

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

**GUILHERME CAMARGO VAZ**

**UM SISTEMA PARA ANÁLISE DE ORDENS DE SERVIÇO**

**PONTA GROSSA**

**2023**

**GUILHERME CAMARGO VAZ**

**UM SISTEMA PARA ANÁLISE DE ORDENS DE SERVIÇO**

**A SYSTEM FOR ANALYSIS OF SERVICE ORDER**

Trabalho de conclusão de curso de graduação apresentada como requisito para obtenção do título de Tecnólogo em Análise e desenvolvimento de sistemas da Universidade Tecnológica Federal do Paraná (UTFPR).

Orientador: Prof. Dr. Richard Duarte Ribeiro

Coorientador: Prof. MSc. Vinícius Camargo Andrade

**PONTA GROSSA**

**2023**



[4.0 Internacional](https://creativecommons.org/licenses/by-nc/4.0/)

Esta licença permite remixe, adaptação e criação a partir do trabalho, para fins não comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

**GUILHERME CAMARGO VAZ**

**UM SISTEMA PARA ANÁLISE DE ORDENS DE SERVIÇOS**

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Tecnólogo em Análise e desenvolvimento de sistemas da Universidade Tecnológica Federal do Paraná (UTFPR).

Data de aprovação: 23/maio/2023

---

Richard Duarte Ribeiro  
Doutorado  
Universidade Tecnológica Federal do Paraná, Campus Ponta Grossa

---

Vinícius Camargo Andrade  
Mestrado  
Universidade Tecnológica Federal do Paraná, Campus Ponta Grossa

---

Luiz Rafael Schmitke  
Mestrado  
Universidade Tecnológica Federal do Paraná, Campus Ponta Grossa

---

Diego Roberto Antunes  
Doutorado  
Universidade Tecnológica Federal do Paraná, Campus Ponta Grossa

**PONTA GROSSA**

**2023**

## RESUMO

O setor de serviços desempenha um papel significativo no Produto Interno Bruto do Brasil, demonstrando sua relevância econômica. No entanto, um nicho pouco explorado, porém de extrema importância, é o de manutenção industrial, que demanda uma análise minuciosa dos custos envolvidos. Essa análise serve como base para orientar as tomadas de decisões de modo estratégico, pois visa reduzir os custos de reparos e melhorar o controle das despesas. Nesse sentido, a ordem de serviço surge como uma ferramenta essencial para essa análise, permitindo descrever, registrar, precificar e especificar as tarefas a serem realizadas para um cliente, levando assim a uma visão mais precisa dos custos envolvidos em cada serviço. Com o intuito de aprimorar a eficiência e modernizar os processos, este trabalho apresenta o desenvolvimento de um sistema customizado de controle de ordens de serviços em uma plataforma Desktop, para uma empresa dos Campos Gerais, no estado do Paraná. A escolha dessa plataforma ocorreu devido ao uso já consolidado da mesma pela empresa consultada. O sistema proposto diferencia-se das abordagens tradicionais, como planilhas de Excel e arquivos físicos, oferecendo uma solução mais eficaz para a gestão e análise de custos. Ao utilizar a ordem de serviço como base, o sistema permite a identificação e categorização dos elementos que compõem os custos dos serviços prestados, incluindo custos diretos, como materiais e mão de obra, e custos indiretos, como alimentação e transporte. O desenvolvimento do sistema foi realizado em uma empresa que utiliza a ferramenta RAD Studio com a linguagem Delphi, devido à familiaridade e experiência com essa tecnologia, garantindo um sistema robusto e de qualidade. Além disso, o sistema inclui a implementação de um *Web Service*, visando facilitar a transição para plataformas Web e Mobile, a fim de proporcionar maior flexibilidade e escalabilidade. Dessa forma, o sistema de controle de ordens de serviços desenvolvido para plataforma Desktop representa uma solução inovadora para a gestão e análise de custos no setor de manutenção industrial. Após o desenvolvimento, o Sistema foi entregue à empresa para testes, aprovação e implementação.

Palavras-chave: manutenção industrial; análise de custo; serviços; orçamento.

## **ABSTRACT**

The service sector plays a significant role in Brazil's Gross Domestic Product, demonstrating its economic relevance. However, a little-explored but extremely important niche is industrial maintenance, which demands a thorough analysis of the costs involved. This analysis serves as a basis for strategic decision-making, as it aims to reduce repair costs and improve expense control. In this sense, the service order emerges as an essential tool for this analysis, allowing to describe, record, price and specify the tasks to be performed for a client, thus leading to a more accurate view of the costs involved in each service. In order to improve efficiency and modernize processes, this work presents the development of a customized service order control system on a Desktop platform, for a company in Campos Gerais, in the state of Paraná. The choice of this platform was due to the already consolidated use of it by the consulted company. The proposed system differs from traditional approaches, such as Excel spreadsheets and physical files, offering a more effective solution for cost management and analysis. By using the service order as a basis, the system allows the identification and categorization of the elements that make up the costs of the services provided, including direct costs, such as materials and labour, and indirect costs, such as food and transportation. The development of the system was carried out in a company that uses the RAD Studio tool with the Delphi language, due to the familiarity and experience with this technology, guaranteeing a robust and quality system. In addition, the system includes the implementation of a Web Service, aimed at facilitating the transition to Web and Mobile platforms, in order to provide greater flexibility and scalability. In this way, the service order control system developed for the Desktop platform represents an innovative solution for the management and analysis of costs in the industrial maintenance sector. After completion, the system was delivered to the company for testing, approval and use.

**Keywords:** industrial maintenance; cost analysis; services; budget.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplo de MVC .....	12
Figura 2 – Exemplo de URL .....	15
Figura 3 – Exemplo de DNS.....	16
Figura 4 – Exemplo de Body .....	17
Figura 5 – Exemplo de JSON.....	19
Figura 6 – Síntese dos diagramas da UML.....	20
Figura 7 – Diagrama de Caso de Uso Ordem de Serviço.....	25
Figura 8 – Diagrama de Estado Ordem de Serviço .....	28
Figura 9 – Tela de Login.....	30
Figura 10 – Menus da Tela Principal .....	30
Figura 11 – Aba de consulta da tela de Pessoa .....	31
Figura 12 – Aba de Cadastro de Pessoa .....	31
Figura 13 – Tela de Cadastro de Itens .....	32
Figura 14 – Tela de Cadastro de País .....	33
Figura 15 – Tela de Cadastro de Ordem de Serviço .....	33
Figura 16 – Aba de Análise de custo .....	34
Figura 17 – Diagrama de Classe <i>Web Service</i> .....	53
Figura 18 – Representação do Modelo Lógico .....	55

## LISTA DE QUADROS

<b>Quadro 1 – Propriedades ACID .....</b>	<b>13</b>
<b>Quadro 2 – Partes da URL .....</b>	<b>15</b>
<b>Quadro 3 – Explicação informações do Header .....</b>	<b>16</b>
<b>Quadro 4 – Tipos de requisições previstas no HTTP .....</b>	<b>18</b>
<b>Quadro 5 – Descrição do Caso de Uso “Cadastrar Produto” .....</b>	<b>26</b>
<b>Quadro 6 – Descrição Textual do caso de uso “Abrir Tela” .....</b>	<b>40</b>
<b>Quadro 7 – Descrição Textual do caso de uso “Sair Tela” .....</b>	<b>40</b>
<b>Quadro 8 – Descrição Textual do caso de uso “Buscar” .....</b>	<b>41</b>
<b>Quadro 9 – Descrição Textual do caso de uso “Definir Modelo” .....</b>	<b>41</b>
<b>Quadro 10 – Descrição Textual do caso de uso “Iniciar” .....</b>	<b>42</b>
<b>Quadro 11 – Descrição Textual do caso de uso “Concluir” .....</b>	<b>42</b>
<b>Quadro 12 – Descrição Textual do caso de uso “Faturar” .....</b>	<b>43</b>
<b>Quadro 13 – Descrição Textual do caso de uso “Reprovar” .....</b>	<b>44</b>
<b>Quadro 14 – Descrição Textual do caso de uso “Aprovar” .....</b>	<b>44</b>
<b>Quadro 15 – Descrição Textual do caso de uso “Cadastrar Custo” .....</b>	<b>45</b>
<b>Quadro 16 – Descrição Textual do caso de uso “Cadastrar Item” .....</b>	<b>46</b>
<b>Quadro 17 – Descrição Textual do caso de uso “Cadastrar Produto” .....</b>	<b>47</b>
<b>Quadro 18 – Descrição Textual do caso de uso “Cadastrar Custos de Funcionários” .....</b>	<b>49</b>
<b>Quadro 19 – Descrição Textual do caso de uso “Analisar Custo” .....</b>	<b>50</b>
<b>Quadro 20 – Descrição Textual do caso de uso “Replicar” .....</b>	<b>51</b>

