

UNIVERSIDADE EVANGÉLICA DE GOIÁS

Graduação em Engenharia de Software

Alexandre Gero Bordori dos Santos – 1911993

Danilo Caetano Martins de Souza – 1910189

Renato Ferreira Nunes – 1810777

Software de Reciclagem e Coleta de Resíduos

Anápolis – Goiás – Brasil

2022.2

Alexandre Gero Bordori dos Santos – 1911993
Danilo Caetano Martins de Souza – 1910189
Renato Ferreira Nunes – 1810777

Software de Reciclagem e Coleta de Resíduos

Trabalho de Conclusão de Curso apresentado ao curso de Engenharia de Software de graduação em TCC da Universidade Evangélica de Goiás.

Orientador: William P. Santos Júnior

Anápolis – Goiás – Brasil

2022.2

RESUMO

A reciclagem consiste em reaproveitar os resíduos produzidos pela sociedade. No Brasil, isso ainda é pouco explorado, e a maior parte deles vão parar em aterros, onde geram poluição da terra, da água e do ar. Para ajudar na resolução desse problema, esta pesquisa propõe o desenvolvimento de um *software* que mostre os benefícios da reciclagem e como se deve fazer o descarte sustentável desses resíduos. O maior diferencial do *software* é a possibilidade de o indivíduo entrar em contato direto com pessoas que trabalham com coleta de resíduos recicláveis, através do chat. Para verificar a viabilidade, fez-se uma pesquisa utilizando um formulário de perguntas que verifica três pontos: a porcentagem de pessoas que já fazem a reciclagem, a quantidade de pessoas que não fazem ou nunca fizeram reciclagem e a porcentagem de pessoas que utilizariam o *software* em seu dia a dia.

Palavras-chave: Reciclagem de resíduos; Descarte sustentável; *Software*.

ABSTRACT

Waste recycling consists of reusing the waste produced by society. In Brazil it is still not very exploitative, and most of them go, where they will make it happen from land in landfill, water and most of it. To help in the sustainable resolution of this problem, this research proposes the development of an software that most benefits from recycling and how to dispose of this waste. The biggest differential of the application is the possibility for the individual to come into direct contact with people who work with the collection of recyclable waste. To try to use the viability of people, a survey was carried out using a form of questions that already do recycling, the amount of people who do not recycle and the verification of people who did or an software never in their day to day.

Keywords: Waste recycling; Sustainable disposal; Software.

SUMÁRIO

1. Introdução	6
2. Fundamentação Teórica	8
2.1. Resíduos	8
2.2. Importância da Reciclagem	9
2.3. Front-end	10
2.3.1. Dart	10
2.3.2. Flutter	10
2.4. Back-end	10
2.4.1. JavaScript	11
2.4.2. Node.js	11
2.4.3. Express.js	11
2.5. Banco de dados	11
2.5.1. MySQL	12
2.6. Outras tecnologias e metodologias	12
2.6.1. Lean Inception	12
2.6.2. Scrum	12
2.6.3. IDE	12
3. Metodologia	13
4. Desenvolvimento	15
4.1. Requisitos não-funcionais	15
4.2. Requisitos funcionais	15
4.3. Regras de negócio	15
4.4. Protótipos	16
4.5. Modelo de Dados	20
4.5.1. Modelo lógico da base de dados	20
4.5.2. Criação física do modelo de dados	21
4.6. Plano de Teste	22
4.6.1. Front-end	22
4.6.2. Back-end	23
5. Resultados Alcançados	24
6. Considerações Finais	34
Glossário	35
Anexos	36
Desenvolvimento Front-end	36
Desenvolvimento Back-end	41
Bibliografia	44

1. Introdução

A reciclagem de resíduos, no Brasil, tem crescido em ritmo muito lento desde os anos 90. Em 1994, cerca de 81 municípios tinham algum tipo de coleta seletiva, e esse número subiu para 1227 em 2018 (CEMPRE, 2018). Visto isso, apenas cerca de 22% dos municípios brasileiros têm algum tipo de coleta seletiva e, desse valor, cerca de 87% está na região Sul e Sudeste do país. No momento, 88% desses resíduos vão parar em lixões que poluem o solo, a água e o ar, trazendo um custo ambiental de cerca de USD 1 bilhão por ano para tratamento de saúde (ABRELPE, 2020). Por outro lado, entre 2010 e 2019, a geração de resíduos sólidos aumentou de 67 milhões para 79 milhões de toneladas por ano, e espera-se um aumento de 50% desse valor até 2050 (ABRELPE, 2020).

Sendo assim, visando melhorar os números de coleta e incentivar a população sobre a importância de reciclar os seus próprios resíduos, propõe-se um *software* que atenda dois públicos-alvo. Para o usuário produtor de resíduos, uma versão que incentiva e apresenta os benefícios da reciclagem e o modo de realizá-la. Caso a reciclagem seja feita, ele pode enviar uma mensagem solicitando a retirada dos resíduos reciclados em seu endereço. Para o usuário coletor de resíduos, é enviado mensagens em uma tela com indicações de locais para a retirada dos resíduos. Uma vez selecionada a mensagem, o coletor tem acesso ao endereço dos resíduos a serem coletados e pode abrir um *chat* para comunicação.

O Brasil ainda tem um longo caminho a percorrer, porque uma grande parte da população não tem acesso à coleta seletiva e nem está ciente dos impactos que o descarte atual tem para o futuro. Como podemos usar, então, a engenharia de *software* para melhorar o sistema de coleta seletiva do Brasil?

Visando ajudar a resolver este problema, é proposto como objetivo geral o desenvolvimento de um *software* que atue como um intermediador entre o coletor e o usuário produtor de resíduos, motivando-o a realizar um descarte consciente de resíduos recicláveis.

As seguintes etapas foram definidas como objetivos específicos:

- Realização de uma pesquisa de aceitação do *software* na sociedade;
- Levantamento de requisitos para o desenvolvimento do *software*;
- Mapeamento do modelo lógico e conceitual da base de dados para o armazenamento correto dos dados;
- Prototipação das telas do *software*;
- Realização de testes para validação do nível de qualidade do *software*.

Como justificativa para o tema de pesquisa, podemos assegurar que a produção de embalagens e produtos descartáveis, assim como a produção de resíduos, aumentaram significativamente a partir da década de 1980 (FONSECA, 2013). Com isso, já são comuns, em várias partes do mundo, atividades como as campanhas de coleta seletiva de reciclagem de alumínio e papel. Na década de noventa, no Brasil, os índices globais de reciclagem foram irrisórios, saltando de 1%, no começo da década, para 4%, em 1999. Na década seguinte, os índices de reciclagem alcançaram 5% no ano 2000 e se elevaram a 13%, em 2008 (FIGUEIREDO, 2010).

Além de preservar o meio ambiente, a reciclagem também gera riqueza. Os materiais mais reciclados são o vidro, o alumínio, o papel e o plástico. Até mesmo muitas indústrias estão reciclando materiais como uma forma de reduzir os custos de produção. Ademais, contribui para a diminuição significativa da poluição do solo, da água e do ar.

2. Fundamentação Teórica

2.1. Resíduos

Em primeiro lugar, é preciso saber que resíduos é tudo que sobra depois de se realizar uma tarefa de produção, sendo de atividades humanas, animais e processos industriais. Grande parte possui algum valor econômico, gerado através de sua reciclagem, fazendo parte da reutilização por indústrias ou coletores de resíduos (ECYCLE, 2018). No meio urbano, esses resíduos são subdivididos por meio de categorias para facilitar a sua separação e gerar um melhor valor para a reciclagem. Nas divisões, estão: as matérias orgânicas, papéis e papelões, plástico, vidro, metais, e também uma categoria considerando eletrodomésticos e roupas (ABRELPE, 2020).

Os resíduos, como foi relatado acima, têm algum valor e podem ser reciclados. Os materiais que não conseguem ter essa mesma finalidade são considerados rejeitos. Estes, podem ser tratados como um tipo de descarte, mas não conseguem ser reaproveitados ou reciclados. Geralmente, o seu fim, segundo a Política Nacional de Resíduos Sólidos (Lei nº 12.305/2010), é o encaminhamento para aterros sanitários (ECYCLE, 2018).

O termo técnico para reciclagem é caracterizado por um processo de transformação de resíduos recicláveis em produtos reutilizáveis, tendo inúmeros benefícios para a população e para o meio ambiente (SIGNIFICADOS, 2011). A fim de chegar ao resultado final da reciclagem, realiza-se a etapa de separação desses resíduos e depois, por meio de indústrias recicladoras, faz-se uma descontaminação com algumas máquinas específicas para esse trabalho. E então, o material é encaminhado para locais onde se fará essa reutilização (ABRELPE, 2020).

Os resíduos recicláveis podem ser divididos em 2 categorias, resíduos sólidos e resíduos orgânicos. É importante conhecer a origem de todos os resíduos sólidos, os tipos que os englobam e suas características. Para isso, cada tipo possui alguma finalidade específica e não pode ser descartado de qualquer forma, já que alguns, no meio ambiente, levariam anos para chegar ao fim de sua decomposição. Por isso, destaca-se a grande importância da realização da coleta seletiva (ABRELPE, 2020).

Os resíduos orgânicos, são aqueles derivados de uma origem biológica, sendo de uma vida animal ou vegetal. As opções mais comuns para facilitar a reciclagem desse tipo é por meio da utilização de um triturador para realizar o processamento ou por meio da compostagem (ECYCLE, 2018).

A compostagem é um processo biológico de decomposição dessas matérias. Pode-se realizar esse processo por meio de minhocas, proporcionando um adubo e biofertilizante de qualidade. Há também caixas chamadas composteiras que, através do empilhamento delas, com furos específicos para o escoamento da parte líquida, armazenam na base o chorume produzido (ECYCLE, 2018).

2.2. Importância da Reciclagem

O aumento da produção de resíduos segue crescendo com o passar dos anos. Com base em dados coletados do Panorama dos Resíduos Sólidos no Brasil 2020, foi constatado em 2010 a geração de 66,7 milhões de toneladas de resíduos. Já em 2019, esse valor passou para 79,1 milhões de toneladas (ABRELPE, 2020). Com o distanciamento social, consequência da pandemia do COVID-19 nos últimos anos, a Associação Brasileira de Empresas de Limpeza Pública e Resíduos Especiais (ABRELPE, 2020) estima o aumento de 15% a 25% de resíduos residenciais produzidos.

