* não sabe quantas nem quais as *interfaces* com o usuário está exibindo seu estado, a *view* não se importa de onde está recebendo os dados, mas ela tem que garantir que sua aparência reflita o estado do modelo, ou seja, sempre que os estados do modelo mudam, o modelo notifica as *view* para que as mesmas atualizem-se. (Baptistella, 2016).

Esta definição do modelo de arquitetura resume-se na figura 4, que pode ser visualizada na página a seguir.

FIGURA 4 – MODELO MVC APLICADO AO ANDROID



O principal benefício dessa arquitetura é a separação de interesses, onde cada camada do MVC é responsável pelo seu próprio trabalho: a *view* cuida da interface com o usuário, a *model* se encarrega dos dados, e a *controller* envia mensagens entre os dois.

## 1.7 TESTES DE SOFTWARE

Os testes de Software têm por objetivo descobrir falhas ou defeitos no *software*, comportamento indesejado ou, não conformidades com a especificação, que pode ser através de testes individuais, ou muitos outros, porém, que devem ser corrigidos antes de ser realizada entrega ao Cliente. Os testes podem variar de acordo com a seguinte classificação:

- Testes de Unidade;
- Testes de Sistema;
- Testes de Release;
- Testes de Validação;
- Testes de Defeitos;
- Testes Exaustivos;
- Testes de *Release*

Tem como finalidade e alvo os clientes, que por sua vez irão aumentar a confiança para com o cliente e verificar se atende ou não os requisitos solicitados pelo mesmo. Nesta etapa o sistema não deve apresentar falhas e precisa mostrar toda a sua funcionalidade, desempenho e a confiabilidade do sistema.

Estes testes são denominados ainda de “testes de caixa preta”, pois seu comportamento é determinado por meio dos estudos de suas entradas e saídas. O foco deste teste está na funcionalidade, e não na implementação do software como os testes de sistema, testes de integração e testes de unidade. DALAMARO, MALDONADO, JINO (2007).

## 1.8 TECNOLOGIAS DE DESENVOLVIMENTO

### 1.8.1 Linguagem Java

Trata-se de uma linguagem de programação expressamente designada para ser usada em ambientes distribuídos da Internet. É uma linguagem de programação orientada a objetos e é multi plataforma, isto é, pode ser utilizada em

diferentes plataformas sem a necessidade de adaptações no código. Java pode ser usado para criar aplicações que possam rodar em um único computador ou ser distribuída através de servers e clientes em uma rede. FURGERI, Sérgio (2008).

### 1.8.2 Android Studio

É a IDE (*Integrated Development Environment*) oficial do *Google*, gratuita para o uso e *download*, serve para o desenvolvimento de aplicativos mobile, desktop e outras tecnologias emergentes para plataforma *Android OS*. A base é IntelliJ IDEA, uma IDE para Java que ajuda a auto completar e gerar código rapidamente usando a funcionalidade código “inteligente”. Tem um rico ambiente de desenvolvimento UI (*User Interface*). O *workflow* do *Android Studio* é construído através do conceito de integração contínua, o que permite que as equipes de desenvolvimento testem o código cada vez que um desenvolvedor atualizar o mesmo. LECHETA, (2010).

## 1.9 OUTRAS FERRAMENTAS

### 1.9.1 Microsoft Office

É um conjunto de programas que compõem um pacote pago da Empresa *Microsoft* utilizados mundialmente para execução de tarefas do cotidiano como, elaboração de textos, planilhas eletrônicas, apresentações de slides entre outros.

### 1.9.2 KanbanFlow

O *KanbanFlow* é uma ferramenta gratuita que auxilia na gestão das tarefas de uma equipe. É um quadro visual onde existem 4 níveis em que as tarefas podem ser divididas:

- *To-do*: Nesta parte são listadas as tarefas a serem desenvolvidas pela equipe.
- *Do Today*: Nesta fase estão as atividades que precisam ser concluídas no dia atual.
- *In Progress*: Aqui as tarefas ainda estão em andamento.
- *Done*: Tarefa concluída.

Cada pessoa é assinalada para uma tarefa e sua responsabilidade é concluir e atualizá-las de acordo com status de progresso.

### 1.9.3 Bizagi

A Ferramenta *Bizagi Modeler* é uma ferramenta de *download* gratuito. É utilizada para modelagem gráfica BPMN (*Business Process Management Notation*) de processos e documentação com facilidade e praticidade. É uma ferramenta utilizada para elaboração de diversos tipos de diagramas, tais como: diagramas de caso de uso, diagramas de classe, diagramas de sequência,

### 1.9.4 GitHub

É uma ferramenta utilizada para um serviço de *Web Hosting*, que por sua vez é compartilhado para projetos que adotam o uso de versionamento. Por meio dele pode-se trabalhar sempre no mesmo diretório, podendo gravar a documentação, realizar alterações no projeto e fazer comentários. A vantagem de se utilizar essa ferramenta é a possibilidade de poder voltar a uma fase anterior caso possa surgir algum problema pela frente. O foco do GitHub é ter um histórico de versionamento caso haja alguma alteração que não deu certo, podendo ser alterado para uma versão anterior, não precisando voltar desde o início do projeto para encontrar e corrigir o problema.

### 1.9.5 QuantUX

É uma ferramenta gratuita *online* para realizar a criação de protótipos. Auxilia no *design* do *software* antes de começar a desenvolver para evitar retrabalho da equipe de desenvolvimento.



## 2 MATERIAL E MÉTODOS

Para que o desenvolvimento do projeto fosse eficaz e atendesse as necessidades do Cliente com qualidade, foram realizadas reuniões mensais entre a equipe de desenvolvimento e a cliente. Estas reuniões tinham por objetivo validar o resultado das respectivas *Sprints* junto ao Cliente e absorver possíveis alterações nos requisitos, e apresentar partes do aplicativo já em funcionamento, conforme a metodologia Ágil *Scrum* preconiza.

No que se refere a escolha do Cliente, este projeto foi categorizado como projeto de prospecção, o que significa desenvolver a aplicação concomitantemente com a definição dos requisitos, possibilitando avaliar as opções tecnológicas para os diversos problemas propostos, entre outros:

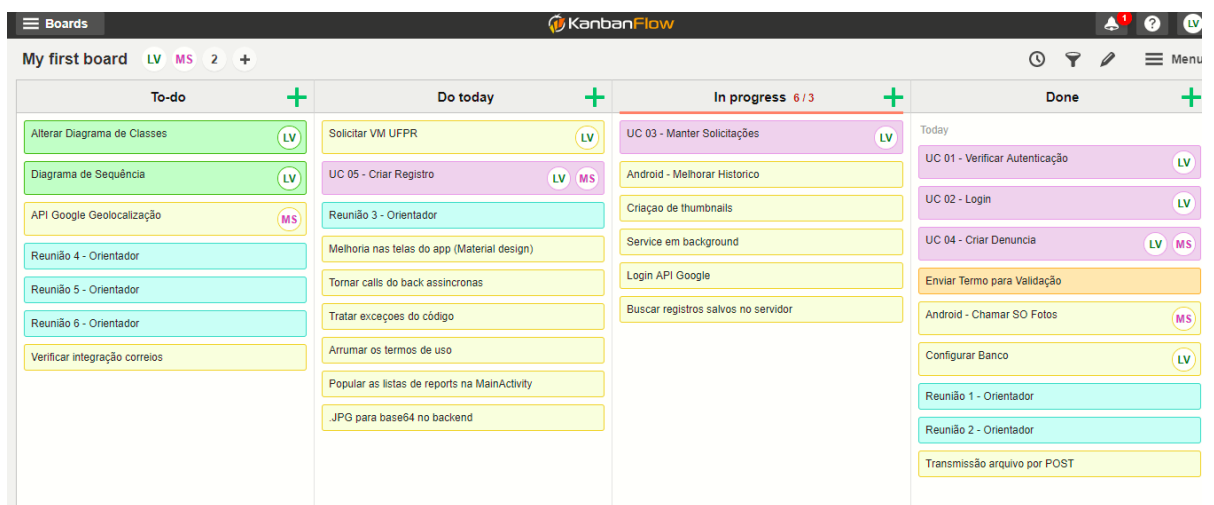
- 1) identificar a localização do árvore (objeto de interesse);
- 2) Obter indicativos para relacionar a precisão/imprecisão dos dados do GPS em relação a distância necessária para enquadrar a árvore, considerando a variação de altura e largura;
- 3) Ter qualidade suficiente para agrupar/classificar as espécies;
- 4) Identificar potenciais riscos a saúde da árvore, tais como raízes expostas, presença de espécies invasoras, infecção de insetos (cupim, formigas) etc;
- 5) Identificação e relacionamento da variação da aparência das árvores nas diferentes estações do ano;
- 6) Identificação da mudança da estrutura da árvore, em decorrência de podas ou outras interferências no desenvolvimento natural;
- 7) Permitir diferentes níveis de detalhamento, onde o registro pode ser de apenas uma foto a uma coleção orientada de fotos, com campos descritivos para informações observadas pelo usuário.