## LISTA DE ABREVIATURAS E SIGLAS

ACID	<i>Atomicity, Consistency, Isolation, and Durability</i>
AJAX	<i>Asynchronous JavaScript and XML</i>
CRUD	<i>Create, Read, Update e Delete</i>
DNS	<i>Domain Name System</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IDE	<i>Integrated Development Environment</i>
JSON	<i>JavaScript Object Notation</i>
LSP	<i>Language Server Protocol</i>
MDA	<i>Model Driven Architecture</i>
MVC	<i>Model-View-Controller</i>
PIB	Produto Interno Bruto
SGBD	Software de Gerenciamento de Banco de Dados
SOAP	<i>Simple Object Access Protocol</i>
SQL	<i>Structure Query Language</i>
UML	<i>Unified Modelling Language</i>
URL	<i>Uniform Resource Locator</i>

## SUMÁRIO

<b>1.</b>	<b>INTRODUÇÃO</b>	<b>10</b>
1.1.	Objetivo Geral	11
1.2.	Objetivos Específicos	11
<b>2.</b>	<b>FERRAMENTAS UTILIZADAS</b>	<b>12</b>
2.1.	RAD Studio	12
2.2.	MVC	12
2.3.	Banco de Dados	13
2.3.1.	MySQL	14
2.4.	HTTP	14
2.4.1.	URL	14
2.4.2.	DNS	15
2.4.3.	Header	16
2.4.4.	Body	17
2.4.5.	Tipos de Solicitações	17
2.5.	JSON	18
2.6.	UML	19
2.7.	StarUML	21
<b>3.</b>	<b>DESENVOLVIMENTO</b>	<b>22</b>
3.1.	Modelagem do Banco de Dados	22
3.1.1.	Regras semânticas	22
3.1.2.	Estrutura do Banco de Dados	23
3.2.	Modelagem dos Caso de Uso	24
3.2.1.	Descrição dos Casos de Uso	26
3.3.	Modelagem dos Diagramas de Classe	28
3.4.	Modelagem dos Diagramas de Máquina de Estados	28
<b>4.</b>	<b>RESULTADO</b>	<b>30</b>
4.1.	Sistema desenvolvido	30
<b>5.</b>	<b>CONCLUSÃO</b>	<b>35</b>
5.1.	Trabalhos Futuros	36

## 1. INTRODUÇÃO

O setor de serviços no Brasil correspondeu a aproximadamente 75% do PIB (Produto Interno Bruto) entre os anos de 2017 e 2019, sendo notável a relevância deste setor no contexto econômico, conforme descrito por Bezerra (2020).

Dentre os nichos de atuação das empresas de serviços, um pouco notado é o de manutenção industrial. Conforme Megiolaro (2015), este setor a cada dia que passa, vem ocupando uma posição de maior destaque na estrutura das organizações produtoras de bens e serviços, que têm como objetivo assegurar altos padrões de confiabilidade e qualidade.

Com isso, observa-se que a manutenção é fundamental para garantir o pleno funcionamento dos equipamentos e a continuidade da produção, conforme analisado por Oliveira (2020). No entanto, é importante ressaltar que essa atividade gera despesas consideráveis para as empresas.

Diante desse contexto, as empresas de prestação de serviço enfrentam o desafio de realizar uma análise minuciosa dos custos envolvidos, levando em conta todos os aspectos que o compõem, desde o serviço prestado até os fatores inerentes à própria manutenção, como transporte, hospedagem, alimentação e horas trabalhadas. Essa análise de custos desempenha um papel fundamental ao avaliar a lucratividade obtida pela empresa e, ainda, serve como base para orientar as tomadas de decisões estratégicas. Em conjunto com essa análise, as empresas prestadoras de serviços podem aproveitar o valor gerado pela manutenção e potencializar seus lucros através de uma eficiente gestão das ordens de serviço.

Enfrentando um desafio, uma empresa dos Campos Gerais, no estado do Paraná, que presta serviços de manutenção industrial para outras empresas da região, identificou a necessidade de melhorar a análise de seus serviços e o controle dos gastos envolvidos em cada um deles. Para solucionar esse problema, a empresa solicitou o desenvolvimento de um sistema que pudesse oferecer maior eficiência na análise e controle dos serviços prestados e dos custos da manutenção industrial.

Em resposta a essa demanda da referida empresa paranaense, este trabalho teve como objetivo desenvolver um Sistema que atendesse à ambas as necessidades em uma plataforma Desktop, devido à familiaridade de uso e pela preferência da empresa solicitante, visando a análise minuciosa dos custos envolvidos na manutenção industrial.

### **1.1. Objetivo Geral**

Desenvolver um sistema gerenciador de ordens de serviço para mensurar as variáveis: tempo de elaboração e custos dos serviços prestados.

### **1.2. Objetivos Específicos**

- Levantar os requisitos do sistema;
- Modelar os diagramas UML;
- Desenvolver o sistema.

## 2. FERRAMENTAS UTILIZADAS

Este capítulo apresenta as ferramentas e tecnologias utilizadas no desenvolvimento deste trabalho.

### 2.1. RAD Studio

O RAD Studio é uma IDE (*Integrated Development Environment*) para a construção de aplicações Windows, Android, IOS e MacOS na linguagem Delphi ou C++ a partir de um único código-fonte (EMBARCADEIRO, 2022).

A ferramenta possui um suporte LSP (*Language Server Protocol*), para oferecer um código-fonte mais rápido e mais confiável, também é possível realizar integrações com serviços *Web SOAP (Simple Object Access Protocol)* para comunicação entre aplicações desenvolvidas em diferentes linguagens e arquiteturas de software. (EMBARCADEIRO, 2022).

### 2.2. MVC

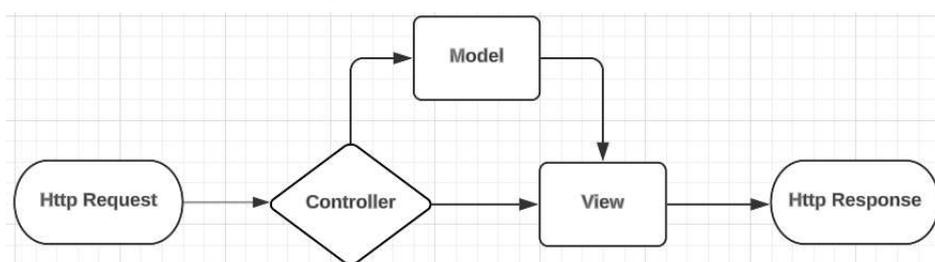
O padrão Modelo, Visão e Controle ou MVC (*Model View Controller*) surgiu na década de 80, para gerenciar o fluxo da aplicação, aumentar a capacidade de reutilização e facilitar a manutenção do projeto (LUCIANO, 2017).

Segundo Luciano (2017), a dinâmica do MVC é simples, a camada de controle processa todas as informações, a camada de modelo acessa o banco de dados, e por fim os resultados são exibidos na camada de visão, conforme a Figura 1.

