

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO
CÂMPUS CAMPINAS

GABRIEL VISELI BARZAGLI

**SISTEMA COLABORATIVO DE INTERAÇÃO PRODUTOR - PESQUISADOR
PARA O DIAGNÓSTICO PRECOCE DE DOENÇAS EM PLANTAS A PARTIR DA
ANÁLISE DE IMAGENS**

CAMPINAS

2017

GABRIEL VISELI BARZAGLI

Sistema colaborativo de interação produtor - pesquisador para o diagnóstico precoce de doenças em plantas a partir da análise de imagens

Trabalho de Conclusão de Curso apresentado como requisito para a obtenção do diploma do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia Câmpus Campinas.

Orientador: Prof. Dr. Andreiwid Sheffer Corrêa.

Coorientador: Prof. Dr. Tiago Carvalho

CAMPINAS

2017

B296s Barzagli, Gabriel Viseli
Sistema colaborativo de interação produtor-pesquisador para o diagnóstico precoce de doenças em plantas a partir da análise de imagens / Gabriel Viseli Barzagli. – Campinas, 2017.
35f. : il.

Orientador: Andreiwid Sheffer Corrêa.
Coorientador: Tiago Carvalho.

Monografia (Graduação) – Instituto Federal de São Paulo – Câmpus Campinas. Curso de Tecnologia em Análise e Desenvolvimento de Sistemas, 2017.

1. Agricultura. 2. Plantas - Doenças. 3. Plantas - Diagnóstico. 4. Aplicativo móvel. 5. Comunicação – Produtores. 6. Comunicação – Pesquisadores. I. Instituto Federal de São Paulo - Câmpus Campinas. Curso de Tecnologia em Análise e Desenvolvimento de Sistemas. II. Título.

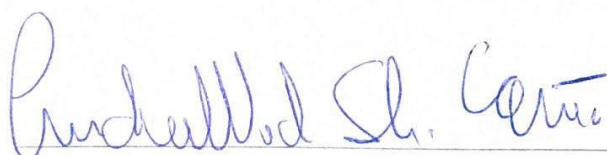
Gabriel Viseli Barzagli

**Sistema colaborativo de interação produtor - pesquisador para o diagnóstico precoce de
doenças em plantas a partir da análise de imagens**

Trabalho de Conclusão de Curso apresentado
como requisito para obtenção do diploma do
Curso de Tecnologia em Análise e
Desenvolvimento de Sistemas do Instituto
Federal de Educação, Ciência e Tecnologia de
São Paulo Câmpus Campinas.

Aprovado pela banca examinadora em: 11 de DEZEMBRO de 2017.

BANCA EXAMINADORA



Prof. Dr. Andreiuid Sheffer Corrêa (orientador)
Instituto Federal de São Paulo Câmpus Campinas

Prof. Me. Glauber da Rocha Balthazar
Instituto Federal de São Paulo Câmpus Campinas

Dra. Luciana Alvim Santos Romani
Embrapa

*Dedico este trabalho aos meus familiares,
colegas de classe, professores e servidores do Instituto
que colaboraram em minha jornada formativa.*

AGRADECIMENTOS

Agradeço aos meus pais que me proporcionaram toda a formação acadêmica necessária para que eu alcançasse o que eu sou hoje.

Agradeço aos meus amigos e colegas de classe que me ajudaram alguma vez nos trabalhos acadêmicos ou na explicação de conteúdo da disciplina, e agradeço a todos os professores e servidores do Instituto Federal de São Paulo (IFSP) Campus Campinas, que contribuíram direta e indiretamente para a conclusão desse trabalho.

Agradeço ao meu orientador Andreiuid Sheffer Corrêa que me auxiliou na elaboração deste trabalho e agradeço ao meu coorientador Tiago Carvalho por todo o auxílio na interação com a Embrapa e outros assuntos referentes ao trabalho inclusive sugestões de melhorias.

Agradeço também às pessoas da Embrapa as quais tive contato durante o desenvolvimento e que me ajudaram quando precisei.

*"Viva como se fosse morrer amanhã.
Aprenda como se fosse viver para sempre."
Mahatma Gandhi*

RESUMO

O crescimento da população mundial traz inúmeras preocupações, entre elas se haverá comida suficiente para suprir a demanda das pessoas nas próximas décadas. Estima-se que parte do problema pode ser resolvido por meio da redução do desperdício de alimentos, sendo que uma das etapas onde ocorre maior desperdício é no cultivo e armazenamento. Este trabalho tem como objetivo desenvolver uma aplicação que proporcione e incentive a comunicação entre produtores e pesquisadores, com a intenção de reduzir o desperdício na cadeia produtiva pelo diagnóstico precoce de doenças. Parte-se de uma solução com o uso de um aplicativo móvel para enviar fotos dos sintomas para uma rede colaborativa de pesquisadores onde se tem mecanismos para analisar e publicar um diagnóstico por meio de uma interface web.

Palavras-chave: Diagnóstico. Doenças. Plantas. Celular.

ABSTRACT

The world's population growth brings numerous concerns; including if there will be enough food to supply the demand of people in the next decades. Part of the problem is estimated to be solved by reducing the food waste, and one of the most wasteful stages is in cultivation and storage. This work aims to develop an application that provides and encourages communication between producers and researchers with the intention of reducing the waste in the productive chain by the early diagnosis of diseases. It starts from a solution with the use of a mobile application to send photos of the symptoms to a collaborative network of researchers, which has mechanisms to analyze and publish diagnosis through a web interface.

Keywords: Diagnostic. Disease. Plants. Mobile.

LISTA DE FIGURAS

Figura 1 – Fluxograma do funcionamento do sistema.....	20
Figura 2 – Ilustração do funcionamento do sistema.....	24
Figura 3 – Diagrama relacional.....	25
Figura 4 - Diagrama de classes do modelo de dados.....	26
Figura 5 - Diagrama de caso de uso do sistema.....	26
Figura 6 – Diagrama de classe da estrutura de envio de fotos.....	27
Figura 7 – Tela de autenticação no sistema.....	31
Figura 8 – Tela de registro no sistema.....	31
Figura 9 – Tela de visualização das postagens.....	32
Figura 10 – Visualização dos diagnósticos.....	32

LISTA DE QUADROS

Quadro 1 – Requisitos funcionais e não funcionais.....	22
Quadro 2 – Classificação de usuários.....	23
Quadro 3 – Retornos do web service.....	30

LISTA DE SIGLAS

CSS	<i>Cascading Style Sheets</i>
DER	Diagrama Entidade-Relacionamento
FAO	<i>Food and Agriculture Organization of the United Nations</i>
HTML	<i>Hyper Text Markup Language</i>
IFSP	Instituto Federal de Educação, Ciência e Tecnologia de São Paulo
ONU	Organização das Nações Unidas
ORM	<i>Object-Relational Mapping</i>
REST	<i>Representational State Transfer</i>
SMS	<i>Short Message Service</i>
SOA	<i>Service Oriented Architecture</i>
SOAP	<i>Simple Object Access Protocol</i>
TCC	Trabalho de Conclusão de Curso

SUMÁRIO

1 INTRODUÇÃO	13
1.1 Objetivos.....	14
<i>1.1.1 Objetivo geral</i>	<i>14</i>
<i>1.1.2 Objetivos específicos</i>	<i>14</i>
1.2 Justificativa	14
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 Desperdício de comida no mundo	16
2.2 Mudança no clima e impactos na produção de comida	16
2.3 Pequenos produtores no Brasil.....	17
2.4 Colaboração social.....	17
2.5 Java	17
2.6 PostgreSQL	18
2.7 SOA	18
2.8 Padrão de projeto MVC.....	18
3 MÉTODO	19
3.1 Levantamento dos requisitos funcionais e não funcionais	19
3.2 Desenvolvimento do método de classificação de pesquisadores	19
3.3 Modelo de dados	21
3.4 Desenvolvimento do servidor.....	21
3.5 Desenvolvimento do cliente web.....	21
4 RESULTADOS	22
5 CONCLUSÕES.....	33
6 TRABALHOS FUTUROS	34
REFERÊNCIAS	35

1 INTRODUÇÃO

Atualmente somos aproximadamente 7,6 bilhões de pessoas no mundo, de acordo com a publicação da ONU (DEPARTMENT OF ECONOMIC AND SOCIAL AFFAIRS OF THE UNITED NATIONS, 2017). Esse fato mostra que a expectativa de vida do ser humano cresceu muito se comparado a 50 anos atrás. Projeções indicam que em 2050 devemos alcançar a marca de 9 bilhões de habitantes, um número que traz a dúvida se será possível sustentar tantas pessoas (DEPARTMENT OF ECONOMIC AND SOCIAL AFFAIRS OF THE UNITED NATIONS, 2017).

