

## **APLICATIVO PARA DESCARTE DE ÓLEO –PROG ÓLEO**

**Gabriel Henrique Silva**

Centro Universitário de Anápolis - UniEVANGÉLICA. gabriel.hqs03@gmail.com

**Danilo Caetano Martins De Souza**

Centro Universitário de Anápolis – UniEVANGÉLICA. danilocaetano09@gmail.com

**Júlio César Caixeta Rodrigues da Cunha**

Centro Universitário de Anápolis - UniEVANGÉLICA. juliocesarcrc@hotmail.com

**Guilherme Henrique dos Santos Godoi**

Centro Universitário de Anápolis - UniEVANGÉLICA. guilherme.henrique.godoi@hotmail.com

**João Paulo Moreira**

Centro Universitário de Anápolis - UniEVANGÉLICA. joaopaulomoreira.app@gmail.com

**Marco Antônio Martins**

Centro Universitário de Anápolis - UniEVANGÉLICA. martins.carvalhoneto@gmail.com

### **Resumo**

O consumo de óleo de cozinha teve um aumento considerável por ser amplamente utilizado na preparação de alimentos, sendo em lanchonetes, restaurantes, casas, entre outros, e muitos não sabem onde e como realizar o descarte adequado desse material. O descarte inadequado pode causar vários problemas graves para o meio ambiente. Com base nessas questões, a plataforma foi projetada em buscar facilidades na hora de identificar pontos-chave para realizar o descarte correto, além de diminuir o tempo gasto na execução do mesmo. O Prog Óleo é um aplicativo que tem como intuito auxiliar no descarte de óleo informando os locais mais próximos. O aplicativo será simples e intuitivo. O sistema de busca utilizará como parâmetros de busca a atual localização do usuário e a categoria de descarte, essas que serão divididas em dois tipos de trocas: A troca do óleo por dinheiro ou benefícios como óleo novo, descontos de água, etc. e doações, que têm o intuito de arrecadar dinheiro e produtos para instituições carentes. O sistema de bate papo possibilita um meio de conversa utilizando criptografia entre o estabelecimento e usuário.

**Palavras-Chave:** Reciclagem de Óleo; Descarte de Óleo; Coleta de Óleo.

### **1. Introdução**

Devido ao óleo de cozinha ser amplamente utilizado na preparação de alimentos em lanchonetes, restaurantes, casas, entre outros existem pessoas que não sabem onde e como fazer o descarte adequado desse material. O descarte inadequado pode causar vários problemas graves para o meio ambiente. A plataforma que será desenvolvida, procura facilitar o encontro de pontos-chave para realizar o descarte correto, além de diminuir o tempo gasto na execução do mesmo.

Este projeto tem por objetivo criar uma plataforma web para facilitar o descarte de óleo para as pessoas e comércios, reduzir a contaminação de mananciais e melhorar o tratamento dos óleos descartáveis, incluindo uma série de ações para sensibilizar as pessoas sobre a importância do descarte correto. Para o incentivo do uso da plataforma e a conscientização, o aplicativo em maneiras de gratificar as pessoas que realizarem o descarte de maneira correta conforme orientados pela plataforma, ou seja, todo o óleo descartado de forma correta deverá ter um retorno diretamente para a pessoa de acordo com a quantidade.

De modo geral, o projeto tem como etapas a escolha da metodologia de desenvolvimento, a construção de requisitos, a concepção do banco de dados e por fim o desenvolvimento do software utilizando todos os artefatos produzidos.

## **2. Metodologia**

### **2.1. Modelo de Desenvolvimento**

Visando alcançar o objetivo, se faz necessário a estruturação do projeto como um todo, podendo suprir todas as necessidades para a entrega do artefato final. Para esta finalidade, estruturamos o desenvolvimento do projeto em etapas, tendo como início a concepção da ideia, passando pelo desenvolvimento e realizando a produção de artefatos que são norteadores para o desenvolvimento.

O primeiro passo a ser dado no projeto foi a escolha de uma metodologia para sistematizar o desenvolvimento do software. Para essa escolha, devemos ter como base os tipos aos quais os autores as classificam em duas modalidades: tradicionais - possuem foco na documentação de cada etapa do desenvolvimento; e ágeis - podendo ser considerada um novo paradigma de desenvolvimento de *software*, visando a redução de formalidades e priorizando a resposta rápida às mudanças de requisitos e ambiente [1].