Essa ferramenta foi utilizada para facilitar a padronização, reutilização e manutenção do código-fonte do *Web Service*, onde cada camada é responsável por:

- Modelo: persistência com o banco de dados;
- Controle: Regra de negócio do sistema;
- Visão: interação com o usuário.

Figura 1 – Exemplo de MVC



Fonte: Autoria Própria (2023)

### 2.3. Banco de Dados

Segundo a empresa Oracle (2023), um banco de dados é um conjunto de informações organizadas, comumente armazenadas em um sistema de arquivos controlado por um SGBD (Software de Gerenciamento de Banco de Dados).

Para a Oracle (2023), existem vários tipos de bancos de dados disponíveis no mercado, utilizados segundo a necessidade do projeto, por exemplo:

- Bancos de dados relacionais;
- Bancos de dados orientados a objetos;
- Bancos de dados NoSQL;
- Bancos de dados de grafos.

Para o desenvolvimento deste trabalho optou-se por um banco de dados relacional, ele utiliza a linguagem SQL (*Structure Query Language*), o MySQL<sup>1</sup>.

Segundo a Oracle (2023), nos bancos de dados relacionais as informações são armazenadas em linhas e colunas, em uma série de tabelas para tornar o processamento dos dados mais eficiente.

Para garantir a consistência das informações, os bancos de dados relacionais utilizam a propriedade ACID (*Atomicity, Consistency, Isolation, and Durability*) que são detalhados no quadro a seguir (ORACLE, 2020).

**Quadro 1 – Propriedades ACID**

<b>Propriedade</b>	<b>Descrição</b>
Atomicidade	Define que toda a transação deve ser executada com sucesso, caso uma falha ocorra, nenhum resultado deve ser refletido na base de dados.
Consistência	Define que uma transação deve levar a base de dados de um estado consistente para outro estado consistente, respeitando toda as regras de integridade definida nas tabelas a serem modificadas.
Isolamento	Um conjunto de regras que permite que vários usuários realizem transações simultâneas e em um único registro obtenham o mesmo resultado como se as transações fossem realizadas uma após a outra.
Durabilidade	Define que os efeitos de uma transação em caso de sucesso devem persistir no banco de dados mesmo em casos de falhas externas, garantindo assim que os dados estarão disponíveis em definitivo.

**Fonte: Adaptado de (Oracle, 2020)**

<sup>1</sup> <https://www.mysql.com/>

### 2.3.1. MySQL

MySQL é um SGBD relacional criado na década de 90 por Michael Widenius, David Axmark e Allan Larsson, de dupla licença (sendo uma de software livre e outra paga) desenvolvido para atender aplicações de grande porte. O MySQL utiliza a linguagem SQL para realizar os comandos de *CRUD* (*Create, Read, Update e Delete*). A ferramenta é multiplataforma tendo compatibilidade com os seguintes sistemas operacionais: Linux, Unix, FreeBSD, Mac OS e Windows (MILANI, 2007).

Para o desenvolvimento deste trabalho optou-se por utilizar a versão 8.0.30.

## 2.4. HTTP

O HTTP (*Hypertext Transfer Protocol*) é um protocolo formado por um conjunto de regras TCP/IP criado em 1990 para transferência de dados na internet, através de *Request/Response* (Requisição/Respostas), entre um cliente e um servidor, permitindo que um conjunto aberto de métodos seja utilizado para indicar a finalidade de um pedido (BERNERS-LEE, 1996).

Por exemplo, o cliente pode efetuar uma solicitação através de uma URL (*Uniform Resource Locator*) para o servidor localizado em um endereço IP, enviando um pacote de dados, contendo um *Header* (Cabeçalho) e *Body* (Corpo), através de uma requisição POST. O servidor deve retornar uma linha de *status* com o retorno da operação solicitada, de 200 a 202 (Sucesso) ou 400 a 499 (Erro).

### 2.4.1. URL

Segundo W3 (2023), a URL é o endereço eletrônico que permite que qualquer site seja encontrado na internet, sendo composto por *scheme* (esquema), *prefix* (prefixo), *domain* (domínio), *port* (porta), *path* (caminho) e *filename* (nome do arquivo), que são detalhados no Quadro a seguir.

Quadro 2 – Partes da URL

Parte	Descrição
<i>scheme</i>	Define qual é o tipo de protocolo de rede utilizado: <ul style="list-style-type: none"> <li>• <b>HTTP/HTTPS</b>: utilizado para comunicação entre cliente e servidor;</li> <li>• <b>DHCP</b>: utilizado para definição de endereço IP de máquinas;</li> <li>• <b>FTP/SFTP/SSH</b>: utilizado para transferência de arquivos;</li> <li>• <b>POP3/SMTP/IMAP</b>: utilizados para transferência de e-mails.</li> </ul>
<i>prefix</i>	O prefixo é utilizado para dividir um website em seções, como por exemplo: <ul style="list-style-type: none"> <li>• <b>WWW</b>: sites gerais, essa seção é a mais comum encontrada na internet;</li> <li>• <b>mail</b>: utilizado login de provedores de email;</li> <li>• <b>receita</b>: utilizado para sites da receita federal do Brasil;</li> <li>• <b>moodle</b>: utilizado para sites voltados para a área acadêmica.</li> </ul>
<i>domain</i>	Define o nome do domínio, por exemplo: google, utfpr, wikipedia.
<i>port</i>	Define o número da porta no servidor, geralmente não sendo visível para o usuário, por exemplo: 80.
<i>path</i>	Define o caminho do servidor, se omitido vai para o diretório raiz do site
<i>filename</i>	Define o arquivo a ser utilizando, geralmente não sendo visível para o usuário, como por exemplo: index.php.

Fonte: Adaptado de (W3, 2023)

A Figura 2 demonstra um exemplo de URL sendo o HTTPS a *scheme*, sistemas2 o *prefix*, utfpr o *domain* e login o *path*.

Figura 2 – Exemplo de URL

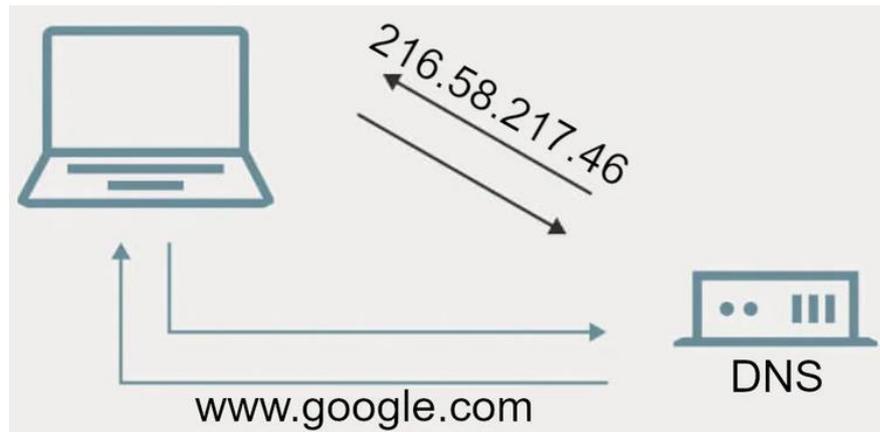


Fonte: Autoria Própria (2023)

#### 2.4.2. DNS

O DNS (*Domain Name System*) é uma ferramenta utilizada na conversão de um nome (URL) legível, por exemplo, www.google.com para um endereço IP interpretado por máquina, como 216.58.217.46, conforme mostra a Figura 3.

Figura 3 – Exemplo de DNS



Fonte: CANALTECH (2022)

### 2.4.3. Header

Segundo Mozilla (2023) o *Header* (cabeçalho) é um componente obrigatório a ser enviado na requisição e na resposta de uma solicitação. O Quadro 3 explica algumas das informações possíveis de existir no *Header*.