Para que seja possível alimentar tamanha população, seria necessário um crescimento na produção em 70% em relação à produção anual de 2007, exigindo que países em desenvolvimento dobrem sua produção alimentícia, de acordo com a Organização das Nações Unidas para a Alimentação e Agricultura (FOOD AND AGRICULTURE ORGANIZATION OF THE UNITED NATIONS, 2009), implicando no aumento significativo na produção de alguns principais produtos.

O Brasil, como país em desenvolvimento, constitui sua principal base econômica a agropecuária desde os primórdios da colonização. Em 2010, nosso país se tornou o terceiro maior exportador de produtos agrícolas do mundo, perdendo apenas para os Estados Unidos da América e União Europeia de acordo com a Organização Mundial do Comércio (LANDIM, 2010). Isso foi devido a uma constante evolução no processo de cultivo das lavouras nos últimos anos, diminuindo perdas e consequentemente aumentando a produtividade.

Apesar da evolução, muito fruto ainda é perdido desde o produtor de grande porte até os pequenos produtores, sendo mais preocupante nos agronegócios familiares, onde a infraestrutura e o conhecimento necessários para prevenir doenças nas lavouras e assim evitar perdas são incipientes. Portanto, torna-se necessário a intervenção de pesquisadores para a realização de um diagnóstico precoce do problema antes que se perca o alimento.

Como o custo de locomoção de pesquisadores até os locais de plantio é alto se torna necessária a utilização de um meio de comunicação que possa oferecer suporte ao diagnóstico de forma remota. Em vista disso, este trabalho busca desenvolver um sistema colaborativo de interação entre produtor e pesquisadores para o diagnóstico precoce de doenças em plantas a partir da análise de imagens.

Assim, espera-se que produtores sem poder aquisitivo suficiente para comprar tecnologia de ponta para a suas produções agrícolas possam ter acesso, pelo menos, ao

conhecimento sobre o problema e que os grandes produtores rurais possam reduzir custos na consultoria do problema de seus cultivos.

Este trabalho foi realizado em parceria com a Embrapa onde foi acordado que o aplicativo para o agricultor fosse desenvolvido pela instituição, enquanto este trabalho ficaria encarregado de desenvolver o servidor e o sistema de diagnóstico colaborativo para criar um meio de comunicação entre as partes interessadas.

1.1 Objetivos

Nesta seção é apresentado os objetivos do trabalho, que consistem em duas subseções: o objetivo geral que trata do que se deseja alcançar e os objetivos específicos que tratam dos objetivos a serem cumpridos para se alcançar o objetivo geral.

1.1.1 Objetivo geral

O objetivo deste trabalho é desenvolver um sistema que permita o envio de fotos de partes de plantas com alguma anomalia e também o diagnóstico colaborativo entre os pesquisadores por meio da análise visual das fotos. Assim que a foto for analisada de forma colaborativa por pesquisadores, o agricultor deve receber o diagnóstico com a informação sobre qual o problema enfrentado.

1.1.2 Objetivos específicos

- Desenvolver servidor para o recebimento e armazenamento das fotos enviadas;
- Desenvolver uma rede colaborativa para o diagnóstico das fotos enviadas.

1.2 Justificativa

A perda de alimento pode ocorrer por vários motivos, entre eles estão as doenças, as pragas e o clima, este último sobretudo causa a falta ou excesso de umidade nas folhas e consequentemente aumenta a vulnerabilidade da planta. Por causa desses problemas, pela falta de informação sobre medidas preventivas ou corretivas e a escassez de mecanismos automatizados em um pequeno agronegócio, muitas colheitas são perdidas gerando diversos prejuízos ao agricultor.

Medidas de detecção semiautomática de doenças já foram desenvolvidas em órgãos de pesquisa como a Embrapa. Porém, nem sempre essas medidas são suficientes para detectar o problema e há uma falta de imagens com boa resolução para que seja possível implementar um algoritmo que faça a classificação automaticamente ou melhorar algum para que faça de maneira mais efetiva e precisa. Logo, torna-se necessário criar um meio de comunicação eficiente entre o agricultor e pesquisadores, permitindo que o agricultor envie o seu problema e receba um diagnóstico sobre o que está afetando sua plantação.

O custo dessa solução é relativamente baixo para o agricultor, havendo apenas a necessidade do agricultor possuir um smartphone compatível e conexão com a internet e para o pesquisador é necessário o acesso à internet e ao sistema de diagnóstico colaborativo. A solução traz benefícios para produtores de qualquer porte, desde o pequeno até o de grande porte com a redução de custos em consultoria.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção é apresentado os assuntos abordados por esse trabalho, baseado em textos de outros autores. Esses assuntos são referentes ao problema relacionado ao desperdício de comida em sua produção e também sobre tecnologias utilizadas no desenvolvimento deste Trabalho de Conclusão de Curso (TCC).

2.1 Desperdício de comida no mundo

Anualmente são desperdiçados em torno de 1,3 bilhões de toneladas de comida de acordo com o relatório da Organização das Nações Unidas para a Alimentação e Agricultura (2013). Baseado nesse valor, Peixoto e Pinto (2016) elencam prejuízos significativos para a sociedade, como, por exemplo, perdas econômicas com a redução de oferta e aumento dos preços do produto. O desperdício pode se dar de duas formas, sendo elas a perda de comida e o descarte de comida.

A perda de comida trata de toda quantia de alimentos que teve decréscimo de forma involuntária em massa ou valor nutricional que era destinada ao consumo por humanos. As perdas geralmente são associadas a ineficiências na cadeia de fornecimento, como infraestrutura pobre, falta de tecnologia ou de conhecimento (FOOD AND AGRICULTURE ORGANIZATION OF THE UNITED NATIONS, 2013).

Já o descarte de comida trata do descarte de comida apropriada para o consumo, sendo rejeitado voluntariamente pelo consumidor final ou pelas empresas por se encontrarem em sua data de validade, por exemplo (FOOD AND AGRICULTURE ORGANIZATION OF THE UNITED NATIONS, 2013).

2.2 Mudança no clima e impactos na produção de comida

A produção de alimento ainda depende bastante do clima apesar de toda tecnologia já empregada no meio agrícola e de melhorias no rendimento das culturas. Isso porque a precipitação, radiação solar e temperatura são os principais requisitos para o crescimento das plantações (ROSENZWEIG et al., 2001).

ROSENZWEIG et al. (2001) também diz que alterações no clima influenciam no fornecimento ou demanda por irrigação e inclusive podem aumentar a vulnerabilidade das culturas a doenças, pragas e sufocamento por ervas daninhas.