Adotando como metodologia a abordagem ágil, tem-se como benefício seus artifícios e práticas para a satisfação de todos os envolvidos no projeto. Dentro desta modalidade, existe um *framework* bastante popular, cujo nome é Scrum, a qual apresenta uma abordagem sustentada pela experiência e pela observação, aplicando alguns conceitos da teoria de gerenciamento e controle de processos [2]. Considerada pertinente para o projeto e pela experiência dos integrantes da equipe, foi escolhido o Scrum para gerir as etapas do projeto.

Dentro de Scrum e de acordo com Franco [3], existem seis papéis que possuem tarefas e propósitos diferentes, sendo eles:

- **Cliente:** participa das tarefas relacionadas à definição da lista de funcionalidade do software sendo desenvolvido ou melhorado, elaborando os requisitos e restrições do produto final desejado.
- **Gerente:** é encarregado pela tomada das decisões finais, utilizando as informações visuais disponibilizadas graficamente pelos padrões e convenções a serem seguidas no projeto. Ele também é responsável por acordar, junto aos Clientes, os objetivos e requisitos do projeto.
- **Equipe Scrum:** é a equipe de projeto que possui autoridade de decidir sobre as ações necessárias e de se organizar para poder atingir os objetivos preestabelecidos. A Equipe Scrum é envolvida, por

exemplo, na estimativa de esforço, na criação e revisão da lista de funcionalidade do produto, sugerindo obstáculos que precisam ser removidos do projeto.

- **Scrum Master:** é responsável por garantir que o projeto esteja sendo conduzido de acordo com as práticas, valores e regras definidas no Scrum e que o progresso do projeto está de acordo com o desejado pelos Clientes. O Scrum Master interage tanto com a Equipe Scrum, como com os Clientes e o Gerente durante o projeto. Ele também é responsável por remover e alterar qualquer obstáculo ao longo do projeto, para garantir que a equipe trabalhe da forma mais produtiva possível.
- **Responsável pelo Produto:** é oficialmente responsável pelo projeto, gerenciamento, controle e por tornar visível a lista de funcionalidade do produto. Ele é selecionado pelo Scrum Master, Clientes e Gerente. Ele também é responsável por tomar as decisões finais referentes às tarefas necessárias para transformar a lista de funcionalidades no produto final, participando na estimativa do esforço de desenvolvimento necessário para o detalhamento das informações referentes à lista de funcionalidade utilizada pela Equipe Scrum.

Seguindo esse raciocínio, foi determinado para cada integrante dentro da equipe um papel. A seleção foi feita também utilizando apenas três papéis, sendo eles o Scrum Master, o *Product Owner* e a Equipe. A separação de papéis ficou como pode ser visto no quadro 1.

Quadro 1: Stakeholders

<b>CARGO</b>	<b>RESPONSABILIDADE</b>	<b>NOME</b>
<b>Scrum Master</b>	<b>Responsável pelo projeto</b>	<b>Gabriel Henrique</b>
<b>Product Owner</b>	<b>Responsável pelos requisitos do projeto</b>	<b>Júlio César</b>
<b>Equipe</b>	<b>Desenvolvedor principal</b>	<b>Danilo Caetano</b>
<b>Equipe</b>	<b>Desenvolvedor</b>	<b>Guilherme</b>
<b>Equipe</b>	<b>Desenvolvedor</b>	<b>João Paulo</b>
<b>Equipe</b>	<b>Desenvolvedor</b>	<b>Marco Antonio</b>

## 2.2. Requisitos

Parte da etapa de elicitação, os requisitos funcionais são todos os problemas e necessidades que devem ser atendidos e resolvidos pelo *software* por meio de funções ou serviços.

Levando em consideração a natureza do projeto e visando facilitar o acesso por pessoas tanto com baixo e com alto conhecimento em tecnologia, os requisitos foram definidos para tornar o acesso simples e

rápido. Com o cliente tendo acesso apenas a interface de cadastro e aos relatórios de trocas da reciclagem. Somente as empresas e administradores terão acesso aos dados dos usuários.

Visando facilitar ainda mais o acesso do cliente ao aplicativo, foi definido que o aplicativo seja responsivo para celular, *tablet*, *notebook* ou *desktop*, se ajustando para modo retrato e modo paisagem em celulares e *tablets*. Para um acesso sempre atualizado em tempo real, o aplicativo se comunicará com o banco de dados em nuvem sempre que for feito um novo acesso ao aplicativo.