Entretanto, o descarte incorreto desses resíduos pode acarretar grandes problemas ao meio ambiente, como a poluição e a degradação da natureza, prejudicando todo o ecossistema local. A alternativa de aterros e lixões também favorece essas consequências. Então, o processo de reciclagem é o melhor meio para evitar danos e gerar renda a cerca de 800 mil e 1 milhão de catadores e catadoras de resíduos recicláveis no Brasil, como um meio de sustento econômico (MNCR, 2014).

Embora o Brasil tenha uma Política Nacional de Resíduos Sólidos, mesmo tendo o potencial de reciclar 30% do total de resíduos coletados, o país recicla apenas 2,1%. De acordo com o SNIS (Sistema Nacional de Informação em Saúde), esse percentual se mantém inalterado há pelo menos 4 anos (SOARES, 2021). No Brasil, há uma grande dificuldade no avanço da reciclagem, porque ainda não se resolveram os principais problemas da região. Um exemplo disso é que apenas 41,4% da população tem acesso à coleta seletiva (SOARES, 2021).

Uma pesquisa do Ibope de 2018 mostrou que, no Brasil, 66% da população sabe pouco ou nada sobre coleta seletiva e 39% não separa seus resíduos. Outra pesquisa realizada pelo Instituto Ipsos, em 2019, mostrou que 54% dos brasileiros desconhecem as práticas de reciclagem em sua área (MARASCIULO, 2020).

2.3. Front-end

É a parte focada na interface gráfica do projeto, ou seja, é a aplicação onde haverá interação com o sistema e, por isso, deve ser desenvolvida com foco na experiência do usuário. O indivíduo responsável por desenvolver essa parte é chamado de *Dev. front-end* e, apesar de formar a parte estética do *software*, ele não tem a incumbência de desenhar essas telas. O responsável pela arte de um software é o *Design*, que irá projetar as telas e o desenvolvedor irá apenas replicá-las através da programação (DEVSKILLER, 2019).

2.3.1. Dart

É uma linguagem de *script* desenvolvida pela *Google* e foi lançada no ano de 2011, na Dinamarca. Seu objetivo inicial foi de substituir o *Javascript* como linguagem principal para navegadores. Em novembro de 2013, foi lançada a primeira versão estável, o Dart 1.0, e em agosto de 2018, foi lançada a 2.0. O novo modelo foi otimizado para uso na web e nos dispositivos móveis e, em maio de 2020, foi lançada sua versão *SDK* (TREINAWEB, 2019).

2.3.2. Flutter

É um kit de desenvolvimento de interface de usuário (*Framework*) de código aberto, criado pela empresa *Google* em 2015. É baseado na linguagem de programação *Dart*, que possibilita a criação de *software* compilado para linguagem nativa. Seu projeto consiste em possibilitar a criação de *software* para *Android*, *iOS*, *Windows*, *Mac*, *Linux* e *Web*. Em dezembro de 2018, a *Google* lançou a versão 1.0 do *Flutter* e, em fevereiro de 2022, foi anunciado a versão 2.10 com várias atualizações, principalmente para o *iOS* (TREINAWEB, 2020).

2.4. Back-end

O *back-end* é considerado a parte oculta de um site, onde acontece as requisições, a estrutura, o gerenciamento dos conteúdos, a organização e o tratamento das informações, para armazená-las no banco de dados. O desenvolvedor *back-end* deve sempre verificar as práticas aplicadas com muito cuidado, já que qualquer falha no código pode gerar brechas na segurança do sistema, palco para possíveis invasões. Portanto, a proteção das práticas é uma parte fundamental na programação de algum *software* (TOTVS, 2020).

2.4.1. JavaScript

O *Javascript* é uma linguagem de programação que teve sua criação por Brendan Eich, juntamente com a empresa Netscape, em meados do ano de 1995. Essa linguagem foi denominada de vários termos ao longo do tempo. Inicialmente, foi batizada de *Mocha*, depois, de *LiveScript*. Então, para melhorar a popularização dessa linguagem, passou a ser chamada de *Javascript*, buscando associar o marketing da outra linguagem de programação *Java*. Entretanto, as duas possuem características bem diferentes uma da outra, trazendo grandes divergências em suas sintaxes e semânticas. O foco da sua utilização é na parte do *front-end*, e para aplicações *back-end*, é necessário o uso de interpretadores de código (MDN, 2021).

2.4.2. Node.js

O *Node.js* é um ambiente de execução da linguagem *Javascript*, criado por Ryan Dahl em 2009, sendo motivado pelo desejo de ver o desenvolvimento de uma barra de progresso, mostrando o *upload* de imagens no *Flickr* (um site de hospedagem de figuras). O *Node.js* é uma abreviação de NPM (*The Node.js Package Manager*) e, em português, Gerenciador de Pacotes *node.js*. Ele pode ser utilizado em qualquer situação, com foco em aplicações escaláveis, consumindo poucos recursos do sistema e facilitando a produtividade. Grandes empresas aderem à essa tecnologia, como o *PayPal*, *LinkedIn*, *Netflix* e outras. Essas características favorecem a base de bibliotecas *web* e *frameworks* (SHAH, 2017).

2.4.3. Express.js

O *express.js* é um *framework* para *softwares web* do *Node.js*, com recursos para aplicativos *web* e móvel, criado em 2010 por TJ *Holowaychuk*. Foi inserido no *GitHub* inicialmente e, em 2014, na empresa *International Business Machines Corporation* (traduzindo para Corporação Internacional de Máquinas de Negócios, sigla: IBM). No ano de 2016, foi transferido para a empresa de Fundação *Node.js*. Esse *framework* é capaz de criar uma interface de programação robusta e simples, fornecendo sistemas de rotas completo e gerenciamento de requisições HTTP (*HyperTextTransferProtocol*, em português: Protocolo de Transferência de Hipertexto) (EXPRESS, 2017).

2.5. Banco de dados

É uma coleção organizada de dados, normalmente armazenada eletronicamente em um sistema de computador. Um banco de dados é geralmente controlado por um sistema de gerenciamento de banco de dados (DBMS). Juntos, os dados e o DBMS, além dos aplicativos

associados a eles, são chamados de Sistema de Banco de Dados, geralmente abreviados para apenas Banco de Dados (ORACLE, 2022).

2.5.1. MySQL

É um sistema *open-source* de gerenciamento de banco de dados relacional de código aberto, suportado pela *Oracle* e baseado em linguagem de consulta estruturada (SQL). O sistema funciona em praticamente todas as plataformas, incluindo *Linux*, *UNIX* e *Windows* (MYSQL, 2022).

2.6. Outras tecnologias e metodologias

2.6.1. Lean Inception

É um método de criação de produtos de forma colaborativa, com foco em obter o Mínimo Produto Viável e encurtar o processo. Um dos principais problemas que as empresas encontram é a demora da criação de um produto, gastando muito tempo e recurso. Muitas vezes, quando o produto é finalizado, o mercado já supriu essa demanda com outra solução, fazendo com que a empresa não tenha o retorno esperado. É por isso que *Lean Inception* é uma ferramenta essencial para o negócio de uma empresa atualmente (LUMIS, 2022).

2.6.2. Scrum

É uma estrutura que ajuda as equipes a trabalharem juntas. O Scrum as estimula a aprenderem com as experiências, a se organizarem enquanto resolvem um problema e a refletirem sobre os êxitos e fracassos para garantir melhoras. Embora seja mais utilizado pelas equipes de desenvolvimento de *software*, os princípios e as lições dessa estrutura podem ser aplicados a todos os tipos de trabalhos em equipe (ATLASSIAN, 2019).

2.6.3. IDE

Uma IDE em português significa Ambiente de Desenvolvimento Integrado. O primeiro conceito foi iniciado por volta da década de 80, facilitando a vida dos programadores, visto que antes a escrita de um código era feita por meio de um editor de textos, normalmente padrão inicial do computador. A IDE surgiu como uma ferramenta para compilar códigos, acessar o terminal, executar *scripts* e editar os códigos, integrada em um ambiente, sendo capaz de apresentar erros e aumentar a produtividade dos desenvolvedores. Entre as IDEs mais utilizadas, de acordo com uma pesquisa do *Stack Overflow*, o *Visual Studio Code* possui 71,06% de uso, o *Visual Studio*, 33,03% e o *IntelliJ*, 28,74% do uso pelos desenvolvedores (LIMA, 2021).

3. Metodologia

Esta pesquisa é de natureza aplicada, porque visa gerar conhecimento para aplicação prática e solucionar um problema específico. Destina-se a resolver problemas sociais através de pesquisa sistemática e investigação. Nesse sentido, a classificação a partir de uma perspectiva alvo é um estudo exploratório, visto que seu objetivo é reunir dados sobre como são coletados os resíduos recicláveis. Serão utilizados métodos de coleta de dados por meio de pesquisas qualitativas e quantitativas e, com base nessas informações, os dados coletados fornecerão informações sobre o conhecimento da população de acordo com o tema e também a facilidade com que se adere ao produto.

O formulário do *Google Forms* foi a ferramenta utilizada para a realização desse tipo de coleta de dados, o qual contou com uma divisão em 4 seções. A primeira seção se vale de uma pergunta contendo três opções de resposta. Com base na resposta do indivíduo, encaminhase a próxima seção, onde surgem novas perguntas de acordo com a resposta escolhida primeiramente. O questionário tem um total de 9 perguntas diferentes de múltipla escolha. Para a divulgação do formulário, disponibilizou-se um link do *Google Forms* com o encaminhamento direto através do *Telegram*, *WhatsApp* e *e-mail*. Objetivou-se um total de 150 indivíduos.

Para o desenvolvimento do *software*, foi escolhida a metodologia ágil *Lean Inception*, juntamente com o *Scrum*, visando à redução de formalidades e priorizando a resposta rápida às mudanças de requisitos e ambientes. Um dos focos no *Scrum* é a sua transparência, inspeção e adaptabilidade, sempre buscando agregar valor ao produto e à resolução do problema. A equipe possui funções bem definidas, com reuniões frequentes, procurando identificar os problemas a fim de corrigi-los rapidamente, seguindo uma frase das metodologias: errar rápido e corrigir rápido.