Quadro 3 – Explicação informações do Header

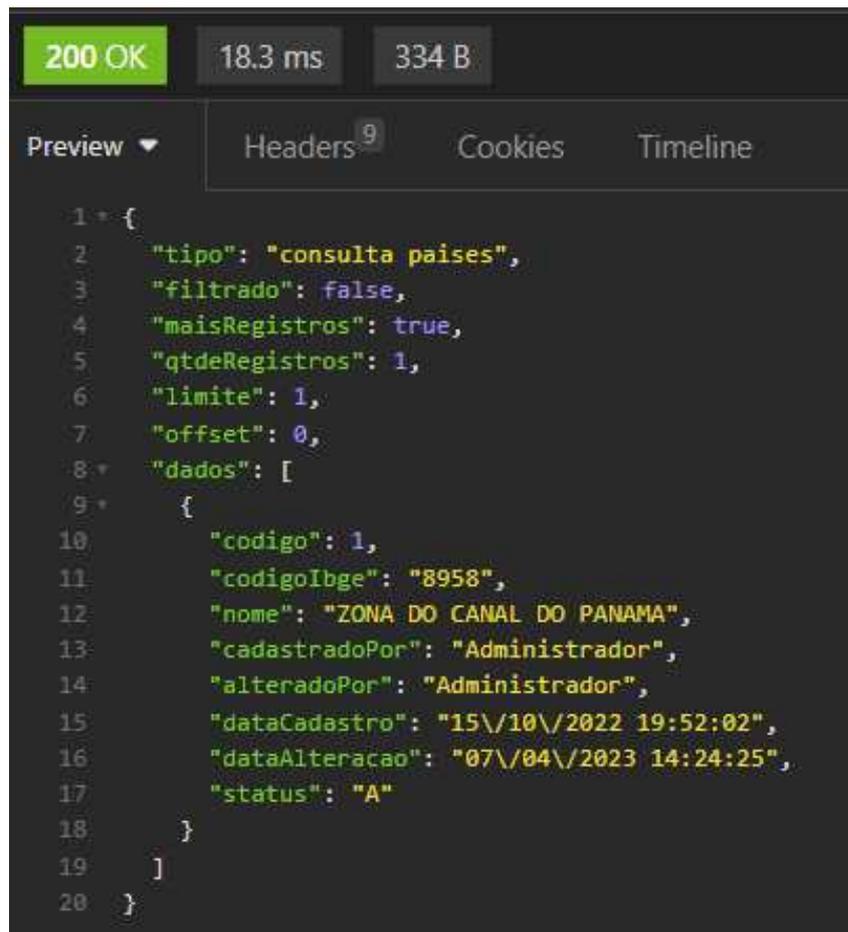
Informação	Descrição
Connection	Define se a conexão de rede permanece aberta após o término da transação atual, exemplo: Close, Open
Content-Type	Indica o tipo de mídia da resposta da solicitação exemplo: application/json; charset=UTF-8
Content-Length	Informa o tamanho do body (corpo da resposta) em bytes, exemplo: 51947
Date	Informa a data e hora que a mensagem foi criada, exemplo: Fri, 21 Apr 2023 12:41:23 GMT
Access-Control-Allow-Origin	Indica se a resposta pode ser compartilhada com o código do solicitante, exemplo: *
Access-Control-Allow-Credentials	Informa aos navegadores se devem expor a resposta ao código JavaScript frontend quando o modo de credenciais do pedido Request.credentials = include e o valor desta propriedade seja True.
Access-Control-Allow-Headers	Indica quais cabeçalhos HTTP podem ser usados durante a solicitação, exemplo: *.
Access-Control-Allow-Methods	Especifica um ou mais métodos permitidos a acessar um recurso em resposta a uma requisição, exemplo: *.
Access-Control-Expose-Headers	Indica quais cabeçalhos de resposta devem ser disponibilizados para scripts em execução no navegador.

Fonte: Adaptado de (Mozilla, 2023)

#### 2.4.4. Body

Segundo Mozilla (2023), componentes opcionais presente nas respostas de requisição a algum servidor, como DELETE, HEAD ou OPTIONS, não devem possuir um *Body*, as demais solicitações como GET, POST ou PUT podem possuir um *Body* a fim de trazer informações ao solicitante. A Figura 4 demonstra a resposta de uma requisição de consulta de países.

Figura 4 – Exemplo de Body



```
200 OK 18.3 ms 334 B
Preview Headers Cookies Timeline
1 {
2   "tipo": "consulta paises",
3   "filtrado": false,
4   "maisRegistros": true,
5   "qtdeRegistros": 1,
6   "limite": 1,
7   "offset": 0,
8   "dados": [
9     {
10      "codigo": 1,
11      "codigoIbge": "8958",
12      "nome": "ZONA DO CANAL DO PANAMA",
13      "cadastradoPor": "Administrador",
14      "alteradoPor": "Administrador",
15      "dataCadastro": "15\10\2022 19:52:02",
16      "dataAlteracao": "07\04\2023 14:24:25",
17      "status": "A"
18    }
19  ]
20 }
```

Fonte: Autoria Própria (2023)

#### 2.4.5. Tipos de Solicitações

Segundo Mozilla (2023), o protocolo HTTP (*Hypertext Transfer Protocol*) define um conjunto de métodos de requisições para indicar a ação que deve ser realizada no servidor. O Quadro 4 mostra todos os métodos previstos no protocolo HTTP.

**Quadro 4 – Tipos de requisições previstas no HTTP**

Requisição	Descrição
GET	O método GET solicita a representação de um recurso específico. Requisições utilizando o método GET devem retornar apenas dados.
POST	O método POST é utilizado para submeter uma entidade a um recurso específico, frequentemente causando uma mudança no estado do recurso ou efeitos colaterais no servidor.
PUT	O método PUT substitui todas as atuais representações do recurso de destino pela carga de dados da requisição.
OPTIONS	O método OPTIONS é usado para descrever as opções de comunicação com o recurso de destino.
PATCH	O método PATCH é utilizado para aplicar modificações parciais em um recurso.
HEAD	O método HEAD solicita uma resposta de forma idêntica ao método GET, porém sem conter o corpo da resposta.
DELETE	O método DELETE remove um recurso específico.
CONNECT	O método CONNECT estabelece um túnel para o servidor identificado pelo recurso de destino.
TRACE	O método TRACE executa um teste de chamada <i>loop-back</i> junto com o caminho para o recurso de destino.

Fonte: Adaptado de (MOZILLA, 2023)

Para o desenvolvimento deste projeto optou-se por utilizar os métodos HTTP GET para consulta, POST para inserção, PUT para atualização e DELETE para exclusão de dados.

## 2.5. JSON

JSON (*JavaScript Object Notation*) foi criado no ano de 2000 por Douglas Crockford com a finalidade de ser utilizado para trabalhar com AJAX (*Asynchronous JavaScript and XML*), permitindo um carregamento de dados assíncrono, sendo um formato para troca de dados baseado na padronização ECMAScript<sup>2</sup>.

Segundo Goyal (2019), um objeto JSON é formado por um conjunto de *keys* (chaves) e *values* (valores), o qual obrigatoriamente deve ser iniciado e finalizado por chaves (“{”, ”}”). As *keys* devem ser informadas em formato de *string* e cercada por aspas duplas. A separação entre as chaves e os valores é feita através de dois pontos “:”. Os *Values* podem assumir os seguintes formatos:

- *Array* (Lista): delimitado por [];
- *Object* (JSON): delimitado por {};
- *String* (Texto): delimitado por “”;

<sup>2</sup> <https://www.ecma-international.org/publications-and-standards/standards/ecma-262/>

- *Boolean* (Verdadeiro ou Falso): sem delimitação;
- *Number* (Número): sem delimitação.