Neste contexto, foram projetados fluxos básicos de telas, feitas descrições iniciais e o desenvolvimento dos primeiros protótipos, onde a aplicação gerencia a qualidade da imagem, a geolocalização (GPS). Nas iterações seguintes, foram incluídas o sistema de identificação, o sistema de operação off-line, o envio para aplicação servidora e o desenvolvimento de aplicação para serviço web, para receber e constituir o banco de registros de fotografias.

## 2.1 GERENCIAMENTO DE PROJETOS

Para este projeto o grupo adotou a Metodologia Ágil com foco no Scrum, e de acordo com esta orientação, foram utilizados alguns artefatos, tais como: *Backlog*, *Sprints*, *Sprint Planning Meeting*, *Sprint Retrospective* por meio de uma equipe auto gerenciável. No projeto foi direcionado o papel de *Scrum Master* a um dos integrantes da equipe, a fim de remover quaisquer impedimentos que viessem a atrapalhar o andamento do projeto. O grupo realizou a divisão do *Backlog* em três *Sprints*. Estas Sprints e suas respectivas atividades eram acompanhadas através do KanBan, que foi hospedado na ferramenta *online KanBanFlow*. Após a realização de cada *Sprint Planning Meeting*, a equipe realizava uma *Sprint Retrospective*, a fim de verificar pontos de melhoria e alterações junto aos requisitos já existentes. A imagem a seguir demonstra o uso da ferramenta para o acompanhamento do projeto.

FIGURA 5 – KANBANFLOW - DIVISÃO DE ATIVIDADES



FONTE: Os autores (2019)

Aos domingos, excepcionalmente foram realizadas reuniões, para alinhar implementação do código, banco de dados e detalhes que seriam implementados ao projeto a partir destas decisões. Abaixo demonstra-se por meio de uma tabela a distribuição de papéis dos integrantes da equipe:

**TABELA 1 – PAPÉIS DOS INTEGRANTES DENTRO DO SCRUM**

<b>Papel</b>	<b>Responsável</b>
<b><i>Product Owner</i></b>	Professor Dieval Guizelini
<b><i>Scrum Master</i></b>	Larissa Valente Lins
<b><i>Scrum Team</i></b>	Larissa Valente Lins, Marcelo Alberto Damaso da Silveira

FONTE: Os autores (2019)

## 2.2 REQUISITOS FUNCIONAIS

Os requisitos funcionais definem as funcionalidades do sistema, esses requisitos foram classificados de acordo com a ordem de desenvolvimento, seguindo a legenda abaixo:

- RF1 - O sistema deve fornecer uma forma simples de login aos usuários.
- RF2 - O sistema deve fornecer a possibilidade de Logout e Alteração para outras contas.
- RF3 - O sistema deverá permitir aos usuários de uma cidade criarem dois tipos de chamado "Catálogo" e "Denúncias" a respeito das árvores presente nas vias públicas.
- RF4 - O sistema deve de forma intuitiva passar para o usuários os passos necessários para o envio dos registros.
- RF5 - O sistema deverá solicitar ao usuário o direito autoral das imagens fotografadas e enviadas em ambos os tipos de chamado. Assim como um termo de responsabilidade pelas imagens enviadas.
- RF6 - O sistema deve possibilitar a captura de uma imagem ou a inclusão de uma imagem já presente na galeria para os registros do usuário.
- RF7 – O sistema deverá conter um tutorial de ajuda, explicando como operar as funcionalidades do aplicativo.
- RF8 – O sistema deve permitir que o usuário envie informações adicionais, além das informações básicas obrigatórias já incluídas.
- RF9 – O sistema deverá deve manter um histórico dos chamados enviados e possibilitar a visualização do usuário aos itens já registrados.
- RF10 – Caso o usuário não esteja conectado a internet o sistema deverá armazenar os registros criados, e envia-los quando a conexão for estabelecida.
- RF11 O sistema deverá enviar a geolocalização do dispositivo móvel no momento do registro para sinalizar a localização da ocorrência.

## 2.3 REQUISITOS NÃO FUNCIONAIS

Este tópico demonstra os requisitos não funcionais do projeto, em outras palavras, o que o sistema deve fazer, tecnologias e *softwares* exigidos pelo Cliente.

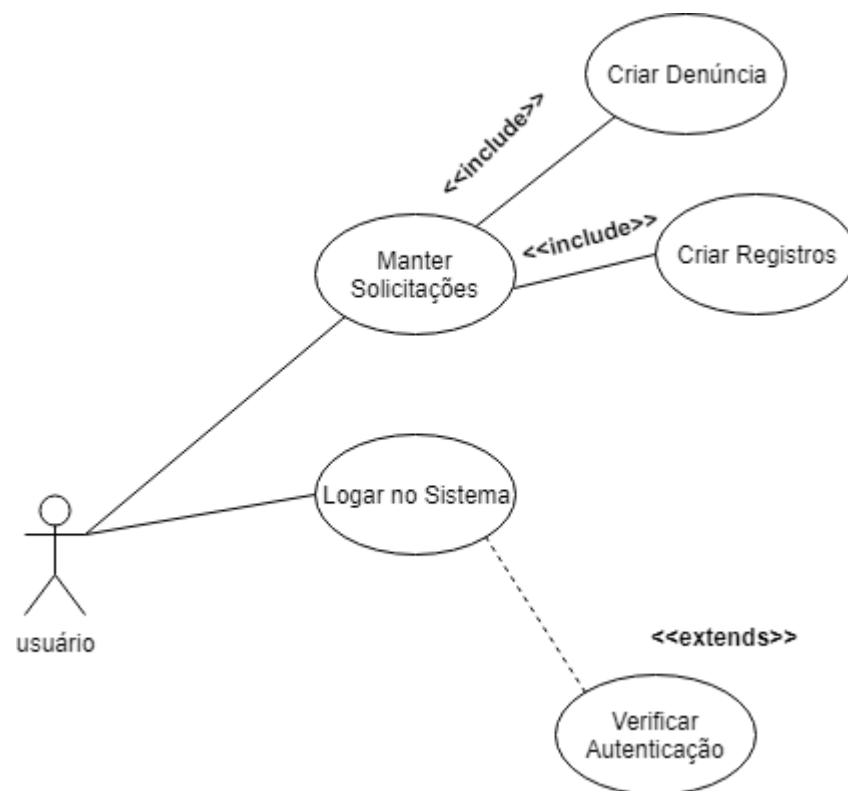
Pode-se verificar estes itens por meio da listagem abaixo:

- RNF1 - O sistema deverá ser desenvolvido na plataforma *Android Studio*;
- RNF2 - O sistema deverá utilizar como local de armazenamento dos registros uma VM disponibilizada pela UFPR.

## 2.4 DIAGRAMA DE CASO DE USO

O caso de uso representado na figura 6 descreve as funcionalidades propostas bem como a interação entre o usuário e o sistema. No diagrama o ator é o usuário, que poderá através do aplicativo: gerenciar empresa, gerenciar avaliação e gerenciar relatório.

FIGURA 6 – DIAGRAMA DE CASO DE USO (GERAL)



FONTE: Os autores (2019)

## 2.5 ESPECIFICAÇÃO DE CASO DE USO

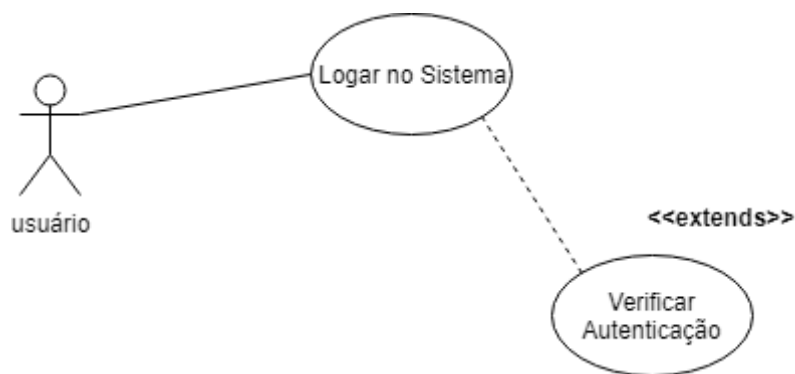
### 2.5.1 UC01 - Verificar Autenticação.