Para a codificação *front-end*, selecionou-se a linguagem *Dart* da *Google*, com a utilização da biblioteca *Flutter*. A ferramenta a ser utilizada como Ambiente de Desenvolvimento Integrado (IDE) nesta parte da codificação será o *Android Studio*, por possuir um emulador de smartphone embutido. Para o *back-end*, escolheu-se a linguagem *Javascript*, porém utilizando o ambiente de execução focado na criação de regras de negócios *NodeJS*, com o uso do *framework Express.js*, por sua capacidade de gerenciar requisições e possuir um sistema de rotas completo. Justifica-se essa escolha por ser uma linguagem de conhecimento da equipe e possuir como vantagem uma flexibilidade para a utilização em qualquer situação, além de ser capaz de gerenciar aplicações em tempo real. A IDE escolhida é o *Visual Studio Code*.

Para o sistema gerenciador de banco de dados, foi selecionado o *MySQL*. Além de ser gratuita e possuir um fácil uso, tem muitas funcionalidades e conta com uma gama de interfaces de usuário que podem ser implementadas. A ferramenta visual de design de banco de dados escolhida foi o *MySQL Workbench*, que é específica para o *MySQL*. O *software* é produzido para atender 2 tipos de usuários, sendo o usuário produtor de resíduos, que é qualquer indivíduo que queira fazer a reciclagem de resíduos e o usuário coletor de resíduos, que é a o indivíduo que trabalham com a coleta de resíduos.

Para o usuário produtor, será desenvolvida uma tela com mensagens de motivação sobre o porquê de se reciclar. Outra tela será desenvolvida com informações sobre a maneira correta de reciclar e quais materiais podem ser utilizados na reciclagem. Outra parte será a tela onde o usuário produtor pode enviar mensagem para que o usuário coletor faça o recolhimento dos resíduos em sua casa. Já o usuário coletor, terá uma tela onde o *software* mostra as mensagens recebidas do usuário produtor. Uma vez selecionada a mensagem, o usuário coletor terá acesso ao endereço do local para coleta e poderá abrir um *chat* com o usuário produtor.

4. Desenvolvimento

4.1. Requisitos não-funcionais

A principal etapa no MVP é a construção de um sistema portátil, que funcione em qualquer *smartphone*, independente de qual seja o seu sistema operacional, sendo essenciais as tecnologias envolvidas. Outro ponto fundamental é a segurança, uma vez que se trata do armazenamento e uso de dados pessoais.

Prioridade	Categoria	ID do Item	Descrição
Essencial	Tecnologia envolvida	RNF01	Portabilidade para aplicativos moveis
Essencial	Segurança	RNF02	Dados pessoais
Poderia	Portabilidade	RNF03	Baixo uso de processamento

4.2. Requisitos funcionais

Os requisitos funcionais são os principais para o MVP, sendo o RF01 importante, mas que pode ser adiado para implementação futura. Já o requisito RF07 é algo interessante para facilitar a comunicação entre o cliente produtor de resíduo e o cliente coletor de resíduo.

Prioridade	ID do Item	Descrição
Poderia	RF01	Mostrar ao usuário como fazer a coleta
Poderia	RF02	Mostrar pontos de coleta
Essencial	RF03	Cadastro de usuário
Essencial	RF04	Autenticação de usuário
Essencial	RF05	Solicitação de para a retirada de coleta
Essencial	RF06	Mostrar endereço do ponto de retirada
Interessante	RF07	Chat para comunicação para ambas as partes

4.3. Regras de negócio

O software possui duas divisões, onde se tem telas que só devem ser acessadas pelo cliente que produz os resíduos e outras telas que só podem ser acessadas pelo cliente que trabalha com a coleta dos resíduos.

Prioridade	Categoria	ID da Regra	Descrição
Essencial	Segurança	RN01	Produtor de resíduos recicláveis não deve ter acesso às telas do coletor
Essencial	Segurança	RN02	O coletor de resíduos recicláveis não deve ter acesso às telas do produtor

4.4. Protótipos

Protótipos são modelos das telas do software feitos em imagens. Assim, é possível validar com o cliente se o design do projeto está de acordo com o que ele quer. Uma vez com os protótipos validados, eles são enviados para o desenvolvedor front-end, que irá criá-los através do código de programação.

O software terá como tela inicial a tela de login. Nessa tela, o usuário pode acessar o tutorial através do maior botão, assim como acessar as telas privadas do sistema através do login feito com e-mail e senha, ou com autenticação feita através da conta do Facebook ou Google, conforme mostrada na imagem 1.

Imagem 1



Fonte: Os autores

Na imagem 2, tem-se a tela de cadastro de nova conta, onde o usuário deve primeiro escolher qual dos usuários ele é, sendo cliente produtor ou cliente coletor de resíduos. Essa seleção é feita através de dois botões com os respectivos nomes. Em seguida, o cadastro deve ser feito com a entrada de nome, e-mail, celular e senha, ou através das contas do Facebook ou Google, sendo necessário fazer o cadastro pessoal nas respectivas plataformas.

Imagem 2



Fonte: Os autores

A imagem 3 mostra a tela de tutorial, onde segue uma lista dos materiais que podem ser reciclados. Uma vez escolhido o material, o usuário é redirecionado para a tela com descrição sobre o tipo do material, o lugar de onde é extraído, a finalidade de utilização e como a sua reciclagem pode ajudar a preservar o meio ambiente, mostrado na imagem 4.

Imagem 3



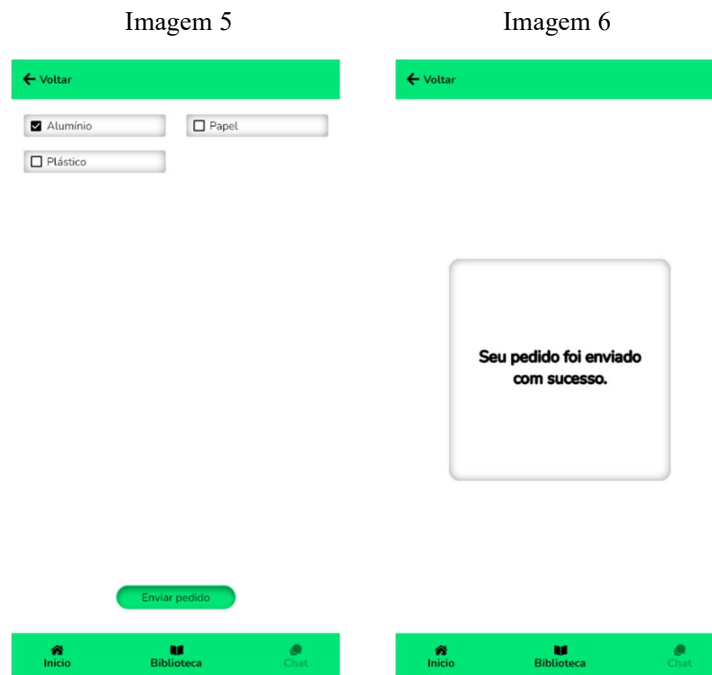
Fonte: Os autores

Imagem 4



Fonte: Os autores

Para o cliente produtor de resíduos, tem-se como tela única a Imagem 5, onde o usuário pode colocar os tipos de materiais que tem em sua casa e os que deseja serem retirados. Uma vez enviado o pedido com sucesso pelo sistema, ele irá redirecionar para a tela com a mensagem de confirmação do envio com sucesso, conforme mostrado na Imagem 6.



Fonte: Os autores

Fonte: Os autores

Para o cliente coletor de resíduos, tem-se como tela única a Imagem 7, onde o usuário pode ver uma lista de pedidos de retirada. Uma vez selecionado, o sistema irá redirecionar para tela do pedido escolhido, onde é possível ver dados como nome, telefone e endereço, assim como os tipos de materiais estão prontos para retirada no local, conforme mostrado na Imagem 8. Confirmado o aceite do pedido, o sistema redireciona o usuário para a tela de confirmação mostrada na Imagem 9.

Imagem 7



Fonte: Os autores

Imagem 8



Fonte: Os autores

Imagem 9



Fonte: Os autores

A imagem 10 é a tela de chat, que só é aberta após a confirmação do aceite do pedido, permitindo que o cliente coletor e o cliente produtor possam se comunicar diretamente.

Imagem 10



Fonte: Os autores

Todas as telas, com exceção da tela de login, possuem na parte superior esquerda um botão de voltar que redireciona o sistema para a tela usada anteriormente. Já na parte inferior, tem-se três botões: o botão iniciar, que envia o usuário para a tela de login; o botão biblioteca,

que envia o usuário para a tela mostrada na Imagem 3; e o botão de chat, que só fica acessível depois que o usuário coletor aceita o pedido mostrado na Imagem 8.

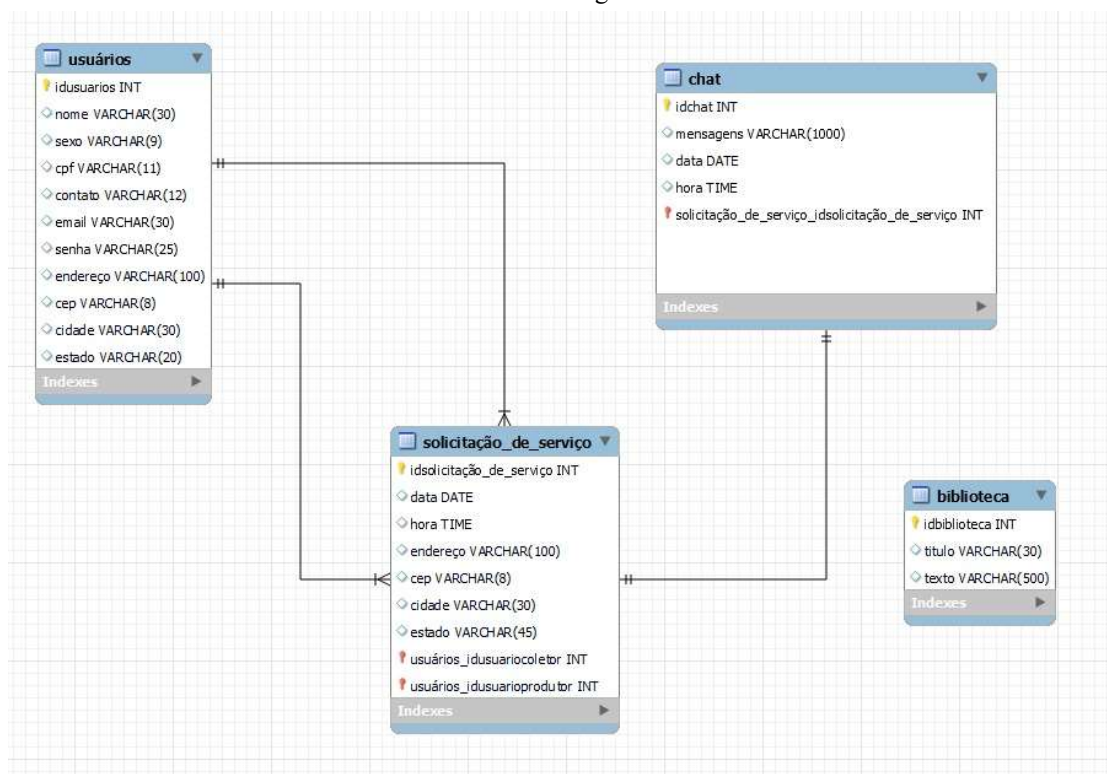
4.5. Modelo de Dados

4.5.1. Modelo lógico da base de dados

Para o Modelo lógico do banco de dados foi feito um Modelo Relacional, como mostra a Imagem 11. Temos 4 entidades, onde o usuário é ligado duas vezes a solicitação de serviço com uma relação de 1:N, ou seja, um único usuário sendo coletor ou produtor podendo realizar inúmeros serviços. Já a entidade de chat é ligada em uma ligação de 1:1, ou seja, um único chat é aberto para cada solicitação de serviço. A entidade de biblioteca não possui nenhuma ligação.