A separação entre um conjunto chave valor deve ser feita através de uma vírgula (",") com exceção do último conjunto, conforme exemplifica a Figura a seguir, demonstrando toda a estrutura de um arquivo JSON.

Figura 5 – Exemplo de JSON

```
{
  "tipo": "consulta estados",
  "filtrado": false,
  "maisRegistros": true,
  "qtdeRegistros": 1,
  "limite": 1,
  "offset": 0,
  "dados": [
    {
      "codigo": 1,
      "codigoPais": 31,
      "nomePais": "BRASIL",
      "codigoIbge": "12",
      "nome": "Acre",
      "cadastradoPor": "Administrador",
      "alteradoPor": "Administrador",
      "dataCadastro": "07\04\2023 13:52:32",
      "dataAlteracao": "07\04\2023 13:52:32",
      "status": "A"
    }
  ]
}
```

Fonte: Autoria Própria (2023)

## 2.6. UML

Segundo Guedes (2014) a UML (*Unified Modeling Language*) é uma linguagem utilizada para realizar a modelagem de sistemas computacionais por meio de paradigmas de orientação a objetos. Seu objetivo é auxiliar engenheiros e analistas na definição das características, requisitos, comportamentos, estrutura lógica e integrações de outros sistemas com seus softwares antes de desenvolvê-los.

A UML é composta por vários diagramas, cada um com o seu objetivo. Com a modelagem é possível determinar a complexidade de um sistema e seus comportamentos, permitindo ao analista ou engenheiro prever possíveis falhas que só seriam encontradas durante o desenvolvimento do software. Esta é dividida em dois grupos: Diagramas Estruturais e Diagramas Comportamentais (GUEDES, 2014).

Os Diagramas Estruturais são responsáveis por toda a estruturação do projeto, compostos dos seguintes diagramas:

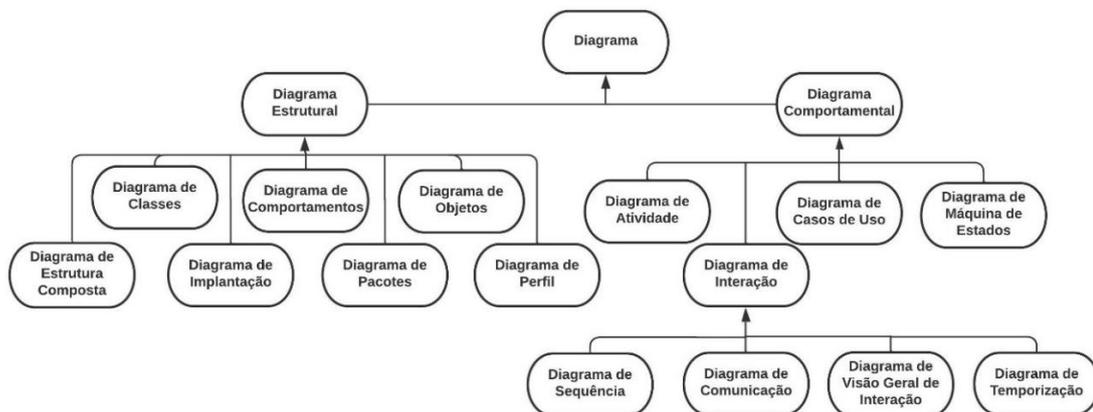
- Diagrama de classes;
- Diagrama de estrutura composta;
- Diagrama de implantação;
- Diagrama de componentes;
- Diagrama de pacotes;
- Diagrama de objetos;
- Diagrama de perfil.

Os Diagramas Comportamentais são responsáveis por analisar as atividades do usuário na aplicação. Esse grupo possui um subgrupo representado pelos diagramas de interações, que representam as interações do usuário com a interface, compostos pelos seguintes diagramas:

- Diagrama de atividade;
- Diagrama de casos de uso;
- Diagrama de máquina de estados;
- Diagrama de interação:
  - Diagrama de sequência;
  - Diagrama de comunicação;
  - Diagrama de visão geral de interação;
  - Diagrama de tempo.

A Figura 6 representa um exemplo da síntese dos diagramas da UML.

**Figura 6 – Síntese dos diagramas da UML**



Fonte: Adaptado de (OMG, 2011)

Para este projeto optou-se por utilizar os diagramas de Classes, Casos de Uso e Máquina de Estados.

## 2.7. StarUML

A ferramenta *StarUML* lançada em 2005, *Open Source*, foi criada para o desenvolvimento de diagramas e MDA (*Model Driven Architecture*), usados para modelagem de sistemas. A ferramenta é multiplataforma, podendo ser utilizada nos sistemas Operacionais Windows, MacOS e Linux, e foi utilizada na versão 5.0.2 para a prototipação das UML (*Unified Modeling Language*) deste projeto, (STARUML, 2022).

### 3. DESENVOLVIMENTO

Para melhor compreensão do presente trabalho, dividiu-se o desenvolvimento do sistema em seções. A seção 3.1 relata a modelagem do banco de dados; a seção 3.2 exibe a modelagem dos Caso de Uso; a seção 3.3 demonstra a modelagem dos Diagramas de Classe e a seção 3.4 apresenta a modelagem dos Diagramas de Máquina de Estados.

#### 3.1. Modelagem do Banco de Dados

Nesta subseção é apresentada uma breve descrição das regras semânticas utilizadas na modelagem da estrutura do banco de dados relacional (seção 3.1.1), em seguida são detalhados a estrutura e os relacionamentos presentes no banco de dados “sistema\_servico” (seção 3.1.2).

##### 3.1.1. Regras semânticas

Nesta seção são detalhados em tópicos as regras de definição de banco de dados utilizados nesse trabalho, conforme as regras do ambiente de trabalho, sendo elas:

- Para nomenclatura de tabelas deve ser utilizada palavra em *Lower Case* (Somente palavras em minúsculo);
- Caso haja mais de uma palavra no nome da tabela utilizar o caractere “\_” para separação das palavras;
- A colação das tabelas é “utf8mb4\_0900\_ai\_ci”;
- O motor das tabelas é “InnoDB”;
- Todas as tabelas têm os seguintes campos:
  - Código Auto incremento;
  - *Status*;
  - Data de Cadastro;
  - Data de Última alteração;
  - Código da sessão que realizou o cadastro;
  - Código da sessão que realizou a última alteração;
- A nomenclatura das colunas de cada tabela deve utilizar palavras em *Upper Case* (palavras em maiúsculo);

- Campos que possuem ligação com outras tabelas devem possuir chave estrangeira;
- Os campos devem possuir uma breve descrição em comentário de qual a sua utilidade;
- As colunas não devem aceitar valores nulos, com a exceção de:
  - Campos do tipo “LongText”;
  - Campos do tipo “Date”;
  - Campos que possuem chave estrangeira que não são de preenchimento obrigatório.

Essas regras foram definidas para garantir a integridade das informações dentro do banco de dados e aplicar a propriedade ACID (*Atomicity, Consistency, Isolation, and Durability*).

### 3.1.2. Estrutura do Banco de Dados

Nesta seção é detalhada a estrutura das tabelas do banco de dados relacional “sistema\_servico” conforme diagrama presente no Apêndice C.

A tabela “pessoa” contém os dados dos: clientes, fornecedores, usuários, funcionários e empresa faturamento. A distinção de cada tipo de pessoa é representada com o relacionamento de modo 1..1 com a tabela “tipo\_pessoa”, pois uma pessoa deve ter somente um tipo. A tabela “tipo\_pessoa”, possui um relacionamento 0..N, com a tabela “pessoa”, sendo assim, um tipo de pessoa pode ter nenhuma ou várias pessoas.

Pelo fato da entidade pessoa possuir muitas informações esses dados foram armazenados separadamente para uma melhoria de desempenho e a diminuição de redundância de informações, sendo criadas as seguintes tabelas filhas:

- “pessoa\_contato”;
- “pessoa\_endereco”;
- “pessoa\_outro\_documento”;

A tabela “ordem\_servico” armazena os dados das ordens de serviço, e possui as seguintes tabelas filhas:

- “ordem\_servico\_custo”;
- “ordem\_servico\_funcionario”;
- “ordem\_servico\_item”;

- “ordem\_servico\_produto”.

