

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO
GRANDE DO NORTE
CAMPUS AVANÇADO LAJES
CURSO TÉCNICO INTEGRADO EM INFORMÁTICA

LAURA CHRISTINA DE OLIVEIRA PAULO

ONDE ESTÁ A LIXEIRA?

LAJES - RN
2022

LAURA CHRISTINA DE OLIVEIRA PAULO

ONDE ESTÁ A LIXEIRA?

Relatório de Prática Profissional apresentado ao Curso Técnico Integrado em informática do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, Campus Avançado Lajes, em cumprimento às exigências legais como requisito parcial à obtenção do título de Técnico em informática

Orientador(a): Prof. Me Fernando Helton Linhares Soares

AGRADECIMENTOS

Agradeço a minha mãe, Maria Antonia de Oliveira Paulo, por sempre me apoiar e está comigo em todos os meus momentos, por conquistar comigo cada pequena vitória, e também por me levantar quando eu achei que não ia conseguir, e que me acompanhou desde do início, assim como meu pai, Francisco Raimundo Paulo e toda a minha família.

Agradeço também aos meus amigos, aqueles que conheci durante o curso, e que melhoraram meus dias durante todos esses anos de caminhada. Mas quero fazer um agradecimento especial à minha melhor amiga, Ana Sara da Silva Clementino, que esteve comigo até a última palavra escrita deste projeto, que me acalmou em todas as crises, e que me ajudou a não desistir de tudo.

E por fim, não posso deixar de agradecer aos meus professores que me doaram seus conhecimentos, para que eu conseguisse realizar este projeto, em específico ao meu orientador, Fernando Helton Linhares Soares, que me ajudou e esteve comigo na realização deste trabalho.

RESUMO

Este projeto busca abordar a problemática da poluição no município de Lajes/RN. A amenização desta problemática é de inteira importância para os moradores da cidade, pois esses são produtores e consumidores de recursos ambientais locais (ex.: pecuária e agronomia). O trabalho visa realizar um mapeamento sobre os índices de poluição e as áreas mais afetadas do município de Lajes/RN e, além disso, propor um sistema de localização de lixeiras que promova o descarte correto de materiais. Para isso, foi realizada uma pesquisa para identificar se a população tem ciência dos locais de descarte da cidade, e se estão utilizando dos mesmos. Em seguida, é proposto o desenvolvimento de um sistema que mostre ao usuário a lixeira mais próxima a ele, de acordo com a necessidade. Espera-se que os resultados deste projeto promovam a conscientização da população sobre a atual situação da poluição do município, e estimule a forma adequada de descartar os materiais rejeitados. Assim, este projeto servirá como um caminho para a obtenção de informações que levem à redução da poluição e preservação ambiental local.

Palavras-chave: Poluição; lixo; descarte; Lajes; java; spring boot;

ABSTRACT

This project seeks to address the problem of pollution in the city of Lajes/RN. The alleviation of this problem is of great importance for the residents of the city, because they are producers and consumers of local environmental resources (livestock and agronomy). This work aims to perform a mapping on the pollution indexes and the most affected areas of the municipality of Lajes/RN and, in addition, propose a dumps location system that promotes the correct disposal of materials. To this end, a survey was conducted to identify whether the population is aware of the city's disposal sites, and whether they are using them. Then, it is proposed to develop a system that shows the user the dump closest to them, according to the need. It is expected that the results of this project promote the awareness of the population about the current situation of the city's pollution, and stimulate the appropriate way to dispose of rejected materials. Thus, this project will serve as a path for obtaining information leading to the reduction of pollution and local environmental preservation.

Keyword: Pollution; Garbage; Discard; Lajes; java; spring boot;

SUMÁRIO

1 INTRODUÇÃO	7
1.1 OBJETIVO GERAL	8
1.2 JUSTIFICATIVA	8
2 DADOS GERAIS DA PESQUISA/EXTENSÃO/ESTÁGIO	9
3 FUNDAMENTAÇÃO TEÓRICA	11
3.1 O DESCARTE DE MATERIAIS SÓLIDOS	11
3.2 HTML, CSS, BOOTSTRAP	11
3.3 SPRING, SPRING BOOT, SPRING SECURITY, SPRING DATA e THYMELEAF	12
4 METODOLOGIA	13
5 ANÁLISE E PROJETO DO SISTEMA PROPOSTO	15
5.1 APLICAÇÃO DO FORMULÁRIO	15
5.2 REQUISITOS E CASOS DE USO	17
5.3 DIAGRAMA DE CLASSES	19
6 ONDE ESTÁ A LIXEIRA?	21
6.1 FUNCIONALIDADES DESENVOLVIDAS	20
6.2 IMPLEMENTAÇÃO	26
7 ANÁLISE E DISCUSSÃO DOS RESULTADOS	38
8 CONSIDERAÇÕES FINAIS	39
9 REFERÊNCIAS	40
ANEXO A – FORMULÁRIO DE IDENTIFICAÇÃO	42

1 INTRODUÇÃO

A preservação ambiental é um assunto de grande importância social. Dessa forma, é importante refletir acerca da poluição e sobre como fazer o descarte correto dos resíduos sólidos, evitando o comodismo social e a indução a uma conduta prejudicial ao meio ambiente. Esses resíduos são consequências das atividades humanas e em sua maioria podem ser reutilizados ou reciclados. Este lixo é classificado em duas categorias: o biodegradável e o não biodegradável.

O primeiro é aquele que a natureza absorve rápido, ou seja, que se degrada facilmente, sendo assim menos prejudicial. Já o segundo é basicamente o que não se degrada, esse lixo não é absorvido pela natureza facilmente, provocando um acúmulo muito grande no meio ambiente e gera consequências negativas à longo prazo. A partir do descarte incorreto de materiais, as pessoas, além de se prejudicarem, prejudicam também o lugar onde vivem.

Partindo da problemática do descarte incorreto do lixo, desenvolvemos a ideia da construção de um sistema que permita a localização de lixeiras próximas ao indivíduo, facilitando assim o descarte adequado do lixo. Através dele, o usuário poderá buscar as lixeiras de acordo com o bairro e rua em que se encontram e pelo tipo de lixo. Após selecionar a lixeira que deseja, poderá ver com mais detalhes informações sobre ela, como por exemplo referências de localidade e imagens para ajudar na sua localização. O usuário também pode cadastrar lixeiras, ou denunciá-las, caso estejam com o endereço errado. O sistema foi pensado para ter uma estética simples e de fácil compreensão, tendo em vista que qualquer pessoa pode acessá-lo.

O projeto, além de ter uma grande importância de formação acadêmica, tem a finalidade de fazer com que a comunidade que tenha acesso ao sistema se conscientize, pois facilita o encontro das lixeiras, induzindo-os ao descarte do lixo de forma correta. Além disso, a partir da propagação deste sistema, outros lugares poderão adotar esse método, contribuindo com a amenização de uma problemática que prejudica a saúde dos moradores e o meio ambiente.

1.1 OBJETIVO GERAL

Realizar a construção de um sistema que auxilie no descarte correto de materiais sólidos, na região de Lajes-RN. Além disso, realizar, também, o estudo da percepção do descarte dos materiais sólidos da população de Lajes, que será a cidade tomada por base para o desenvolvimento deste sistema.

1.2 JUSTIFICATIVA

É comum as pessoas descartarem incorretamente os materiais sólidos, muitas vezes por não entenderem sobre o assunto e não saberem a problemática que esse descarte inadequado pode ocasionar. Mas também podem apenas não saber onde descartar o lixo, e acabar descartando em qualquer lugar. O sistema desenvolvido serve para conscientizar as pessoas e ajudá-las a encontrarem um ponto de descarte mais próximo de si, fazendo com que a população seja induzida à prática do descarte correto, diminuindo a poluição que os descartes incorretos de materiais sólidos ocasionam.

2 DADOS GERAIS DA EXTENSÃO

Título do projeto: Onde está a lixeira

Período de realização: 10/10/2020 a 31/10/2021

Total de horas: 340 horas.

Orientador: Fernando Helton Linhares Soares

Função: Aluno Voluntário

Formação profissional: Técnico em informática

Abaixo seguem as atividades desenvolvidas ao longo do período de realização do projeto destacando a carga horária empenhada e os resultados alcançados em cada uma das atividades.

Quadro 1 – Síntese das Atividades do Aluno no Projeto.

CARGA HORÁRIA	ATIVIDADES DESENVOLVIDAS	RESULTADOS ALCANÇADOS
30	Levantamento de Requisitos	Todos os requisitos funcionais do portal foram levantados e o diagrama de caso de uso foi desenvolvido
60	Projeto de arquitetura do sistema	Desenvolvimento do diagrama de classe e do diagrama de sequência;
400	Implementação do sistema	Finalização da implementação do portal proposto com todos os requisitos funcionais levantados anteriormente ;
30	Escrita do relatório de atividades	Relatório final concluído
30	Reuniões do acompanhamento do projeto	As reuniões ocorriam nas salas da própria instituição, mas, devido a pandemia, passaram a ser realizadas

		por vídeo chamadas através do Google meets.
--	--	---

Fonte: autoria própria (2020).

