

IntranetCyclePlastic: Aplicação Web para Gestão de Coleta e Venda de Material Plástico

Abner Henrique Genova de Oliveira¹, Michele Cristiani Barion²

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas – Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) – *Campus Hortolândia* – São Paulo – SP – Brasil

a.genova@aluno.ifsp.edu.br¹, michele_barion@hotmail.com²

Abstract. *Considering the importance of an information system to manage input and output of products in a company that recycles plastics, the objective of this work is the development of a web system to computerize the main processes of this company. As the tool was developed using the incremental model using the PHP language and relational database, the result was satisfactory with the implementation of the system in the company with functionality tests. The approach of the work is based on a first version of the system, considering improvements in future works as a second version of the proposal.*

Resumo. *Considerando a importância de um sistema de informação para gerenciar entrada e saída de produtos em uma empresa que recicla plásticos, o objetivo deste trabalho é o desenvolvimento de um sistema web para informatizar os principais processos dessa empresa. Sendo a ferramenta desenvolvida usando o modelo incremental com o uso da linguagem PHP e banco de dados relacional, o resultado foi satisfatório com a implantação do sistema na empresa com testes das funcionalidades. A abordagem do trabalho se baseia em uma primeira versão do sistema, considerando melhorias em trabalhos futuros como a segunda versão da proposta.*

1. Introdução

Conforme dados divulgados no *site* da Associação Brasileira da Indústria de Plástico [ABIPLAST 2021], no Brasil com o fechamento do ano de 2021, destaca-se que apenas 23.1% do plástico pós-consumo foram reciclados em 2020, ou seja, a maioria desse tipo de material pode ter sido descartado de maneira incorreta, como ter sido depositado junto com lixo orgânico, estando em aterros sanitários. Se for considerada a produção desse material, em 2017, foram mais de 6 toneladas e esse valor está associado devido ao plástico ser uma matéria-prima muito usada para fabricação de diversos produtos [RODRIGUES; VASSAR; ALMEIDA 2020].

A norma NBR 13.230:2008, é um documento que “estabelece os símbolos para identificação das resinas termoplásticas utilizadas na fabricação de embalagens e acondicionamento plásticos, visando auxiliar na separação e posterior reciclagem dos materiais de acordo com a sua composição”, separando os tipos de plásticos em sete conjuntos: PET, PEAD, PVC, PEBD, PP, PS, além da opção outros para identificar produtos fabricados por meio de diversas resinas e materiais [NORMAS 2008]. A Figura 1 traz a

simbologia para identificação de produtos e matérias-primas plásticas, considerando a norma mencionada neste parágrafo.

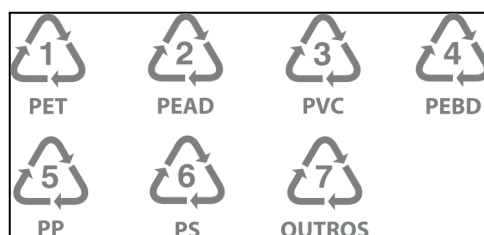


Figura 1. Embalagens e acondicionamentos plásticos recicláveis - Identificação e simbologia

Nesse artigo não serão abordados os tipos apresentados, todavia vários *sites* fazem essas abordagens, tais como da [Recicloteca 2020]¹.

É factível que o plástico seja um material que pode ser reciclado ou reutilizado e que empresas de reciclagem acabam tendo grande importância nesses processos. Como informação, o Sindicato da Indústria de Material Plástico, Transformação e Reciclagem de Material Plástico do Estado de São Paulo (Sindiplast), destaca que em 2019 haviam 319 empresas de reciclagem no estado, estando em quarto lugar como maior empregador dentre os setores da indústria de transformação brasileira [SINDIPLAST 2021].

Uma das empresas nesse segmento, surge em 2020 na cidade de Sumaré, com o intuito de auxiliar no processo de reciclagem de materiais plásticos. No início da empresa todo o gerenciamento administrativo era feito manualmente com o uso de anotações em papéis e pequenas planilhas eletrônicas desenvolvidas no computador. Considerando o aumento de entrada e saída diária de materiais plásticos, tornou-se evidente a necessidade de um sistema de informação para informatizar esses processos e, assim, ter mais resultados para possibilitar ações e decisões gerenciais. É aquele conceito muito conhecido na área de sistemas que diz: transformar dados em informações.

Um sistema de informação pode ser definido tecnicamente como um conjunto de componentes inter relacionados que coletam, processam, armazenam e distribuem informações destinadas a apoiar a tomada de decisões, a coordenação e o controle em uma organização.[...] Os sistemas de informação contêm informações sobre pessoas, locais e itens significativos para a organização ou para o ambiente que a cerca [LAUDON; LAUDON 2014].

Diante dessa relação entre uma empresa que gerencia compra e venda de materiais plásticos e a importância da implantação de um sistema de informação, surge o objetivo geral deste trabalho que é o desenvolvimento de um sistema de informação na plataforma *web* para uma empresa do ramo de gestão de resíduos, na categoria materiais de plástico recicláveis, tendo como objetivos específicos:

¹ Maiores informações dos tipos de plásticos podem ser consultados na Recicloteca através do *link* <https://www.recicloteca.org.br/material-reciclavel/plastico/>.

- A centralização dos dados em uma base local;
- Funcionalidades para administrar funcionários que terão acesso ao sistema;
- Funcionalidades para administrar clientes, produtos e estoque;
- Relatórios gerenciais para apresentação de resultados, considerando o controle do estoque quanto às diversas categorias de produtos que são armazenados.

Para abordagem dos objetivos expostos, além desta introdução, o presente trabalho está organizado em seis seções ordenadas em: fundamentação teórica, metodologia, desenvolvimento, conclusão, trabalhos futuros e relação das referências utilizadas tanto na pesquisa sobre as metodologias adotadas quanto das ferramentas que foram fundamentais para o desenvolvimento.

2. Referencial Teórico

Esta seção aborda a pesquisa bibliográfica realizada para gerar o embasamento teórico que fundamenta o uso de cada escolha para realização do presente trabalho.

2.1. Reciclagem de plásticos

Como apresenta [Miranda 2003], a origem da palavra plástico é originária palavra grega “Plastikos” que significa “capaz de ser moldado”. É um material cujo elemento essencial é constituído por ligações moleculares orgânicas, que resultam da síntese ou através da transformação de produtos naturais. São por via de regra deformáveis plasticamente por meio de manufatura sob determinadas condições (calor e pressão). A matéria-prima para os polímeros é chamada de monômetro e é obtidas principalmente do petróleo e gás natural.

Os plásticos são divididos em categorias, que levam em consideração as estruturas de suas macromoléculas e também pelo tipo de ligação entre elas. A Figura 2 ilustra os tipos de plásticos existentes, sendo apresentadas as principais características de cada grupo.