2.3 Pequenos produtores no Brasil

Como escreve MELLO (2007, p. 1),

agricultura familiar é uma forma de produção através da interação entre gestão e trabalho; são os próprios agricultores que dirigem o processo produtivo, trabalhando com a diversificação e utilizando o trabalho familiar, eventualmente completando com o trabalho assalariado.

De acordo com o Censo Agropecuário 2006 Brasil, Grandes Regiões e Unidades da Federação (IBGE, 2006), 84,4% do total dos estabelecimentos agropecuários pertencem a grupos familiares, sendo a base econômica de 90% dos municípios brasileiros com até 20 mil habitantes e corresponde a 35% do produto interno bruto nacional. O termo agricultor familiar se estende inclusive a silvicultores, aquicultores, extrativistas, pescadores, indígenas, quilombolas e assentados de reforma agrária.

2.4 Colaboração social

Colaboração social é um processo que auxilia no compartilhamento de informações entre várias pessoas ou grupos para alcançar objetivos comuns. A colaboração usada corretamente amplia a exposição ao conhecimento, simplifica o fluxo de ideias e faz com que todos pensem sobre como resolver um problema ou melhorar processos, produtos e serviços (CARR, 2014).

Existem diversos sistemas que utilizam desse conceito, entre eles pode-se citar o Stack Overflow, Yahoo! Answers, Wikipedia, Github e Quora.

2.5 Java

A linguagem, de acordo com o texto “*The History of Java Technology*” escrita pela Oracle, foi criada em 1995 e surgiu com a promessa de funcionar em todos dispositivos existentes, coisa que até então nenhuma linguagem ou tecnologia fornecia.

Ela é uma linguagem compilada e interpretada, onde seu código é, em um primeiro momento, compilada em bytecode e depois é interpretada pela JVM (*Java Virtual Machine*).

Atualmente, Java ainda é bastante utilizada em diversos softwares e tecnologias que usamos, sejam eles celulares Android, servidores Web e aplicações *desktop*.

Este trabalho usará a linguagem juntamente com o framework Spring pelas diversas facilidades que juntos fornecem.

2.6 PostgreSQL

É um banco de dados relacional de código aberto que ganhou notoriedade por possuir grande confiabilidade, integridade de dados e correção de erros, podendo rodar em todos os grandes sistemas operacionais, como Linux, UNIX e Windows.

O BD é totalmente compatível com ACID (*Atomicity, Consistency, Isolation, Durability*), tem suporte para chaves estrangeiras, *joins*, *views*, *triggers*, procedimentos e implementa os padrões do SQL92 assim como muitas outras funcionalidades do SQL99 (DOUGLAS, Korry; DOUGLAS, Susan, 2003).

2.7 SOA

Service-Oriented Architecture é um modelo de arquitetura de software onde visa a criação de serviços web interoperáveis que podem ser compartilhados livremente entre diversas aplicações, independentemente da plataforma onde estiverem rodando.

Dois tipos de SOA são usados atualmente: o SOAP (*Simple Object Access Protocol*) que utiliza de XML para a transferência de dados entre as partes e o REST (*Representational State Transfer*) que utiliza de JSON para a transferência de dados.

O trabalho utiliza de REST pois entre as principais diferenças entre ela e a arquitetura SOAP, é que ela é mais simples de ser desenvolvida e usa menos banda ao se transmitir dados, de acordo com WAGH e THOOL (2012).

2.8 Padrão de projeto MVC

O *Model-View-Controller* é um padrão de projeto de software que divide o software em módulos, sendo um para a camada de negócios (*Model*), um para a exibição de conteúdo (*View*) e um para o controle de fluxo de operações entre as duas partes (*Controller*).

3 MÉTODO

Nesta seção, descreve-se como este trabalho foi desenvolvido de forma detalhada em etapas. Essas etapas buscam construir um conjunto de ferramentas para auxiliar o desenvolvimento do projeto.

O projeto foi desenvolvido utilizando o método em cascata para desenvolvimento de software (PRESSMAN, 2005, p. 47-48), realizado em cinco etapas sequenciais, sendo elas: levantamento de requisitos funcionais e requisitos não funcionais, desenvolvimento do método de classificação de pesquisadores, definição do modelo de dados, desenvolvimento do servidor e desenvolvimento do cliente do pesquisador.

3.1 Levantamento dos requisitos funcionais e não funcionais

Nesta etapa foram levantados requisitos funcionais e não funcionais do sistema de acordo com as necessidades da Embrapa. Os requisitos funcionais consistem nas características do que o software deve fazer, enquanto os requisitos não funcionais se referem a especificações técnicas que devem ser cumpridas para que o software funcione.

Os resultados dessa etapa foram obtidos por meio de conversas com pessoas da Embrapa sobre quais funcionalidades o sistema deveria abranger e também quais são as limitações tecnológicas que devem ser respeitadas de acordo com a disponibilidade de recursos.

3.2 Desenvolvimento do método de classificação de pesquisadores

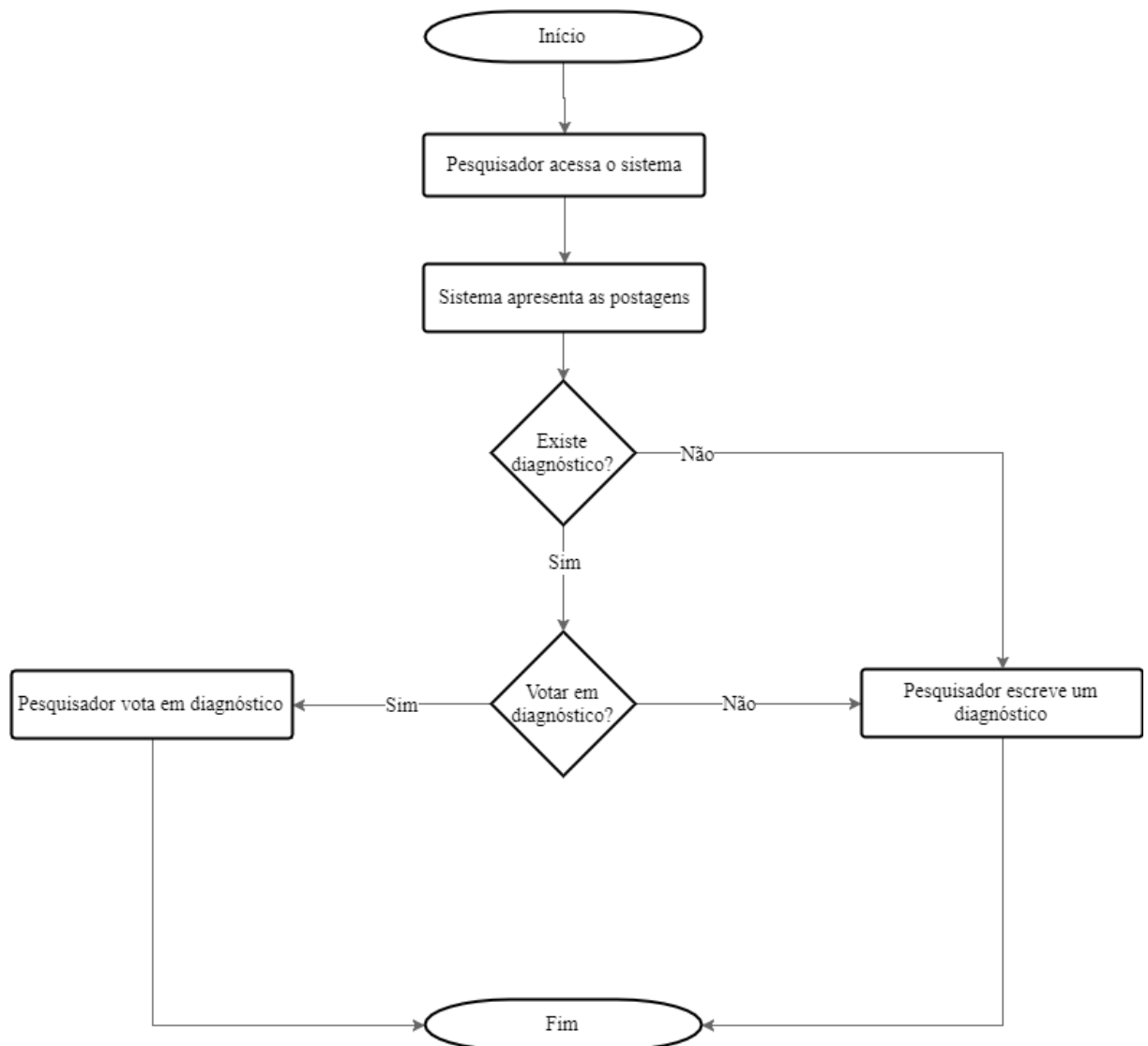
Nesta etapa foi desenvolvido o método de classificação dos usuários da rede colaborativa de diagnósticos com o intuito de destacar os melhores contribuidores e estabelecer a base para um sistema de reputação mútua baseada em votos.