De forma generalizada todas as tabelas filhas se relacionam com as tabelas pais de modo 1..1, ou seja, todo registro deve pertencer a um pai. Já a tabela pai se relaciona com suas tabelas filhas de 0..N, sendo assim, ela não precisa possuir nenhum registro filho, porém, pode ter vários.

### **3.2. Modelagem dos Caso de Uso**

Para a criação do sistema foram levantados os seguintes requisitos básicos para o funcionamento da aplicação:

- Login;
- Sair;
- Cadastro de Usuários;
- Cadastro de Clientes;
- Cadastro de Empresas Faturamentos;
- Cadastro de Fornecedores;
- Cadastro de Funcionários;
- Cadastro de Ordens de Serviço;
- Cadastro de Pais;
- Cadastro de Estados;
- Cadastro de Cidades;
- Cadastro de Tipos de Documentos;
- Cadastro de Tipos de Endereços;
- Cadastro de Funções;
- Cadastro de Grupos.

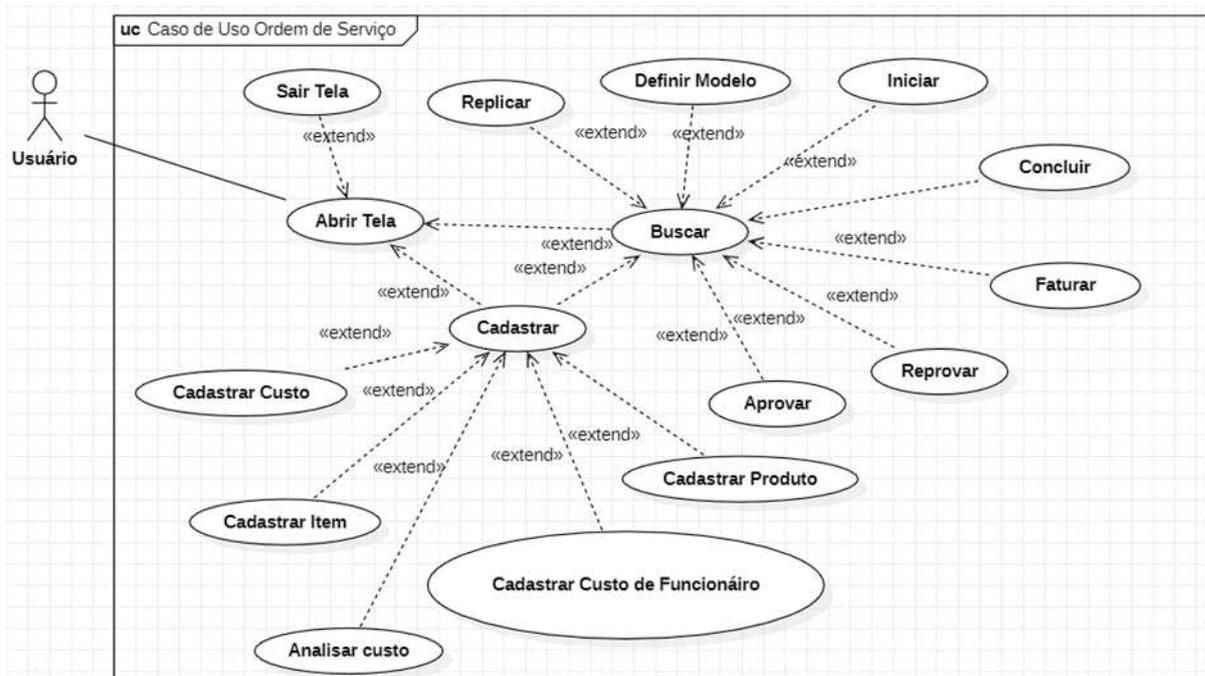
Com os requisitos gerais do sistema definidos, elaborou-se o diagrama de caso de uso do cadastro de ordens de serviço, o qual possui os seguintes requisitos funcionais:

- Abrir a tela;
- Sair da tela;
- Cadastrar;
- Buscar;
- Cadastrar Item;

- Cadastrar Produto;
- Cadastrar Custo;
- Cadastrar Custo de Funcionário;
- Aprovar;
- Reprovar;
- Definir Modelo;
- Iniciar;
- Concluir;
- Faturar;
- Analisar Custo.

A Figura 7 ilustra o diagrama de caso de uso da ordem de serviço.

**Figura 7 – Diagrama de Caso de Uso Ordem de Serviço**



Fonte: Autoria Própria (2023)

Conforme representado na Figura 7, o caso de uso “Abrir Tela” tem como pré-requisito realizar login na tela principal do sistema. Caso o usuário não tenha um cadastro no sistema, deve solicitar ao administrador para realizar o mesmo.

Se o usuário estiver logado no sistema, o mesmo pode abrir a tela de ordem de serviço, que exibe todas as ordens de serviços já cadastradas no sistema. No caso de uso “Sair tela”, o usuário é redirecionado para a tela principal do sistema.

No caso de uso “Buscar” o usuário pode efetuar uma busca das ordens de serviço conforme os filtros disponíveis em tela. Ao localizar uma ordem de serviço o usuário pode alterar e/ou cadastrar uma nova ordem através do caso de uso “Cadastrar”.

Um usuário pode cadastrar itens, produtos, custos gerais e custos com funcionários através dos casos de usos “Cadastrar Item”, “Cadastrar Produto”, “Cadastrar Custo” e “Cadastrar custo com Funcionários” respectivamente. Após todas as informações preenchidas o usuário tem dados suficientes para realizar a análise correta dos custos e avaliar a necessidade de alteração nos valores dos itens/produtos para obter uma margem de lucro melhor.

Um usuário pode replicar uma ordem de serviço através do caso de uso “Replicar”, copiando todos os dados da ordem selecionada e gerando uma nova ordem de serviço.

Os demais casos de usos são modificadores da situação atual da ordem de serviço, sendo eles:

- “Definir Modelo”: muda a situação para “MODELO”;
- “Iniciar”: muda a situação para “EXECUTANDO”;
- “Concluir”: muda a situação para “CONCLUÍDO”;
- “Faturar”: muda a situação para “FATURADO”;
- “Reprovar”: muda a situação para “REPROVADO”;
- “Aprovar”: muda a situação para “APROVADO”.

### 3.2.1. Descrição dos Casos de Uso

Nesta subseção é abordado as descrições do Diagrama de Caso de Uso. O Quadro 5 demonstra o caso de uso “Cadastrar Produto”.