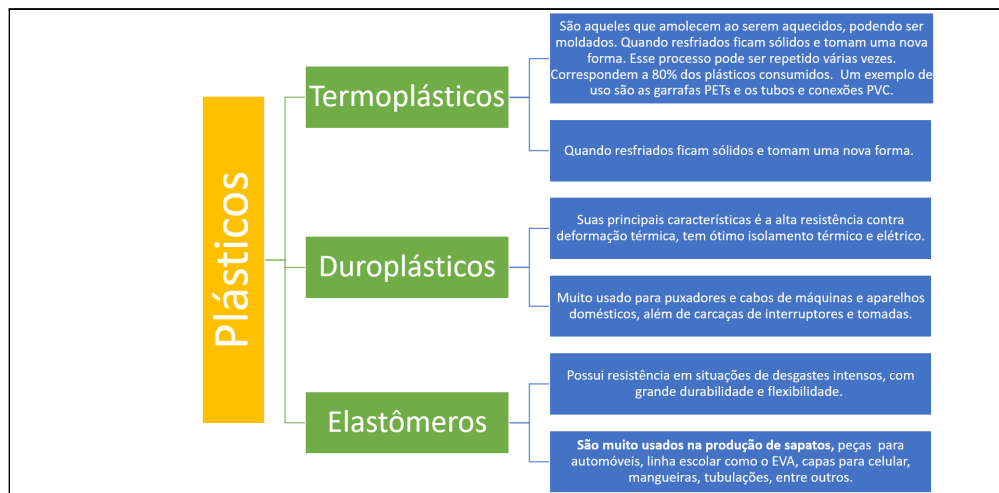


Figura 2. Tipos de plásticos [Michaeli 1995]

- Termoplásticos: constituem a maior parte dos polímeros existentes. São fundíveis podendo ser moldados em formas que são mantidas após seu resfriamento e também ser refundidos diversas vezes. Dependendo de suas estruturas, à temperatura ambiente os termoplásticos podem ser maleáveis, rígidos ou frágeis. Esse grupo corresponde a cerca de 80% dos plásticos consumidos [RECILOTTECA 2020].

- Duroplásticos: devido sua estrutura intermolecular ser extremamente encadeada em todas as direções, essa categoria é extremamente estável à variações de temperaturas e não podem ser fundidos. Nas décadas de 40 e 50 houve um grande desenvolvimento comercial em espumas flexíveis, tintas e vernizes. Possui alta resistência à tração e compressão, sendo ideal para produção de peças que exijam grande durabilidade [MIRANDA 2003].

- Elastômeros: podem ser termoplásticos ou duroplásticos. A diferença entre os elastômeros e os outros tipos de plásticos, é que ele possui um campo de deformação bem acima dos anteriores.

Devido à alta durabilidade dos plásticos e os impactos negativos ao meio ambiente quanto ao descarte ilegal dos mesmos, o reaproveitamento desses resíduos vem sendo cada vez mais discutido. A principal função do reaproveitamento é não permitir que esses lixos fiquem sem utilização, inserindo-os através da preparação para uma nova produção. A reciclagem (cycle = ciclo; re = repetir) conforme escrito por [Piva e Wiebeck 2004] é “a coleta, processamento, comercialização e uso dos materiais considerados lixo” e definem o processo pelo qual passa um mesmo material já utilizado para fazer o mesmo produto ou um produto equivalente. Isso leva à redução da quantidade de lixo e também reduz a dependência da matéria-prima.

Como enfatiza [Miranda 2003], "a reciclagem de plásticos começou a ser realizada pelas próprias indústrias, para o reaproveitamento de suas perdas de produção. Quando o material passou a ser recuperado em maior quantidade, separado do lixo, formou-se um novo mercado, absorvendo modernas tecnologias para possibilitar a produção de artigos com percentual cada vez maior de plástico reciclado".

2.2. Modelo incremental

Segundo [Sommerville 2019] o desenvolvimento incremental parte do levantamento de requisitos funcionais que indicam o propósito do sistema de informação a ser desenvolvido [FIGURA 3 - 1], partindo da implementação inicial, ter *feedback* dos usuários ou terceiros e fazer o *software* evoluir através de várias versões, até alcançar os objetivos da aplicação. As etapas de especificação, desenvolvimento e validação são alternadas tendo um rápido retorno dos usuários ao longo de todas elas [FIGURA 3 - 2].

Quanto às entregas, essas são divididas em partes, cada qual sendo avaliada pelos usuários e, se for necessário realizar algum ajuste, será criado um novo incremento para o

desenvolvimento. Esse ciclo se repetirá até o sistema ser desenvolvido por completo [FIGURA 3 - 3].

Diante desses conceitos, pode-se destacar que uma das vantagens desse modelo é que os requisitos não precisam estar todos definidos na etapa inicial. Assim que o projeto estiver em desenvolvimento, novos requisitos podem ser inseridos e, assim, vão sendo implementados.

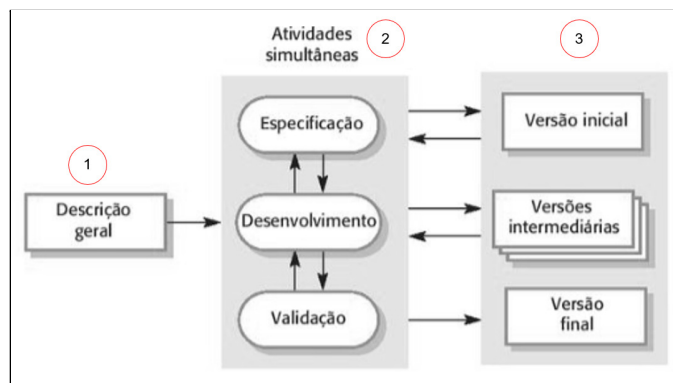


Figura 3. Modelo incremental [SOMMERVILLE 2019]

2.3. Levantamento de Requisitos

Segundo [Pressman 2011] a engenharia de requisitos é uma importante ação dentro da engenharia de *software* que se inicia nas atividades de comunicação e continua na modelagem, servindo como ponte entre o projeto e sua construção. “O levantamento de requisitos (também chamado elicitación de requisitos) combina elementos de resolução de problemas, elaboração, negociação e especificação”.

A fase de levantamento de requisitos é fundamental para que a equipe de analistas e desenvolvedores possam abstrair informações que gerarão os propósitos do sistema. Há muitas ferramentas que podem ser usadas para descobrir essas necessidades. Dessas ferramentas, duas podem ser exploradas conforme [Sommerville 2019], sendo: entrevista e observação.

- A entrevista pode ser gerada através de questões fechadas ou abertas, por meio de um roteiro predefinido. O propósito é que colete requisitos para uma melhor compreensão acerca das necessidades do usuário, podendo ser um bate-papo documentado. É uma técnica muito usada, fazendo com que o usuário se sinta atuante no desenvolvimento do sistema, pois há um diálogo entre as partes envolvidas no projeto.

- A observação é uma técnica interessante na visão do desenvolvedor, afinal permite a sua inserção no ambiente que há os requisitos importantes para coleta. Assim, responsáveis pelo desenvolvimento poderão observar o dia-a-dia do sistema real com a finalidade de anotar as tarefas rotineiras dos usuários envolvidos.