Quando um pesquisador acessa o sistema, ele se depara com uma série de postagens com fotos de partes das plantas com alguma anomalia. Então, o pesquisador é encorajado a diagnosticar tal foto usando a caixa de texto presente no sistema sobre qual é o problema da folha de acordo com a sua análise visual da foto.

Um diagnóstico é representado no sistema como um comentário na foto. Nele deve apresentar a conclusão da análise do pesquisador a respeito do problema que é apresentado na foto. Ao se deparar com um diagnóstico satisfatório, o pesquisador pode votar

no comentário do outro pesquisador, através de um botão específico no sistema, para que tal diagnóstico fique em evidência. Caso ele não concorde com nenhum diagnóstico, o usuário pode fornecer a sua análise usando a caixa de texto. A Figura 1 mostra um fluxograma para explicar o funcionamento desse sistema.

Figura 1 - Fluxograma do funcionamento do sistema



Fonte: elaborado pelo autor

A quantidade de votos e diagnósticos que um pesquisador possui são usados para o cálculo de uma pontuação a fim de classificá-los. A pontuação (**P**) se baseia na quantidade de votos (**V**) e na quantidade de diagnósticos (**D**), onde: $P = 10 * V + 5 * D$. Então, de acordo com determinado intervalo de valores pré-determinados e configuráveis, o usuário é

classificado com uma das possíveis classificações disponíveis e tal informação aparece ao lado do nome do usuário.

O resultado do diagnóstico é obtido por meio de uma compilação dos três melhores diagnósticos ranqueados por meio da plataforma para os pesquisadores, estando presente assim que o agricultor abrir o aplicativo móvel.

3.3 Modelo de dados

Nesta etapa foram desenvolvidos os modelos de dados utilizados na aplicação para melhor atender aos requisitos elencados, assim como toda a arquitetura do sistema. O modelo será responsável por armazenar informações sobre as postagens dos agricultores, assim como os diagnósticos dos pesquisadores para envio como resposta à postagem.

3.4 Desenvolvimento do servidor

Para suprir a necessidade de receber fotos de um aplicativo de celular, foi necessário a construção de *web services* numa arquitetura SOA para controlar o fluxo de operações.

Portanto, a arquitetura REST será usada no desenvolvimento junto com o framework Java chamado Spring que facilita muito a codificação das funcionalidades por permitir a utilização de padrões de projeto e conceitos como *dependency injection* (injeção de dependência), arquitetura MVC e *data binding* (ligação de dados).

Dessa forma, o servidor consegue interoperar com o sistema Android desenvolvido pela Embrapa e também operar com o cliente *web* descrito abaixo para o diagnóstico cooperativo.

3.5 Desenvolvimento do cliente web

Para possibilitar o diagnóstico colaborativo, foi desenvolvido um cliente web usando HTML5, JavaScript, CSS3, com auxílio de alguns frameworks como JQuery e MaterializeCSS.

O sistema precisa ter uma tela de registro de pesquisadores, uma tela de autenticação do pesquisador, uma tela para visualizar as postagens de fotos e em cada foto deve ser possível a adição de um diagnóstico ou votação de um diagnóstico já presente.

4 RESULTADOS

Como resultado da etapa de levantamento dos requisitos funcionais e não funcionais, temos eles categorizados na Tabela 1 e uma breve descrição dos requisitos funcionais.

Quadro 1 - Requisitos funcionais e não funcionais

Requisitos funcionais	Requisitos não funcionais
Permitir o envio de fotos	Servidor Linux Ubuntu
Armazenar as fotos de forma catalogada	Banco de dados PostgreSQL 9.5
Cadastro do pesquisador	OpenJDK 8
Acessar o sistema	Servidor de aplicação Apache Tomcat 8
Visualizar fotos postadas	
Possibilitar o diagnóstico colaborativo	
Enviar diagnóstico ao agricultor	

Fonte: elaborado pelo autor

Permitir o envio de fotos

O sistema deve possibilitar que o agricultor envie fotos das plantas com anomalias presentes em sua plantação.

Armazenar as fotos de forma catalogada

O sistema deve armazenar as fotos enviadas de forma catalogada, para que auxilie futuramente no desenvolvimento ou aprimoramento de algoritmos forenses.

Cadastro do pesquisador

O sistema deve permitir que os pesquisadores se cadastrem, fornecendo seus dados, como nome, e-mail e senha.

Acessar no sistema

O sistema deve permitir o acesso no sistema mediante a autenticação do usuário por meio de e-mail e senha.

Visualizar fotos postadas

O sistema deve permitir a visualização das fotos postadas pelos agricultores logo na página principal do sistema.

Possibilitar o diagnóstico colaborativo

O sistema deve permitir que mais de um pesquisador forneça um diagnóstico da mesma foto, possibilitando também a concordância em um diagnóstico por meio de um voto.

Enviar diagnóstico ao agricultor

O sistema deve enviar os diagnósticos ao agricultor como resposta a imagem postada.

Na etapa do desenvolvimento de metodologia de classificação de pesquisadores foram definidos alguns valores pré-definidos para a classificação do usuário. Na Tabela 2, está presente a relação dos intervalos das pontuações iniciais sugeridas e as suas respectivas denominações.