3 FUNDAMENTAÇÃO TEÓRICA

3.1 O DESCARTE DE MATERIAIS SÓLIDOS:

Segundo a Política Nacional de Resíduos Sólidos(PNRS), resíduos sólidos são definidos como “todo material, substância, objeto ou bem descartado resultante de atividades humanas em sociedade”. Logo, podemos dizer que são partes de produção e bens de consumo que após a sua utilização, são descartados. A maioria desses resíduos são produzidos em grandes centros urbanos, principalmente em residências pessoais, escolas, hospitais, indústrias e outros meios de grande porte, localizados nas grandes cidades. É importante se preocupar como está acontecendo esse descarte de lixo, pois, se feito de maneira errada pode mais prejudicar do que ajudar. (ProteGEEr, 2022)

Alguns materiais são altamente prejudiciais para o meio ambiente. Muitos, na maioria das vezes, demoram anos para se decompor. Isso acaba gerando, principalmente em aterros sanitários, uma grande devastação causada pela poluição. Muitos desses resíduos poderiam ser reciclados, mas são apenas deixados de lados gerando mais e mais entulho, sem nenhuma resolução (IPEA, 2020)

3.2 HTML, CSS, BOOTSTRAP

O HTML (Linguagem de Marcação de Hipertexto), é uma linguagem utilizada para a construção de páginas web. Além de ser de fácil entendimento, nos proporciona realizar o desenvolvimento e estruturação de conteúdos que serão postos nas respectivas páginas. Os padrões HTML são gerenciados pela W3C (*World Wide Web*) e atualmente encontra-se na versão 5, que foi lançada em 2014. (W3C, 2021)

O CSS (Style Sheets), identifica-se como folhas de estilo em cascata que permite, ao desenvolvedor, exibição e ação da página HTML. Sua particularidade é definir como os elementos que estruturam uma página web serão exibidos. Desta forma, é através do CSS que podemos desenvolver, páginas “dinâmicas” e interativas, com funcionalidades que vão além da utilização apenas do HTML.(W3C, 2021)

3.3 SPRING, SPRING BOOT, SPRING SECURITY, SPRING DATA e THYMELEAF

O Spring é um framework, que é utilizado principalmente em sistemas de linguagem java, que tem como objetivo, facilitar o desenvolvimento das aplicações. Com a sua concepção de inversão de controle e injeção de dependências, podemos ter acesso a uma gama de funcionalidades, que são adicionadas a partir de bibliotecas. Abaixo temos uma descrição das que foram utilizadas e suas principais funções.(Spring, 2021)

O Spring Boot é usado para configurar o Spring e os ambientes de desenvolvimento e execução, trazendo uma grande facilidade e velocidade na configuração inicial das aplicações. Graças a ele é possível utilizar outras bibliotecas do Spring com pouca ou nenhuma configuração adicional. Uma delas é o Spring Security que traz recursos de segurança para a sua aplicação, possui uma fácil configuração e diversas funcionalidades como gerenciamento de login, logout, cargos e autorização. Outra biblioteca bastante utilizada também é o Spring Data, que facilita o acesso e a conexão à ferramentas e tecnologias de armazenamento de dados, permitindo a interação entre a sua aplicação e um banco de dados de maneira simples, tornando o ambiente de desenvolvimento mais produtivo. (Spring, 2021)

O Thymeleaf é um *template engine*, utilizado em sistemas com linguagem Java, que tem como função, criar páginas HTML dinâmicas, fazendo uma conexão entre o frontend e o backend, dando inúmeras possibilidades para o desenvolvedor. (Thymeleaf, 2018)

4 METODOLOGIA

O método utilizado para o desenvolvimento do projeto foi o pesquisa-ação, que segundo Thiollent (2009), em Metodologia da Pesquisa-ação, é um método que utiliza a pesquisa social com base empírica que é elaborada e realizada em junção com uma ação ou com a resolução de um problema coletivo no qual os pesquisadores e os participantes representativos da situação ou do problema estão envolvidos de modo cooperativo ou participativo.

Sendo assim, foi realizada uma pesquisa para entender a real importância e a necessidade da criação do sistema proposto. Para conseguir atingir um maior número de pessoas de maneira eficiente, foi desenvolvido um questionário utilizando o *Google Forms*, por se tratar de um serviço gratuito gerador de formulários *online* de fácil acesso, bom retorno de dados e grande segurança de informações. As perguntas foram direcionadas aos alunos do campus avançado Lajes, prioritariamente aos moradores do município de Lajes, uma vez que esta é a localidade objeto de estudo do trabalho, sendo utilizado uma linguagem mais popular para garantir a compreensão do público alvo.

Após a obtenção dos resultados, analisamos as respostas e concluímos que a maioria dos alunos do campus Lajes e residentes do município desconheciam a localização das lixeiras da cidade, especialmente as lixeiras destinadas à coleta seletiva. Portanto, direcionamos o sistema para as reais necessidades da população, tornando a visualização da localização das lixeiras a principal funcionalidade proposta.

Com os dados do questionário analisados foi dado início ao planejamento do sistema, criando os diagramas de caso de uso e de classe, visando uma melhor organização e entendimento das funcionalidades a serem desenvolvidas na execução do projeto. O sistema foi implementado com a linguagem Java, utilizando para tal o *framework Spring Boot*, junto com a *template engine thymeleaf*. Essas ferramentas foram escolhidas devido sua praticidade, eficiência e adequação aos requisitos do sistema proposto. O Eclipse foi a IDE (*Integrated Development Environment*) para desenvolvimento do sistema devido a sua grande variedade de ferramentas, funcionalidades e sua popularidade no mercado de desenvolvimento Java. Para gerenciar o banco de dados da aplicação, o MySQL foi escolhido pela

sua praticidade e eficiência no gerenciamento de dados. Também foi utilizado o *Spring Security*, devido seus recursos avançados e de simples configuração, responsável pela autenticação e autorização dos usuários da aplicação. No *front-end* foi utilizado a linguagem de marcação HTML juntamente com o bootstrap, que é um *framework* CSS de código aberto que disponibiliza componentes prontos deixando as páginas do site visualmente mais harmônicas e bem vistas.

Para se ter uma melhor segurança no armazenamento código-fonte do sistema foi utilizado o GitHub, plataforma que armazena código fonte e arquivos, que além de hospedar a aplicação, também é possível ambos integrantes do grupo contribuírem ao projeto de forma simultânea, estando cada um em sua cidade.

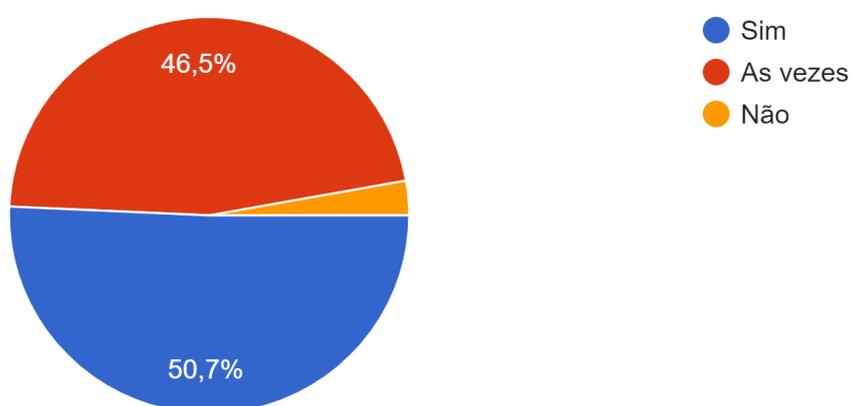
Utilizamos o padrão MVC devido sua ampla utilização nos projetos desenvolvido para *web*, usado para organizar a aplicação, dividindo o projeto em camadas bem definidas. As camadas são três: o *Model*, responsável pela manipulação de dados da aplicação; *View*, onde acontece a interação do usuário com o sistema, lá ficam as interfaces que mostram as informações do *model* para o usuário; e por último a camada *Controller*, responsável por controlar e ligar o *model* a *view*.

5 ANÁLISE E PROJETO DO SISTEMA PROPOSTO

5.1 APLICAÇÃO DO FORMULÁRIO

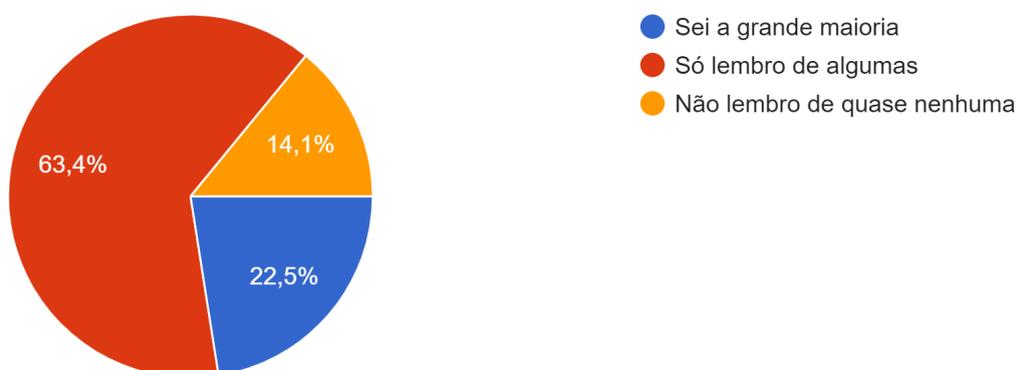
Para concretizar a ideia do projeto, e fazer um levantamento do entendimento dos possíveis usuários do sistema proposto, foi aplicado um formulário de forma online à alguns alunos do IFRN Campus Avançado Lajes-RN. Abaixo podemos ver as perguntas e imagens dos gráficos com as respostas obtidas:

Você descarta o seu lixo de maneira minimamente correta(joga o lixo no lixo) ?