Os requisitos podem ser classificados em duas categorias, sendo os funcionais e os não funcionais. Os requisitos funcionais abordam o que o sistema deve fazer, suas operações tais como: cadastrar um usuário, cadastrar um novo cliente, etc. Já os requisitos não funcionais referem-se às características que o sistema deve ter, tais como restrições, velocidade de operação, validações, usabilidade, desenho, portabilidade e outras atividades e funções [PRESSMAN 2011].

2.3.1. UML e diagrama de caso de uso

Unified Modeling Language (UML) é uma linguagem gráfica para visualização, especificação, construção e documentação de sistemas complexos de *software*. Como cita [Pressman 2011], assim como arquitetos e engenheiros criam plantas para auxiliar no desenvolvimento de um projeto, o profissional da área de sistemas de informação cria diagramas UML para auxiliar no desenvolvimento e construção de um *software*.

O diagrama UML usado para documentar os requisitos funcionais é o caso de uso. O seu desenvolvimento permite a apresentação geral de como funciona o sistema, possibilitando determinar classes, operações, casos de teste de aceitação e roteiros para o manual do usuário [PRESSMAN 2011]. Como aborda o autor [Guedes 2018], esse diagrama “procura apresentar uma linguagem simples e de fácil compreensão para que os usuários possam ter uma ideia geral de como o sistema vai se comportar. Procura identificar os atores (usuários, outros sistemas ou até mesmo algum *hardware* especial) que utilizaram, de alguma forma, o *software*, bem como os serviços, ou seja, as funcionalidades que o sistema disponibilizará a esses atores, conhecidas neste diagrama como casos de uso”.

2.4. Banco de Dados Relacional

Segundo os autores [Elmasri e Navathe 2018] “um banco de dados é uma coleção de dados relacionados. Com dados queremos dizer fatos conhecidos que podem ser registrados e possuem significado implícito”. Um banco de dados sempre está relacionado a um mini mundo, ou seja, a um aspecto do mundo real associado para uma aplicação específica.

Para manter um banco de dados é necessário possuir um Sistema Gerenciador de Banco de Dados (SGBD), que tem como funcionalidades o processo de definição, construção, manipulação e compartilhamento.

Como apresentam [Elmasri e Navathe 2018], as operações em banco de dados são:

- Definir um banco de dados é a ação de especificar os tipos, estruturas e restrições de dados que serão armazenados. Essa definição descritiva é armazenada em uma base de dados na forma de um catálogo chamado de metadados.

- Construir um banco de dados é o processo de armazenar os dados em algum meio controlado pelo SGBD.

- Manipular um banco de dados é a ação de incluir dados, realizar a consulta dos mesmos e gerar relatórios com bases nesses dados inseridos.

- Compartilhar um banco de dados é a ação que permite que usuários e programas acessem esse banco de dados simultaneamente.

Para uso de um SGBD há opções denominadas *opensource* e outras que são pagas. Para consulta, o *site* [DB-Engines 2022] apresenta mensalmente uma média que é calculada por várias variáveis que são coletadas por diferentes fontes e, assim, gerando um *ranking* dos sistemas mais populares nesta área. Um exemplo é apresentado na Figura 4, na qual observa-se que o Oracle ocupa a primeira posição, sendo um exemplo de SGBD proprietário, ou seja, é pago pelo seu uso, além do MySQL que está em segundo lugar sendo um SGBD gratuito.

Rank			DBMS	Database Model	Score		
Jan 2022	Dec 2021	Jan 2021			Jan 2022	Dec 2021	Jan 2021
1.	1.	1.	Oracle 	Relational, Multi-model 	1266.89	-14.85	-56.05
2.	2.	2.	MySQL 	Relational, Multi-model 	1206.05	+0.01	-46.01
3.	3.	3.	Microsoft SQL Server 	Relational, Multi-model 	944.81	-9.21	-86.42
4.	4.	4.	PostgreSQL  	Relational, Multi-model 	606.56	-1.66	+54.33

Figura 4. *Ranking* dos principais bancos de dados relacionais [DB-ENGINES 2022]

2.4.1. Modelagem de Banco de Dados Relacional

Como há o diagrama caso de uso para modelar os requisitos funcionais, conforme abordado na subseção 2.3.1., para modelar um banco de dados relacional pode ser criado um Diagrama Entidade Relacionamento (DER) para apresentar as estruturas das tabelas, além do relacionamento que ocorrerá entre elas.

Como apresentam [Elmasri e Navathe 2018], um “DER é composto de entidades, relacionamentos, seus atributos e cardinalidades. As entidades representam um objeto ou conceito da realidade e os relacionamentos exibem qual a ligação entre elas, utilizando das cardinalidades para definir a sua interação”.

A representação gráfica de um DER possibilita uma visão geral de como será composta a base de dados que será implementada no SGBD através da linguagem *SQL* (*Structured Query Language*).

2.4.2. Linguagem *SQL*

Nos anos 1970 o departamento de pesquisas da empresa IBM desenvolveu a linguagem *SQL* para servir como uma interface para o sistema de banco de dados relacional. Inicialmente a linguagem foi chamada de *SEQUEL* (*Structured English QUery Language*), mas em 1977 foi renomeada para *SQL*. A linguagem, em conjunto com a *American National Standards*

Institute (ANSI) e a *Internacional Standards Organization (ISO)* segue um padrão para manipulação de dados em bancos de dados relacionais [OLIVEIRA 2002].

Como apresentam os autores [Puga, França e Goya 2013], a linguagem *SQL* classifica seus comandos em categorias de instrução, sendo elas:

- *Data Definition Language (DDL)*: estão os comandos associados à definição da estrutura dos dados.

- *Data Manipulation Language (DML)*: associa os comandos voltados à manipulação dos dados que foram armazenados nas estruturas.

- *Data Query Language (DQL)*: possui apenas um comando para realização de consulta aos dados que pertencem a uma estrutura.

Há ainda mais duas categorias, sendo *Transactional Control Language (TCL)* e *Data Control Language (DCL)*, porém elas não serão abordadas nesse trabalho.

A Figura 5, extraída do *site* da [Devmedia 2022], traz uma visão dos comandos *SQL* organizados por suas categorias.