#### Descrição

Este caso de uso permite o sistema verificar a autenticação do usuário que acessou a plataforma.

#### Diagrama de Caso de Uso

FIGURA 7 – DIAGRAMA DE CASO DE USO (UC01)



FONTE: Os autores (2019)

#### Pré-condições

Este caso de uso pode iniciar somente se:

1. O usuário tiver tentado acesso ao sistema.

#### Pós-condições

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

1. Avaliar o usuário como autenticado.

#### Ator

Usuário: Qualquer pessoa com acesso a aplicação.

**Fluxo básico**

FB 1. O usuário acessa o sistema.

FB 2. O sistema realiza a validação de autenticação do usuário.

FB 3. O sistema valida que o usuário se adequa as verificações de segurança [FE1]

FB 4. O sistema redireciona o usuário para o UC03 – Manter Solicitações. [RN01]

FB 5. O caso de uso é finalizado.

**Fluxo alternativo**

Não se aplica.

**Fluxo de exceção**

FE 1 - Usuário não encontrado

FE 1. O sistema valida que o usuário não se adequa as verificações de segurança.

FE 2. O sistema redireciona o usuário para o UC02 – Login.

FE 3. O caso de uso é finalizado.

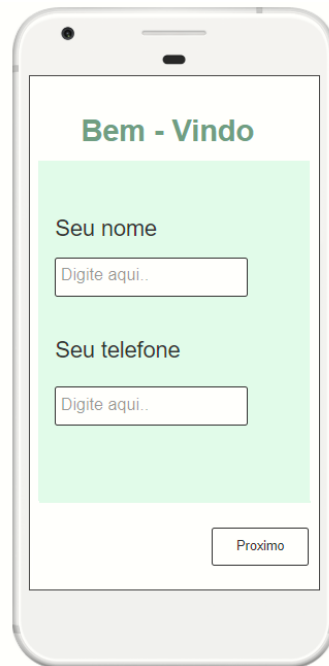
**Regras de Negócio**

RN01 – Apenas usuários autenticados tem acesso ao UC – Manter Solicitações.





FIGURA 9 – T01 – TELA DE LOGIN



FONTE: Os autores (2019)

### **Pré-condições**

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC 01 – Verificar Autenticação

### **Pós-condições**

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

1. Redirecionar o usuário para o UC 03 – Manter Solicitações.

### **Ator**

Usuário: Qualquer pessoa com acesso a aplicação.

### **Fluxo básico**

- FB 1. O usuário acessa o sistema.
- FB 2. O sistema realiza a verificação de autenticação.
- FB 3. O sistema identifica que o usuário não está autenticado. [FA1] [RN01]
- FB 4. O sistema redireciona para a tela de Login. [T01]
- FB 5. O usuário informa os dados necessários para logar.
- FB 6. O sistema valida os dados informados. [FE1] [FE2] [RN02] [RN03]
- FB 7. O sistema redireciona o usuário o UC03 – Manter Solicitações.
- FB 8. O caso de uso é finalizado.

### **Fluxo alternativo**

#### FA 1 - Usuário já autenticado

- FA 1. O sistema identifica que o usuário já está autenticado.
- FA 2. O sistema redireciona o usuário ao UC03 – Manter Solicitações
- FA 3. O caso de uso é finalizado.

### **Fluxo de exceção**

#### FE 1 - Dados Inválidos

- FE 1. O sistema valida que os dados inseridos pelo usuário não correspondem as máscaras dos campos.
- FE 2. O sistema apresenta mensagem [MSG01]
- FE 3. O caso de uso é finalizado.

#### FE 2 - Dados em branco

- FE 1. O sistema valida que os dados não foram inseridos pelo usuário
- FE 2. O sistema apresenta mensagem [MSG02]
- FE 3. O caso de uso é finalizado.

## **Regras de Negócio**

RN01 – O usuário sem exceção deve passar pela verificação de autenticação  
UC01 – Verificar Autenticação

RN02 – Os campos obrigatórios devem ser preenchidos com dados validos

RN03 – Apenas usuários autenticados anteriormente pelo UC01 – Verificar Autenticação devem ter acesso a esta funcionalidade

## **Mensagens**

MSG01 – “Dados informados em formato inválido! Digite apenas letras no campo Nome e apenas números no campo Telefone”

MSG02 – “Os campos obrigatórios não podem estar em branco”

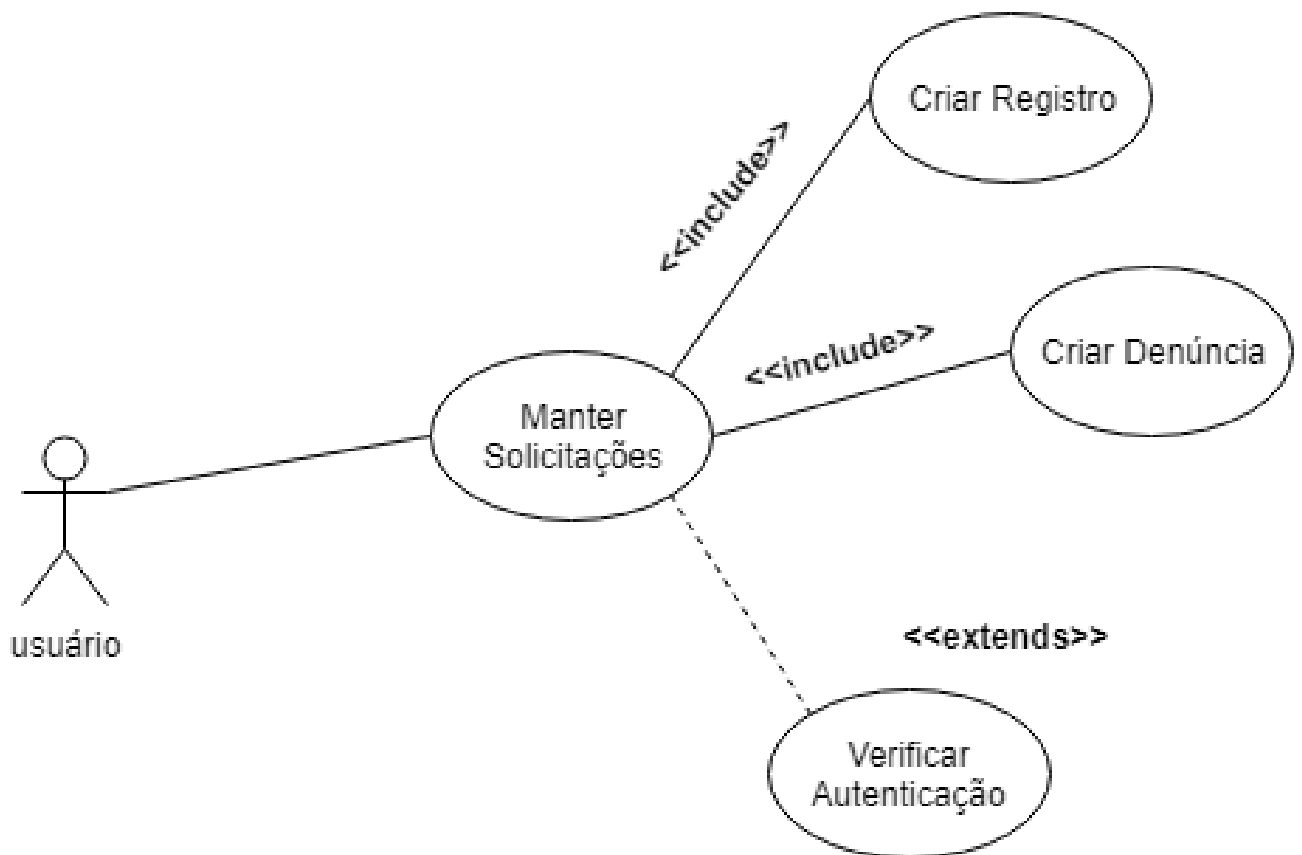
### 2.5.3 UC03 - Manter Solicitações.

#### Descrição

Este caso de uso permite ao usuário verificar as solicitações já enviadas/pendentes, suas informações e imagens. Assim como criar novas solicitações do tipo registro/denúncia.