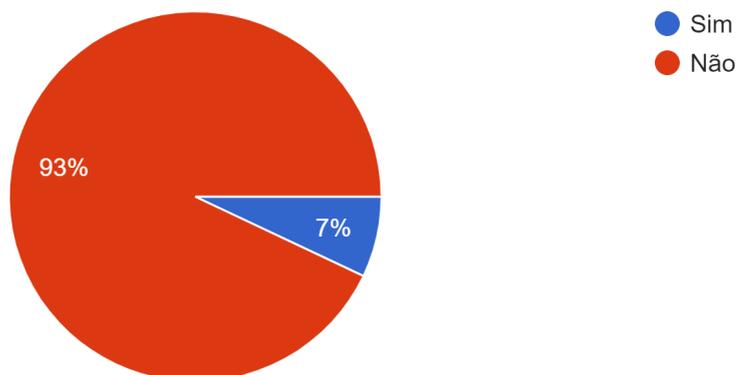
(Figura 1 - Referente a pergunta "Você descarta o seu lixo de maneira minimamente correta?")

Você sabe onde ficam localizadas as lixeiras dos ambientes URBANOS que costuma frequentar ?



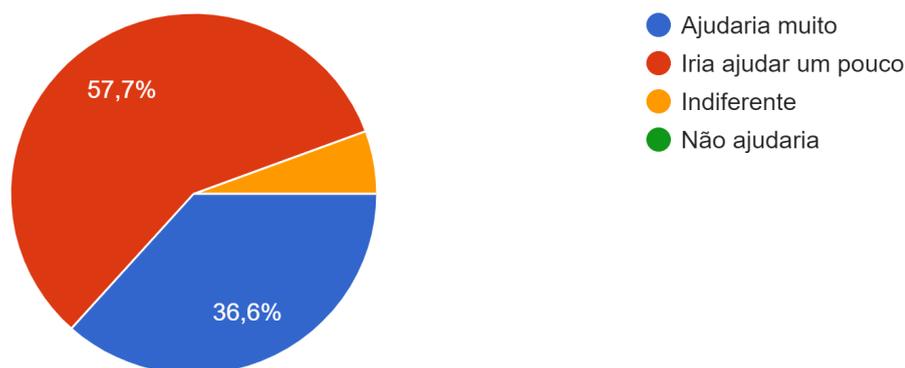
(Figura 2 - Referente a pergunta "Você sabe onde ficam localizadas as lixeiras dos ambientes URBANOS que costuma frequentar ?")

Você sabe onde ficam as lixeiras de coleta seletiva da cidade ?



(Figura 3 - Referente a pergunta "Você sabe onde ficam as lixeiras de coleta seletiva da cidade ?")

Você acha que um app/site que mostra as lixeiras mais próximas a você ajudaria ?



(Figura 4 -Referente a pergunta "Você acha que um app/site que mostra as lixeiras mais próximas a você ajudaria ?")

Diante das respostas obtidas, apenas 50,7% das pessoas descartam o lixo minimamente correto, como podemos ver na Figura 1, resultado preocupante tendo em vista que 46,5% dos questionados não fazem o básico. Ao serem também questionados sobre a localidade das lixeiras em ambientes onde eles costumam frequentar apenas 22,5% disseram que sabiam onde encontrar a maioria das lixeiras(Figura 2). Já na Figura 3 ao serem interrogados se sabiam onde ficavam localizadas as lixeiras seletivas da cidade, a maioria (93%) responderam que não sabiam. E por último foi questionado sobre a construção de site que mostrasse a

localidade das lixeiras mais próximas de si, e 36,6% disse que ajudaria muito e 57,7% disse que ajudaria um pouco (Figura 4). Portanto, analisamos e chegamos a conclusão que é benéfico o desenvolvimento de um projeto com a finalidade de conscientizar e ajudar na localidade das lixeiras.

5.2 REQUISITOS E CASOS DE USO

A primeira etapa a ser realizada para a construção de um sistema é o levantamento dos requisitos, que serve como guia no decorrer do desenvolvimento. Dessa forma, foi elaborado um conjunto de funcionalidades, descritas como requisitos, que o sistema proposto deve atender. Abaixo podemos ver essas funcionalidades e suas respectivas descrições: :

- **RF01:** O sistema deve permitir o cadastro de lixeiras:
 - Pré-condição: para realizar essa funcionalidade o usuário já deve estar cadastrado
 - Pós-condição: A nova lixeira deve estar armazenada no banco de dados
 - Autores: Usuário
- **RF02:** O sistema deve permitir que o usuário visualize as lixeiras próximas a sua localização
 - Pré-condições: A lixeira deve estar cadastrada para poder ser visualizada
- **RF03:** O sistema deve permitir o cadastro do usuário
 - Pré-condições: para realizar essas funcionalidade o usuário deve preencher os campos de informação
 - Pós-condições: o novo usuário deve estar armazenado no banco de dados
- **RF04:** O sistema deve permitir o login do usuário
 - Pré-condições: para realizar essa funcionalidade o usuário deve estar cadastrado
 - Pós-Condições: O usuário tem que estar armazenado no banco de dados

Após a listagem dos requisitos, iniciamos a criação do diagrama de casos de uso, que tem a finalidade de descrever as principais funcionalidade do sistemas na visão do usuário, também descreve a interação dessas funcionalidades com os atores do mesmo sistema. Na Figura 5 podemos visualizar as funções que podem ser realizadas e o que cada ator pode fazer.

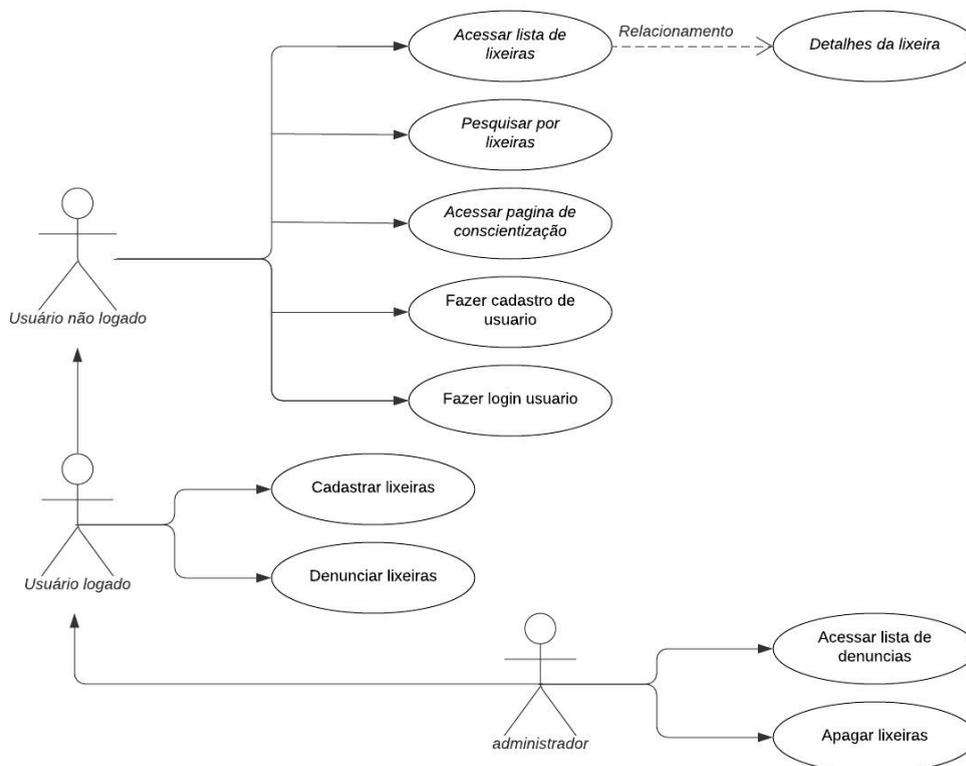


Figura 5 - Diagrama de Casos de uso

Como podemos ver no diagrama o *Usuário não logado* é o usuário que pode navegar pelo site sem precisar estar logado, ele tem acesso a lista das lixeiras, podendo pesquisar e achar a lixeira mais próxima de si e ver detalhes sobre ela, pode também visualizar a página de conscientização, se cadastrar ou fazer login. Já o *Usuário logado* tem acessos a todas as funções que o *Usuário não logado* tem, além de, quando logado no site ele pode cadastrar lixeiras, assim como também pode denunciá-las. E por fim o *Administrador* que é o responsável pelas denúncias das lixeiras, é o único que pode ver a lista das denúncias, podendo também deletar a lixeira do banco de dados após três denúncias. Somente os desenvolvedores do site têm acesso a essa conta.

5.3 DIAGRAMA DE CLASSES

Na Figura 6 podemos visualizar o diagrama de classe usado para documentar a estrutura do sistema. Esse diagrama descreve a estrutura de classes que o projeto terá e os relacionamentos entre essas classes.