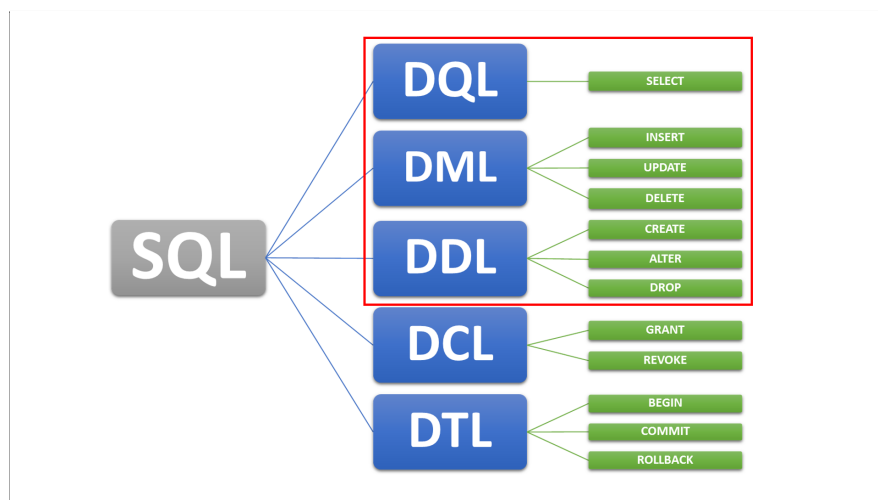


Figura 5. Subdivisões da linguagem SQL [DEVMEDIA 2022]

A referência da Devmedia² também foi usada para consultar as sintaxes dos comandos *SQL* que foram usados na implementação da proposta, considerando as três primeiras camadas apresentadas na Figura 5.

2.5. PHP

² Os comandos possuem sintaxe, ou seja, a forma correta que devem ser executados. Assim, para facilitar o uso dos comandos *SQL* foi utilizado o livro do autor [Oliveira 2002] e o link <https://www.devmedia.com.br/guia/guia-completo-de-sql/38314> do *site* da [Devmedia 2022].

A linguagem *PHP*³ foi criada na década de 90, sendo muito conhecida para desenvolvimento de *sites* dinâmicos.

Como apresenta [Niederauer 2017], o *PHP* tem como principais características:

- É gratuito: o *PHP* pode ser baixado e instalado através do *site* <http://www.php.net>, além de ter inúmeras comunidades para auxiliar no uso e exemplos de códigos;

- Embutido no *HTML*: os arquivos dessa linguagem possuem extensão *.php*. Todavia, quando o desenvolvedor está editando um arquivo de programação *PHP*, esse pode ser mesclado junto com *HTML*. O código *PHP* é delimitado pelas instruções de processamento (*tags*) de início e fim `<?php` e `?>` que permitem que você entre e saia do "modo *PHP*".

- Baseado no servidor: a linguagem *PHP* é executada no servidor. O navegador é usado para apresentar os resultados que foram codificados em *PHP*, todavia parece um *HTML*. Só que a execução de cada linha dos códigos *PHP* quem está sendo responsável pelo processamento é o servidor Apache⁴.

- Bancos de Dados: diversos bancos de dados são suportados pelo *PHP*, sejam de código aberto ou proprietário. Um exemplo é o SGBD MySQL, que será usado no desenvolvimento deste trabalho.

- Portabilidade: o *PHP* pode ser executado em diferentes sistemas operacionais, tais como Linux, Unix ou Windows.

2.6. *Framework* Bulma⁵

Antes de abordar o Bulma enfatiza-se que, conforme apresenta [Minetto 2007], *framework* é um conjunto integrado de artefatos de *softwares* (como classes, objetos e componentes) que colaboram para proporcionar uma arquitetura reutilizável para uma família de aplicações relacionadas e, assim, podendo ser citadas como vantagens a manutenção, reaproveitamento de códigos e separação de *layout* e lógica. Uma funcionalidade específica pode ser implementada através de um *framework* por meio de uma configuração durante a programação de uma aplicação.

³ Para consultar os comandos *PHP* no desenvolvimento da proposta, foram utilizados como referência o *site* oficial do *PHP* através do link <https://www.php.net/> e o livro do [Niederauer 2017].

⁴ O Apache é um programa denominado servidor *web*. Ele é instalado para que as páginas da Internet requisitadas através de um navegador, como Internet Explorer, Chromes entre outros, sejam visualizadas pelo usuário em sua estação de trabalho, podendo ser um *desktop* ou um dispositivo móvel.

⁵ O uso do *framework* pode ser consultado através do link <https://bulma.io/documentation/>.

O Bulma é um exemplo de *framework CSS*⁶, inspirado no Bootstrap e baseado em Flexbox, tendo compatibilidade com a maioria dos navegadores de páginas *web* usados pelos usuários. Contém todos os elementos mais comuns como botões, formulários, menus, tabelas, títulos, notificações, barras de progresso e um simples sistema de *grid*. Também se destaca pela facilidade no manuseio dos seus recursos.

3. Metodologia e Materiais

A proposta é um estudo de caso aplicado em uma pequena empresa de coleta e venda de plástico localizada na cidade de Sumaré, Estado de São Paulo.

Para entender os diferentes tipos de plásticos existentes, sendo esse o principal produto a ser armazenado na base de dados e que irá gerar relatórios estratégicos à empresa, foi feita uma pesquisa bibliográfica através de *sites* que fundamentam o entendimento desse tipo de produto. Além disso, livros, apostilas e manuais também foram usados para gerar a consulta e entendimento dos conceitos associados à teoria e à prática para o desenvolvimento da proposta.

A construção da proposta seguiu o modelo incremental, sendo apresentada na seção de Desenvolvimento. Salienta-se que cada incremento consiste no código implementado e nos testes da funcionalidade correspondente.

Requisitos funcionais e não funcionais também serão abordados no desenvolvimento, considerando os padrões UML.

O sistema proposto é voltado para o ambiente *web*, fazendo o uso da linguagem PHP com banco de dados relacional. Já as telas seguem a implementação através do *framework* Bulma.

4. Desenvolvimento

Esta seção retrata as fases do desenvolvimento do projeto como um todo, sendo elas: estrutura do sistema, levantamento e documentação dos requisitos, modelo de dados do sistema e implementação da proposta.

A Figura 6 apresenta as etapas do trabalho por meio do diagrama de atividades. A cada funcionalidade implementada serão realizados os testes e após *feedback* dos usuários que irão manusear a ferramenta, a rotina será disponibilizada ou retornará para possíveis ajustes, considerando os requisitos levantados. Os incrementos podem ser considerados para cada

⁶ CSS que é uma abreviação de *Cascading Style Sheets* é um mecanismo para adicionar estilo a uma página *web*. O código CSS pode ser aplicado diretamente nas *tags* ou ficar contido dentro das *tags* `<style>`. Outra possibilidade em usá-lo é criando um *link* para um arquivo externo do tipo CSS contendo os estilos que serão aplicados nas páginas *web*. Informações sobre o uso do CSS podem ser extraídas do *site* <https://portalwebdesigner.com/programacao/css/>.

funcionalidade que será implementada com *feedback* através de testes. Essa rotina será seguida até que sejam concluídas as funcionalidades propostas, como são apresentadas no diagrama de caso de uso através da Figura 6. Essa figura também indica a relação com os casos de uso que estão sendo identificados durante a coleta de requisitos.