Quadro 2: Requisitos (Funcionais e não funcionais)

RF.001	Interface com fácil visualização de forma objetiva. (Usar como exemplo a plataforma de <i>Streaming Twitch</i> )
RF.002	Cadastrar Usuários (Realizador do descarte e coletor)
RF.003	Gerar relatórios. (Relatório de reciclagem diário)

### Não Funcionais

Quadro 3: Segurança

RNF01.01	O sistema deverá permitir o login
RNF01.02	Somente os responsáveis da reciclagem terão acesso aos dados dos usuários

Quadro 4: Sistema

RNF02.01	O sistema deve ser executável através de celular, tablet, notebook ou desktop
RNF02.02	O sistema deve ser desenvolvido em React Native

Quadro 5: Comunicação

RNF03.01	O sistema deverá se comunicar com o banco SQL Server.
----------	---

Quadro 6: Usabilidade

RNF04.01	O sistema deve rodar em ambiente web e a interface deverá se comportar adequadamente independente do front-end que será utilizado para acesso – Browser, Smartphone ou Tablet.
----------	--

### **2.3. Caso de Uso**

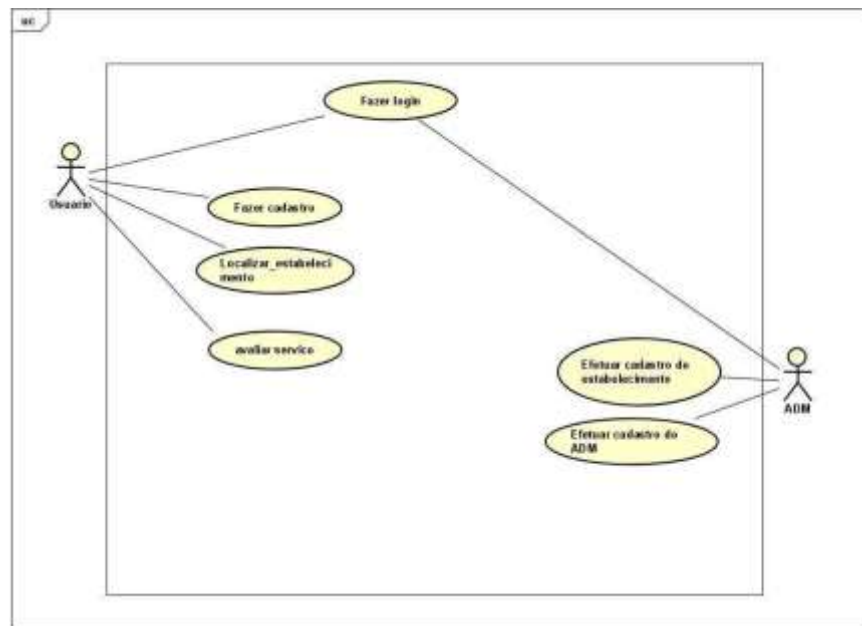
A Unified Modeling Language (UML) possui uma série de diagramas, cada um com uma finalidade específica com suas respectivas regras e restrições, que são usadas para aperfeiçoar um sistema. Esses esquemas ilustram o funcionamento de um projeto de forma organizada, dessa forma, facilitando a compreensão da equipe e a representação do trabalho, portanto os diagramas e modelos são fundamentais para a organização das funções e definição do foco do trabalho.

No decorrer do progresso do projeto foram desenvolvidos diagramas de todos os tipos para complementar o que é produzido, dentre eles está o de “Caso de Uso”, que serve para representar como os casos de uso interagem com o usuário e outros atores no decorrer do uso do programa, ou seja, esse diagrama é utilizado para ilustrar as funcionalidades de um sistema. Ele é composto por três elementos principais: Ator, Caso de Uso e Relacionamento.

O diagrama que representa o sistema do “Prog Óleo” possui dois atores: Usuário e ADM, com seus respectivos casos e relações que os ligam aos UCs (Use Cases). O ator “Usuário” será capaz de Fazer Cadastro/Login no aplicativo para manter seus dados salvos em usos posteriores, Localizar o Estabelecimento para efetuar o descarte do lixo, e Avaliar o Serviço prestado, estes são seus casos de uso. Já o ator ADM possui como casos de uso Fazer Login, Efetuar Cadastro do Estabelecimento para que o Usuário possa encontrá-lo, e Efetuar o Cadastro do ADM do próprio para que possa usufruir do aplicativo.