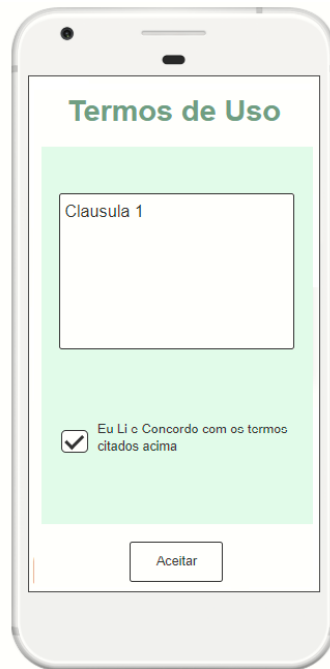
#### Diagrama de Caso de Uso

FIGURA 10 – DIAGRAMA DE CASO DE USO (UC03)



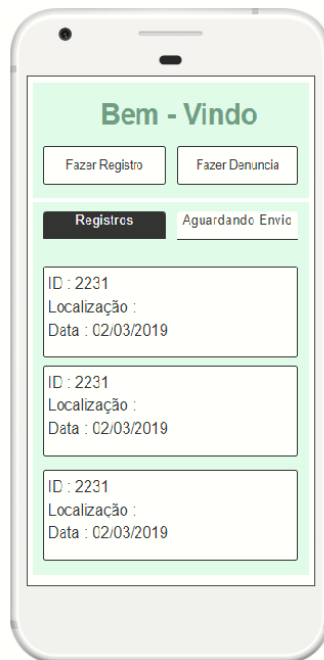
FONTE: Os autores (2019)

FIGURA 11 – T02 – TELA DE TERMOS



FONTE: Os autores (2019)

FIGURA 12 – T03 – TELA DE MANTER SOLICITAÇÕES



FONTE: Os autores (2019)

### **Pré-condições**

Este caso de uso pode iniciar somente se:

1. O sistema identificar que o usuário faz parte de um registro ativo já autenticado

### **Pós-condições**

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

1. Ter o termo de concessão de direitos autorais aceito pelo usuário ativo e autenticado da sessão.

### **Ator**

Usuário: Qualquer pessoa com acesso a aplicação que já passou pela etapa de autenticação UC01 e UC02.

### **Fluxo básico**

- FB 1. O sistema valida o aceite do termo pelo usuário. [FA1] [RN01]
- FB 2. O sistema carrega os campos e apresenta a tela Termos. [T02] [RN02]
- FB 3. O usuário visualiza o termo obrigatório referente aos direitos autorais e responsabilidade fotográfica. [ANEXO 1]
- FB 4. O usuário marca o checkbox e aciona o botão <ACEITO>. [FA2] [FE1]
- FB 5. O sistema valida o aceite, salva o registro de autorização do usuário.
- FB 6. O sistema redireciona o usuário para a tela Manter Solicitações [T03]
- FB 7. O usuário aciona a opção <CRIAR DENÚNCIA> [FA4][FA5][FA6]
- FB 8. O sistema redireciona o usuário para o UC04 - Criar Denúncia
- FB 9. O caso de uso é finalizado.

### **Fluxo alternativo**

- FA 1 –Termo já aceito anteriormente

- FA 1. O sistema redireciona o usuário para a tela de manter solicitações [T03]
- FA 2. O caso de uso é finalizado.

#### FA 2 – Opção voltar é acionada

- FA 1. O usuário aciona a opção <Voltar>
- FA 2. O sistema redireciona o usuário para a tela anterior.
- FA 3. O caso de uso é finalizado.

#### FA 3 – Opção criar registro é acionada

- FA 1. O usuário aciona a opção <CRIAR REGISTRO>
- FA 2. O sistema redireciona o usuário para o UC05 - Criar Registro.
- FA 3. O caso de uso é finalizado.

#### FA 4 – Opção histórico é acionada

- FA 1. O usuário aciona a opção <HISTÓRICO>
- FA 2. O sistema redireciona o usuário para a tela Manter Solicitações [T03]
- FA 3. O caso de uso é finalizado.

### **Fluxo de exceção**

#### FE 1 – Opção “Li e aceito os termos” não foi marcada

- FE 1. O sistema apresenta a mensagem [MSG01]
- FE 2. O sistema não redireciona o usuário para tela Manter Solicitações [T03]
- FE 3. O caso de uso é finalizado.

### **Regras de Negócio**

RN01 – É obrigatório o termo de direitos autorais e responsabilidade ser aceito pelo usuário

RN02 – O termo caso já aceito não é apresentado novamente para o usuário.



RN03 – Caso haja mudanças nos termos, a opção deverá ser apresentada ao usuário, mesmo que estes já tenham aceitado a versão anterior. As imagens enviadas antes da alteração do termo não perdem o direito de uso pelo app

## Mensagens

MSG01 “O termo é obrigatório para prosseguir.”

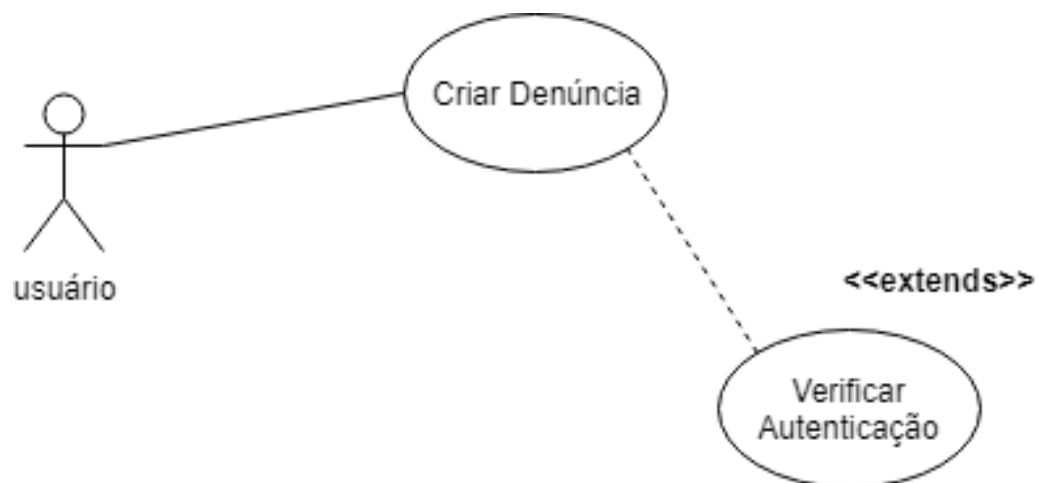
### 2.5.4 UC04 - Criar Denuncia.

## Descrição

Este caso de uso permite ao usuário criar uma solicitação do tipo denúncia, e enviar a requisição junto a suas imagens para o sistema.

## Diagrama de Caso de Uso

FIGURA 13 – DIAGRAMA DE CASO DE USO (UC04)



FONTE: Os autores (2019)

FIGURA 14 – T04 – TELA DE INSTRUÇÃO DENUNCIA



FONTE: Os autores (2019)

FIGURA 15 – T05 – TELA DE PREVIA DENUNCIA



FONTE: Os autores (2019)

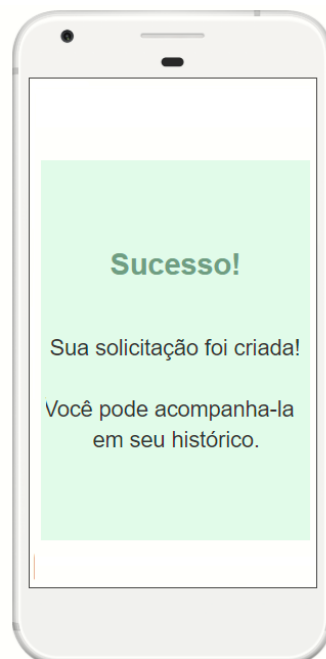
FIGURA 16 – T06 – TELA DE MAIS INFORMAÇÕES



The image shows a smartphone screen with a light green background. At the top, the title "Mais informações" is displayed in a bold, dark green font. Below the title, there are four input fields, each with a label above it: "Endereço", "Especie", "Detalhes", and "Problema". Each input field contains the placeholder text "Digite aqui..".