Para a criação do diagrama foram feitos estudos para o desenvolvimento do mesmo. A classe na programação é onde é armazenado características e comportamentos de um objeto, os atributos é onde definimos essas características, ou seja, são as propriedades desse objeto, conhecido também como variáveis. E os métodos que são as ação ou procedimentos desse objeto, podendo também ser utilizado para a interação com outros. Abaixo veremos como foi aplicado cada conceito na criação do diagrama.

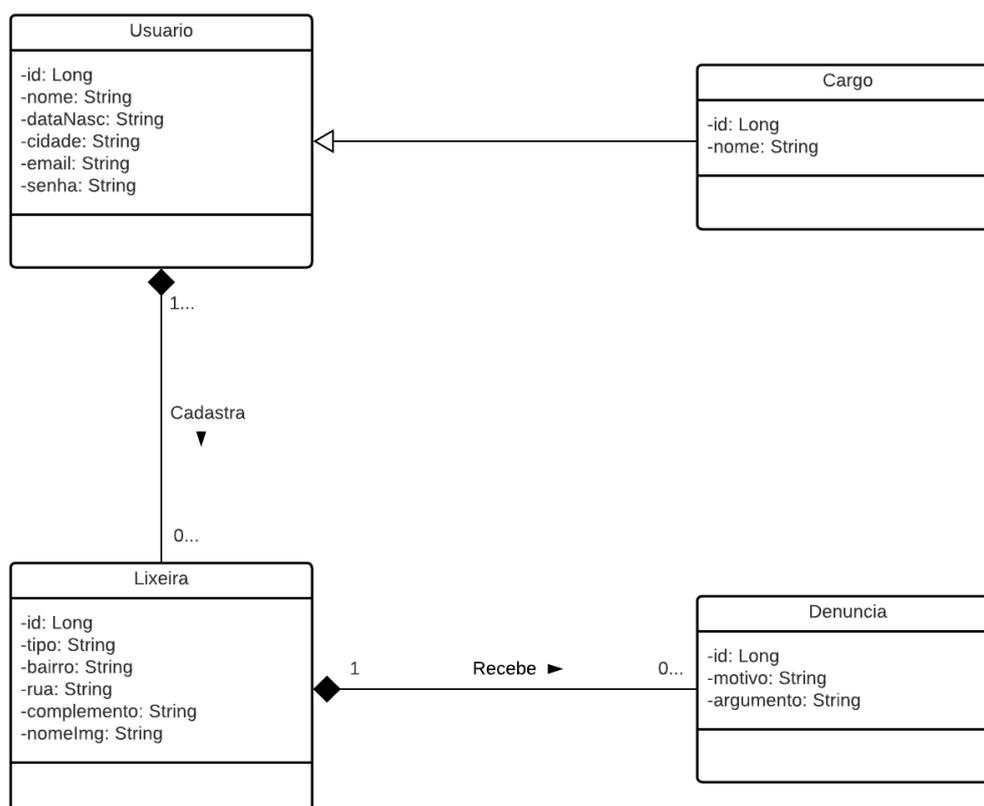


Figura 6 - Diagrama de Classe

A classe *Usuário* é responsável por definir os dados dos usuários que irão se cadastrar no site, esta classe tem como atributos o id, nome, dataNasc (referente a data de nascimento), cidade, email e senha. A classe *Cargo* foi desenvolvida apenas para os administradores do site, onde é possível ter acesso a lista de

denúncias, e a permissão para a exclusão das lixeiras, a classe tem como atributos o id e nome. A classe *Lixeira*, é a classe onde é definido os dados para o cadastro das lixeiras no banco de dados, ela depende da classe *Usuario*, ou seja as lixeiras só podem ser cadastradas pelo usuário que está no sistema, um usuário pode ter várias lixeiras cadastradas, a classe tem como atributos, id, nome, tipo, bairro, rua, complementos e nomeImg (referente a imagem da lixeira) . e Por fim a classe *Denúncia* que foi criada para denunciar lixeiras que não estão com endereço correto, ou algo do tipo, e assim ser avaliado pelos ADM, que podem excluir ou não a lixeira do sistema, é uma que depende da classe *Lixeira*, as denúncias só irão existir se a lixeira também existir, por isso a dependência, e uma lixeira pode receber várias denúncias, os atributos são id, motivo e argumento.

6 ONDE ESTÁ A LIXEIRA?

O portal Onde está a lixeira, foi desenvolvido para facilitar a localização de lixeiras na cidade de Lajes/RN, com o intuito de promover o descarte consciente de resíduos sólidos, visando uma cidade mais sustentável. Possuindo uma interface de fácil entendimento e manuseio, assim como funções simples e diretas, utiliza uma linguagem simples que traz ainda mais facilidade ao navegar. A intenção é que qualquer pessoa possa acessar e buscar um local de descarte, independente de seu nível de conhecimento.

6.1 FUNCIONALIDADES DESENVOLVIDAS

Ao acessar o sistema, a primeira coisa que temos acesso é uma lista com lixeiras, cada uma delas mostra qual o tipo de descarte, bairro, rua e complemento, para facilitar a localização. Caso o sistema esteja sendo acessado por um administrador também aparecerá o id da lixeira.

Através do botão “Detalhes” temos acesso a mais informações sobre a mesma, além de uma foto para ser usada de referência do local. Acima da lista(Figura 7 - Página inicial) podemos encontrar filtros para facilitar a busca, pode-se utilizar três tipos de buscas, rua, bairro e tipo. Clicando no botão “Limpar” o filtro será reiniciado e todas as lixeiras irão aparecer novamente.

Ainda na tela inicial podemos encontrar um botão chamado “Adicionar lixeira”, localizado acima dos filtros, ao acessar, o usuário é redirecionado a tela de cadastro de lixeira. Além disso, na barra mais acima, encontramos o cabeçalho do site, ele está presente em todas as outras telas, sempre que clicar na logo que fica à esquerda, o usuário será redirecionado para o menu principal. Além disso tem acesso a uma página de conscientização e orientação sobre descarte adequado e consciente, para acessá-la, basta clicar no botão “Conciete-se”.

Ainda no cabeçalho encontramos os botões “cadastre-se” que leva o usuário para uma tela de cadastro onde ele poderá criar uma conta no site, e o botão “entrar” que leva até a tela de login onde é possível acessar sua conta caso tenha uma. Porém após efetuar o login esses botões são substituídos, aparecerá um botão “sair” que permite sair de sua conta. Além disso, se a conta que estiver

acessando for uma conta de administrador do sistema, também ficará disponível um botão “Lixeiras com denúncias” que permite o administrador acessar uma lista com as lixeiras que foram denunciadas. Para melhor visualizar essas mudanças no cabeçalho veja as figuras.

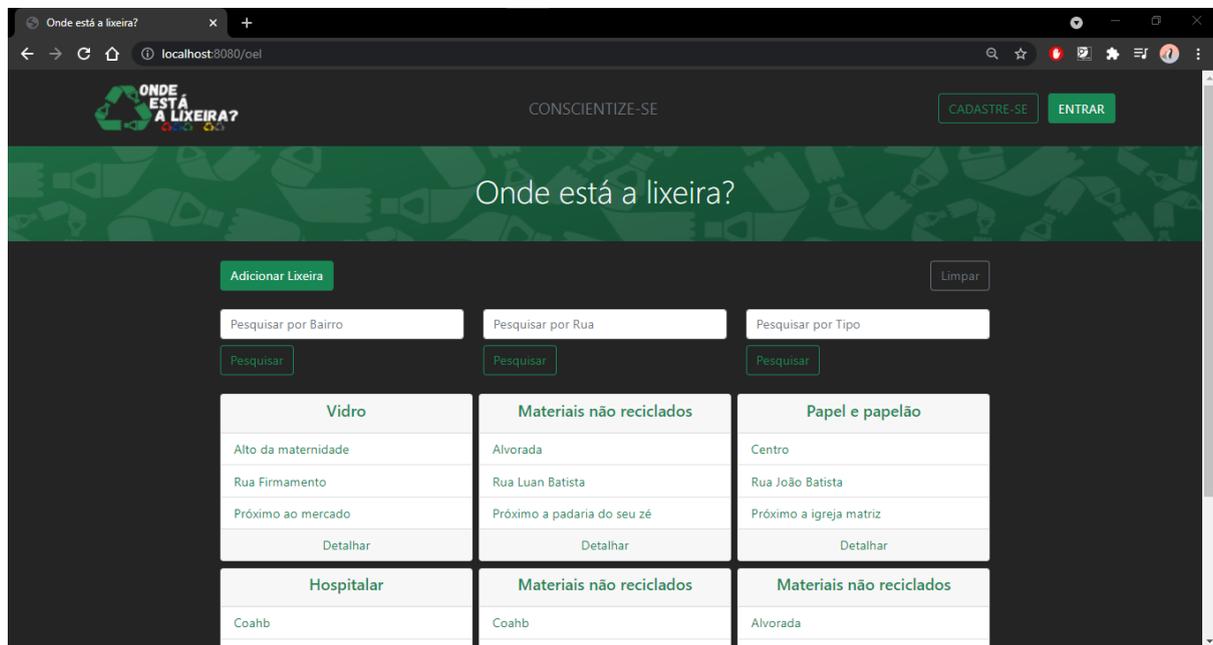


Figura 7 - Página inicial

Após acessar um lixeira específica, clicando em "detalhes", o sistema leva o usuário para uma nova página(Figura 8 - Detalhes da lixeira) onde se tem acesso a mais informações sobre a lixeira, o principal ponto dessa funcionalidade é a aparição de uma imagem do local, que é de grande auxílio para uma melhor localização. Caso o usuário verifique que a lixeira não existe mais, ou não se encontra com as informações que estão sendo mostradas, é possível realizar uma denuncia sobre ela clicando no botão “solicitar exclusão”, que se encontra no canto inferior direito. Caso esteja acessando o sistema através de uma conta de administrador, também aparecerá uma tabela abaixo das informações, contendo todas as denúncias feitas para aquela lixeira, outra funcionalidade acrescentada ao cargo, é botão de excluir, que delata a lixeira do sistema.