**Quadro 5 – Descrição do Caso de Uso “Cadastrar Produto”**

<b>Caso de Uso:</b>	Cadastrar
<b>Descrição:</b>	Cadastrar/Alterar uma ordem de serviço
<b>Precondição:</b>	Estar com uma ordem de serviço selecionada ou situação da mesma deve ser "ORÇAMENTO", "APROVADO", "EXECUTANDO", "MODELO", caso o evento seja de alteração
<b>Curso de Eventos Básicos</b>	
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>

1.Usuário Clica no botão "Cadastrar"	2.O sistema habilita os campos para inserção dos dados
3.O Usuário informa os dados da ordem de serviço	
4.O Usuário clica no botão "Confirmar"	5.O sistema faz uma requisição para o <i>Web Service</i> do tipo POST
	6.O <i>Web Service</i> valida os dados recebidos e caso os dados informados estejam corretos o sistema insere no banco de dados
	7.O <i>Web Service</i> envia a resposta ao sistema
	8.O sistema recebe a resposta do <i>Web Service</i>
	9.O sistema bloqueia os campos da ordem de serviço
10. O usuário clica em "Alterar"	
	11.O sistema verifica se a situação da ordem é "ORÇAMENTO", "APROVADO", "EXECUTANDO", "MODELO"
	12.O sistema verifica se está com uma ordem de serviço selecionada
	13.Sistema habilita os campos para edição de dados
14.O Usuário altera os dados da ordem de serviço	
15.O Usuário clica no botão "Confirmar"	16.O sistema faz uma requisição para o <i>Web Service</i> do tipo PUT
	17.O <i>Web Service</i> valida os dados recebidos e caso os dados informados estejam corretos o <i>Web Service</i> atualiza o registro no banco de dados
	18.O <i>Web Service</i> envia a resposta ao sistema
	19.O sistema recebe a resposta do <i>Web Service</i>
	20.O sistema bloqueia os campos da ordem de serviço
<b>Cursos Alternativos:</b>	
Usuário clica no botão "Cancelar"	
A registro selecionado está com a situação "CONCLUÍDO", "FATURADO", "REPROVADO" em caso de alteração	
O <i>Web Service</i> localiza uma divergência nos dados informados e retorna um erro para o sistema	
<b>Pós-Condição:</b>	As informações são persistidas no banco de dados

Fonte: Autoria Própria (2023)

As descrições textuais dos demais Casos de Uso da tela de ordem de serviço encontram-se em sua totalidade no Apêndice A.

### 3.3. Modelagem dos Diagramas de Classe

O Diagrama de Classe da ordem de serviço que foi utilizado como base para a criação do *Web Service*, encontra-se no Apêndice B.

Para a criação dos atributos das Classes do *Web Service* foi utilizado como referência o modelo de banco de dados descrito na seção 3.1 Modelagem do banco de dados.

As Classes possuem métodos padronizados, sendo eles:

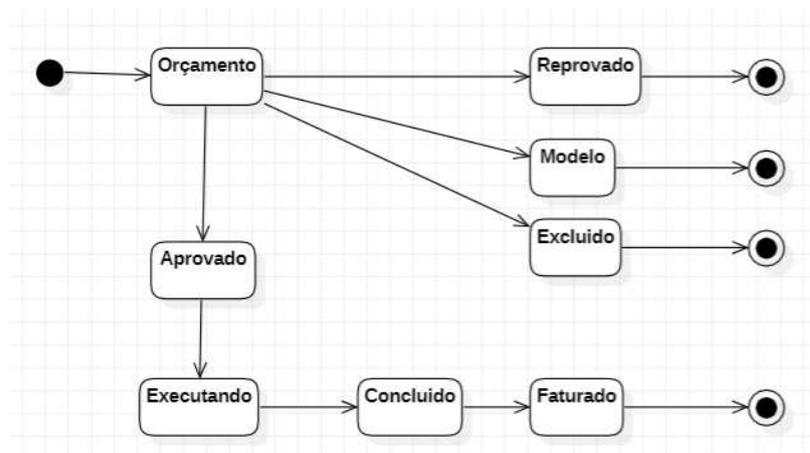
- “consultarCodigo”: consulta um único registro com base na sua chave primária;
- “montar”: converte os dados recuperados do banco de dados em objeto;
- “limpar”: limpa todas as variáveis do objeto;
- “consultar”: consulta vários registros com base nos filtros estabelecidos;
- “cadastrar”: insere um novo registro no banco de dados;
- “alterar”: altera um registro existente no banco de dados;
- “inativar”: altera o *status* de um registro existente no banco de dados.

A Nomenclatura de algum desses métodos acrescenta o nome da Classe para facilitar a localização.

### 3.4. Modelagem dos Diagramas de Máquina de Estados

A Figura 8 ilustra o Diagrama de Estado da ordem de serviço.

Figura 8 – Diagrama de Estado Ordem de Serviço



Fonte: Autoria Própria (2023)

Toda ordem de serviço tem sua situação inicial definida como orçamento, podendo ser alterada para: “Reprovado”, “Modelo”, “Excluído”, “Aprovado”, “Executando”, “Concluído” ou “Faturado”, que são detalhados a seguir:

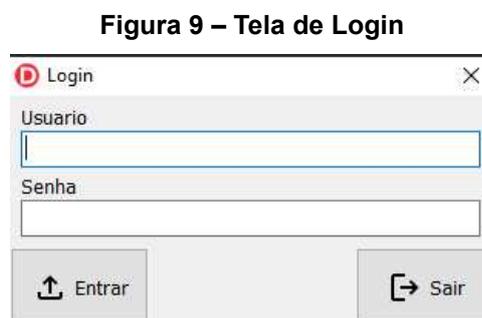
- **Reprovado:** caso o cliente não aceite os valores informados ou desista do pedido.
- **Modelo:** criada para facilitar o trabalho dos usuários, criando ordens de serviços que sejam padrão para determinado cliente que presta um serviço recorrentemente ou um serviço realizado com frequência, sendo assim, o usuário replica o modelo para situação “Orçamento”, sem a necessidade de preencher todos os campos.
- **Excluído:** caso o usuário crie uma ordem de teste ou preencha incorretamente os valores e deseje manter para fins de histórico.
- **Aprovado:** caso o orçamento seja enviado para o cliente e ele aprove, podendo dar início a ordem de serviço.
- **Executando:** define o início do processo da prestação de serviço no cliente.
- **Concluído:** define o término do serviço no cliente.
- **Faturado:** define que o serviço foi cobrado do cliente.

## 4. RESULTADO

A prototipação do aplicativo é apresentada a seguir.

### 4.1. Sistema desenvolvido

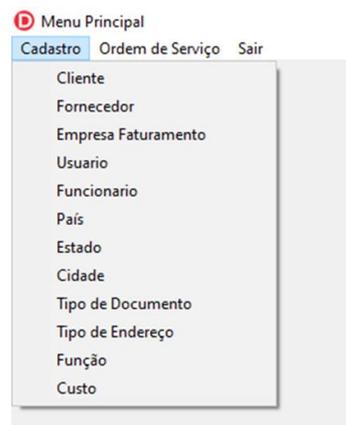
Para que um usuário acesse o sistema é necessário que tenha um cadastro realizado pelo administrador do sistema. A Figura 9 demonstra a tela de login, onde o usuário deve inserir o nome do usuário e a sua senha. Após preencher os campos é necessário pressionar o botão “Entrar”.



**Fonte: Aatoria Própria (2023)**

Após realizar o Login, o sistema mostra a tela principal. No canto superior esquerdo é possível acessar todas as outras telas do sistema, através dos menus, mostrados na Figura 10.

### Figura 10 – Menus da Tela Principal



**Fonte: Aatoria Própria (2023)**

A tela de pessoa, é uma generalização para as telas de clientes, fornecedor, Empresa Faturamento, Usuário e Funcionário devido à semelhança de informação, assim diminuindo código repetido, ajustando as especificações de cada tipo de pessoa ao abrir a tela. A Figura 11 mostra a aba de consulta de Pessoas.

O usuário pode realizar uma consulta pela razão social ou nome fantasia do cliente ao clicar no botão “Consultar”. Para fechar a tela de cadastro de pessoa basta o usuário clicar em “Fechar”.

**Figura 11 – Aba de consulta da tela de Pessoa**

Codigo	Documento	Nº Documento	Razão Social	Nome Fantasia	Telefone	Email
4	CPF	106.244.742-4	GUILHERME C	GUILHERME V	(42) 9024-5780	GUILHERME@GMAIL.COM
2	CPF	106.244.749-25	Guilherme Vaz	Guilherme Vaz	(42) 9 8402-4250	guilherme@gm

**Fonte: Autoria Própria (2023)**

Ao clicar no botão “Cadastrar” ou “Alterar” os campos do cadastro de cliente são desbloqueados para que o usuário insira e/ou altere as informações da pessoa, como, documento, número do documento, razão social, nome fantasia, telefone, email, outros documentos, endereços e contatos. Ao clicar em “Inativar” é mostrado uma confirmação de inativação, e caso o usuário confirme o campo “Status” é alterado de “A” para “I”.