É notável a diferença na organização e entendimento que esse diagrama traz ao projeto. O Caso de Uso é uma das formas de especificar as funcionalidades de um software, facilitando seu desenvolvimento, é fundamental para a produção dos casos de teste, e auxilia na especulação de esforço/prazo/custo, além de ser um importante elemento na documentação do sistema.

Figura 1: Diagrama de Casos de Uso



## 2.4. Histórias de Usuário

Foi realizado um levantamento utilizando a técnica de história de usuário (user story) do Scrum. Para o melhor entendimento dos requisitos foi listado, conforme os quadros 7, 8, 9 e 10, cada critério de aceitação, que possuem objetivo similar ao caso de uso, ou seja, definir quais as ações necessárias que o sistema precisa desempenhar de acordo com possíveis usuários.

As HU 's são escritas em primeira pessoa do ponto de vista do usuário, contendo as ações que ela deseja realizar no sistema e qual a finalidade. Com essas informações, foram extraídos os critérios de aceitação, podendo assim ter noção do processo. A HU do quadro 7 discorre sobre a demanda do cliente tendo como necessidade a realização de acesso ao sistema, além de possíveis problemas como a validação de dados no login.

O quadro 8 orienta a respeito da principal funcionalidade do software,

HU – História de Usuário  
 CA – Critério de Aceitação

Quadro 7: História de Usuário 1

<b>HU.001</b>	<b>Efetuar Login</b>
História de Usuário:	<b>COMO</b> cliente <b>QUERO</b> efetuar login
	<b>PARA</b> ter a autorização necessária para executar as atividades dentro do sistema.
CA.001.0 1	<b>Acesso ao sistema</b> <b>DADO</b> que estou na tela de entrada do sistema; <b>E</b> estou cadastrado no sistema; <b>QUANDO</b> informo os dados para a autenticação; <b>E</b> clico em entrar; <b>ENTÃO</b> ocorre a validação os dados; <b>E</b> visualizo que a autenticação foi realizada com sucesso.
CA.001.0 2	<b>Problema na validação dos dados de login</b> <b>DADO</b> que estou na tela de entrada do sistema; <b>E</b> estou cadastrado no sistema; <b>QUANDO</b> cliquei em entrar; <b>E</b> ocorreu erro de validação de dados; <b>ENTÃO</b> os dados de login e senha são apagados; <b>E</b> visualizo a mensagem "Login ou senha incorretos!".

Quadro 8: História de Usuário 2

<b>HU.002</b>	<b>Localizar estabelecimento</b>
História de Usuário:	<b>COMO</b> cliente <b>QUERO</b> localizar estabelecimento <b>PARA</b> efetuar o descarte de óleo.
CA.002.0 1	<b>Encontrar local de descarte</b>



	<p><b>DADO</b> que estou na tela de localização de estabelecimento; <b>E</b> estou cadastrado no sistema; <b>QUANDO</b> procuro um estabelecimento; <b>ENTÃO</b> aparece o mapa com os locais dos estabelecimentos marcados;</p>
CA.002.0 2	<p><b>Local de descarte não localizado</b></p> <p><b>DADO</b> que estou na tela de localização de estabelecimento; <b>E</b> estou cadastrado no sistema; <b>QUANDO</b> procuro um estabelecimento; <b>E</b> não encontra nenhum estabelecimento; <b>Então</b> aparece uma mensagem dizendo “Ops... Não foi encontrado nenhum estabelecimento. Tente novamente mais tarde”</p>

Quadro 9: História de Usuário 3

<b>HU.003</b>	<b>Avaliar estabelecimento</b>
História de Usuário:	<p><b>COMO</b> cliente <b>QUERO</b> avaliar o estabelecimento <b>PARA</b> deixar a minha opinião.</p>
CA.003.0 1	<p><b>Deixar opinião</b></p> <p><b>DADO</b> que estou na tela de avaliação; <b>E</b> estou online no sistema; <b>QUANDO</b> clico em avaliar; <b>ENTÃO</b> aparece uma tela para deixar a minha avaliação;</p>