Quadro 2 - Classificação de usuários

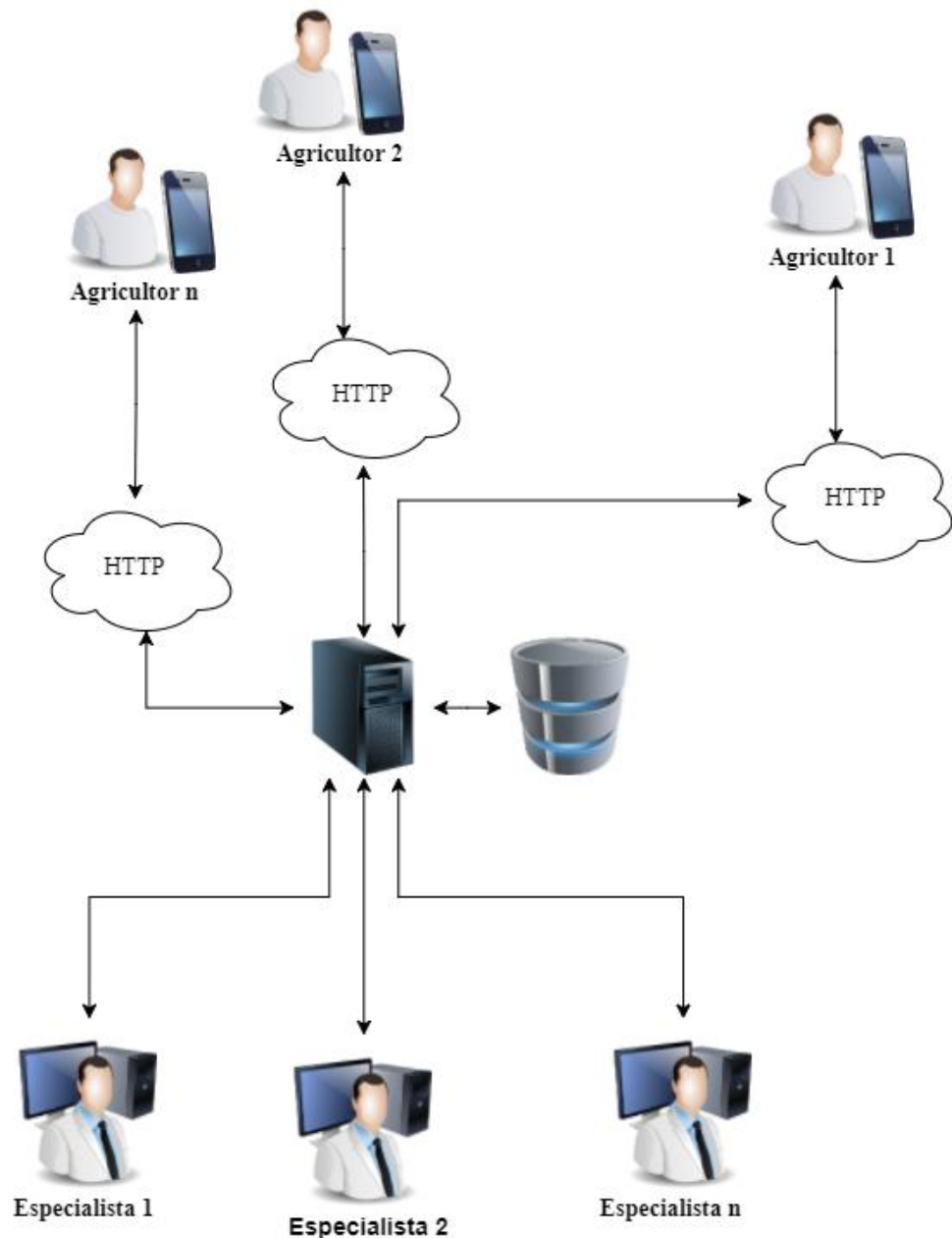
Classificação	Pontuação (P)
Pesquisador	$0 \leq P < 50$
Pesquisador I	$50 \leq P < 100$
Pesquisador II	$100 \leq P < 500$
Pesquisador III	$500 \leq P < 1000$
Pesquisador IV	$P \geq 1000$

Fonte: elaborado pelo autor

O pesquisador é avaliado de acordo com a pontuação que fizer baseado na fórmula especificada na seção método deste documento. Todos os pesquisadores iniciam como usuários neutros do sistema e vão progredindo nas classificações seguintes conforme for aumentando a quantidade de votos recebidos em seus diagnósticos.

Na etapa de modelo de dados foram elaborados diagramas para a elaboração da arquitetura do sistema. Como se pode observar na Figura 2, o sistema funciona com o envio de fotos e informações para um servidor por meio de serviços *web* que ao recebe-las, armazena suas informações em um banco de dados relacional.

Figura 2 - Ilustração do funcionamento do sistema

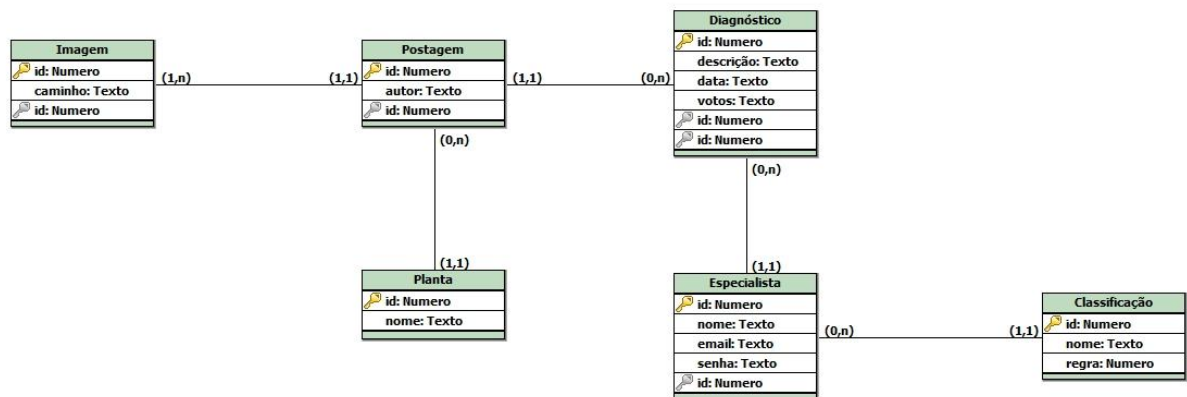


Fonte: elaborado pelo autor

Assim que estiver disponível as postagens dos agricultores, uma equipe de pesquisadores podem acessar o sistema por meio de um módulo cliente-servidor *web* por meio de uma URL direto de seus navegadores. Já a Figura 3 retrata o diagrama relacional da modelagem de dados para garantir que o sistema funcione.

O diagrama detalha as tabelas e suas relações, mostrando como os dados estão relacionados. Assim possibilita-se o envio de diversas fotos para uma mesma postagem e também o envio de mais de um diagnóstico para uma postagem.

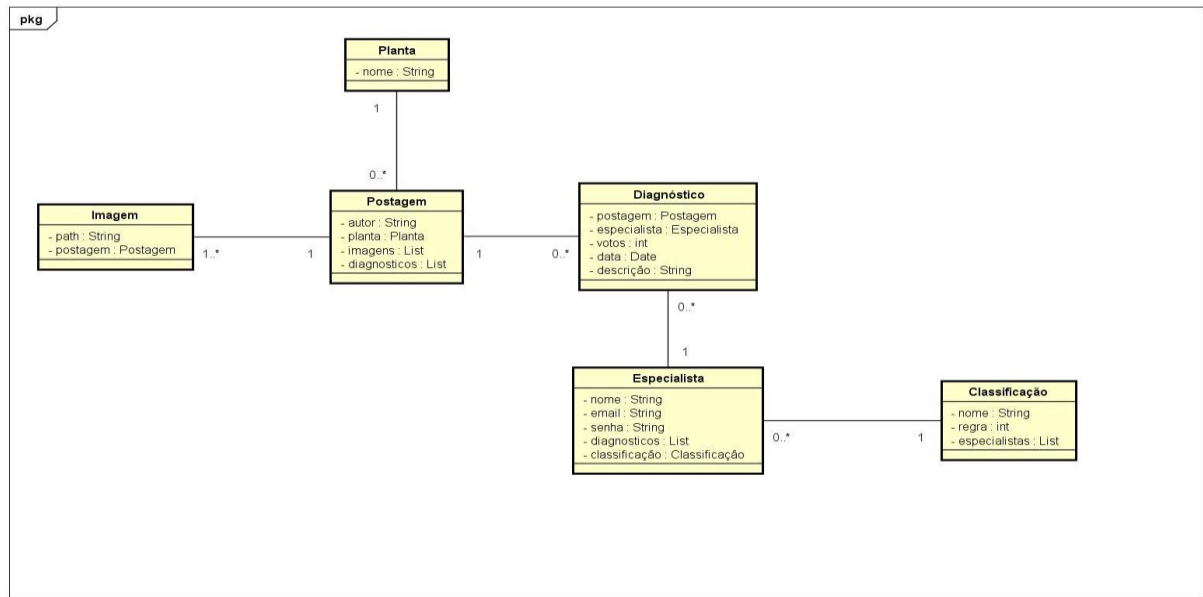
Figura 3 - Diagrama relacional



Fonte: elaborado pelo autor

A partir do diagrama, é possível definir uma estrutura de classes do sistema para a representação dos dados. O diagrama de classes abaixo (Figura 4) mostra como as classes estão relacionadas e seus respectivos atributos possibilitando que uma entidade consiga recuperar o valor de outra.