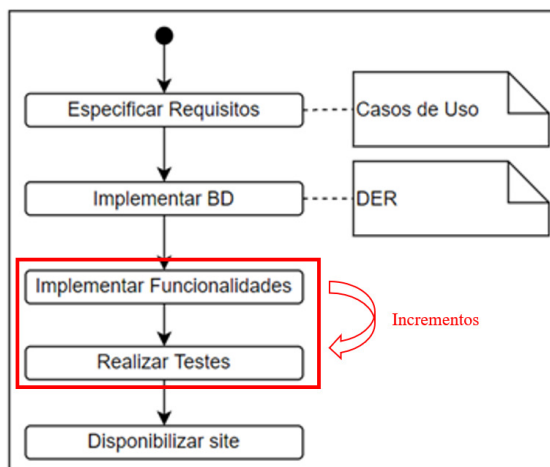


Figura 6. Atividades da proposta

4.1. Requisitos Não-Funcionais

Para a presente versão do sistema proposto, foram levantados os seguintes requisitos não-funcionais, descritos na Tabela 1.

Tabela 1. Requisitos Não-Funcionais

RNF	Descrição
RNF-1	O servidor de banco de dados e o sistema ficarão instalados em uma máquina de acesso para apenas um usuário. Será implantado um servidor local. Este é um requisito de disponibilidade.
RNF-2	Os serviços de servidor serão executados através do sistema operacional Windows. Este é um requisito de compatibilidade.
RNF-3	Facilidade de manusear as páginas da aplicação, considerando ser um sistema que irá rodar em ambiente <i>web</i> . Todas as páginas serão criadas em português. Este é um requisito de usabilidade.
RNF-4	Para cadastro de produtos devem ser respeitados os tipos de plásticos que são discriminados através da norma NBR 13.2030/ABNT. Em atendimento à norma mencionada, esse é um requisito legal.
RNF-5	Quando o usuário efetuar <i>logout</i> o sistema deverá fazer <i>backup</i> da base de dados em uma pasta que deverá ficar na nuvem, considerando como nome a data e a hora em que foi gerado. Este é um requisito de segurança.
RNF-6	A máquina que irá rodar a aplicação deverá ter acesso à Internet para possibilitar o armazenamento do <i>backup</i> e o uso de uma <i>API</i> da Google para geração de gráficos.

4.2. Requisitos Funcionais

Para alcançar os objetivos propostos, a Tabela 2 demonstra os requisitos funcionais, considerando o cenário do usuário e a primeira versão desta aplicação a ser implantada no ambiente de estudo.

Tabela 2. Requisitos Funcionais

RF	Descrição
RF-1	O sistema deve ter um único usuário de acesso ao sistema.
RF-2	O sistema deve permitir as operações básicas em um banco de dados relacional para gerenciamento de clientes.
RF-3	O sistema deve permitir as operações básicas em um banco de dados relacional para gerenciamento de produtos.
RF-4	O sistema deve permitir as operações básicas para gerenciamento de estoque dos produtos cadastrados.
RF-5	O sistema deve permitir o registro de vendas.
RF-6	O sistema deve possibilitar gráficos e relatórios gerenciais, considerando dados de produtos e vendas.

4.3. Diagrama de Caso de Uso

Para coleta de requisitos, foram feitas entrevistas com os funcionários da empresa em que a aplicação será implantada. A partir desse levantamento foi feita uma análise e elaborado um diagrama de caso de uso, conforme é apresentado na Figura 7, descrevendo as funcionalidades e características do *software* sob o ponto de vista do usuário. Já a Tabela 2 traz resumidamente os casos de uso apresentados na Figura 7.

É importante enfatizar que antes que o usuário tenha acesso às funcionalidades, ele deverá realizar login no sistema. Caso ele ainda não tenha esse acesso, há a opção de efetuar o cadastro, como apresenta a Figura 7. As funcionalidades associadas ao “gerenciar” incluem as ações incluir, excluir, alterar e consultar.

Quanto a funcionalidade “Atualizar estoque”, indica uma rotina implementada para manter o estoque atualizado com a confirmação de vendas, afinal gera saída de produtos.

Já com o *logout* do sistema pelo usuário, a ação será executar a funcionalidade “Gerar backup” que consiste em exportar uma cópia atualizada do *database* que se encontra no servidor. Assim, apresenta-se BD como um ator.

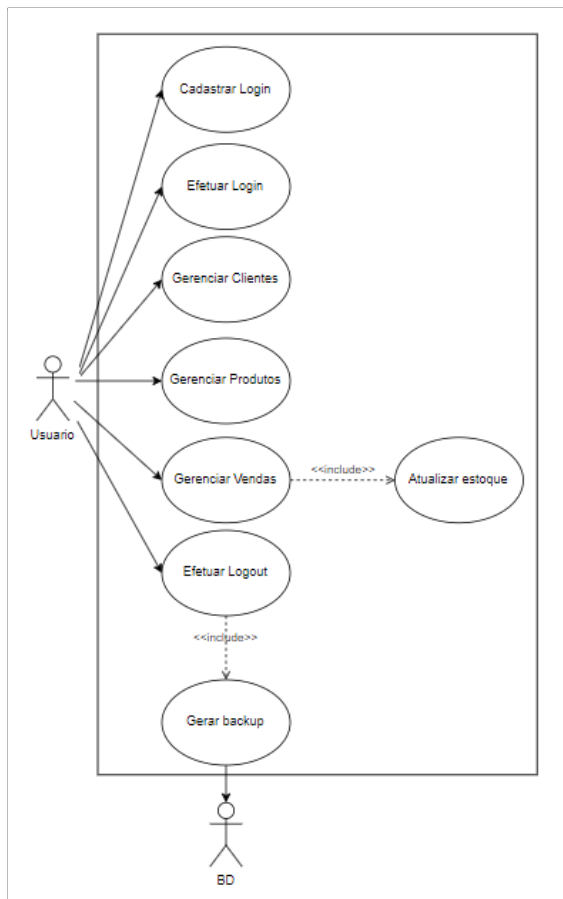


Figura 7. Caso de Uso

4.4. Diagrama de Entidade-Relacionamento

A Figura 8 que traz esta subseção demonstra o Diagrama Entidade Relacionamento (DER) após a abstração dos requisitos funcionais levantados. Considerando que neste trabalho está sendo usado o modelo relacional, sendo escolhido o PHPMyAdmin para gerenciar a base de dados. Observa-se no DER que há cinco tabelas, sendo uma associada a clientes, outra aos usuários do sistema *web* e três vinculadas ao cadastro de produtos e gerenciamento de estoque e vendas.