Como atributos, a tabela de usuários recebe informações pessoais básicas e seu endereço. Para a solicitação de serviço, recebe horário e local do serviço a ser prestado. O chat recebe todos os dados de uma mensagem. A tabela de biblioteca recebe título e conteúdo. Todas as entidades recebem um ID como chave primária e solicitação de serviço e chat recebem algumas chaves estrangeiras.

Imagem 11



Fonte: Os autores

4.5.2. Criação física do modelo de dados

Para a criação física do modelo de dados, foi feito um script de banco mostrado nas Imagens 12 e 13 abaixo. Por meio de códigos, o script é responsável para criação das tabelas do banco de dados com seus respectivos atributos como mostrado na Imagem 11, no modelo relacional.

Na Imagem 12, é feita a criação das tabelas de chat e solicitação de serviço e também seus atributos e as especificações deles, além da ligação das tabelas por meio de uma chave estrangeira.

Imagem 12

```
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';

CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
USE `mydb` ;

-----
-- Table `mydb`.`chat`
-----
CREATE TABLE IF NOT EXISTS `mydb`.`chat` (
  `idchat` INT NOT NULL,
  `mensagens` VARCHAR(1000) NULL,
  PRIMARY KEY (`idchat`))
ENGINE = InnoDB;

-----
-- Table `mydb`.`solicitação_de_serviço`
-----
CREATE TABLE IF NOT EXISTS `mydb`.`solicitação_de_serviço` (
  `idsolicitação_de_serviço` INT NOT NULL,
  `data` DATE NULL,
  `hora` TIME NULL,
  `endereço` VARCHAR(100) NULL,
  `cep` VARCHAR(8) NULL,
  `cidade` VARCHAR(30) NULL,
  `estado` VARCHAR(45) NULL,
  `usuários_idusuarios` INT NOT NULL,
  `chat_idchat` INT NOT NULL,
  PRIMARY KEY (`idsolicitação_de_serviço`),
  INDEX `fk_solicitação_de_serviço_usuários1_idx` (`usuários_idusuarios` ASC) VISIBLE,
  INDEX `fk_solicitação_de_serviço_chat1_idx` (`chat_idchat` ASC) VISIBLE,
  CONSTRAINT `fk_solicitação_de_serviço_usuários1`
    FOREIGN KEY (`usuários_idusuarios`)
      REFERENCES `mydb`.`usuários` (`idusuarios`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  CONSTRAINT `fk_solicitação_de_serviço_chat1`
    FOREIGN KEY (`chat_idchat`)
      REFERENCES `mydb`.`chat` (`idchat`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

Fonte: Os autores

Já na Imagem 13, é feita a criação da tabela de usuários e biblioteca e seus atributos, além da ligação da tabela de usuários para solicitação de serviço por meio da chave estrangeira.

Imagem 13

```
-----  
-- Table `mydb`.`usuários`  
-----  
CREATE TABLE IF NOT EXISTS `mydb`.`usuários` (  
  `idusuarios` INT NOT NULL,  
  `nome` VARCHAR(30) NULL,  
  `sexo` VARCHAR(9) NULL,  
  `cpf` VARCHAR(11) NULL,  
  `contato` VARCHAR(12) NULL,  
  `email` VARCHAR(30) NULL,  
  `endereco` VARCHAR(100) NULL,  
  `cep` VARCHAR(8) NULL,  
  `cidade` VARCHAR(30) NULL,  
  `estado` VARCHAR(20) NULL,  
  `solicitação_de_serviço_idsolicitação_de_serviço` INT NOT NULL,  
  PRIMARY KEY (`idusuarios`),  
  INDEX `fk_usuários_solicitação_de_serviço_idx` (`solicitação_de_serviço_idsolicitação_de_serviço` ASC) VISIBLE,  
  CONSTRAINT `fk_usuários_solicitação_de_serviço`  
    FOREIGN KEY (`solicitação_de_serviço_idsolicitação_de_serviço`)  
    REFERENCES `mydb`.`solicitação_de_serviço` (`idsolicitação_de_serviço`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;  
  
-----  
-- Table `mydb`.`biblioteca`  
-----  
CREATE TABLE IF NOT EXISTS `mydb`.`biblioteca` (  
  `idbiblioteca` INT NOT NULL,  
  `titulo` VARCHAR(30) NULL,  
  `texto` VARCHAR(500) NULL,  
  PRIMARY KEY (`idbiblioteca`))  
ENGINE = InnoDB;  
  
SET SQL_MODE=@OLD_SQL_MODE;  
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;  
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

Fonte: Os autores

4.6. Plano de Teste

4.6.1. Front-end

O Teste de Unidade será executado em cada parte do sistema. Uma vez realizado com o uso de objetos falsos que simulam objetos reais, pode-se saber o estado da unidade, já que se houver erro, permite identificar onde e qual objeto se encontra com problema.

Outro teste é o de widget, conhecido também como teste de componente, que possibilita testar as funções de um widget, criado ao unir duas ou mais unidades. Uma vez detectado o erro, pode-se consertar com mais facilidade, já que o widget em questão é único.

Após o teste de widget passar sem erro, executa-se o teste de integração, que deve ser realizado em uma parte ou em todo o sistema. A execução dos testes desde a unidade faz com que os custos sejam menores, visto que os testes de unidade e widget são mais simples, rápidos e mais baratos do que os de integração. Com essas testagens feitas, os testes de integração serão mais baratos por terem menor chance de erro.

4.6.2. Back-end

A aplicação das testagens no *software* no desenvolvimento do *back-end* foi com a utilização dos testes unitários, baseando-se no Desenvolvimento Orientado à Testes, conhecido como *Test Driven Development* (TDD), técnica que consiste em escrever primeiro o teste da aplicação e depois implementar o código, fazendo com que funcionem.

Para a realização de testes no *Node.js*, utilizou-se o *framework Jest* por possuir um suporte focado na linguagem *Javascript*. A funcionalidade do *software* construída por meio do TDD foi a função de *login*, cadastro e autenticação do usuário, colocando os dados a serem inseridos nos campos e escrevendo em forma de código o resultado esperado, de acordo com a ação feita, por meio da automação com o teste. Nas funcionalidades, ao ser necessária a realização de requisições no *back-end*, utiliza-se a biblioteca *Supertest*, colocando os cenários de acordo às situações, bem-sucedidas ou mal-sucedidas.

5. Resultados Alcançados

O formulário de pesquisa teve como foco descobrir a porcentagem de indivíduos que fazem reciclagem, que nunca fizeram reciclagem e que não sabem o que é reciclagem. Após isso, pesquisou-se a importância dada à reciclagem e a possibilidade de uso do *software* proposto, buscando uma melhor validação com a produção dos requisitos propostos. A aplicação do formulário teve uma duração de tempo de 11 dias ininterruptos, coletando respostas de 106 indivíduos. As informações da base de dados estão descritas na tabela 1 abaixo.

Tabela 1: Pesquisa de validação da viabilidade do *software* proposto.

PERGUNTA	FAZ RECICLAGEM	NUNCA FEZ RECICLAGEM	NÃO SABE O QUE É RECICLAGEM
Marque a pergunta que se encaixa com você.	34,9%	64,1%	0,9%
Qual o nível de importância você daria para a reciclagem no Brasil?	62,2% Alta	64,7% Alta	
	21,6% Média	23,5% Média	
	16,2% Baixa	11,8% Baixa	
Existe coleta seletiva em sua cidade?	67,6% Sim	50% Sim	
	32,4% Não	50% Não	
Se tiver a chance de aumentar a reciclagem no Brasil, você participaria do processo?	94,6% Sim		
	2,7% Não		
	2,7% Benefício próprio		
Você usaria um <i>software</i> que mostre os locais onde você pode levar o lixo reciclado para coleta?	89,2% Sim	82,4% Sim	
	10,8% Não	17,6% Não	
Você usaria um <i>software</i> que permite avisar um coletor para que ele retire o lixo reciclável em sua casa?	89,2% Sim	94,1% Sim	
	10,8% Não	5,2% Não	
Você acha útil um <i>software</i> que tenha um tutorial sobre o que pode ser reciclado e como fazer essa reciclagem?	97,3% Sim	89,7% Sim	
	2,7% Não	10,3% Não	
Qual nível de importância você daria para reciclagem do lixo, sabendo que ele polui a terra, a água e o ar e que, ao reciclar, você também estará cuidando do planeta?	86,5% Importante	77,9% Importante	
	13,5% Mediano	22% Mediano	
Estaria disposto a fazer a reciclagem de seu lixo, caso tenha oportunidade?		85,3% Sim	
		1,5% Não	
		13,2% Benefício próprio	

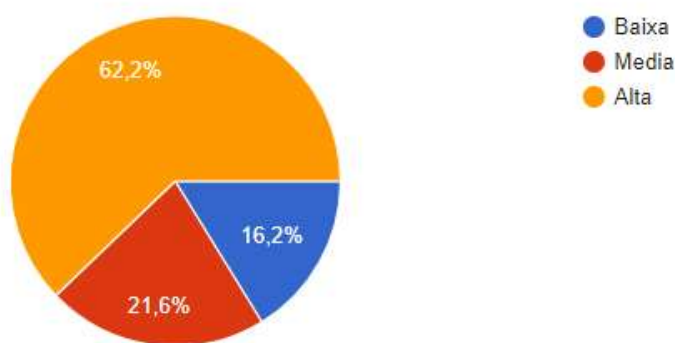
Fonte: Os autores

Com a pesquisa realizada através da ferramenta *Google Forms*, foi possível perceber que a maior parte da população nunca fez reciclagem de resíduos e que existe uma pequena parcela que nem mesmo sabe o que é reciclagem. Com uma pergunta inicial foi obtido o seguinte resultado: 64,9% nunca fizeram reciclagem; 34,09% fazem ou já fizeram reciclagem em algum momento; e 0,9% não sabe o que é reciclagem.