ONDE ESTÁ A LIXEIRA? CONSCIENTIZE-SE LIXEIRA COM DENÚNCIAS SAIR

Tipo de lixo: Materiais não reciclados
Bairro: Coahb
Rua: Rua Fernando Caxias
Complemento: Próximo ao mercado são José

Solicitar exclusão apagar

id	Motivo	Argumento
3	Lixeira não corresponde ao tipo que está na descrição	a lixeira é não mais de lixo reciclável
4	Lixeira não corresponde ao tipo que está na descrição	a lixeira é não mais desse tipo

Figura 8 - Detalhes da lixeira

Caso seja necessário solicitar a exclusão da lixeira, após verificar que suas informações realmente estão equivocadas, o usuário poderá acessar um formulário de denúncia, que possui dois campos, o primeiro requer que seja informado o motivo da realização da denúncia, e utiliza opções pré definidas pelo sistema, já o outro campo, pede argumentos para exclusão de preferência que situações atual seja descrita. Após a realização da denúncia os administradores do sistema irão avaliar se esta denúncia é recorrente e irão tomar as devidas providências.

ONDE ESTÁ A LIXEIRA? CONSCIENTIZE-SE SAIR

Denunciar Lixeira

Qual o motivo da denuncia?:
 Escolha um motivo...

Argumente:
 Ex: a lixeira está posicionada no outro lado da rua, em frente a casa x

Limpar Denunciar

Figura 9 - Formulário de denúncias

Caso um usuário tenha conhecimento de um local de descarte seletivo de qualquer tipo, e até mesmo de uma lixeira comum, ele mesmo pode cadastrar uma nova lixeira. Para isso basta acessar o formulário de cadastro de lixeira, através do menu iniciar clicando em “Adicionar Lixeira”. Após o acesso, é necessário preencher cinco campos, os dois primeiros são referentes ao tipo de lixeira e ao bairro, ambos possuem opções pré definidas que abrangem todas as opções quando se trata do contexto da cidade de Lajes. As próximas duas opções são campos de texto para que seja informado a rua e o complemento. Já o último é para que seja anexado uma foto do local para que haja uma melhor visualização do local.



Figura 10 - Formulário de cadastro de lixeira

Algumas das funcionalidades do sistema só ficam disponíveis caso o usuário esteja cadastrado, para realizá-lo, basta clicar no botão “cadastre “ no canto superior direito. Após isso, ele será redirecionado para o formulário de cadastro, onde terá que preencher cinco campos de texto, que são: nome, data de nascimento, cidade, email e senha. Após isso, basta clicar no botão “entrar”, que também fica no canto superior direito, então, será redirecionado para tela de *login*, onde encontrará os campos: email e senha, basta preencher com os mesmo dados utilizados no cadastro.

The screenshot shows the 'Cadastrar Usuário' (Register User) form on a website. The header includes the logo 'ONDE ESTÁ A LIXEIRA?' on the left, the text 'CONSCIENTIZE-SE' in the center, and buttons for 'CADASTRE-SE' and 'ENTRAR' on the right. The form itself is set against a dark green background with a pattern of recycling symbols. It contains the following fields: 'Nome:' with an example 'Ex: Maria Laura da Silva'; 'Data de Nascimento:' with an example 'Ex: 24/07/2000'; 'Cidade:' with an example 'Ex: Lajes-RN'; 'Email:' with an example 'Ex: laura@gmail.com'; and 'Senha:' with an example 'Ex: laurinha@123'. At the bottom of the form are two buttons: 'Limpar' (Clear) and 'Cadastrar' (Register).

Figura 11 - Formulário de cadastro de usuário

The screenshot shows the 'Login' page on the same website. The header is identical to the registration page. The main content is a white login box centered on the green background. The box is titled 'Login' and contains two input fields: 'Email:' and 'Senha:'. Below these fields are two buttons: 'Limpar' (Clear) and 'Entrar' (Login). At the bottom of the box, there is a link: 'Não é cadastrado? [Cadastre-se](#)'.

Figura 12 - Tela de acesso

Ao clicar na palavra conscientize-se, que fica no centro do cabeçalho, o usuário será redirecionado para uma página de informações referente ao descarte seletivo. A página apresenta a cor referente a cada tipo de lixo, fala como descartar cada um deles e ainda dá dicas gerais de descarte.



Figura 13 - Conscientize-se

6.2 IMPLEMENTAÇÃO

Para a implementação das funcionalidades a qual fiquei responsável, foi necessário usar recursos que já foram mencionados anteriormente na metodologia, como spring boot, bootstrap, HTML, Thymeleaf.

Inicialmente foi criado um projeto em spring, usado para desenvolver e configurar de forma mais fácil o desenvolvimento de um framework, o projeto foi baixado já com todas as dependências necessárias para dar início ao desenvolvimento da aplicação. Após criar e baixar o spring e configurar com as dependências necessárias, foi iniciada a implementação do projeto, como podemos ver abaixo, na figura 14, foi feita uma classe de entrada que é a partir dela que Spring Boot irá inicializar todos os beans necessários. A própria configuração do projeto entende que se trata de uma classe de entrada devido a anotação `@SpringBootApplication` e também por possuir o método `main()`. Na linha 10 é usado `SpringApplication.run` para a inicialização da aplicação com Spring Boot.

```
OelApplication.java
1 package pi.projeto.oel;
2
3 import org.springframework.boot.SpringApplication;
4
5
6 @SpringBootApplication
7 public class OelApplication {
8
9     public static void main(String[] args) {
10         SpringApplication.run(OelApplication.class, args);
11         System.out.println("Servidor iniciado");
12     }
13
14 }
```

Figura 14 - Classe de entrada

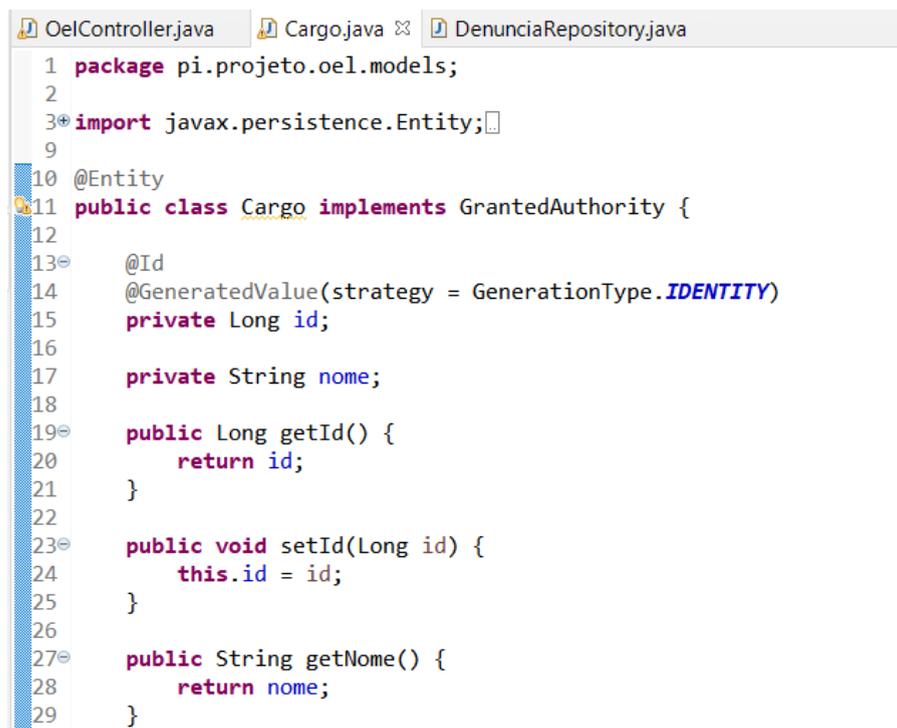
Após a execução da classe de entrada, foi criado um pacote com o nome `pi.projeto.oel.controller`, onde iremos criar a classe controller, responsável por gerenciar as requisições do projeto. No pacote criamos a classe *IndexController*, através da anotação `@Controller` o programa já identifica que a classe é um Controller no padrão MVC, e na linha 9 usamos a anotação `@RequestMapping` para mapear a página, e na linha 12 usamos o `return "redirect:/oel"` pois, iremos redirecionar a página para outra, na qual irá conter todos os métodos, necessários para o desenvolvimento do projeto.

```
OelApplication.java | IndexController.java
1 package pi.projeto.oel.controller;
2
3 import org.springframework.stereotype.Controller;
4
5
6 @Controller
7 public class IndexController {
8
9     @RequestMapping("/")
10    public String inicio() {
11        System.out.println("Chamou o método index");
12        return "redirect:/oel";
13    }
14
15 }
```