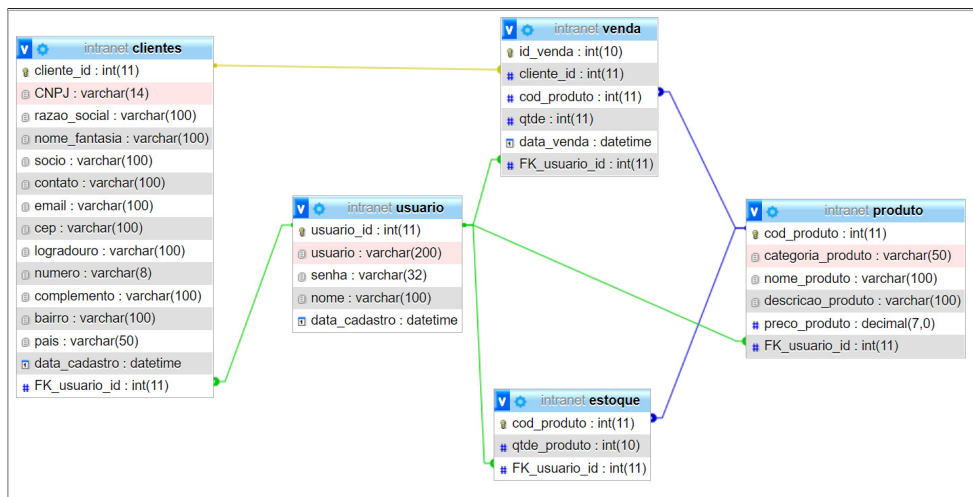


Figura 8. Estrutura do Banco de Dados

4.5. Implementação das Funcionalidades

Esta subseção irá abordar as rotinas que foram desenvolvidas através das funcionalidades apresentadas. Como já apresentado na subseção 3, a proposta é para plataforma *web*, a qual a escolha das tecnologias se resume na facilidade de desenvolvimento com a linguagem PHP e seu banco de dados PHP MySQL e a integração com o *framework* Bulba.

As funcionalidades serão apresentadas através dos casos de uso que estão documentados na Figura 7.

A Figura 9 aborda a tela em que o usuário efetuará o seu *login* no sistema, verificando se o usuário e senha digitados estão cadastrados na tabela usuário (contempleta o RF-1). Caso seja verdadeiro, o sistema irá levar o usuário à tela inicial, conforme é apresentado na Figura 11. Caso não encontre, não permitirá esse acesso, todavia o novo usuário terá a opção de efetuar o cadastro através do *link* denominado “Cadastrar”. A Figura 10 traz a tela para cadastro de usuário.



A imagem mostra a interface de login de um sistema. No topo, há o título "Sistema de Login" e "Intranet - CYCLE PLAST". Abaixo, há dois campos de entrada: o primeiro contém o texto "samuel.oliveira" e o segundo contém pontos, indicando uma senha. Abaixo dos campos, há dois botões: "Cadastrar" e "Entrar".

Figura 9. Tela da funcionalidade “Efetuar Login”

A Figura 10 aborda a tela em que o usuário efetuará o cadastro do usuário no sistema. Será necessário incluir o nome, identificação para o *login* e a senha desejada. Após clicar no botão cadastrar, o *website* fará o *input* dessas informações no BD e levará o usuário para a tela de *login*. Enfatiza-se que todos os campos estão parametrizados como obrigatórios.

Com o *login* efetuado, o usuário terá acesso ao menu principal [FIGURA 11]. Serão apresentados alguns dados quantitativos como a quantidade de clientes cadastrados, produtos e vendas efetuadas, além de um pequeno gráfico abordando o estoque. Enfatiza-se que para apresentação do gráfico como também o mapa, foram utilizadas *APIs* externas por meio de *links* da empresa Google. O [Anexo 1] traz os trechos das rotinas que foram implementadas para apresentação do mapa e dos gráficos. Ressalta-se que a implementação desses resultados contempla funcionalidades definidas para o RF-5.

Figura 10. Tela da funcionalidade “Cadastrar usuário”

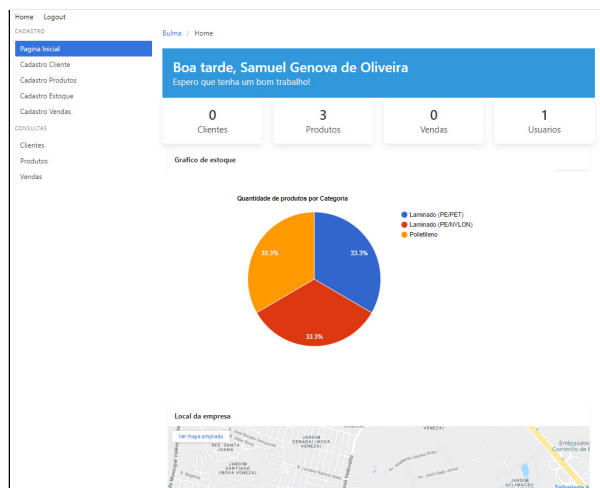


Figura 11. Tela apresentada apenas o usuário efetuar login

A Figura 12 aborda a tela em que o usuário realizará o cadastro de um novo cliente, considerando os campos CNPJ (os clientes são do tipo pessoa jurídica, sendo assim, não foi criado o campo CPF), o endereço completo e as informações de contato. Enfatiza-se que os clientes são empresas que fazem o processo de reciclagem. As ações para gerenciamento de clientes contempla o RF-2.

Figura 12. Tela da funcionalidade “Cadastrar Clientes”

A Figura 13 retrata a tela em que o usuário realizará o cadastro de produtos. Será necessário preencher o código do produto, sua descrição e valores. Ao clicar em “Cadastrar” o sistema irá persistir os dados na tabela produto e também gerará um produto na tabela estoque. As ações para gerenciamento de produtos contempla o RF-3.



Figura 13. Tela da funcionalidade “Cadastrar Produtos”

Na Figura 14 está a tela para adicionar produtos no estoque. O botão “Cadastrar” aciona a operação para a persistência dos dados na Tabela “estoque”. O sistema identifica a quantidade atual do produto que será adicionado e realiza a ação *update* na tabela, adicionando a quantidade informada no formulário. As ações para gerenciamento de estoque de produtos contempla o RF-4.



Figura 14. Tela da funcionalidade “Cadastrar Clientes”

A Figura 15 demonstra o formulário para lançar vendas no sistema. O botão “Cadastrar” aciona a operação para a persistência dos dados na tabela “vendas”. O sistema identifica a quantidade atual do produto que será adicionado como vendido e realiza a ação *update* na tabela “estoque” e, assim, confirmando a venda.

Figura 15. Tela da funcionalidade “Lançamento de Vendas”

Para representar o fluxo do lançamento de vendas, a Figura 16 traz o diagrama de atividade. Observa-se que antes da confirmação do processo de venda, será analisado o estoque do produto que está sendo registrado. Havendo um saldo positivo no estoque quanto a quantidade solicitada na venda, será confirmada a ação com atualização do estoque e o cálculo da venda efetuada. As ações para gerenciamento de vendas contempla o RF-5.