Depois disso, o formulário foi dividido entre os que fazem reciclagem e os que não fazem. Para os dois grupos, fez-se a seguinte pergunta: “Qual nível de importância você daria para reciclagem no Brasil?”.

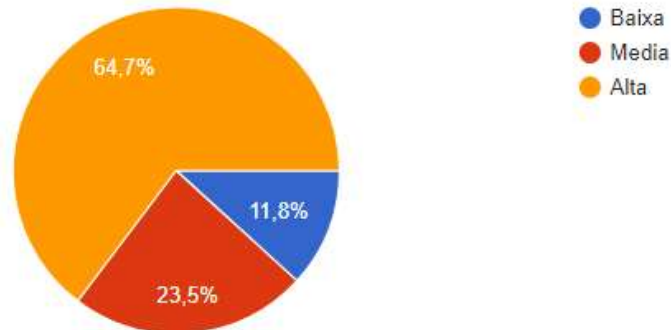
Com o grupo que faz reciclagem, obteve-se o seguinte resultado: 62,2% consideram de alta relevância; 21,6%, de média relevância; 16,2%, de baixa relevância, como é mostrado no gráfico 1. Já com o grupo que não faz reciclagem, obteve-se o seguinte resultado: 64,7% consideram de alta relevância; 23,5%, de média relevância; 11,8% de baixa relevância, como é mostrado no gráfico 2.

Gráfico 1 – Grupo que faz reciclagem
Qual o nível de importância você daria para a reciclagem no Brasil?



Fonte: Os autores

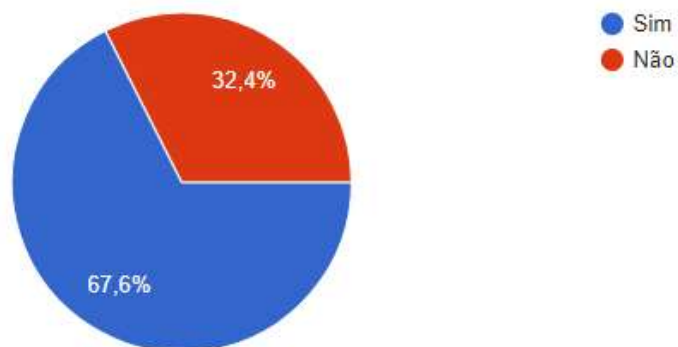
Gráfico 2 – Grupo que não faz reciclagem
Qual o nível de importância você daria para a reciclagem no Brasil?



Fonte: Os autores

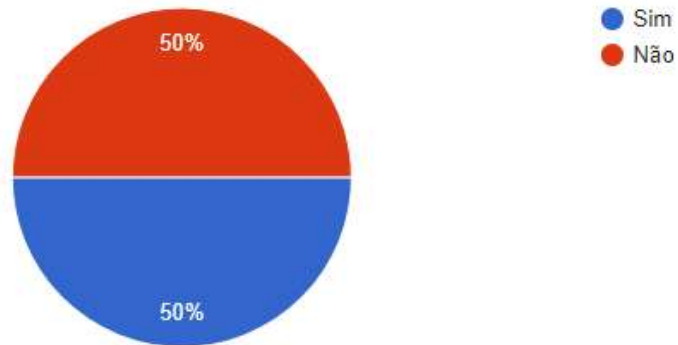
A próxima pergunta é sobre a existência de coleta seletiva onde o indivíduo mora. Para o grupo que faz reciclagem, 67,6% possuem coleta em sua residência, enquanto 32,4% não possuem a coleta seletiva. Já no grupo que não faz reciclagem, é dividido em 50% possuindo cobertura da coleta seletiva, enquanto 50% não tem a coleta. De acordo com os gráficos 3 e 4 abaixo, nota-se que muitos lugares não realizam a coleta seletiva nas cidades, mostrando como o Brasil está com uma capacidade baixa para a reciclagem.

Gráfico 3 – Grupo que faz reciclagem
Existe coleta seletiva em sua cidade



Fonte: Os autores

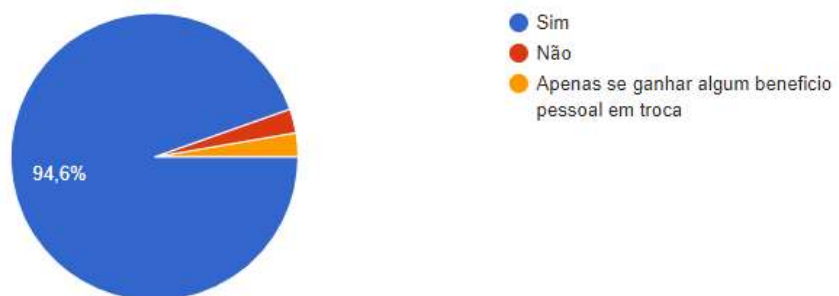
Gráfico 4 – Grupo que não faz reciclagem
Existe coleta seletiva em sua cidade



Fonte: Os autores

A pergunta a seguir é sobre se o indivíduo estaria disposto a ajudar a melhorar o sistema de reciclagem de resíduos. Entre o grupo que faz reciclagem, 94,6% indicaram que participariam do processo, enquanto 2,4% relataram que não participariam e 2,4% só participariam se ganhassem algo em troca, conforme é mostrado no gráfico 5. No grupo que não faz reciclagem, 85,3% estão dispostos a fazerem reciclagem caso tenham oportunidade, enquanto 13,2% fariam essa reciclagem se pudessem conseguir algum benefício em troca e 1,5% não participaria, como demonstrado pelo Gráfico 6. Pode-se notar que muitas pessoas estão dispostas a reciclar, porém é percebido que existe também indivíduos que só estão dispostos a fazer essa mudança ao ganharem algo em troca.

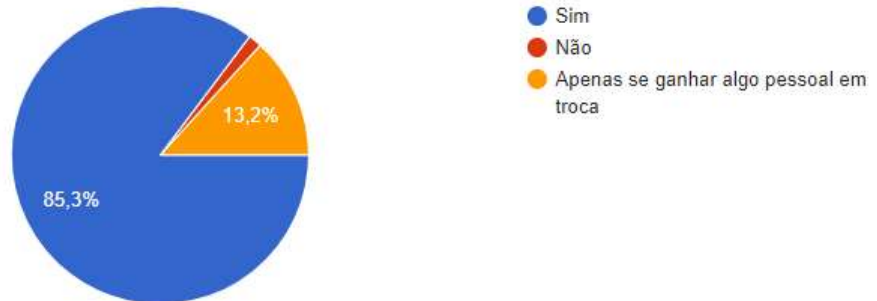
Gráfico 5 – Grupo que faz reciclagem
Se tiver a chance de aumentar a reciclagem no Brasil, você participaria do processo?



Fonte: Os autores

Gráfico 6 – Grupo que não faz reciclagem

Estaria disposto a fazer a reciclagem de seu lixo, caso tenha a oportunidade?

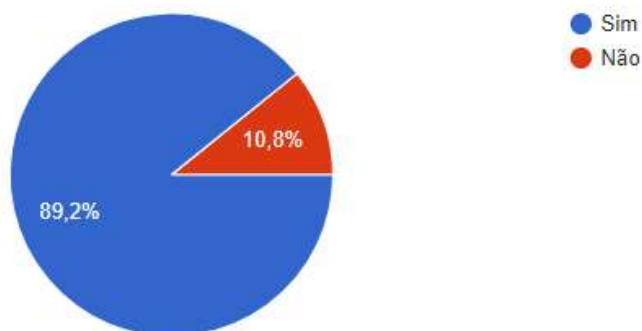


Fonte: Os autores

Na pergunta que discorre sobre a disposição em usar um software que mostre o local dos postos de coleta seletiva, 89,2% do grupo que faz reciclagem fariam uso, enquanto 10,8% não aplicariam no dia a dia, conforme mostra no gráfico 7. No grupo que não faz reciclagem, 82,4% fariam o uso do software, enquanto 17,6% não usariam, conforme mostra no gráfico 8. Comparando com o gráfico 7, é possível notar que as pessoas que não fazem reciclagem consideram esse requisito menos importante do que as pessoas que já fazem a reciclagem.

Gráfico 7 – Grupo que faz reciclagem

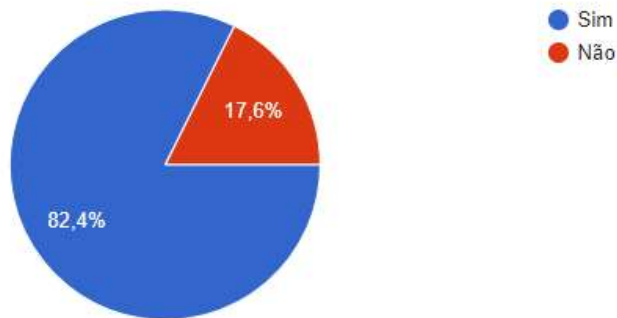
Você usaria um aplicativo que mostre os locais onde você pode levar o lixo reciclado para coleta?



Fonte: Os autores

Gráfico 8 – Grupo que não faz reciclagem

Você usaria um aplicativo que mostre os locais onde você pode levar o lixo reciclado para coleta?