Quadro 10: História de Usuário 4

<b>HU.004</b>	<b>Efetuar cadastro estabelecimento</b>
História de Usuário:	<p><b>COMO</b> administrador <b>QUERO</b> cadastrar um estabelecimento <b>PARA</b> exibir aos clientes os locais de descarte de óleo.</p>
CA.004.0 1	<b>Campos preenchidos corretamente</b>



	<p><b>DADO</b> que estou na tela de cadastrar estabelecimento;  <b>E</b> logado como administrador;  <b>QUANDO</b> clico em salvar;  <b>E</b> todos os campos estão preenchidos corretamente;  <b>ENTÃO</b> o estabelecimento será salvo na base de dados;</p>
CA.004.0 2	<p><b>Campos não preenchidos corretamente</b></p> <p><b>DADO</b> que estou na tela de cadastrar estabelecimento;  <b>E</b> logado como administrador  <b>QUANDO</b> clico em salvar;  <b>E</b> caso algum campo não for preenchido corretamente;  <b>Então</b> aparece uma mensagem dizendo “Campo não foi preenchido corretamente. “Por favor verifique”</p>

## 2.5. Banco de Dados

O objetivo do modelo conceitual é criar um modelo de forma gráfica, sendo este chamado de Diagrama Entidade e Relacionamento (DER), que identifica todas as entidades e relacionamentos de uma forma global. Aqui é evitado qualquer detalhamento específico do modelo de BD, sua principal finalidade é capturar os requisitos de informação e regras de negócio sob o ponto de vista do negócio. No desenvolvimento de soluções é o primeiro modelo que deve ser desenvolvido. Na fase de levantamento de requisitos. Feito geralmente pelo Gestor de Dados de Negócio ou outro profissional acompanhado de sua supervisão/orientação. É independente de *hardware* ou *software*, ou seja, não depende de nenhum tipo de servidor de banco de dados tais como Sql Server, MySql, Oracle e Postgresql. Portanto, qualquer alteração no *software* ou *hardware*, não terão efeito no nível conceitual.

O modelo lógico já leva em conta algumas limitações e implementa recursos como adequação de padrão e nomenclatura, define as chaves primárias e estrangeiras, normalização, integridade referencial, entre outras. A modelagem lógica é principalmente associada à coleta de necessidades de negócios, e não ao design do banco de dados. As informações que precisam ser coletadas são sobre unidades organizacionais, entidades de negócios e processos de negócios. A Figura 2 descreve como os dados serão armazenados no banco e seus relacionamentos. Os modelos lógicos basicamente determinam se todos os requisitos do negócio foram reunidos. Ele é revisado pelos desenvolvedores, pelo gerenciamento e, por fim, pelos usuários finais para ver se é necessário coletar mais informações antes do início da modelagem física.