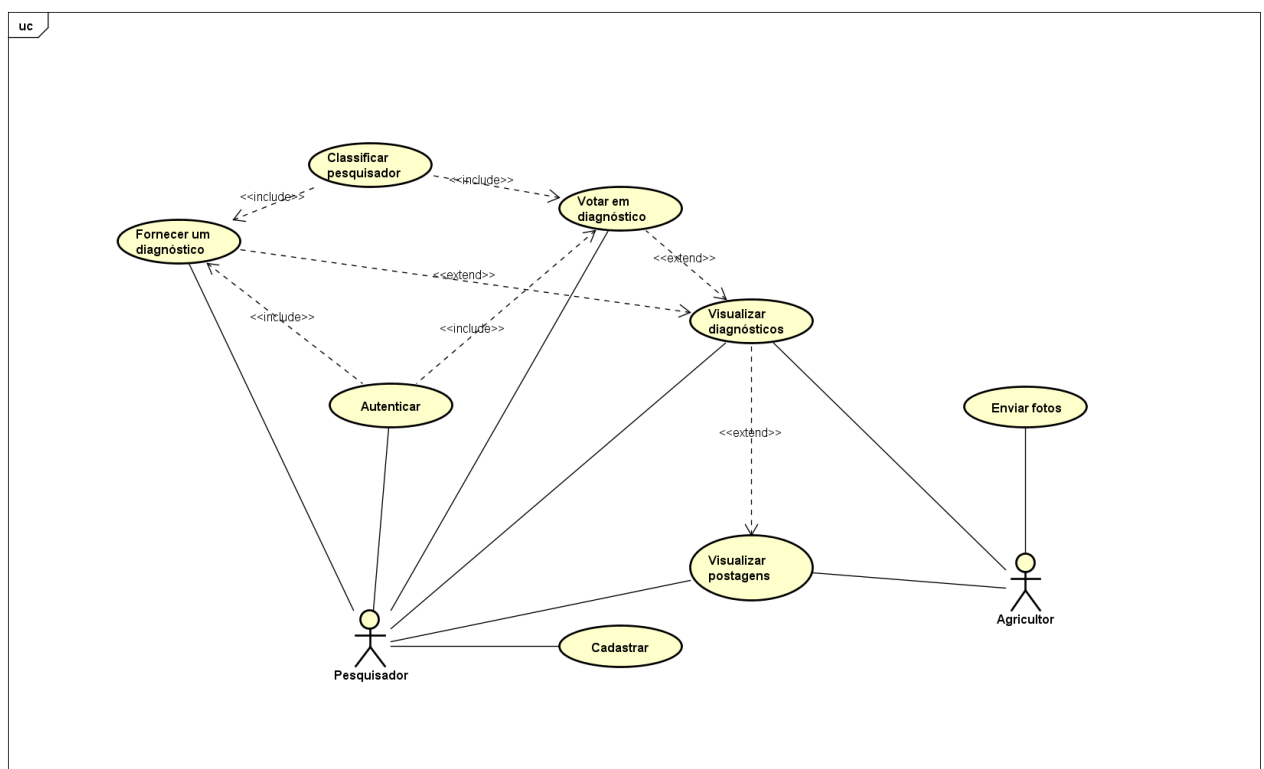
Figura 4 - Diagrama de classes do modelo de dados



Fonte: elaborado pelo autor

O modelo de dados possibilita que os usuários possam realizar uma série de operações que foram definidas na etapa de requisitos funcionais. A partir dos requisitos, é então criado o diagrama de caso de uso do sistema (Figura 5) que mostra a relação entre os atores do sistema e funcionalidades da mesma.

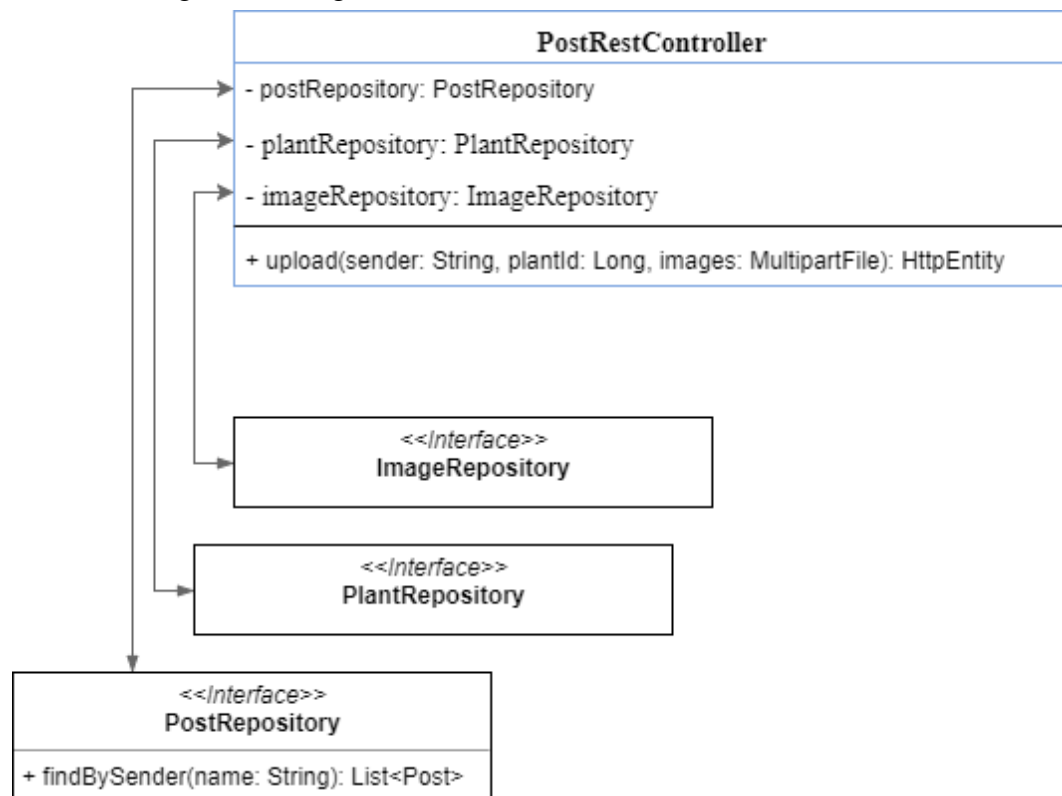
Figura 5 - Diagrama de caso de uso do sistema



Fonte: elaborado pelo autor

A etapa do desenvolvimento do servidor faz uso da linguagem de programação Java juntamente com o framework Spring para a criação da estrutura para recebimento de fotos enviadas pelo aplicativo. A Figura 6 apresenta as classes responsáveis pelo recebimento e envio de diagnóstico em um diagrama de classe.