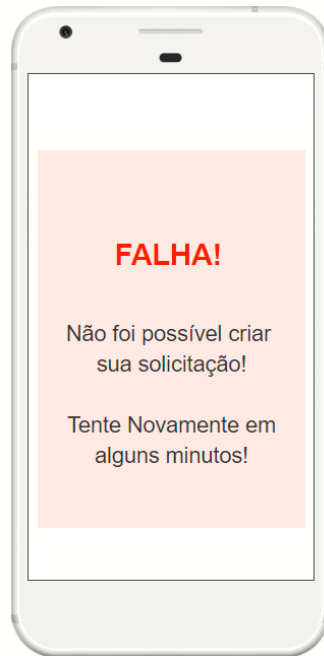
FONTE: Os autores (2019)

FIGURA 17 – T07 – TELA DE SUCESSO



FONTE: Os autores (2019)

FIGURA 18 – T08 – TELA DE FALHA



FONTE: Os autores (2019)

### **Pré-condições**

Este caso de uso pode iniciar somente se:

1. O sistema identificar que o usuário faz parte de um registro ativo já autenticado

### **Pós-condições**

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

1. Enviar para o banco de dados da aplicação uma solicitação do tipo denúncia junto as imagens adicionadas pelo usuário.

### **Ator**

Usuário: Qualquer pessoa com acesso a aplicação que já passou pela etapa de autenticação UC01 e UC02.

## Fluxo básico

FB 1. O usuário aciona a opção <Criar Denúncia> na tela Manter Solicitações.

FB 2. O sistema redireciona o usuário para a tela de instrução de denúncia.

[T04]

FB 3. O sistema apresenta a instrução para o primeiro registro.

FB 4. O usuário aciona o botão <OK> [FA1]

FB 5. O sistema chama a função câmera do dispositivo e redireciona o usuário para a tela de captura de imagens do sistema operacional.

FB 6. O usuário posiciona a câmera conforme instrução anterior e captura a imagem. [FA2]

FB 7. O sistema operacional retorna a imagem capturada para o sistema.

FB 8. O sistema apresenta a prévia da imagem capturada para o usuário. [T05]

FB 9. O usuário visualiza a imagem e aciona a opção <ENVIAR> [FE1]

FB 10. O sistema armazena a imagem obtida e redireciona para a próxima instrução. [T04]

FB 11. O usuário conclui todas as instruções e aciona <ENVIAR>.

FB 12. O sistema carrega a tela de mais informações. [T06]

FB 13. O usuário adiciona informações e aciona <ENVIAR>.

FB 14. O sistema valida os dados fornecidos pelo usuário. [RN01] [FE2]

FB 15. O sistema salva e armazena a denúncia no banco de dados. [FE3]

FB 16. O sistema carrega e redireciona o usuário para a confirmação [T07]

FB 17. O usuário realiza um clique no dispositivo.

FB 18. O sistema redireciona o usuário para o UC03 – Manter Solicitações

FB 19. O caso de uso é finalizado.

## Fluxo alternativo

FA 1 – Opção voltar é acionada

FA 1. O usuário aciona a opção <Voltar>

FA 2. O sistema redireciona o usuário para a tela anterior.

FA 3. O caso de uso é finalizado.

FA 2 – Opção pular é selecionada

FA 1. O usuário aciona a opção <Pular>

FA 2. O sistema redireciona o usuário para a próxima instrução

FA 3. O caso de uso é finalizado.

### **Fluxo de exceção**

FE 1 – Opção tentar novamente é acionada

FE 1. O usuário não deseja enviar a imagem anteriormente capturada.

FE 2. O usuário aciona a opção <Tentar Novamente>

FE 3. O sistema descarta a imagem anteriormente capturada.

FE 4. O sistema chama a função câmera do dispositivo e redireciona o usuário para a tela de captura de imagens do sistema operacional.

FE 5. O usuário posiciona a câmera conforme instrução anterior e captura a imagem.

FE 6. O sistema operacional retorna a imagem capturada para o sistema.

FE 7. O caso de uso é finalizado.

FE 2 – Nenhuma imagem foi adicionada a denúncia

FE 1. O sistema valida a inclusão de imagens na denúncia.

FE 2. O sistema apresenta a mensagem [MSG02]

FE 3. O sistema não encaminha a denúncia para o BD/Serviço

FE 4. O caso de uso é finalizado.

FE 3 – Sem conexão com a internet

FE 1. O sistema valida que o usuário não tem acesso a rede WI-FI.

FE 2. O sistema armazena a denúncia na memória do dispositivo

FE 3. O sistema disponibiliza a denúncia criada na aba aba 'Aguardando Envio' presente no UC03 – Manter Solicitações.

FE 4. O sistema carrega e redireciona o usuário para a confirmação [T08]

FE 5. O sistema mantém o envio pendente da denúncia. [RN02]

FE 5. O caso de uso é finalizado.

## Regras de Negócio

RN01 Ao menos uma foto/instrução deve ser realizada.

RN02 O sistema encaminha o chamado ao BD/Serviço quando a conexão do usuário a uma rede WI-FI for restabelecida.

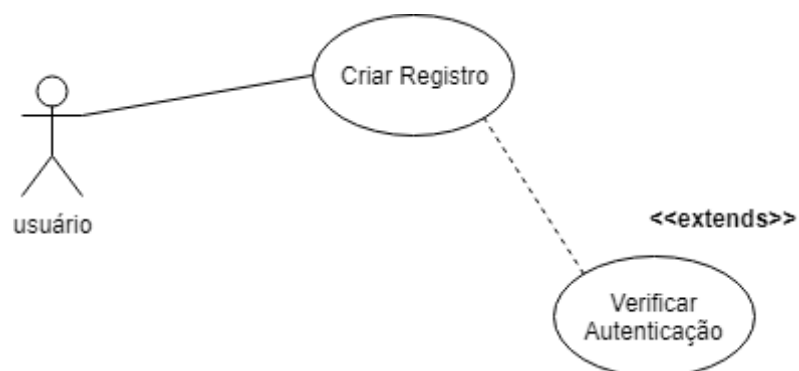
### 2.5.5 UC05 - Criar Registro 2.0.

## Descrição

Este caso de uso permite ao usuário criar uma solicitação do tipo denúncia, e enviar a requisição junto a suas imagens para o sistema.

## Diagrama de Caso de Uso

FIGURA 19 – DIAGRAMA DE CASO DE USO (UC05)



FONTE: Os autores (2019)

FIGURA 20 – T09 – TELA DE INSTRUÇÃO REGISTRO



FONTE: Os autores (2019)

FIGURA 21 – T10 – TELA DE PREVIA REGISTRO





FONTE: Os autores (2019)

FIGURA 22 – T11 – TELA DE MAIS INFORMAÇÕES



**Mais informações**

Endereço  
Digite aqui..

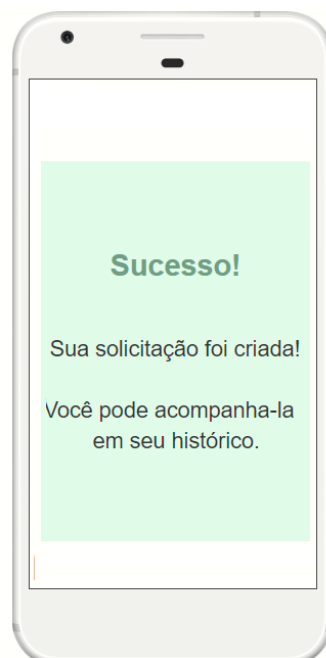
Especie  
Digite aqui..

Detalhes  
Digite aqui..

Problema  
Digite aqui..

FONTE: Os autores (2019)

FIGURA 23 – T12 – TELA DE SUCESSO

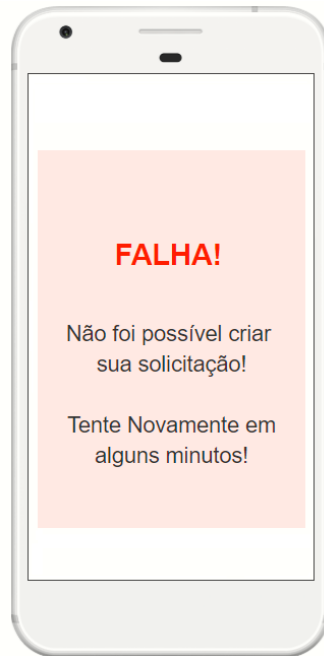


**Sucesso!**

Sua solicitação foi criada!

Você pode acompanhá-la  
em seu histórico.

FONTE: Os autores (2019)  
FIGURA 24 – T13 – TELA DE FALHA



FONTE: Os autores (2019)

### **Pré-condições**

Este caso de uso pode iniciar somente se:

1. O sistema identificar que o usuário faz parte de um registro ativo já autenticado

### **Pós-condições**

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

1. Enviar para o banco de dados da aplicação uma solicitação do tipo denúncia junto as imagens adicionadas pelo usuário.

### **Ator**

Usuário: Qualquer pessoa com acesso a aplicação que já passou pela etapa de autenticação UC01 e UC02.