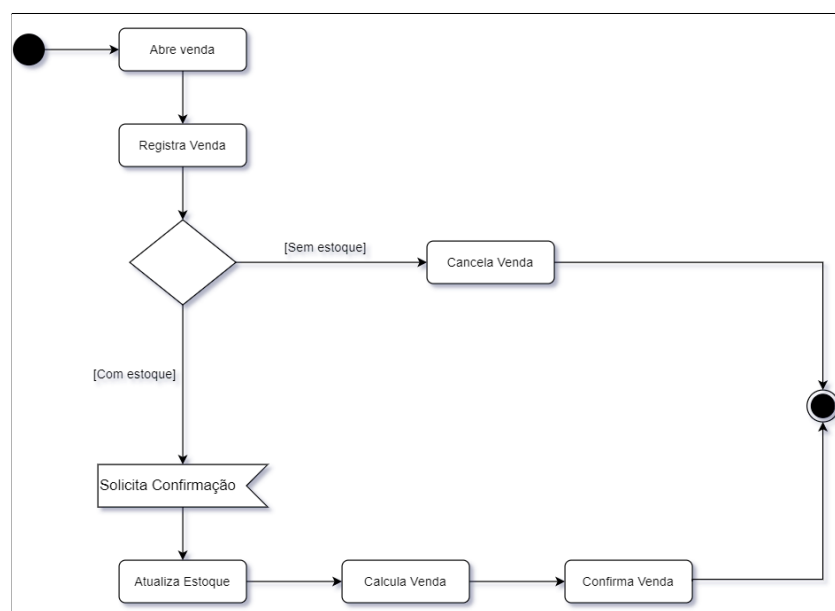



Figura 16. Diagrama de Atividade “Lançamento de Vendas”

Para consulta de dados e, conseqüentemente, contemplar o RF-5, as Figuras 17, 18 e 19 traz essa abordagem.

A Figura 17 apresenta a opção de consulta dos clientes que foram cadastrados. Nessa mesma tela há possibilidade do usuário executar as ações *update* e *delete* na base de dados, todavia respeitando as regras para manter a integridade da base de dados.

Base de Clientes




CNPJ	Razão Social	Nome Fantasia	Sócio	Contato	E-mail	CEP	Logradouro	Numero	Complem
36832944000172	SAMUEL GENOVA DE OLIVEIRA	CYCLE PLAST	SAMUEL GENOVA DE OLIVEIRA	(19) 9 9999-999	samugenova@hotmail.com	13181-610	AV AUGUSTA DIOGO AYALA	21	APT M21

Figura 17. Tela da funcionalidade "Consulta Clientes"

A Figura 18 também é uma opção de consulta, porém dos produtos que foram cadastrados. Nessa mesma tela estão disponíveis as opções de alterar e excluir algum produto na base de dados.

Consultar Produtos



Codigo	Categoria	Nome	Descrição	Preço	Editar	Deletar
1001	Laminado (PE/PET)	Bobina de PE com PET	Bobinas laminadas sem impressão	14	Editar	Deletar
1002	Laminado (PE/NYLON)	Bobina de Polinylon	Bobinas de PE com Nylon	9	Editar	Deletar
1003	Polietileno	Bobina de Polietileno	Polietileno puro	8	Editar	Deletar

Figura 18. Tela da funcionalidade "Consultar Produtos"

A Figura 19 apresenta a listagem de vendas para consulta do usuário sendo possível realizar a edição e a exclusão das vendas realizadas.

Base de Vendas



ID	Nome do Cliente	Cod	Nome Produto	QTDE	Data da Venda	Editar	Deletar
21	CYCLE PLAST	1001	Bobina de PE com PET	10	2022-03-17 15:35:16	Editar	Deletar

Figura 19. Tela da funcionalidade "Consultar Vendas"

A última funcionalidade a ser abordada é “gerar *backup*”, sendo essa executada através do *logout* do usuário no sistema. Foi implementada uma função que o *backup* da base de dados do sistema ocorra quando houver a desativação do usuário. A Figura 20 apresenta a listagem da implementação dessa funcionalidade, sendo desenvolvida em PHP com execução interna de comandos em SQL.

```
<?php
$dbHost = 'localhost';
$dbUsername = 'root';
$dbPassword = '';
$dbName = 'intranet';
$tables = '*';

$return = null;
$db = new mysqli($dbHost, $dbUsername, $dbPassword, $dbName);

if($tables == '*'){
    $tables = array();
    $result = $db->query("SHOW TABLES");
    while($row = $result->fetch_row()){
        $tables[] = $row[0];
    }
}else{
    $tables = is_array($tables)?$tables:explode(',',$tables);
}
foreach($tables as $table){
    $result = $db->query("SELECT * FROM $table");
    $numColumns = $result->field_count;

    $return .= "DROP TABLE $table;";

    $result2 = $db->query("SHOW CREATE TABLE $table");
    $row2 = $result2->fetch_row();

    $return .= "\n\n".$row2[1].";\n\n";

    for($i = 0; $i < $numColumns; $i++){
        while($row = $result->fetch_row()){
            $return .= "INSERT INTO $table VALUES(";
            for($j=0; $j < $numColumns; $j++){
                $row[$j] = addslashes($row[$j]);
                if (isset($row[$j])){
                    $return .= "'".$row[$j]."' ";
                } else {
                    $return .= "teste";
                }
                if ($j < ($numColumns-1)) {
                    $return.= ',';
                }
            }
            $return .= ");\n";
        }
    }

    $return .= "\n\n\n";
}
$handle = fopen('db-backup-' .time().'.sql','w+');
fwrite($handle,$return);
fclose($handle);
?>
```

Figura 20. Listagem da implementação da funcionalidade “gerar *backup*”

5. Resultados

As reuniões com o responsável da empresa Cycle Plast ocorreram periodicamente, ocorrendo de acordo com cada incremento finalizado. Os testes eram realizados na máquina de desenvolvimento sendo anotadas as melhorias que deveriam ser implementadas. De modo geral, os *feedbacks* foram excelentes levando em consideração os requisitos levantados para esta primeira versão do sistema de informação proposta.

Em relação ao modelo de desenvolvimento incremental utilizado, os testes de cada funcionalidade foram realizados ao final de cada incremento e foram aprovadas pelo cliente à medida que ficaram prontas.

Uma intercorrência no processo de implantação foi a solicitação, por parte do cliente, de acessar o sistema conectado em uma rede de internet fora da rede do escritório da empresa. Para solucionar o problema foi utilizada a ferramenta “Cisco Anyconnect” que permite acessar a rede de hospedagem do sistema e realizar o *login*.

6. Conclusão

O objetivo deste trabalho foi desenvolver um sistema *web* voltado para a gestão de estoque de uma pequena empresa buscando oferecer uma centralização de dados e informações da companhia através de cadastro, gerenciamento de estoque e lançamento de vendas. O sistema desenvolvido atendeu aos requisitos levantados, considerando os testes na implantação que seguiu o modelo incremental. Todavia, os testes por serem executados apenas com um usuário não foram documentados devido o *checklist* feito a cada incremento.

Para desenvolvimento deste trabalho e até mesmo destacando as competências e habilidades adquiridas no decorrer do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, houve aplicação de conceitos associados às disciplinas de Algoritmo e Lógica de Programação, Desenvolvimento de Sistemas *Web*, Banco de Dados, Engenharia de *Software*, além da Metodologia de Pesquisa e Projeto de Sistemas para planejamento e desenvolvimento teórico-técnico.