Figura 6 - Diagrama de classe da estrutura de envio de fotos



Fonte: elaborado pelo autor

As interfaces “*Repository*” fazem parte do *framework* como um substituto ao padrão DAO (*Data Access Object*), encapsulando toda a lógica referente ao acesso ao banco e a manipulação de seus dados.

O código abaixo é pertencente ao *web service* de envio das fotos usado pelo aplicativo Android. Como pode se observar, os arquivos são enviados numa requisição e a partir dela é feito todo um tratamento dos dados para a armazenagem.

```

/**
 * POST endpoint to upload a image to our server
 *
 * @param sender
 *         email of the producer
 * @param plant
 *         id of the corresponding plant
 * @param images
 *         jpeg file of the plant's leaf
 * @return HTTP response status: <br>
 *         <ul>
 *         <li>201 Post created</li>
 *         <li>400 Missing some information in the request</li>
 *         <li>500 Server internal error</li>
 *         </ul>
 */
@RequestMapping(method = RequestMethod.POST)
public HttpEntity<Object> upload(@RequestParam String sender, @RequestParam Long
plantId, @RequestParam MultipartFile files) {
    if ((sender != null && !sender.isEmpty()) && (files != null && !files.isEmpty()) &&
        (plantId != null)) {
        ZipInputStream zip = null;
        FileOutputStream fos = null;
        try {
            List<Image> images = null;
            zip = new ZipInputStream(files.getInputStream());
            ZipEntry entry = zip.getNextEntry();
            while (entry != null) {
                String imageName = Constants.PREFIX_IMAGE_NAME +
                    new Date().getTime() + Constants.SUFIX_JPEG_NAME;
                File file = new File(imageName);
                if (!file.exists()) {
                    file.createNewFile();
                }
            }
        }
    }
}

```

```

        fos = new FileOutputStream(file);
        IOUtils.copy(zip, fos);
        fos.flush();
        fos.close();
        entry = zip.getNextEntry();
        Image image = new Image();
        image.setPath(file.getAbsolutePath());
        if (images == null) {
            images = new ArrayList<>();
        }
        images.add(image);
    }
    Plant plant = plantRepository.findOne(plantId);
    Post post = new Post(sender, plant, images);
    postRepository.save(post);
} catch (IOException e) {
    return new
    ResponseEntity<>(HttpStatus.INTERNAL_SERVER_ERROR);
} finally {
    try {
        if (fos != null) {
            fos.close();
        }
    } catch (IOException e) {
    }
}
} else {
    return new ResponseEntity<>(HttpStatus.BAD_REQUEST);
}
return new ResponseEntity<>(HttpStatus.CREATED);
}

```

Como pode ser observado, o código guarda as informações sobre a postagem feita pelo agricultor no banco e envia uma resposta de volta, informando que a postagem foi criada. Entretanto, caso ocorra algum erro no meio do processo, é enviado *status* de erro para o usuário. A Tabela 3 lista os possíveis códigos de retorno que podem acontecer em cada fluxo do algoritmo.

Quadro 3 - Retornos do web service

Código	Descrição
201	Postagem criada
400	Falta parâmetros na requisição
404	Serviço não encontrado
500	Erro interno do servidor

Fonte: elaborado pelo autor

Cada status de retorno do elencado acima existe um motivo para ser exibido em cada fluxo. O de código 201 que significa “Postagem criada”, só será exibido se a inserção no banco for bem sucedida, enquanto o código 400 significa que faltam informações a serem enviadas para que seja possível o processamento da requisição.

No caso do código 404, o problema pode ser que o servidor esteja *offline* ou que o endereço digitado não seja igual ao especificado no desenvolvimento da API. E por fim, o código 500 acontece somente se um erro catastrófico como a corrupção das imagens na hora do armazenamento.

Na etapa de desenvolvimento do cliente web foram desenvolvidas as telas e toda a lógica de negócio por trás da rede colaborativa para pesquisadores analisarem de forma colaborativa as fotos de plantas. A Figura 6 mostra a tela de autenticação, a Figura 7 mostra a tela de registro no sistema, a Figura 8 mostra a tela de visualização das postagens e a Figura 9 mostra a visualização dos diagnósticos.