## Fluxo básico

- FB 1. O usuário aciona a opção <Criar Registro> na tela Manter Solicitações.
- FB 2. O sistema redireciona o usuário para a tela de instrução de denúncia.
- [T04]
- FB 3. O sistema apresenta a instrução para o primeiro registro.
- FB 4. O usuário aciona o botão <OK> [FA1]
- FB 5. O sistema chama a função câmera do dispositivo e redireciona o usuário para a tela de captura de imagens do sistema operacional.
- FB 6. O usuário posiciona a câmera conforme instrução anterior e captura a imagem. [FA2]
- FB 7. O sistema operacional retorna a imagem capturada para o sistema.
- FB 8. O sistema apresenta a prévia da imagem capturada para o usuário. [T05]
- FB 9. O usuário visualiza a imagem e aciona a opção <ENVIAR> [FE1]
- FB 10. O sistema armazena a imagem obtida e redireciona para a próxima instrução. [T04]
- FB 11. O usuário conclui todas as instruções e aciona <ENVIAR>.
- FB 12. O sistema carrega a tela de mais informações. [T06]
- FB 13. O usuário adiciona informações e aciona <ENVIAR>.
- FB 14. O sistema valida os dados fornecidos pelo usuário. [RN01] [FE2]
- FB 15. O sistema salva e armazena a denúncia no banco de dados. [FE3]
- FB 16. O sistema carrega e redireciona o usuário para a confirmação [T07]
- FB 17. O usuário realiza um clique no dispositivo.
- FB 18. O sistema redireciona o a usuário para o UC03 – Manter Solicitações
- FB 19. O caso de uso é finalizado.

## Fluxo alternativo

FA 1 – Opção voltar é acionada

- FA 1. O usuário aciona a opção <Voltar>
- FA 2. O sistema redireciona o usuário para a tela anterior.
- FA 3. O caso de uso é finalizado.

FA 2 – Opção pular é selecionada

FA 1. O usuário aciona a opção <Pular>

FA 2. O sistema redireciona o usuário para a próxima instrução

FA 3. O caso de uso é finalizado.

### **Fluxo de exceção**

FE 1 – Opção tentar novamente é acionada

FE 1. O usuário não deseja enviar a imagem anteriormente capturada.

FE 2. O usuário aciona a opção <Tentar Novamente>

FE 3. O sistema descarta a imagem anteriormente capturada.

FE 4. O sistema chama a função câmera do dispositivo e redireciona o usuário para a tela de captura de imagens do sistema operacional.

FE 5. O usuário posiciona a câmera conforme instrução anterior e captura a imagem.

FE 6. O sistema operacional retorna a imagem capturada para o sistema.

FE 7. O caso de uso é finalizado.

FE 2 – Nenhuma imagem foi adicionada a denúncia

FE 1. O sistema valida a inclusão de imagens na denúncia.

FE 2. O sistema apresenta a mensagem [MSG02]

FE 3. O sistema não encaminha a denúncia para o BD/Serviço

FE 4. O caso de uso é finalizado.

FE 3 – Sem conexão com a internet

FE 1. O sistema valida que o usuário não tem acesso a rede WI-FI.

FE 2. O sistema armazena a denúncia na memória do dispositivo

FE 3. O sistema disponibiliza a denúncia criada na aba aba 'Aguardando Envio' presente no UC03 – Manter Solicitações.

FE 4. O sistema carrega e redireciona o usuário para a confirmação [T08]

FE 5. O sistema mantém o envio pendente da denúncia. [RN02]

FE 5. O caso de uso é finalizado.

### **Regras de Negócio**

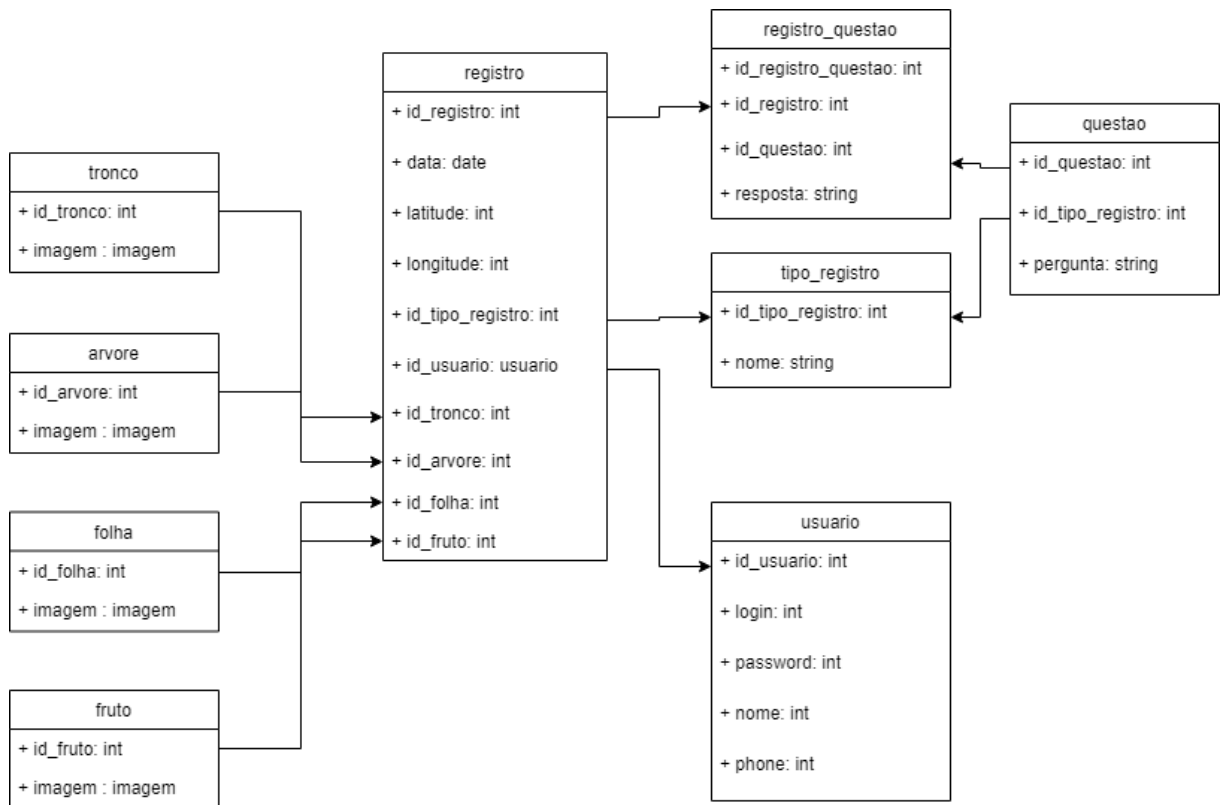
RN01 Ao menos uma foto/instrução deve ser realizada.

RN02 O sistema encaminha o chamado ao BD/Serviço quando a conexão do usuário a uma rede WI-FI for restabelecida.

## 2.6 DIAGRAMA DE CLASSES

O diagrama da figura 25 representa as classes existentes no aplicativo: tais como seus relacionamentos.