7. Trabalhos futuros

Para os trabalhos futuros, sugere-se:

- Desenvolver a integração entre o sistema *Intranet* da empresa CyclePlastic e o sistema da Receita Federal, com o objetivo de consultar o *status* dos clientes, assim como os dados cadastrados. Deve ser incluído essas informações na tabela cliente e fazer com que essa consulta seja realizada periodicamente.

- A integração com o sistema SINTEGRA para gerar as notas fiscais. O principal objetivo será auxiliar a empresa no faturamento das vendas. Em complemento, implementar também o envio dessa nota ao *e-mail* do cliente cadastrado na tabela clientes.

- O desenvolvimento de uma agenda com a possibilidade de cadastrar tarefas que irão emitir lembretes e alertas.

- Geração e exportação de relatórios em formato de planilha eletrônica e PDF.

A informatização desses processos permitirá facilitar tarefas rotineiras com maior eficiência.

8. Referências

ABIPLAST. (2021). “Estudo aponta que 23.1% dos resíduos plásticos pós-consumo foram reciclados em 2020 no Brasil”.

<http://www.abiplast.org.br/noticias/estudo-aponta-que-231-dos-residuos-plasticos-pos-consumo-foram-reciclados-em-2020-no-brasil/>. [Online: acessado em 10 de dezembro de 2021].

DB-Engines. (2022). “DB-Engines Ranking”. <https://db-engines.com/en/ranking>. [Online: acessado em 5 de janeiro de 2022].

Devmedia. (2022). “Guia Completo de SQL”.

<https://www.devmedia.com.br/guia/guia-completo-de-sql/38314>. [Online: acessado em 5 de janeiro de 2022].

Elmasri, R., Navathe, S. B. (2018). “Sistemas de Banco de Dados”. Editora Pearson, 7ª Edição.

Guedes, G. T. A. (2018). “UML 2 - Uma Abordagem Prática”. Editora Novatec, 3ª Edição.

Heuser, C. A. (2009). “Projeto de Banco de Dados”. Editora Bookman, 6ª Edição.

Laudon, K. C.; Laudon, J. P. (2014). "Sistemas de Informação Gerenciais". Editora Pearson Education do Brasil, 11ª Edição.

Minetto, E. L. (2007). “Frameworks para Desenvolvimento em PHP”. Editora Novatec.

Miranda, I. P. (2003). “Diagnóstico da Produtividade Sistêmica: uma análise da cadeia produtiva dos resíduos plásticos recicláveis do projeto reciclando”.

https://repositorio.ufc.br/bitstream/riufc/16286/1/2003_dis_ipmiranda.pdf. [Online: acessado em 7 de dezembro de 2021].

Niederauer, J. (2017). “PHP para quem conhece PHP”. Editora Novatec, 5ª Edição.

Oliveira, C. H. P. (2002). “SQL - Curso Prático”. Editora Novatec.

Piva, A. M.; Wiebeck, H. (2004). “Reciclagem do Plástico - Como Fazer da Reciclagem um Negócio Lucrativo”. Editora Artliber.

Pressman, R. S. (2011). “Engenharia de Software: uma abordagem profissional”. Editora McGraw Hill Brasil, 7ª Edição.

Puga, S.; França, E.; Goya, M. (2013). “Banco de dados: implementação em SQL, PL/SQL e Oracle 11g.”. Editora Pearson.

Recicloteca. (2020). “Plástico: história, composição, tipos, produção e reciclagem”. <https://www.recicloteca.org.br/material-reciclavel/plastico/>. [Online: acessado em 7 de dezembro de 2021].

Rodrigues, I. S.; Vassar, M. P. B; Almeida, F. S. (2020). “Reciclagem, reutilização e substituição dos plásticos”. <https://itr.ufrj.br/determinacaoverde/reciclagem-reutilizacao-e-substituicao-dos-plasticos/>. [Online: acessado em 10 de janeiro de 2022].

SINDIPLAST. (2021). “A indústria de plástico no Estado de São Paulo”. <http://www.sindiplast.org.br/dados-do-setor/>. [Online: acessado em 20 de dezembro de 2021].

Sommerville, I. (2019). “Engenharia de Software”. Editora Pearson, 10ª Edição.

Anexos

[ANEXO 1]: Códigos implementados em PHP para:

- Apresentação do mapa da empresa.

```
<tbody>
  <tr>
    <td width="10%"><i class=""></i></td>
    <iframe
      src="https://maps.google.com/maps?q=AV%20AUGUSTA%20DIOGO%20AYALA%2021&t=z=15&ie=UTF8&iwloc=&output=embed"
      width="979" height="450" style="border:0;" allowfullscreen=""
      loading="lazy">
    </iframe>
  </tr>
</tbody>
```

- Apresentação dos gráficos.

```
<?php
$servidor = "127.0.0.1";
$usuario = "root";
$senha = "";
$dbname = "intranet";

//Criar a conexao
$conn = mysqli_connect($servidor, $usuario, $senha, $dbname);
$x = 0;
$y = 0;
$z = 0;

//percorre o banco
$result_niveis_ava = "SELECT * FROM produto";
$resultado_niveis_ava = mysqli_query($conn, $result_niveis_ava);
while($row_niveis_ava = mysqli_fetch_assoc($resultado_niveis_ava)){
    if($row_niveis_ava['categoria_produto'] == "Laminado (PE/PET)"){
        $x++;
    } else if($row_niveis_ava['categoria_produto'] == "Laminado (PE/NYLON)"){
        $y++;
    } if($row_niveis_ava['categoria_produto'] == "Polietileno"){
        $z++;
    }
}
}

<html>
<head>
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
google.charts.load('current', {'packages': ['corechart']});
google.charts.setOnLoadCallback(drawChart);

function drawChart() {

    var data = google.visualization.arrayToDataTable([
        ['Task', 'Hours per Day'],
        ['Laminado (PE/PET)', <?=$x?>], //copia $x
        ['Laminado (PE/NYLON)', <?=$y?>], //copia $y
        ['Polietileno', <?=$z?> ] //copia $z
    ]);
    //titulo do gráfico
    var options = {
        title: 'Quantidade de produtos por Categoria'
    };

    var chart = new google.visualization.PieChart(document.getElementById('piechart'));
    chart.draw(data, options);
}
</script>
</head>
<body>
<div id="piechart" style="width: 900px; height: 500px;"></div>
</body>
```

Documento Digitalizado Restrito

Versão Final do TCC

Assunto: Versão Final do TCC
Assinado por: Michele Barion
Tipo do Documento: Outro
Situação: Finalizado
Nível de Acesso: Restrito
Hipótese Legal: Documento Preparatório - documentos de tomada de decisão ou do ato administrativo. (Art. 7º, § 3º, da Lei nº 12.527/2011)
Tipo do Conferência: Documento Original e Cópia

Documento assinado eletronicamente por:

- Michele Cristiani Barion, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 28/03/2022 21:38:22.

Este documento foi armazenado no SUAP em 28/03/2022. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifsp.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 929233

Código de Autenticação: ef29dddcae