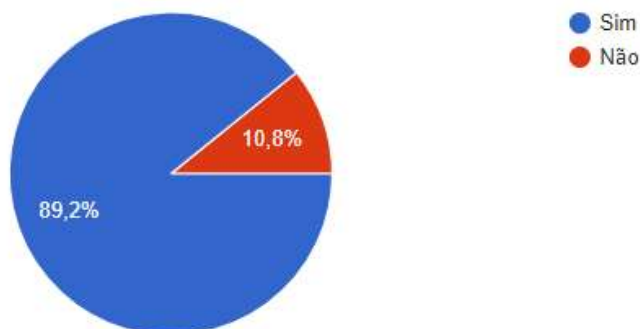


Fonte: Os autores

Com a pergunta “Você faria o uso de um *software* que permite avisar um coletor, para que ele retire os resíduos em sua casa?”, no grupo que já faz reciclagem, 89,2% das pessoas utilizariam esse *software* e somente 10,8% não fariam o uso, conforme mostrado pelo Gráfico 9. Para o grupo que nunca fez reciclagem, 94,1% gostariam de um *software* para avisar um coletor para a retirada de resíduos e 5,9% não utilizariam, conforme o Gráfico 10. Analisando os Gráficos 9 e 10, nota-se que as pessoas que não fazem reciclagem consideram mais útil um *software* que avise o coletor para retirar os resíduos em sua casa, devido a uma maior facilidade na hora do descarte.

Gráfico 9 – Grupo que faz reciclagem

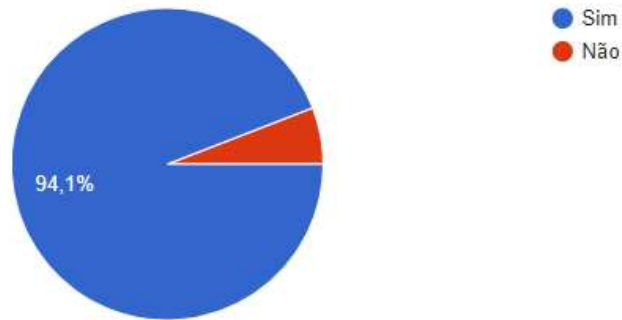
Você usaria um aplicativo que permite avisar um coletor, para que ele retire o lixo reciclável em sua casa?



Fonte: Os autores

Gráfico 10 – Grupo que não faz reciclagem

Você usaria um aplicativo que permite avisar um coletor, para que ele retire o lixo reciclável em sua casa?

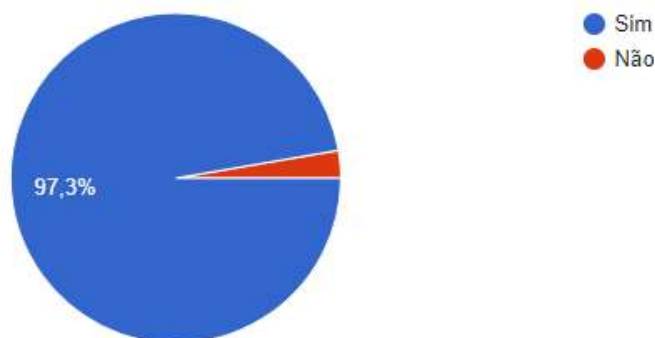


Fonte: Os autores

No grupo que já faz reciclagem, 97,3% consideram útil um *software* que contenha uma biblioteca explicativa sobre quais produtos são recicláveis e como fazer essa reciclagem. Já 2,7% não acham importante esse tipo de informação, conforme o Gráfico 11. Já no grupo que nunca fez reciclagem, 89,7% concordam que seja importante uma biblioteca desse formato e 10,3% desconsideram útil essa informação, de acordo com o Gráfico 12. A utilização da biblioteca facilitaria para as pessoas reciclarem, podendo esclarecer muitas dúvidas dos usuários e economizar o tempo de busca e pesquisa para solucionar alguma questão.

Gráfico 11 – Grupo que faz reciclagem

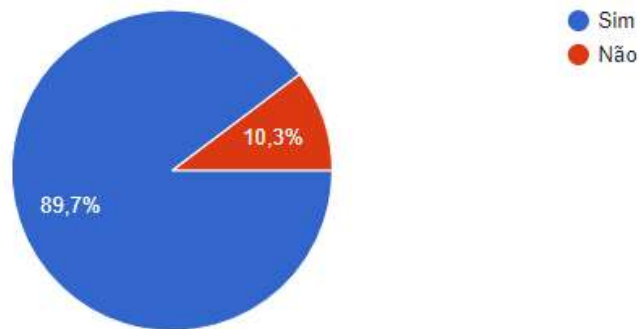
Acha útil um aplicativo que tenha uma pequena biblioteca sobre o que pode ser reciclado, e como fazer essa reciclagem.



Fonte: Os autores

Gráfico 12 – Grupo que não faz reciclagem

Acha útil um aplicativo que tenha uma pequena biblioteca sobre o que pode ser reciclado, e como fazer essa reciclagem.

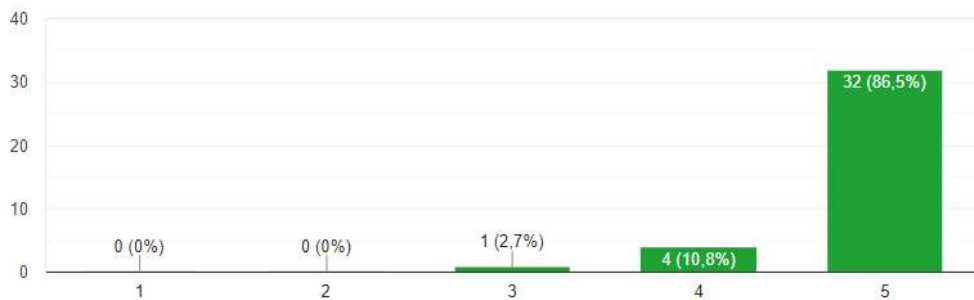


Fonte: Os autores

Perguntou-se sobre o nível de importância que as pessoas dariam em relação a reciclagem dos resíduos, sendo informados dos prejuízos ao não realizarem e dos benefícios que se pode trazer com esse método. No grupo que já faz reciclagem, 86,5% consideram o nível de importância 5, enquanto 10,8% acreditam que seja de nível 4 e somente 2,7% de nível 3, com base no Gráfico 13. Já no grupo que não faz reciclagem, 77,9% consideram o nível de importância 5, enquanto 13,2% acham que o nível é 4 e somente 8,8% consideram de nível 3, de acordo com o Gráfico 15. É possível ver que apesar de opiniões diferentes, nenhum indivíduo considera a reciclagem totalmente sem importância.

Gráfico 13 – Grupo que faz reciclagem

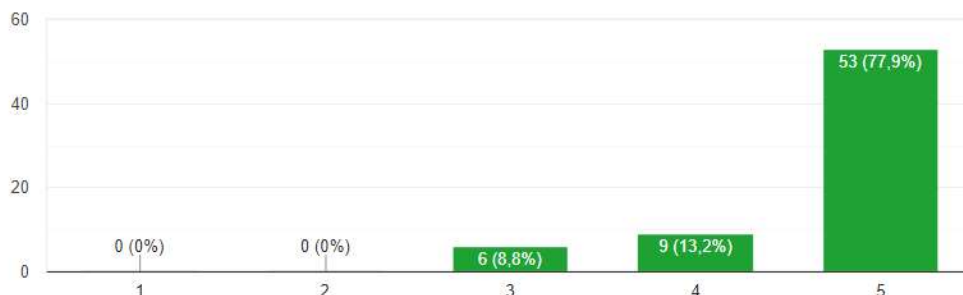
Qual o nível de importância você daria para reciclagem do lixo, sabendo que ele polui a terra, a água e o ar, e que ao reciclar você também está cuidando do planeta e das futuras gerações. Sendo 1 para nada e 5 para muito importante.



Fonte: Os autores

Gráfico 14 – Grupo que não faz reciclagem

Qual o nível de importância você daria para reciclagem do lixo, sabendo que ele polui a terra, a água e o ar, e que ao reciclar você também está cuidando do planeta e das futuras gerações. Sendo 1 para nada e 5 para muito importante.



Fonte: Os autores

Com base nas respostas do formulário, verificou-se que a grande maioria sabe o que é reciclagem, porém a maior parte nunca fez. Mesmo com esse cenário, as pessoas estão dispostas a reciclarem materiais caso tenham oportunidade.

O grupo que já faz reciclagem considera o requisito de mostrar o endereço dos postos de coleta mais útil do que o requisito de avisar um coletor para que retire os resíduos em sua casa. Por outro lado, o grupo que não faz reciclagem considera o requisito de avisar um coletor para que retire os resíduos em sua casa mais importante do que o requisito de mostrar o endereço dos postos de coleta. Portanto, foi considerada viável a produção do *software* proposto.

6. Considerações Finais

Ao fim do desenvolvimento do *software* foi possível considerar que alguns fatores são necessários melhorar e adaptar, tanto na parte tecnológica e na reciclagem. Conclui-se que o *software* pode agregar muito na parte relacionada ao aumento da reciclagem e como uma forma de conscientização de sua importância. Analisando os dados, nota-se que grande parte não possui conhecimento sobre locais, ou formas de reciclagem de sua região. Com isso o *software* desenvolvido tem como intuito facilitar e melhorar esses aspectos para os produtores e coletores de resíduos.

A reciclagem no Brasil ainda tem muito a crescer, a população precisa ser conscientizada e ensinada que a reciclagem não é algo para se deixar em segundo plano, e sim colocado em prioridade. O prejuízo ambiental continua a se acumular, e caso não seja feito nada, pode chegar a níveis drásticos. A reciclagem pode não só ajudar a sustentabilidade do planeta e melhorar a saúde da população, como também gerar renda para diversas famílias e o país.

Com isso, conseguimos concluir que, assim como é preciso incentivar as pessoas a reciclar, também é preciso conectá-las com as pessoas ou empresas que fazem essa coleta. Já na construção do aplicativo, foi percebido como o controle e a execução constante das metodologias ágeis são realmente necessários. A falha nessa parte, assim como na comunicação, acarretou atrasos, lentificando e estressando o processo final do produto.

Para melhorias futuras, o sistema pode receber uma localização automática por mapa e GPS (Sistema de Posicionamento Global), facilitando o processo de retirada dos coletores. E para manter o incentivo do uso desse software pelos usuários, a implementação de um sistema de *ranking*, entre coletores e produtores, pode gerar uma bonificação para os melhores colocados.

Glossário

Aterro: Local destinado à decomposição final de resíduos sólidos gerados pela atividade humana, é muito bem estruturado para que reduza o dano ao meio ambiente.

Back-end: Desenvolvimento responsável pela estrutura que possibilita a operação do sistema.

Banco de dados: Conjunto de arquivos relacionados entre si com registros sobre pessoas, lugares ou coisas.

Chat: É uma forma estrangeira que designa aplicações de conversação em tempo real, permite que duas ou mais pessoas façam uma conversa online.