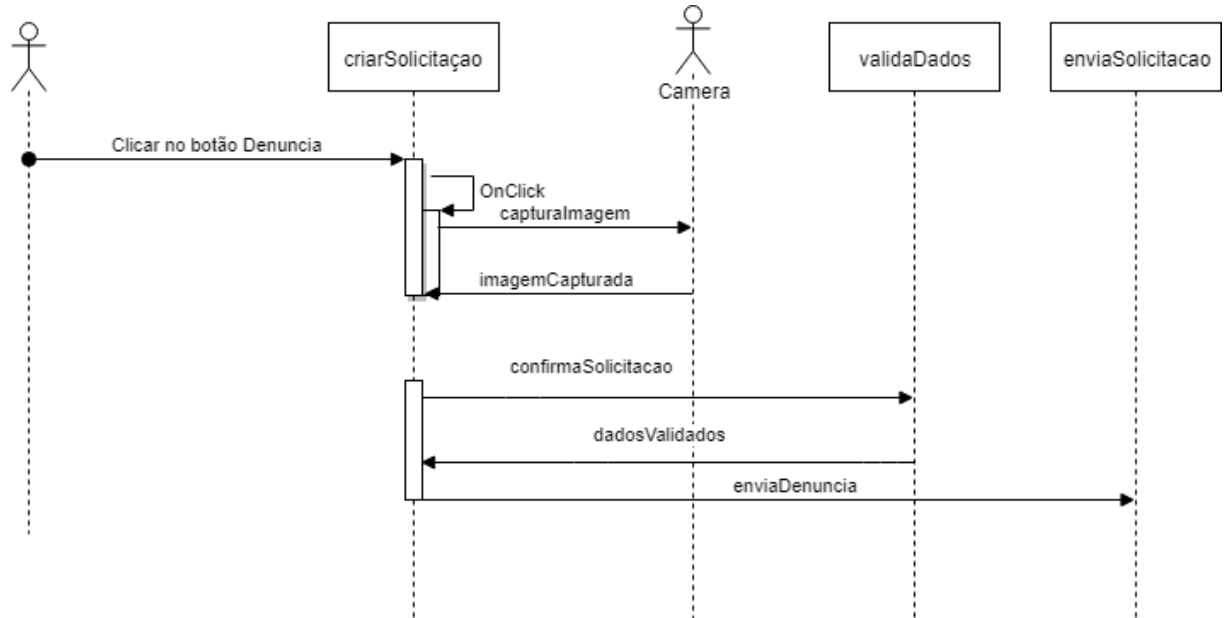
FIGURA 25 – T13 – DIAGRAMA DE CLASSES



FONTE: Os autores (2019)

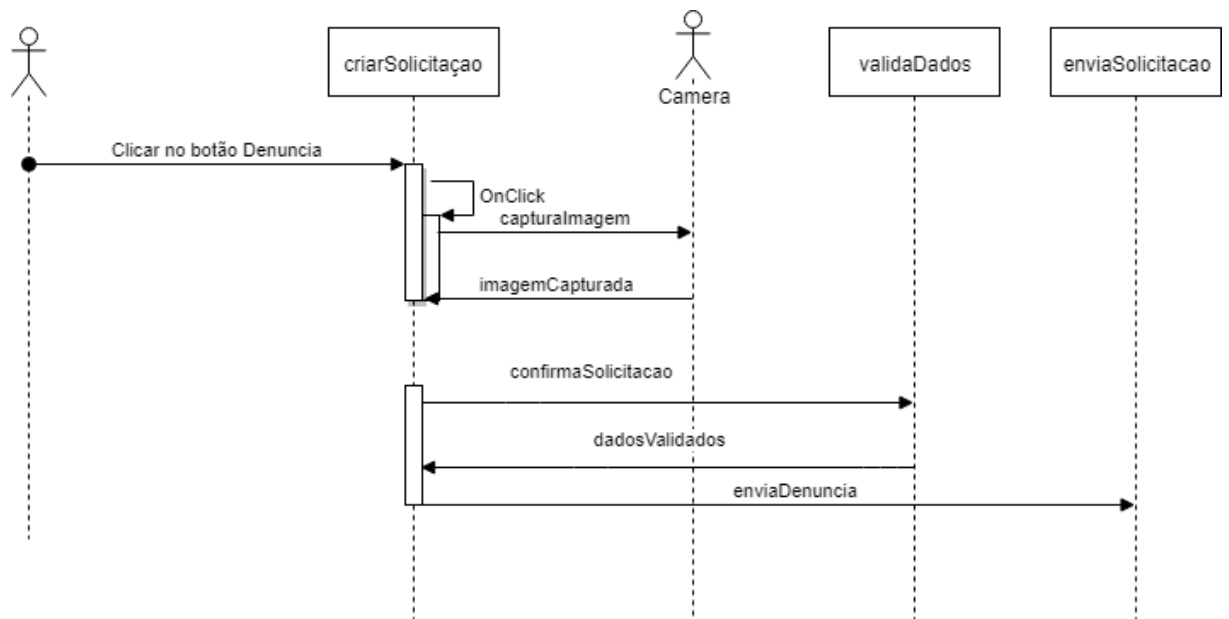
## 2.7 DIAGRAMA DE SEQUENCIA

FIGURA 26 – DIAGRAMA DE SEQUENCIA – CRIAR REGISTRO



FONTE: Os autores (2019)

FIGURA 27 – DIAGRAMA DE SEQUENCIA – CRIAR DENUNCIA



FONTE: Os autores (2019)

### 3 CONCLUSÃO

Desde as primeiras reuniões com o Cliente foi enfatizado o quanto é essencial ter um aplicativo funcional ao término do projeto, não somente para atender as necessidades do mesmo, mas também para disponibilizar as imagens em banco de dados para as equipes avaliadoras e para a continuidade do projeto.

Optou-se por aplicar a metodologia Ágil, implantada de forma gradual, e o framework de gerenciamento de projetos *SCRUM* que com poucas práticas pode aplicar ao desenvolvimento de software. O que nos proporcionou algumas entregas ao longo do cronograma, gerando assim valor ao Cliente.

Após compreender as principais necessidades da Cliente, acreditamos que o objetivo inicial do projeto foi atingido, implementação de um sistema que realizasse o registro fotográfico e de observações das espécies de árvores urbanas, em um único aplicativo, versão *mobile*.

Utilizamos diversas tecnologias gratuitas para o desenvolvimento do aplicativo, sendo todas elas de extrema importância para a construção e funcionamento do produto. Foi implementada a metodologia Scrum,



## **4 CONSIDERAÇÕES FINAIS**

Como parte de um processo, o presente trabalho representa a conclusão de uma das etapas previstas, onde esperamos que seja aplicado e testado por diferentes pessoas, especialistas e pessoas comuns, com diferentes modelos de celular, antes a efetiva disponibilização e implantação. Cabe o importante registro, como previsto no projeto principal, sem o envolvimento e compromisso da Prefeitura em usar os dados do aplicativo para atender as solicitações e a demanda da população, é melhor que o aplicativo não seja disponibilizado, para não produzir falsas expectativas e inviabilizar um importante mecanismo para a sociedade e para os órgãos públicos, em outras palavras, quando for dado publicidade para o uso do aplicativo, é fundamental, que as instituições estejam preparadas para dar as respectivas resposta aos cidadãos

### **4.1 RECOMENDAÇÕES PARA TRABALHOS FUTUROS**

Avaliar a inclusão de filtros, que controlem o envio e recebimento das fotografias, impedindo que o sistema seja utilizado para propagação de imagens fora do escopo proposto nesse trabalho;

Com a efetiva utilização da aplicação, o sistema poderá classificar os usuários, para aumentar a prioridade das denúncias proveniente de usuários, que apresentaram denúncias reais, em detrimento dos usuários eventuais ou com baixa taxa de contribuição na prestação de informações que favoreçam a tomada de decisão pelas equipes técnicas.

## REFERÊNCIAS

ÁGIL. Disponível em: <http://www.desenvolvimentoagil.com.br/scrum>. Acesso em: 31 mar. 2019.

AGILE. Disponível em <http://www.allaboutagile.com/10-good-reasons-to-do-agile-development/> >

AMBLER, Scott W. **Modelagem Ágil: Práticas eficazes para a Programação extrema e o Processo Unificado**. Porto Alegre: Bookman, 2004.

BAPTISTELLA, Adriano José. **Abordando a arquitetura MVC, e Design Patterns: Observer, Composite, Strategy**. Disponível em <http://www.linhadecodigo.com.br/artigo/2367/abordando-a-arquitetura-mvc-e-design-patterns-observer-composite-strategy.aspx#ixzz4IUzrnUbq> > Acesso em: 26 mar. 2019.

BIZAGI. **Bizagi BPMN Modeler**. Disponível em <http://www.bizagi.com/pt/produtos>>. Acesso em: 25 mar 2019.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML Guia do Usuário**. 2ª ed. Rio de Janeiro. Editora Campus, 2006. 463p.

BROOKLINE, Benjamin, <http://slideplayer.com/slide/6604918/> - **Presentation on theme: "Coaching Skills for Scrum Masters & The Self-Organizing Team."** - **Coaching Skills for Scrum Masters & The Self-Organizing Team**. Acesso em 20 ago. 2019.

COCKBURN Alistair, HIGHSMITH Jim - **Agile Software Development: The People Factor** - IEEE Computer, v. 34, n 11, p 131-133, nov. 2001. Disponível em <http://www.uml.org.cn/softwareprocess/pdf/IEEEArticle2Final2.pdf>>. Acesso em: 24 abr. 2019.