Figura 15 - Index Controller

Já com os pacotes de inicialização e controle do projeto criados, foi feito

também o pacote model, com o nome pi.projeto.oel.models, nesse pacote estão incluso todos os modelos de classe que foram utilizados no projeto, no caso temos as classes Cargo, Denúncia, Lixeira e Usuário. como podemos ver abaixo nas figuras elas seguem o mesmo padrão de criação, o que muda são os objetos e os nomes a qual está usando para identificá-la, também é necessário gerar os *Getters* e *setters* das classes, com finalidade de proteger os dados, pois é a partir das classes que irão ser armazenados os dados da aplicação. É usado a anotação *@Entity* para informar ao programa que a classe também é uma entidade, e a partir disso o próprio programa irá estabelecer ligações entre as entidades e uma tabela com o mesmo nome no banco de dados. As anotações *@Id* usada para identificação daquela variável como única, usamos também juntamente a anotação *@GeneratedValue* para gerar esse número de identificação automaticamente pelo banco de dados.



```
OelController.java Cargo.java DenunciaRepository.java
1 package pi.projeto.oel.models;
2
3 import javax.persistence.Entity;
9
10 @Entity
11 public class Cargo implements GrantedAuthority {
12
13     @Id
14     @GeneratedValue(strategy = GenerationType.IDENTITY)
15     private Long id;
16
17     private String nome;
18
19     public Long getId() {
20         return id;
21     }
22
23     public void setId(Long id) {
24         this.id = id;
25     }
26
27     public String getNome() {
28         return nome;
29 }
```

Figura 16 - Classe Cargo

Nas classes abaixo foi utilizada a seguinte anotação, para a validação e verificação dos objetos, no caso foi usada *@NotBlank*, que serve para verificar se objeto é nulo

```

OelController.java DenunciaRepository.java Cargo.java Denuncia.java
1 package pi.projeto.oel.models;
2
3 import javax.persistence.Entity;
9
10 @Entity
11 public class Denuncia {
12
13     @Id
14     @GeneratedValue(strategy = GenerationType.IDENTITY)
15     public Long id;
16
17     @NotBlank
18     public String motivo;
19     @NotBlank
20     public String argumento;
21
22     @ManyToOne
23     private Lixeira lixeira;
24
25     public Long getId() {
26         return id;
27     }
28
29     public void setId(Long id) {
30         this.id = id;
31     }

```

Figura 17 - Classe Denúncia

```

OelController.java DenunciaRepository.java Cargo.java Denuncia.java Lixeira.java
1 package pi.projeto.oel.models;
2
3
4 import javax.persistence.Entity;
9
10
11
12 @Entity
13 public class Lixeira {
14
15     @Id
16     @GeneratedValue(strategy = GenerationType.IDENTITY)
17     private Long id;
18
19     @NotBlank
20     private String tipo;
21
22     @NotBlank
23     private String bairro;
24
25     @NotBlank
26     private String rua;
27
28     @NotBlank
29     private String complemento;
30
31     private String nomeImg;
32

```

Figura 18 - Classe Lixeira

```
OelController.java DenunciaRepository.java Cargo.java Denuncia.java Lixeira.java Usuario.java x
1 package pi.projeto.oel.models;
2
3 import java.util.Collection;
17
18 @Entity
19 public class Usuario implements UserDetails{
20
21     @Id
22     @GeneratedValue(strategy = GenerationType.IDENTITY)
23     private Long id;
24
25     @NotBlank
26     private String nome;
27
28     @NotBlank
29     private String dataNasc;
30
31     @NotBlank
32     private String cidade;
33
34     @NotBlank
35     private String email;
36
37     @NotBlank
38     private String senha;
--
```

Figura 19 - Classe Usuario

A partir da criação das classes, foi dado início a criação das páginas HTML responsáveis pelo cadastro das lixeiras e dos usuários, mas antes no pacote `pi.projeto.oel.controller` foi criado um método responsável pelo cadastro do usuário, e outro para o cadastro da lixeira. A página de controle é responsável por redirecionar a interface do formulário, e para ter o controle de que os dados cadastrados no banco de dados foram salvos com sucesso. É usada a anotação `@PostMapping` para mapear o método que vai ser manipulado, e também responsável por chamar a página HTML do cadastro. Quando os dados são inseridos para a realização do cadastro e em seguida é necessário clicar no botão “salvar” para finalizar o cadastro e salvar as informações no banco de dados, como é visto na linha 77 e na linha 102 usamos `.save()` para encaminhar a informações para o repositório, onde será salvo todos os dados.

```

65Ⓞ @PostMapping("/usuario")
66 public String cadastrarUsuario(@Valid Usuario usuario, BindingResult result) {
67     System.out.println("daleee " + usuario);
68
69     if (result.hasErrors()) {
70
71         System.out.println("erro");
72         return formUsuario(usuario);
73     }
74
75     System.out.println(usuario);
76     usuario.setSenha(new BCryptPasswordEncoder().encode(usuario.getSenha()));
77     ur.save(usuario);
78
79     return "/oel/cadastroUsuario";
80 }

82Ⓞ @PostMapping("/lixeria")
83 public String cadastrarLixeira(@Valid Lixeira lixeira, BindingResult result,
84     @RequestParam("file") MultipartFile arquivo) {
85
86     if (result.hasErrors()) {
87         System.out.println("erro");
88         return formLixeira(lixeria);
89     }
90
91     System.out.println(lixeria);
92     lr.save(lixeria);
93
94     try {
95         if (!arquivo.isEmpty()) {
96             byte[] bytes = arquivo.getBytes();
97             Path caminho = Paths
98                 .get(caminhoImg + String.valueOf(lixeria.getId()) + "-" + arquivo.getOriginalFilename());
99             Files.write(caminho, bytes);
100
101             lixeira.setNomeImg(String.valueOf(lixeria.getId()) + "-" + arquivo.getOriginalFilename());
102             lr.save(lixeria);
103         }
104     } catch (IOException e) {
105         e.printStackTrace();
106     }
107
108     return "/oel/cadastroLixeira";
109 }
110 }

```

Figura 20 e 21 - Controller

Já com o *controller* feito, foi criada a página html com o formulário de cadastro chamada cadastroLixeira.html, para a criação desta página, assim como as outras páginas html. foram utilizados recursos já prontos do BootStrap. Nesta página irá conter o formulário, com os espaços de preenchimento dos dados, e botões com finalidade de adicionar uma nova lixeira ao banco de dados, e de limpar as caixas de respostas caso seja necessário. Na linha 31 estamos fazendo referência ao *controller* e a classe que vamos utilizar para inserir e salvar as informações no banco de dados de acordo com o nome, sendo assim direcionando as informações para a tabela correta.

```

31  <form action="/oel/Lixeira" enctype="multipart/form-data"
32  method="post" th:object="{lixreira}"
33  <input type="hidden" th:name="{_csrf.parameterName}"
34  th:value="{_csrf.token}" />
35
36
37  <div class="row">
38
39      <label class="col-md-auto col-form-label">Tipo:</label>
40      <div class="mb-1">
41          <select class="form-control" name="tipo">
42              <option>Escolha uma opção...</option>
43              <option>Papel e papelão</option>
44              <option>Vidro</option>
45              <option>Plástico</option>
46              <option>Metal</option>
47              <option>Orgânico</option>
48              <option>Materiais não reciclados</option>
49              <option>Hospitalar</option>
50              <option>Outros</option>
51          </select>
52          <p th:errors="{tipo}"></p>
53      </div>
54  </div>
55
56  <div class="row">
57      <label class="col-md-auto col-form-label">Bairro:</label>
58      <div class="mb-1">
59          <select class="form-control" name="bairro" >
60              <option>Escolha uma opção...</option>
61              <option>Coahb</option>
62              <option>Alto da maternidade</option>
63              <option>Nova Esperança</option>
64              <option>Centro</option>
65              <option>Alvorada</option>
66              <option>Ceac</option>
67              <option>São Judas</option>
68              <option>Boa Esperança</option>
69
95  <div class="row">
96      <label class="col-md-auto col-form-label">Adicione uma
97      foto da lixeira:</label>
98      <div class="mb-1">
99          <input type="file" name="file">
100     </div>
101  </div>
102
103  <div class="row justify-content-end my-1">
104      <div class="col-auto">
105          <div class="d-grid gap-2 d-md-block">
106              <button class="btn btn-outline-light my-2" type="reset">Limpar</button>
107              <button class="btn btn-light my-2" type="submit">Adicionar</button>
108