Dart: É uma linguagem de programação multi-paradigma utilizada para desenvolvimento de aplicações web, aplicações móveis e de desktop.

E-mail: Sistema que permite compor, enviar e receber mensagens através de um sistema computacional online de comunicação de forma assíncrona.

Express.js: Framework para aplicativos NodeJS mínimo e flexível com recursos para construção de servidores.

Flutter: É um framework desenvolvido pelo Google na linguagem Dart, permite o desenvolvimento de aplicações nativas tanto para Android quanto para iOS a partir da composição de Widgets.

Framework: Abstração que une códigos comuns entre vários projetos de software provendo uma funcionalidade genérica.

Front-end: Desenvolvimento responsável pela parte visual do sistema.

Google Forms: Aplicativo de gerenciamento de formulários para pesquisas online.

IDE: Ambiente de Desenvolvimento Integrado, é um programa de computador que reúne características e ferramentas de apoio ao desenvolvimento de software com o objetivo de agilizar este processo.

Intellij: IDE utilizada na produção do Front-end desse projeto.

Javascript: Linguagem de programação para implementação de funcionalidades mais complexas em páginas web.

Lean Inception: Workshop coletivo que visa alinhar o entendimento das áreas de negócio e técnicas sobre um produto em seus aspectos mais fundamentais.

Lixão: Área de disposição final de resíduos a céu aberto, sem qualquer planejamento ou medidas de proteção ao meio ambiente e à saúde pública.

MySQL: É um sistema gerenciador de banco de dados relacional de código aberto, faz uso da linguagem SQL (Linguagem de Consulta Estruturada), que é a linguagem mais popular para inserir, acessar e gerenciar o conteúdo armazenado num banco de dados.

MySQL Workbench: IDE utilizada na produção do banco de dados desse projeto.

NodeJS: Ambiente de execução para construir aplicativos de redes escaláveis.

Reciclagem: Processo de conversão de desperdício em materiais ou produtos de potencial utilidade.

Scrum: Estrutura para desenvolver, entregar e sustentar produtos em um ambiente complexo.

Software: É uma sequência de instruções escritas em códigos, para serem interpretadas por um computador, com o objetivo de executar tarefas específicas. Também pode ser chamado de outros nomes, como os programas usados para trabalho, aplicativos para smartphone, jogos, etc.

Telegram: Aplicativo multiplataforma de mensagens instantâneas e chamadas de voz.

Visual Studio Code: IDE utilizada na produção do Back-end desse projeto.

WhatsApp: Aplicativo multiplataforma de mensagens instantâneas e chamadas de voz.

Anexos

Desenvolvimento Front-end

Imagem 14

```
import ...

void main() {
  runApp(const CorpoPrincipal());
}

class CorpoPrincipal extends StatefulWidget {...}

class _CorpoPrincipalState extends State<CorpoPrincipal> {
  int _selectedIndex = 0;
  static const TextStyle optionStyle =
    TextStyle(fontSize: 30, fontWeight: FontWeight.bold);
  static const List<Widget> _widgetOptions = <Widget>[...]; // <Widget>[]

  void _onItemTapped(int index) {...}

  @override
```

Fonte: Os autores

```

@override
Widget build(BuildContext context) {
  return MaterialApp(
    home: Scaffold(
      appBar: AppBar(
        backgroundColor: Colors.greenAccent,
        titleTextStyle: const TextStyle(
          color: Colors.black, fontSize: 20, fontWeight: FontWeight.bold),
        title: SizedBox.fromSize(
          child: const Text("Voltar"),
        ), // SizedBox.fromSize
        leading: IconButton(
          onPressed: () {},
          icon: const Icon(
            Icons.arrow_back,
            color: Colors.black,
          )), // Icon, IconButton
      ), // AppBar
      body:
        Center(
          child: _widgetOptions.elementAt(_selectedIndex),
        ), // Center
    ),
  );
}

```

Fonte: Os autores

Imagem 16

```
bottomNavigationBar: BottomNavigationBar(  
  unselectedItemColor: Colors.black,  
  backgroundColor: Colors.greenAccent,  
  selectedItemColor: Colors.indigo,  
  onTap: _onItemTapped,  
  currentIndex: _selectedIndex,  
  items: const <BottomNavigationBarItem>[  
    BottomNavigationBarItem(  
      icon: Icon(Icons.home),  
      label: 'Início',  
    ), // BottomNavigationBarItem  
    BottomNavigationBarItem(  
      icon: Icon(Icons.menu_book),  
      label: 'Biblioteca',  
    ), // BottomNavigationBarItem  
    BottomNavigationBarItem(  
      icon: Icon(Icons.chat),  
      label: 'Chat',  
    ), // BottomNavigationBarItem  
  ], // <BottomNavigationBarItem>[]  
), // BottomNavigationBar  
, // Scaffold  
); // MaterialApp  
}
```

Fonte: Os autores

Imagem 17

```
import 'package:flutter/material.dart';

class ListaParaReciclagem extends StatefulWidget {...}

class _ListaParaReciclagemState extends State<ListaParaReciclagem> {
  bool isChecked = false;

  @override
  Widget build(BuildContext context) {
    return Column(
      children: [
        CheckboxListTile(
          checkColor: Colors.white,
          title: const Text("Aluminio"),
          value: isChecked,
          onChanged: (bool? value) {
            setState(() {
              isChecked = value!;
            });
          },
          controlAffinity: ListTileControlAffinity.leading,
        ), // CheckboxListTile
        CheckboxListTile(...), // CheckboxListTile
        CheckboxListTile(...), // CheckboxListTile
        CheckboxListTile(...), // CheckboxListTile
      ],
    ); // Column
  }
}
```

Fonte: Os autores

Imagem 18

```
import 'package:flutter/material.dart';

class Botao extends StatefulWidget {...}

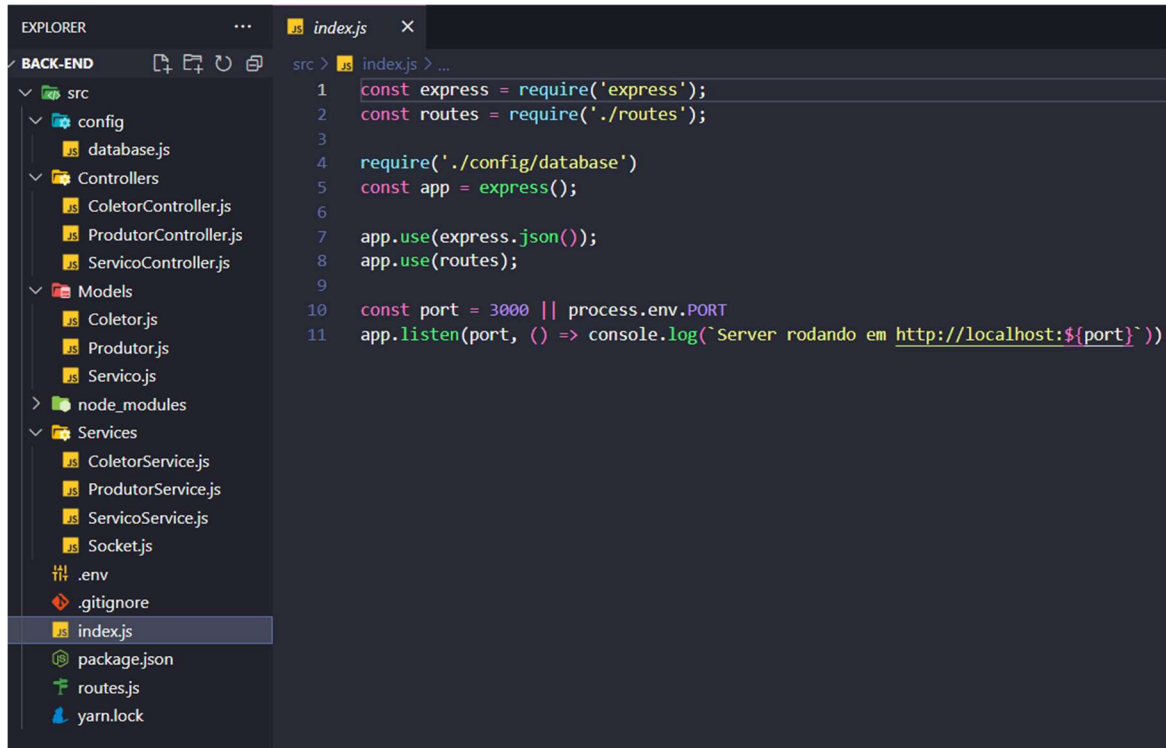
class _BotaoState extends State<Botao> {
  @override
  Widget build(BuildContext context) {
    final ButtonStyle style =
      ElevatedButton.styleFrom(textStyle: const TextStyle(fontSize: 20));

    return Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.min,
        children: <Widget>[
          const SizedBox(height: 340),
          ElevatedButton(
            style: style,
            onPressed: () {},
            child: const Text('Enviar'),
          ), // ElevatedButton
        ], // <Widget>[]
      ), // Column
    ); // Center
  }
}
```

Fonte: Os autores

Desenvolvimento Back-end

Imagem 19



The image shows a code editor interface with a file explorer on the left and a code editor on the right. The file explorer is titled 'EXPLORER' and shows a project structure under 'BACK-END'. The code editor is titled 'index.js' and shows the following code:

```
src > .js index.js > ...
1  const express = require('express');
2  const routes = require('./routes');
3
4  require('./config/database')
5  const app = express();
6
7  app.use(express.json());
8  app.use(routes);
9
10 const port = 3000 || process.env.PORT
11 app.listen(port, () => console.log(`Server rodando em http://localhost:${port}`))
```