Figura 2: Modelo Conceitual

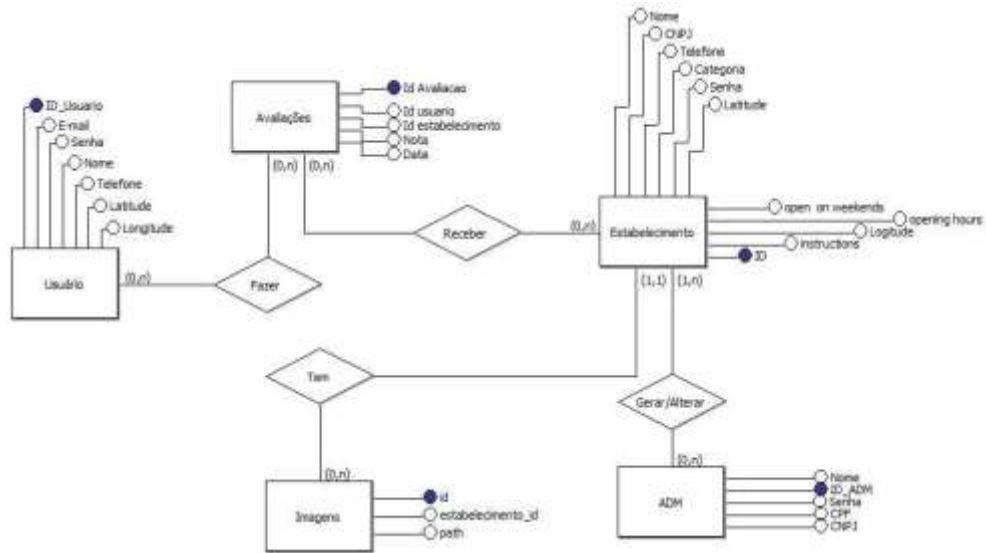
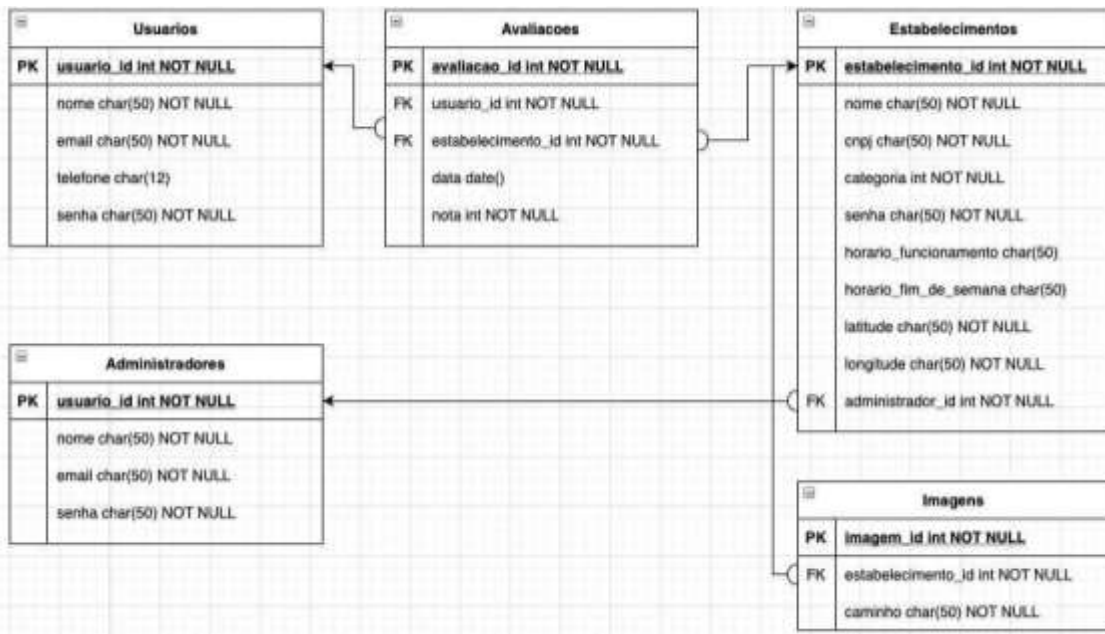


Figura 3: Modelo Lógico



### 3. Resultados e discussão

O resultado esperado do projeto é a plataforma rodando em produção. A etapa a qual o projeto se encontra é a prototipação das telas, ao qual foi utilizado a ferramenta Figma e as boas práticas de UI e UX para a criação das mesmas.

#### **4. Conclusão**

O projeto trouxe bastante conhecimento por ter a possibilidade de mesclar artefatos em um projeto e realizar o desenvolvimento do mesmo, principalmente na disciplina de Análise de Projeto de Software por ter maior parte teórica e voltada principalmente na criação dos casos de usos e outros diagramas comportamentais e estruturais. Em banco de dados também houve a necessidade e o aprimoramento na criação dos três tipos de modelos, o modelo lógico e conceitual. O desenvolvimento da plataforma irá trazer benefícios para as pessoas na parte de aproveitamento do óleo, conseguindo ajudar o meio ambiente e na troca ganhando recompensas.

#### **5. Referências**

- [1] SCHWABER, K. BEEDLE, M. **Agile Software Development with Scrum**. Prentice Hall, 2002.
- [2] SCHWABER, K. **Agile Project Management with Scrum**. Microsoft Press. 1a Edição, 2004.
- [3] FRANCO, E. F.: **Um modelo de gerenciamento de projetos baseado nas metodologias ágeis de desenvolvimento de software e nos princípios da produção enxuta**. Escola Politécnica da Universidade de São Paulo (Dissertação de Mestrado). 2007.
- [4] VENTURA, Plínio. **O que é UML (Unified Modeling Language)**. 31 de Jan. de 2019. Disponível em: <https://www.ateomomento.com.br/diagramas-uml/>. Acesso em: 01 de Jun. 2021.