Figura 7 - Tela de autenticação no sistema



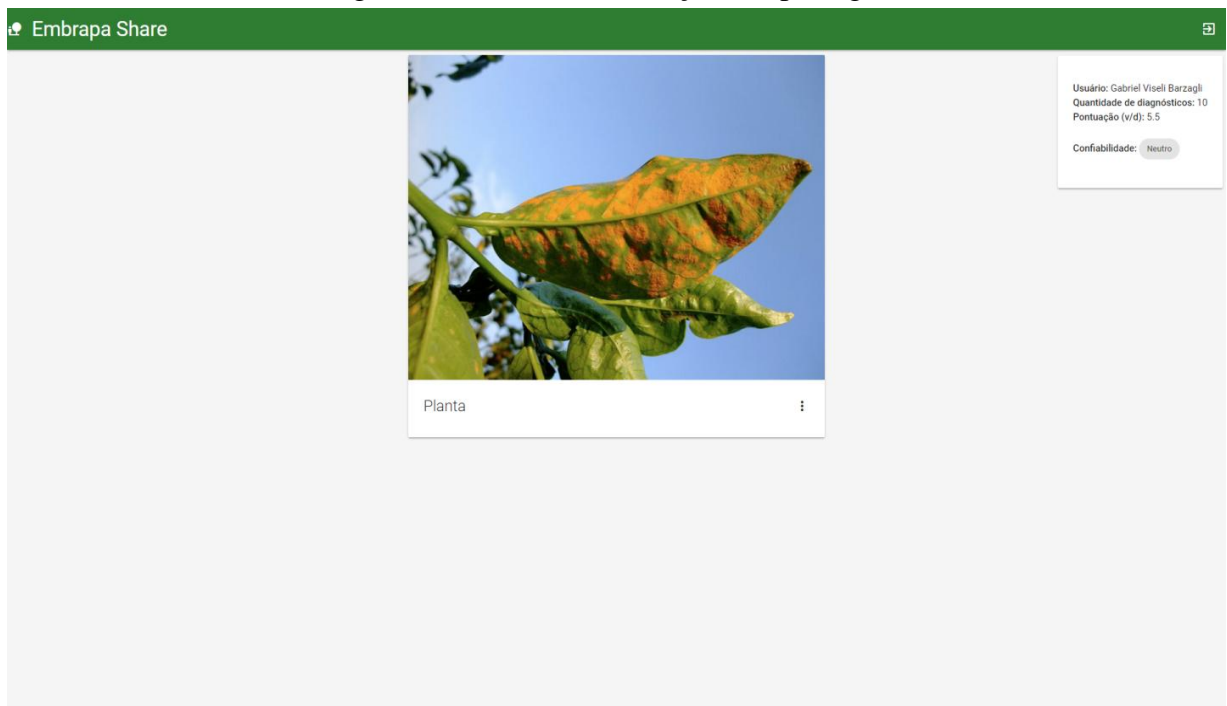
Fonte: elaborado pelo autor

Figura 8 - Tela de registro no sistema



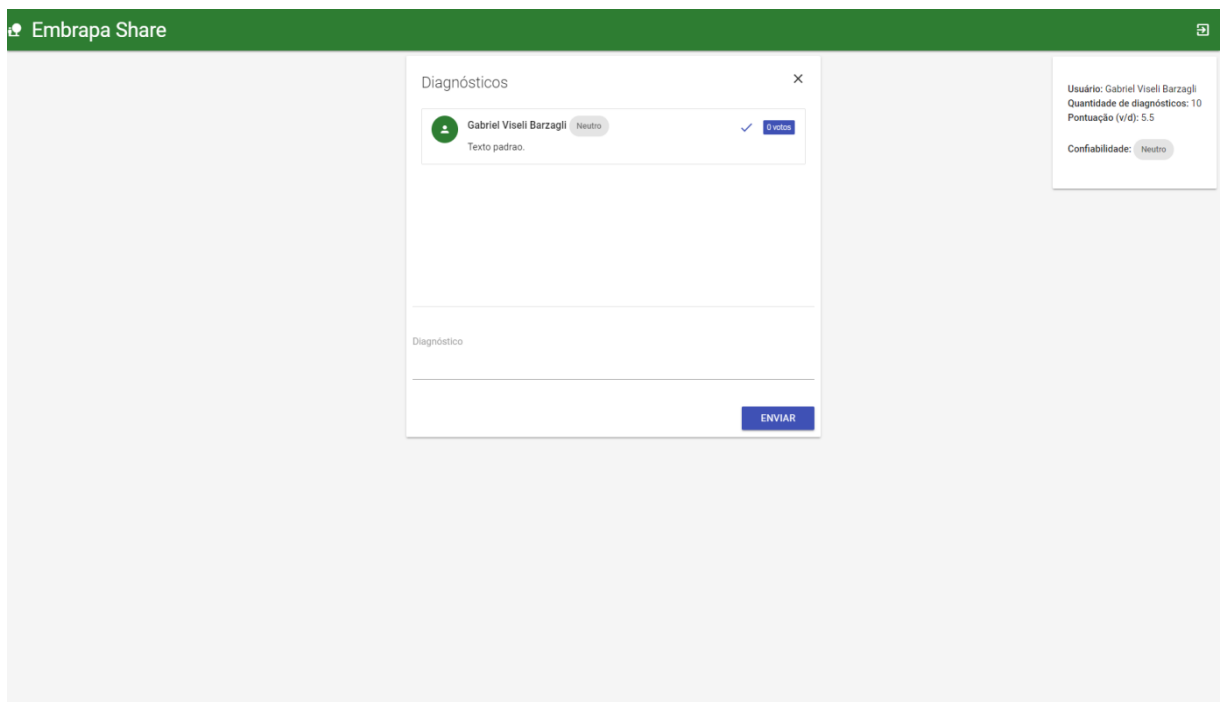
Fonte: elaborado pelo autor

Figura 9 - Tela de visualização das postagens



Fonte: elaborado pelo autor

Figura 10 - Visualização dos diagnósticos



Fonte: elaborado pelo autor

5 CONCLUSÕES

O desperdício de alimento no mundo é um problema que afeta muitas pessoas, principalmente no âmbito econômico. Esse desperdício pode ser causado pelo clima, por doenças ou por pragas. Logo, o objetivo desse trabalho foi construir um meio de diminuir a desconhecimento dos problemas que o agronegócio familiar enfrenta e diminuir custos em consultoria dos grandes produtores rurais.

O levantamento de requisitos foi executado e auxiliou no desenvolvimento do produto como um todo. O servidor de armazenamento de imagens foi criado com boas práticas de programação e uma API de upload de imagens foi criada para atender o sistema como um fornecedor de postagens de imagens para serem diagnosticadas. Ela possibilita também a integração do servidor com o aplicativo Android produzido pela Embrapa.

O método de classificação dos pesquisadores da rede colaborativa de diagnósticos faz uso de uma escala de valores baseado numa pontuação que é calculada a partir da quantidade de votos e do número de diagnósticos que o pesquisador recebeu. A nomenclatura adotada para cada classificação auxilia no engajamento dos usuários a fornecer cada vez mais diagnósticos.

A aplicação web foi desenvolvida usando boas práticas de design e funciona perfeitamente como uma rede colaborativa para que pesquisadores compartilhem conhecimento a respeito de determinados sintomas em plantas.

A solução permite ainda que seja possível a configuração dos valores usados em cada uma das classificações através de uma tabela no banco de dados relacional e, permite também que se tenha um banco de dados de imagem para um possível trabalho futuro de criação ou melhoria de algoritmos de análise forense para detecção automática de doenças em plantas.

6 TRABALHOS FUTUROS

Como trabalhos futuros pode-se elencar a melhoria do método de classificação e recompensa de usuários ao diagnosticar uma foto a fim de reter o máximo de usuários possíveis e considerar o acréscimo do voto negativo.

É necessário desenvolver o método de envio dos diagnósticos melhor ranqueados ao agricultor e uma base histórica de diagnósticos para que o agricultor possa realizar consultas sobre os problemas no passado.

Outra possibilidade é o desenvolvimento ou a melhoria de um algoritmo de análise forense de imagens baseado nas fotos coletadas por meio do sistema para identificar os sintomas nas fotos das plantas.

REFERÊNCIAS

CARR, David F. **Social collaboration for dummies**. 1. ed. Hoboken: John Wiley & Sons, 2014.

DEPARTMENT OF ECONOMIC AND SOCIAL AFFAIRS OF THE UNITED NATIONS. **World Population Prospects: The 2017 Revision**. Disponível em: <https://esa.un.org/unpd/wpp/Publications/Files/WPP2017_KeyFindings.pdf>. Acesso em: 20 set. 2017.

DOUGLAS, Korry; DOUGLAS, Susan. **PostgreSQL: a comprehensive guide to building, programming, and administering PostgreSQL databases**. SAMS publishing, 2003.

FOOD AND AGRICULTURE ORGANIZATION OF THE UNITED NATIONS. **Food Wastage Footprint: Impacts on Natural Resources: Summary Report**. FAO, 2013.

FOOD AND AGRICULTURE ORGANIZATION OF THE UNITED NATIONS. **Global agriculture towards 2050**. Disponível em: <http://www.fao.org/fileadmin/templates/wsfs/docs/Issues_papers/HLEF2050_Global_Agriculture.pdf>. Acesso em: 20 set. 2017.

IBGE. **Censo Agropecuário 2006 Brasil, Grandes Regiões e Unidades da Federação**. Rio de Janeiro: MPOG, 2006. Disponível em: <https://biblioteca.ibge.gov.br/visualizacao/periodicos/51/agro_2006.pdf>. Acesso em: 22 set. 2017.

LANDIM, Raquel. **Brasil já é o terceiro maior exportador agrícola do mundo**. Jornal Estadão, Mar, 2010.

MELLO, Roxane Lopes de. **Agricultura familiar sustentabilidade social e ambiental**. 2007.

PEIXOTO, M.; PINTO, H. S. **Desperdício de Alimentos: questões socioambientais, econômicas e regulatórias**. Brasília: Núcleo de Estudos e Pesquisas/CONLEG/ Senado, fev. 2016 (Boletim Legislativo nº 41, de 2016). Disponível em: <<http://www.senado.leg.br/estudos>>. Acesso em: 27 out. 2017.

PRESSMAN, Roger S. **Software engineering: a practitioner's approach**. 6 ed. New York: McGraw-Hill, 2005. p. 47-48.

ROSENZWEIG, C. et al. Climate Change and Extreme Weather Events; Implications for Food Production, Plant Diseases, and Pests. **Global Change and Human Health**, [S.l.], v. 2, p. 90-104, dez. 2001. Disponível em: <<https://doi.org/10.1023/A:1015086831467>>. Acesso em: 13 dez. 2017.

WAGH, Kishor; THOOL, Ravindra. A comparative study of soap vs rest web services provisioning techniques for mobile host. **Journal of Information Engineering and Applications**, [S.l.], v. 2, n. 5, p. 12-15, 2012.