```

Figura 22 e 23 - página HTML cadastro de lixeira

A mesma lógica segue na página de cadastro de usuário, o que muda é o direcionamento, ou seja para onde as informações serão mandadas, no caso do cadastro de usuário as informações serão mandadas para a tabela do banco de dados com nome usuario.

```

29@ <form action="/oel/usuario" method="post" th:object="{usuario}">
30 <input type="hidden" th:name="{_csrf.parameterName}" th:value="{_csrf.token}" />
31
32
33@ <div class="row">
34 <label class="col-md-auto col-form-label">Nome:</label>
35@ <div class="mb-1">
36 <input type="text" name="nome" class="form-control"
37 placeholder="Ex: Maria Laura da Silva">
38 <p th:errors="{nome}"></p>
39 </div>
40 </div>
41@ <div class="row">
42@ <label class="col-sm-4 col-form-label">Data de
43 Nascimento:</label>
44@ <div class="mb-1">
45 <input type="text" name="dataNasc" class="form-control"
46 placeholder="Ex: 24/07/2000">
47 <p th:errors="{dataNasc}"></p>
48 </div>
49 </div>
50@ <div class="row">
51 <label class="col-sm-2 col-form-label">Cidade:</label>
52@ <div class="mb-1">
53 <input type="text" name="cidade" class="form-control"
54 placeholder="Ex: Lajes-RN">
55 <p th:errors="{cidade}"></p>
56 </div>
57 </div>

```

Figura 24 - página HTML cadastro de usuário

A página inicial do site, é composta pela lista de todas as lixeiras cadastradas no sistema, onde lá você pode fazer a pesquisa e encontrar a lixeira que procura, assim como pode clicar no botão e adicionar uma nova lixeira, caso já seja um usuário do site. Nesta página fiquei responsável apenas pelo layout, na qual já foi falado acima, utilizando os padrões dos bootStrap.

```

31@ <div class="p-4" style="background-color: #242424;">
32@ <div class="container">
33@ <div class="row justify-content-md-center">
34@ <div class="col-md-9">
35@ <div class="container-fluid">
36@ <div class="row justify-content-between">
37@ <div class="col-auto">
38@ <a href="/oel/Lixeira" class="btn btn-success mb-2"
39 role="button" data-bs-toggle="button">Adicionar Lixeira</a>
40 </div>
41@ <div class="col-auto">
42@ <a href="/" class="btn btn-outline-secondary mb-2" role="button"
43 data-bs-toggle="button">Limpar</a>
44 </div>
45 </div>
46 </div>
47 </div>
48@ <div class="container-fluid">
49@ <div class="row justify-content-end my-3">
50@ <div class="col">
51@ <form action="/oel/pesquisa" method="get">
52@ <div class="col-sm-auto">
53@ <input name="nomepesquisa" class="form-control mb-2"
54 type="search" placeholder="Pesquisar por Bairro"
55 aria-label="Pesquisar">
56 </div>
57@ <div class="col-sm-3">
58@ <button class="btn btn-outline-success mb-2" type="submit">Pesquisar</button>
59 </div>
60 </form>
61 </div>
62 </div>

```

Figura 25 - página inicial

Na figura 25, podemos ver a implementação de um botão “adicionar lixeira” que ao clicar o usuário será direcionado para a página de cadastro de lixeira, também foi colocado as barras de pesquisas, que são responsáveis por buscar no banco de dados e mostrar ao usuário as lixeiras pesquisadas de acordo com seu tipo, nome ou endereço.

E por fim foi adicionado os *cards*, listando todas as lixeiras cadastrada no sistema, na linha 91 (figura 26) é utilizado o *th:each* para pegar as informações que foram inseridas no banco de dados e imprimir na página html. No card contém um botão com o nome detalhar, que irá direcionar ao usuário a página onde vai ter todos os detalhes da lixeira, na linha 105 (figura 26), quando usado o *th:href* sendo feito o mapeamento da lixeira na qual iremos ver todos os detalhes.

```

90@         <div class="row row-cols-1 row-cols-md-3 g-2 mb-4">
91@             <div class="col" th:each="l : ${lixeiros}">
92@                 <div class="card h-100 text-success">
93@                     <div class="card-header">
94@                         <h5 class="card-title text-center" th:text="${l.tipo}">Tipo</h5>
95@                     </div>
96@                     <ul class="list-group list-group-flush">
97@                         <li sec:authorize="hasRole('ADM')" class="list-group-item" th:text="${l.id}" >id</li>
98@                         <li class="list-group-item" th:text="${l.bairro}">Bairro</li>
99@                         <li class="list-group-item" th:text="${l.rua}">rua</li>
100@                        <li class="list-group-item" th:text="${l.complemento}">complemento</li>
101@                    </ul>
102@
103@                    <div class="card-footer text-center">
104@                        <a class="card-link text-success text-decoration-none "
105@                            th:href="@{/oel/detalhes/{id}(id=${l.id})}">Detalhar </a>

```

Figura 26 - página inicial

Na página de detalhamento, também foi colocado um botão para solicitar a exclusão daquela lixeira, ao clicar o usuário irá ser redirecionado para uma página com o formulário a ser respondido, explicando o motivo da solicitação.

```

17@ <div class="p-2"
18     style="background-image: url('/imagens/img4.png'); background-size: 100% 100%;">
19
20@ <div class="container" style="height: 100%;">
21@   <div class="row">
22@     <div class="col-12 text-center p-2 text-light">
23@       <h1 class="display-6">Detalhes da lixeira</h1>
24@     </div>
25@   </div>
26
27@   <div class="container">
28@     <div class="row justify-content-md-center">
29@       <div class="col-md-9">
30
31@         <div class="card bg-light mb-3" style="max-width: 100rem;">
32@           <h5 class="card-header">Lixeira - <span
33@             th:text="{Lixeira.id}">id</span></h5>
34
35@           
38
39@           <div class="card-body ">
40
41@             <p class="card-text">
42@               <strong>Tipo de lixo:</strong> <span th:text="{Lixeira.tipo}">Tipo
43@                 de lixo</span>
44@             </p>
45@             <p class="card-text">
46@               <strong>Bairro:</strong> <span th:text="{Lixeira.bairro}">bairro</span>
47@             </p>

```

Figura 27 - detalhes da lixeira

```

58@ <div class="row justify-content-end my-3">
59@   <div class="col-auto">
60@     <a th:object="{Lixeira}"
61@       th:href="@{/oel/{idLixeira}/fazerDenuncia(idLixeira={Lixeira.id})}"
62@       class="btn btn-light" role="button" data-bs-toggle="button">Solicitar
63@       exclusão</a>
64@   </div>

```

Figura 28 - botão de de solicitar exclusão da lixeira

```

26@ <div class="container">
27@   <div class="row justify-content-md-center">
28@     <div class="col-md-6">
29
30@     <form th:action="@{/oel/{idLixeira}/fazerDenuncia(idLixeira={Lixeira.id})}"
31@           method="post" th:object="{denuncia}">
32
33@     <div class="row">
34@       <label class="col-md-auto col-form-label">Qual o motivo
35@         da denuncia?</label>
36@       <div class="mb-3">
37@         <select class="form-control" name="motivo">
38@           <option>Escolha um motivo...</option>
39@           <option >Lixeira não existe mais!</option>
40@           <option >Lixeira não corresponde ao tipo que está na
41@             descrição</option>
42@           <option >Outros</option>
43@         </select>
44@       </div>
45@     </div>

```

Figura 29 - formulário de solicitação de exclusão da lixeira

A página de conscientização também foi implementada por mim, nela contém informações sobre os tipos de lixeira e dicas de descarte correto, para a sua

implementação também foram utilizados recursos já pronto do bootstrap, basicamente apenas organizei e formatei o texto da melhor forma.

```

17  <div class="container-fluid text-light p-4"
18      style="background-image: url('/imagens/text.png'); height: 340px; background-size: 100% 100%;">
19  </div>
20
21
22  <div class="p-5 text-white"
23      style="background-image: url('/imagens/fundo1.png'); background-size: 100% 100%; background-color: #242424;">
24  <div class="container">
25  <div class="row">
26  <div class="container">
27  <div class="col-7 text-center my-5 p-5">
28  <h2>
29  <em>Separe resíduos por categorias</em>
30  </h2>
31  <p>
32  <em><h5>A maneira mais recomendada para descartar
33      corretamente o lixo doméstico é separá-los por materiais
34      recicláveis (garrafas de vidro, garrafas plásticas, latas, etc)
35      e materiais não recicláveis (espelhos, adesivos, fotografias,
36      etc), assim facilitará a separação dos resíduos pelos lixeiros
37      e centros de reciclagem.</h5> </em>
38  </p>
39  <p>
40  <em>
41  <h5>Além disso você pode também, separar lixos orgânicos
42      dos lixos secos, de modo que os secos poderão ser encaminhados
43      à centros de reciclagem, e os orgânicos usados para
44      compostagem, ou seja, serão transformados em adubo.</h5>
45  </em>
46  </p>
47  </div>
48  </div>