CRUZ Fabio - **Scrum e PMBOK unidos no Gerenciamento de Projetos** – Rio de Janeiro: Brasport 2013. Disponível em <https://books.google.com.br/books?hl=pt-BR&lr=&id=SJA37S2QGR0C&oi=fnd&pg=PA1&dq=papeis+do+scrum&ots=lvM3v1xTny&sig=MU1MH9oeQYLXsgqjbvMc tqSwE#v=onepage&q&f=false>>. Acesso em 26 abr. 2019.

DALAMARO, MALDONADO, JINO. **Introdução ao Teste de Software**. 13ª Tiragem. Rio de Janeiro. Elsevier. 2007.

DURELLI, V. H. S.; VIANA, M.C.; PENTEADO, R. A. D. **Uma Proposta de Réuso de Interface Gráfica com o Usuário Baseada no Padrão Arquitetural MVC**. XVIII Simpósio Brasileiro de Engenharia de Software (SBES). 2008.

Disponível em:

<[http://s3.amazonaws.com/academia.edu.documents/30662249/0024.pdf?AWSAccessKeyId=AKIAJ56TQJRTWSMTNPEA&Expires=1470014734&Signature=4AqizbJesi u25UpJsiAG0xZMxfs%3D&response-content-disposition=inline%3B%20filename%3DUma Proposta de Reuso de Interface Gr a fi.pdf](http://s3.amazonaws.com/academia.edu.documents/30662249/0024.pdf?AWSAccessKeyId=AKIAJ56TQJRTWSMTNPEA&Expires=1470014734&Signature=4AqizbJesi u25UpJsiAG0xZMxfs%3D&response-content-disposition=inline%3B%20filename%3DUma+Proposta+de+Reuso+de+Interface+Gr a fi.pdf)>. Acesso em: 31 abr. 2019.

FADEL Aline Cristine, SILVEIRA Henrique da Mota - **Metodologias Ágeis no Contexto de Desenvolvimento de Software: XP, Scrum e Lean**. Limeira, 2010. Disponível em

<[http://www.ft.unicamp.br/liag/Gerenciamento/monografias/Lean%20Agil\\_v8.pdf](http://www.ft.unicamp.br/liag/Gerenciamento/monografias/Lean%20Agil_v8.pdf)>. Acesso em 25 abr. 2019.

FURGERI, Sérgio. **Java 6: Ensino Didático: Desenvolvendo e Implementando Aplicações**. São Paulo: Érica, 2008. 17-19 p.

GITHUB. <<https://github.com/>>. Acesso em: 31 abr. 2019.

JANA. <<https://blog.jana.com/blog>>. Acesso em: 01 abr. 2019.

JOHNSON, Thienne M. **Java para Dispositivos Móveis: desenvolvendo aplicações com J2ME**. São Paulo: Ed. Novatec, 2007.

LECHETA, Ricardo R. **Google Android: aprenda a criar aplicações para dispositivos móveis com o Android SDK**. 2. ed. rev. e ampl. São Paulo: Novatec, 2010. 608 p.

MANIFESTO ÁGIL. Disponível em: <<http://www.manifestoagil.com.br/>>. Acesso em 23 mai. 2019.

ORMLITE FRAMEWORK. Disponível em: <<http://ormlite.com/>>. Acesso em 30 mai. 2019.

PAIVA, Samuel. **Astah Community, um software para trabalha com UMLs!**– Disponível em <<http://www.plantaonerd.com/blog/2011/04/18/astah-community-um-software-para-trabalha-com-umls/>>. Acesso em: 23 mai. 2019.

PRESSMAN, Roger S. **Engenharia de Software**. São Paulo. Makron Books, 2011. 58-63 p.

SILVA, F. G.; HOENTSCH, S. C. P.; SILVA, L. **Uma análise das Metodologias Ágeis FDD e Scrum sob a Perspectiva do Modelo de Qualidade MPS.BR**. Publicação: SCIENTIA PLENA, 2009, p. 3. Disponível em:<[http://www.scientiaplina.org.br/sp\\_v5\\_121301.pdf](http://www.scientiaplina.org.br/sp_v5_121301.pdf)>. Acesso em: 29 mai. 2019.

SCRUM. Disponível em <<http://scrumtrainingseries.com/>> Acesso em 20 mai. 2019.

SOKOLOVA K, LEMERCIER M, GARCIA L (2013) **Android passive MVC: a novel architecture model for the android application development**. In: Proceedings of the fifth international conference on pervasive patterns and applications (PATTERNS'13). IARIA, pp 7–12.

SOMMERVILLE, Ian. **Engenharia de Software**. 9ª ed. São Paulo. Pearson Prentice Hall, 2011. 521p.

SQLITE. Disponível em <<https://www.sqlite.org/>>. Acesso em 30 jul. 2016.

STANGARLIN, Lize. **Lista de Avaliação para Boas Práticas em Serviços de Alimentação Rdc 216**. Editora Varela, 2006. 47p.

TECHTUDO .Disponível em <<http://www.techtudo.com.br/tudo-sobre/google-play.html>> Acesso em Acesso em 17 mai. 2019.

USER INTERFACES (UI) in Android applications. Disponível em <<http://www.cs.ccsu.edu/~stan/classes/CS355/notes/03-AndroidUI.html>> Acesso em: 2 mai. 2019.

## ANEXO 1 – TERMO DE AUTORIZAÇÃO DE USO DE IMAGENS E RESPONSABILIDADE FOTOGRÁFIC

Pelo presente instrumento as partes celebram acordo que caracteriza a concessão de licença de reprodução de Obras Fotográficas, conforme o abaixo relacionado, e se obrigam por si e por seus eventuais sucessores ao cumprimento das cláusulas aqui dispostas.

1. CONCEDENTE Nome: **Nome do usuário** Telefone: **(00) 00000-0000**

2. AUTORIZADA Neste ato representadas por seus prepostos, identificados acima como os Autores do projeto **AS ÁRVORES DA CIDADE**.

3. OBRA(S) OBJETO DA LICENÇA Qualquer imagem fotográfica anexada por esta ferramenta em um ou mais registros enviados pela plataforma **AS ÁRVORES DA CIDADE**.

4. DIREITOS E OBRIGAÇÕES DA AUTORIZADA:

- A AUTORIZADA tem permissão para reproduzir a(s) obra(s) fotográfica(s) discriminada(s) neste instrumento, no(s) produto(s) e/ou evento(s) indicado(s) nas cláusulas anteriores.
- A AUTORIZADA não se compromete a não efetuar cessão ou transferências dos direitos autorais inerentes ao objeto do presente termo, não permanecendo o CONCEDENTE como único e exclusivo titular deste.
- A AUTORIZADA não obriga-se a indicar, em qualquer utilização, a autoria da obra licenciada, fazendo expressa menção ao nome dos autores e ao Banco de Imagens.
- A AUTORIZADA não se responsabiliza por imagens as quais a CONCEDENTE enviou. Sejam estas de conotação sexista, racista, homofobia ou possa gerar danos a terceiros.

5. DIREITOS E OBRIGAÇÕES DO ÓRGÃO CONCEDENTE:

- O CONCEDENTE declara ser o titular dos direitos autorais e patrimoniais das obras objeto desta Licença.
- A concessão de licença objeto deste termo importará na cessão e transferência dos direitos autorais, dos quais o CONCEDENTE não permanece como único e exclusivo titular.
- O CONCEDENTE obriga-se a disponibilizar gratuitamente à AUTORIZADA, durante tempo indeterminado, a contar da aceitação deste termo, o material descrito no item 3 do presente ajuste.
- A AUTORIZADA poderá reproduzir a(s) obra(s) fotográfica(s) discriminada(s) neste instrumento, no(s) produto(s) indicado(s) no item 4, em todas suas versões impressas e eletrônicas (internet), em qualquer idioma e/ou área geográfica, por um prazo indeterminado a partir da aceitação deste termo, deste que se comprometa a não divulgar imagens e/ou informações com conotação sexista, racista, homofobia ou possa gerar danos a terceiros.

- A AUTORIZADA poderá ceder, transferir ou sublicenciar a reprodução das obras a terceiros, desde que o CONCEDENTE tenha aceitado este termo ao realizar a autenticação na plataforma **AS ÁRVORES DA CIDADE**.

6. VALOR E FORMA DE PAGAMENTO: O presente termo será celebrado gratuitamente, sem quaisquer custas, repasses orçamentários ou dispêndio pecuniário, por quaisquer das partes.

7. PENALIDADES: Caso sejam enviadas obras de conotação sexista, racista, homofobia ou possa gerar danos a terceiros a CONCEDENTE será cobrado unicamente dos valores processuais.