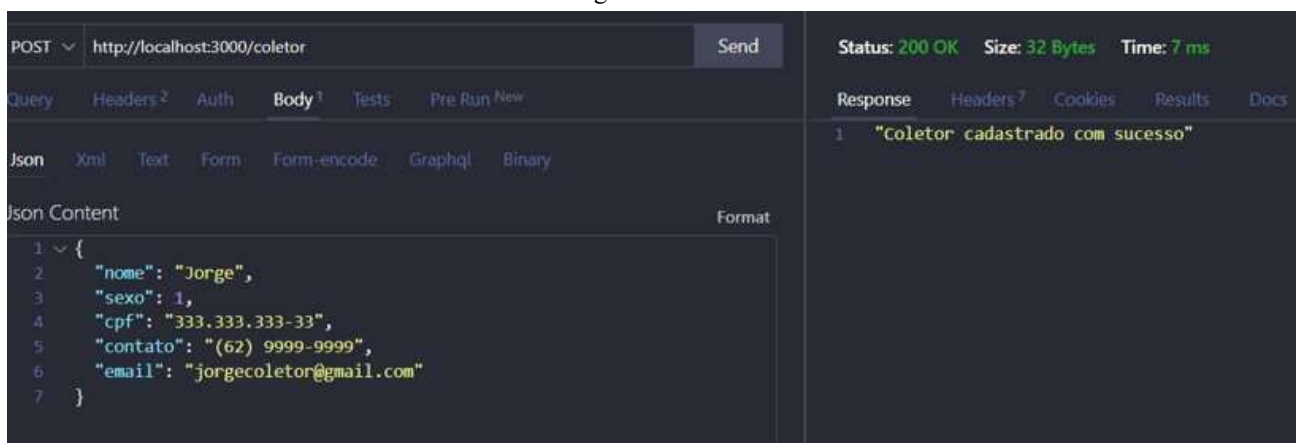
Fonte: Os autores

Imagem 20

```
routes.js x
src > routes.js > ...
1  const express = require('express');
2  const routes = express.Router();
3
4  //Importando controllers
5  const produtorController = require('./Controllers/ProdutorController')
6  const coletorController = require('./Controllers/ColetorController')
7  const servicoController = require('./Controllers/ServicoController')
8
9  //Rotas do empresa/pessoa coletora de resíduos
10 routes.get('/coletor', coletorController.FindAll)
11 routes.get('/coletor/:id', coletorController.FindByKey)
12 routes.post('/coletor', coletorController.Create)
13 routes.put('/coletor/:id', coletorController.Update)
14 routes.delete('/coletor/:id', coletorController.delete)
15
16
17 //Rotas de usuario produtor de resíduos
18 routes.get('/produtor', produtorController.FindAll)
19 routes.get('/produtor/:id', produtorController.FindByKey)
20 routes.post('/produtor', produtorController.Create)
21 routes.put('/produtor/:id', produtorController.Update)
22 routes.delete('/produtor/:id', produtorController.delete)
23
24 //Rotas de Serviço (Produtor-Coletor)
25 routes.get('/coletor/:id/produtor', servicoController.FindAll)
26 routes.get('/coletor/:id/produtor/:id_produtor', servicoController.FindByKey)
27 routes.post('/coletor/:id/produtor/:id_produtor', servicoController.Create)
28
29
30 module.exports = routes
```

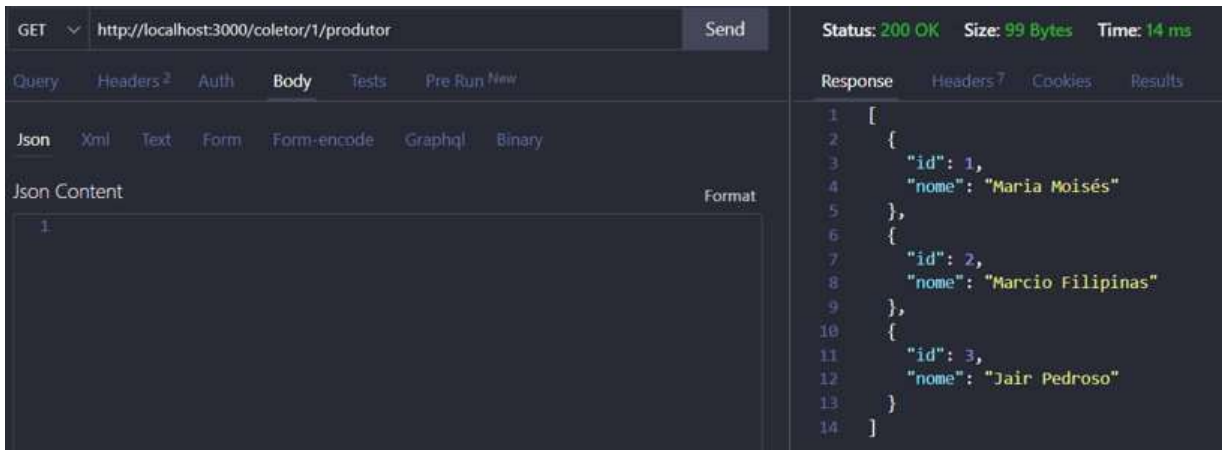
Fonte: Os autores

Imagem 21



Fonte: Os autores

Imagem 22



Fonte: Os autores

Imagem 23



Fonte: Os autores

Bibliografia

- Abrelpe. **Panorama dos resíduos sólidos no Brasil 2020**. Associação brasileira das empresas de limpeza pública e resíduos especiais, 2020. Disponível em: <<https://abrelpe.org.br/panorama/>>. Acesso em: 18 de março de 2022.
- ATLASSIAN. **Scrum: saiba como usar da melhor forma**, 2019. Disponível em: <<https://www.atlassian.com/br/agile/scrum>> Acesso em: 01 de julho de 2022.
- BRASIL, Lei Nº 12.305 de 02 de agosto de 2010 - **Política Nacional de Resíduos Sólidos (PNRS)**. European Commission, (1996).
- Cempre. Pesquisa Ciclossoft 2018: **Radiografando a coleta seletiva**. Cempre, 2020. Disponível em: <<https://cempre.org.br/wp-content/uploads/2020/08/Pesquisa-Ciclossoft-2018.pdf>>. Acesso em: 22 de março de 2022.
- eCycle. **Você sabe a diferença entre resíduo e rejeito?** eCycle, 2018. Disponível em: <<https://www.ecycle.com.br/residuo-e-rejeito/#Residuo>>. Acesso em: 03 de abril de 2022.
- DART, **Paint your UI to life**, Disponível em: <<https://dart.dev/>>. Acesso em 30 de junho de 2022.
- DEVSKILLER. **Descrição do trabalho do desenvolvimento de front-end**. 29 de outubro de 2019. Disponível em: <<https://devskiller.com/pt/front-end-developer-job-description/>>. Acesso em 30 de junho de 2022.
- Equipe TOTVS. **O que é back-end e qual seu papel na programação?** TOTVS, 2020. Disponível em: <<https://www.totvs.com/blog/developers/back-end/>>. Acesso em: 30 junho 2022.
- EXPRESS. **Express – framework de aplicativo da web Node.js.**, 2017. Disponível em: <<https://expressjs.com/pt-br/>>. Acesso em 30 de junho de 2022.
- FIGUEIREDO, Fábio Fonseca. 2012. **O desenvolvimento da indústria da reciclagem dos materiais no Brasil: motivação econômica ou benefício ambiental conseguido com a atividade?** Disponível em: <https://www.researchgate.net/publication/332232829_o_desenvolvimento_da_industria_da_reciclagem_dos_materiais_no_brasil_motivacao_economica_ou_beneficio_ambiental_conseguido_com_a_atividade>. Acesso em: 07 de abril de 2022.
- FLUTTER, **Build apps for any screen**, Disponível em: <<https://flutter.dev/>>. Acesso em 30 de junho de 2022.
- FONSECA, Lúcia Helena Araújo. 2013. **Reciclagem: o primeiro passo para a preservação ambiental**. Disponível em: <<https://semanaacademica.org.br/system/files/artigos/reciclagem.pdf>>. Acesso em: 06 de abril de 2022.
- LIMA, Guilherme. **O que é uma IDE**. Alura, 2021. Disponível em: <<https://www.alura.com.br/artigos/o-que-e-uma-ide>>. Acesso em: 30 junho 2022.
- LUCIMARE TELLECHEA RODRIGUES, L.; ETZ, P.; B. SOARES, C.; D. O GONÇALVES, V.; C. SOARES, M. **Projeto de Conscientização Sobre a Importância da Reciclagem do Lixo para o Meio Ambiente**. Anais do Salão Internacional de Ensino, Pesquisa e Extensão, v. 4, n. 3, 15 março de 2013.
- LUMIS. **Lean Inception: entenda como adotar essa metodologia**, 2022. Disponível em: <<https://www.lumis.com.br/a-lumis/blog/lean-como-adotar.htm>> Acesso em: 01 de julho de 2022.
- MARASCIULO, Marília. **Porque o Brasil recicla tão pouco (e produz muito lixo)?** Galileu, 2020. Disponível em: <<https://revistagalileu.globo.com/Ciencia/Meio-Ambiente/noticia/2020/02/por-que-o-brasil-ainda-recicla-tao-pouco-e-produz-tanto-lixo.html>>. Acesso em: (02/05/2022).
- MDN. **JavaScript**. Mdn web docs, 2021. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Glossary/JavaScript/>>. Acesso em: 30 de junho de 2022.
- MNCR – Movimento Nacional Dos Catadores de Materiais Recicláveis. **Programa de luta e organização nas bases do movimento**, [s.d.]. Disponível em: <<https://www.mnrc.org.br/>>. Acesso em: 19 de março de 2022.
- MYSQL, **MySQL**, 2022. Disponível em: <<https://www.mysql.com/>>. Acesso em 30 de junho de 2022.
- ORACLE. **O que é um banco de dados**. Oracle, 2022. Disponível em: <<https://www.oracle.com/br/database/what-is-database/>> Acesso em: 01 de julho de 2022.
- SHAH, Hezbollah. **Node.js Challenges in Implementation**. Global Journal of Computer Science and Technology, [S.l.], may 2017. ISSN 0975-4172. Disponível em: <<https://computerresearch.org/index.php/computer/article/view/1735>>. Acesso em: 30 junho 2022.

Significados. **Significado de Reciclagem.** Significados, c2011. Disponível em: <<https://www.significados.com.br/reciclagem/>>. Acesso em: 03 de abril de 2022

SOARES, Gabriella. **Reciclagem no Brasil atinge apenas 2,1% de tudo que é coletado.** Poder360, 2021. <Disponível em: <https://www.poder360.com.br/brasil/reciclagem-no-brasil-atinge-apenas-21-de-tudo-que-e-coletado/>>. Acesso em: (02/05/2022).

TREINAWEB. **O que é DART?**, 2019. Disponível em: < <https://www.treinaweb.com.br/blog/o-que-e-dart>>. Acesso em 30 de junho de 2022.

TREINAWEB. **O que é FLUTTER?**, 2020. Disponível em: < <https://www.treinaweb.com.br/blog/o-que-e-flutter>>. Acesso em 30 de junho de 2022.