```

figura 30 - página de conscientização

O menu e o rodapé das páginas, foram feitos de forma isolada, sendo assim com uma tag podemos utilizar o mesmo menu e rodapé em todas as páginas ao mesmo tempo sem necessariamente ter que repetir todas as tags usadas para fazê-los. Usamos o th:fragment, para isolar um trecho de código em um arquivo separado e então incluí-los em outras páginas de acordo com a necessidade, como é visto na linhas 11 (figura 31). E para reaproveitar esse fragmento uso a tag th:replace para substituir pelo conteúdo do fragmento, como é visto na figura 32, e essa tag é utilizada em todas as pagina que eu quero reaproveitar o menu

```

11Ⓣ <nav th:fragment="menu" class="navbar col-12 sticky-top"
12   style="background-color: ■ #242424;">
13Ⓣ   <div class="container">
14Ⓣ     <a class="navbar-brand" href="/"> 
16     </a> <a class="navbar-brand text-secondary" href="/oel/conscientizacao">CONSCIENTIZE-SE</a>
17
18
19
20Ⓣ     <div class="d-grid gap-2 d-md-block">
21
22Ⓣ       <a href="/oel/ListDenuncia" class="btn btn-outline-success me-2" role="button"
23         sec:authorize="hasRole('ADM')" data-bs-toggle="button">LIXEIRA COM DENÚNCIAS</a>
24
25Ⓣ       <a href="/oel/usuario" class="btn btn-outline-success me-2" role="button"
26         sec:authorize="!isAuthenticated()" data-bs-toggle="button">CADASTRE-SE</a>
27
28Ⓣ       <a href="/Login" class="btn btn-success me-2" role="button"
29         sec:authorize="!isAuthenticated()" data-bs-toggle="button">ENTRAR</a>
30
31Ⓣ       <a href="/Logout" th:href="@{/Logout}"
32         class="btn btn-outline-success me-2" role="button"
33         sec:authorize="isAuthenticated()" data-bs-toggle="button">SAIR</a>
34
35     </div>
36   </div>
37 </nav>

```

Figura 31 - Menu

```

14Ⓣ <body>
15   <div th:replace="~{menu :: menu}"></div>

```

figura 32 - tag th:replace do menu

O mesmo padrão de reutilização do fragmento foi utilizado para o rodapé

```

8Ⓣ <body>
9
10Ⓣ   <footer th:fragment="rodape" class="p-2 " style="background-color: ■ #242424; ">
11Ⓣ     <div class="container my-4 text-center text-secondary">
12Ⓣ       <p class="footer-copyright">© 2021 Copyright - Lauris & Pêagá </p>
13     </div>
14   </footer>
15
16 </body>

```

Figura 33 - rodapé

```

255   <div th:replace="~{rodape :: rodape}"></div>

```

Figura 34 - tag th:replace do rodapé

7 ANÁLISE E DISCUSSÃO DOS RESULTADOS

O projeto onde está a lixeira foi criado com a finalidade de trazer a população de forma dinâmica e de fácil acesso uma tecnologia que ajudasse a conscientizar e melhorar o descarte correto em nossa cidade, facilitando para os usuários onde descartar o lixo, e onde encontrar uma lixeira mais próxima. A implantação do projeto conseguiu seguir todos os requisitos de acordo com o que foi planejado previamente para seu desenvolvimento, onde as funções de cadastrar usuários, e adicionar lixeiras foram desenvolvidas e tendo um desempenho satisfatório, assim como todo o layout do projeto que foi desenvolvido de forma satisfatória, na qual os usuários conseguem navegar pelo site facilmente, de forma dinâmica e bem explicativa.

Vale ressaltar que o projeto não está 100%, muitas dificuldades surgiram no percurso e algumas coisas não foram implementadas e algumas funcionalidades também não foram desenvolvidas com eficácia, como foi o caso da utilização de recurso gráfico em forma de mapa, para orientar os usuários em relação a localização das lixeiras, mas conforme o projeto foi se desenvolvendo, essa funcionalidade foi substituída pelos filtros de bairro, tipo de lixo e rua. Entretanto, a funcionalidade do mapa não foi descartada e pode ser uma futura adição ao projeto.

Chegando ao final do projeto é satisfatório ver que ele foi concluído e desempenhado da melhor forma, onde conseguimos atingir nossos objetivos e entregando um bom resultado. Sabe-se que o projeto tem aspectos a melhorar, e sendo assim fica em aberto melhorias futuras, funcionalidades como a utilização de um recurso gráfico em forma de mapa, para uma melhor localização, assim como também transformar o site em um aplicativo, que a princípio era umas dos nossos principais objetivos, mas por motivos de tempo e também não ser um conteúdo que não tivemos em nossa grade curricular, foi deixado para depois. Mais todo o conhecimento que adquirimos durante o curso, foi utilizado de forma eficaz e eficiente para o desenvolvimento deste projeto.

8 CONSIDERAÇÕES FINAIS

O presente trabalho pretendeu planejar um método que tornasse fácil o descarte correto de resíduos sólidos, tendo em vista uma cidade mais sustentável no futuro. Ao chegarmos no resultado final, vimos que é possível ajudar o meio ambiente com uma das ferramentas mais importantes para o homem atual: a tecnologia. Por ser um dos assuntos mais trabalhados por cientistas e estudiosos, sentimos que ao desenvolver este portal, estamos deixando nossa contribuição para as gerações vindouras, o que foi e é extremamente gratificante, não esquecendo de mencionar a conquista de termos utilizado o conhecimento que nos adquirido durante nossa passagem pelo Instituto Federal de Ciência e Tecnologia.

Discorrendo sobre a metodologia que utilizamos para a coleta de dados, podemos perceber que nos faltam políticas públicas voltadas para o descarte correto e até mesmo o meio ambiente em geral. Através dessa observação enfatizamos ainda mais a necessidade do nosso portal para melhor orientar a população, e até mesmo o nosso campus.

Para melhorias futuras, poderiam ser adicionados artifícios como mapas para facilitar o acesso e visualização da comunidade, podendo ser um projeto a ser levado às demais escolas do município ou até mesmo a câmara municipal como proposta mais abrangente, não deixando de citar a acessibilidade como fator determinante para uma sociedade inclusiva. Também vimos que é possível atingir o máximo de pessoas, deixando de ser apenas um portal web, passando a ser um aplicativo. Acreditamos que com essas melhorias o portal pode cumprir um importante dever para a sociedade.

REFERÊNCIAS

ProteGEEr, projeto de cooperação técnica entre o Brasil e a Alemanha para promover uma gestão sustentável e integrada dos resíduos sólidos urbanos: O que são resíduos sólidos ?. <<http://protegeer.gov.br/rsu/o-que-sao>> Acesso em: 7 de março de 2022

IPEA (Instituto de Pesquisa Econômica Aplicada). Resíduos sólidos urbanos no Brasil: desafios tecnológicos, políticos e econômicos. <<https://www.ipea.gov.br/cts/pt/central-de-conteudo/artigos/artigos/217-residuos-solidos-urbanos-no-brasil-desafios-tecnologicos-politicos-e-economicos>> Acesso em: 15 novembro de 2020

IBGE (instituto brasileiro de geografia e estatística). Lajes/RN | Cidades e Estados <<https://www.ibge.gov.br/cidades-e-estados/rn/lajes.html>> Acesso em: 16 novembro de 2020

W3C (*World Wide Web*). <<https://www.w3.org/TR/html5/introduction.html#introduction>> Acesso em: 15 março de 2021

W3C (*World Wide Web*) Descriptions of all CSS specifications <<https://www.w3.org/Style/CSS/specs.en.html>> Acesso em: 15 março de 2021

SPRING. Spring projects <<https://spring.io/projects>> Acesso em: 16 março de 2021

SPRING. Spring Boot <<https://spring.io/projects/spring-boot>> Acesso em: 16 março de 2021

SPRING. Spring Security <<https://spring.io/projects/spring-security>> Acesso em: 16 março de 2021

SPRING, Spring Data <<https://spring.io/projects/spring-data>> Acesso em: 16 março de 2021

THYMELEAF, Tutorial: Using Thymeleaf
<<https://www.thymeleaf.org/doc/tutorials/3.0/usingthymeleaf.html#introducing-thymeleaf>> Acesso em: 16 março de 2021

ANEXO A – FORMULÁRIO DE IDENTIFICAÇÃO

Dados do Relatório Científico	
Título e subtítulo: Onde está a lixeira	
Tipo de relatório: Relatório de prática profissional	Data: 24/05/2022
Título do projeto/ programa/ plano: Onde está a lixeira	
Autor(es): Laura Christina de Oliveira Paulo	
Instituição e endereço completo: Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte	
<p>Resumo: Este projeto busca abordar a problemática da poluição no município de Lajes/RN. A amenização desta problemática é de inteira importância para os moradores da cidade, pois esses são produtores e consumidores de recursos ambientais locais (ex.: pecuária e agronomia). O trabalho visa realizar um mapeamento sobre os índices de poluição e as áreas mais afetadas do município de Lajes/RN e, além disso, propor um sistema de localização de lixeiras que promova o descarte correto de materiais. Para isso, foi realizada uma pesquisa para identificar se a população tem ciência dos locais de descarte da cidade, e se estão utilizando dos mesmos. Em seguida, é proposto o desenvolvimento de um sistema que mostre ao usuário a lixeira mais próxima a ele, de acordo com a necessidade. Espera-se que os resultados deste projeto promovam a conscientização da população sobre a atual situação da poluição do município, e estimule a forma adequada de descartar os materiais rejeitados. Assim, este projeto servirá como um caminho para a obtenção de informações que levem à redução da poluição e preservação ambiental local.</p>	
Palavras-chave/descriptores: Poluição; lixo; descarte; Lajes; java; spring boot;	
Nº de páginas: 43 páginas	
Jornada de trabalho: Segunda a sexta, período matutino	Horas semanais: 15 horas
Total de horas: 340 horas	

Observações/notas