Ao clicar no botão “Cancelar” as alterações feitas são canceladas e os campos do cadastro são bloqueados. Ao clicar no botão “Confirmar” os dados são enviados para o *Web Service* onde é feita uma validação de todos os campos obrigatórios, para serem persistidos no banco de dados, e caso a resposta seja de sucesso os campos são bloqueados.

As lógicas utilizadas nos botões “Cadastrar”, “Alterar”, “Inativar”, “Cancelar”, “Confirmar”, “Consultar” e “Sair” são iguais para todas as telas do sistema. A Figura 12 exemplifica a aba de cadastro de pessoas.

**Figura 12 – Aba de Cadastro de Pessoa**

Código	Tipo Documento	Documento	Data Emissão	Data Vencimento	Observação	Cadastrado
4	CPF	106.244.742-00				

**Fonte: Autoria Própria (2023)**

Na aba de cadastro de pessoa consta as abas de “Outros Documentos”, “Endereços” e “Contatos”. Em cada uma das abas existe um botão “Cadastrar” e “Remover”. Ao clicar no botão “Cadastrar”, é aberta uma tela para cadastrar o respectivo item. Caso o usuário clique no botão “Remover”, é mostrada uma mensagem de confirmação de inativação, se clicado em “Sim”, o campo “Status” é alterado para “I”, e o mesmo é oculto da tela, desde que o campo “Check box” na parte inferior da aba do item, denominado “Mostrar Inativo”, esteja desmarcado.

Para confirmar o cadastro do item escolhido, o usuário deve clicar em “Confirmar”. Caso clique em “Cancelar” as informações não são salvas. Caso o usuário deseje alterar alguma das informações de seu item, deve selecionar a aba respectiva, selecionar o registro na tabela e clicar duas vezes. Assim é aberta a tela de cadastros com todas as informações em seus devidos campos, conforme exemplifica a Figura 13, o cadastro de endereços, as demais telas têm características semelhantes, mudando somente seus campos.

**Figura 13 – Tela de Cadastro de Itens**

Cadastro de Endereços

Descrição  CEP

Longradouro

Numero  Bairro  Complemento

Cidade  Estado

País  Tipo Endereço

Observação

Ativo

**Fonte: Autoria Própria (2023)**

Para os cadastros mais simples, foi desenvolvido uma tela com somente uma aba principal que é dividida ao meio, sendo o lado direito da tela com campos para realizar filtros de buscar e mostrar os registros já cadastrados na base de dados e a esquerda os campos para informar os dados, esta estrutura de tela foi utilizada para a criação das telas de: País, Estados, Cidades, Tipos de Documentos, Tipo de Endereço, Função e Custo. A Figura 14 demonstra esta estrutura a partir do cadastro de País.

**Figura 14 – Tela de Cadastro de País**

Codigo	Codigo IBGE	Nome	Cadastrado Por
1	8958	ZONA DO CANAL DO PANAMA	Administrador
38	1457	CAMAROES	Administrador
39	1414	CAMBOJA	Administrador
40	1490	CANADA	Administrador
41	1511	CANARIAS, ILHAS	Administrador

Fonte: Autoria Própria (2023)

A tela de Ordem de Serviço foi desenvolvida com características semelhantes à tela de Pessoas, porém a sua aba de cadastro possui algumas funcionalidades extras, que foram detalhadas na seção 3.2 (Modelagem dos Caso de Uso) e no Apêndice A (Descrição Textual para os Casos de Uso do Sistema), a Figura 15 mostra a tela de cadastro de Ordem de Serviço.

**Figura 15 – Tela de Cadastro de Ordem de Serviço**

Codigo	Descrição	Quantidade	Valor Unitário	Valor Total	Desconto %	Valor Desconto	Valor Final
1	INSTALAÇÃO DA TORRE PRINCIPAL	1,00	\$ 50.000,00	\$ 50.000,00	2,00 %	R\$ 1.000,00	\$ 49.000,00

Fonte: Autoria Própria (2023)

A tela de Cadastro de Ordem de Serviço, especialmente na aba de cadastro, possui uma aba na parte inferior denominada “Resumo Financeiro”, a qual apresenta ao usuário de forma analítica os custos envolvidos na Ordem de serviço, conforme exemplifica a figura 16.

**Figura 16 – Aba de Análise de custo**

Serviços	Produtos	Custos Gerais	Custo com Funcionarios	Resumo Financeiro	
		Descrição	Valor Bruto	Desconto	Valor Final
▶	Serviço		R\$ 50.000,00	R\$ 1.000,00	R\$ 49.000,00
	Produto		R\$ 100.000,00	R\$ 10.000,00	R\$ 90.000,00
	<b>Total</b>		<b>R\$ 150.000,00</b>	<b>R\$ 11.000,00</b>	<b>R\$ 139.000,00</b>

	Descrição	Itens	Quantidade	Valor Total
▶	ALIMENTAÇÃO	AUXILIAR	20,00	R\$ 1.000,00
	<b>Total Custo Geral</b>		<b>1,00</b>	<b>R\$ 1.000,00</b>
	MECÂNICO		1,00	R\$ 10.235,00
	<b>Total Custo Funcionario</b>		<b>1,00</b>	<b>R\$ 10.235,00</b>
	<b>Total Custos</b>		<b>2,00</b>	<b>R\$ 11.235,00</b>
	<b>Lucro/Prejuízo R\$</b>		<b>91,92</b>	<b>R\$ 127.765,00</b>

Fonte: Autoria Própria (2023)

## 5. CONCLUSÃO

O sistema proposto de controle de ordem de serviço oferece uma solução abrangente e eficiente para empresas que buscam aprimorar seus processos de gerenciamento de ordens de serviço. Por meio da implementação desse sistema, é possível obter uma série de benefícios em comparação às abordagens convencionais, como planilhas de Excel, Access, Word e arquivo físico.

Uma das principais vantagens desse sistema é a redução do tempo de elaboração da proposta de ordem de serviço. Ao automatizar tarefas e fluxos de trabalho, a elaboração das propostas se torna mais ágil e eficiente, possibilitando um aumento da produtividade e uma resposta mais rápida aos clientes.

Além disso, o sistema proporciona uma análise minuciosa dos custos envolvidos na prestação do serviço. Por meio da análise de custo, é possível ter uma visão clara dos gastos associados a cada ordem de serviço, identificando possíveis áreas de melhoria e permitindo uma tomada de decisão eficiente e estratégica.

A disponibilidade de acesso à informação é outro aspecto fundamental do sistema. Com todas as informações relevantes centralizadas em um único local, os usuários têm acesso rápido e fácil aos dados das ordens de serviço, facilitando a consulta e a análise dos processos em andamento.

Além disso, o sistema oferece uma robusta gestão de dados, garantindo que as informações sejam armazenadas e organizadas de forma estruturada. Isso possibilita uma gestão mais eficaz das ordens de serviço, permitindo a identificação de tendências e um planejamento estratégico com base em dados confiáveis e atualizados.

Em suma, o sistema de controle de ordem de serviço proposto representa um avanço significativo em relação às formas tradicionais de controle. Com recursos de automação, gestão financeira, disponibilidade de acesso à informação e controle de dados, ele oferece uma solução completa e eficiente para as empresas que desejam otimizar seus processos, melhorar a eficiência operacional e aprimorar a qualidade do serviço prestado.

Ao término do desenvolvimento do Sistema, o mesmo foi repassado para a empresa solicitante para que fossem realizados testes pelos usuários responsáveis pelas ordens de serviço. Caso se prove adequado e atenda as necessidades da

empresa, o Sistema deverá ser aprovado e implementado através do setor responsável.

### **5.1. Trabalhos Futuros**

O Sistema desenvolvido pode ter suas funcionalidades melhoradas e expandidas através de trabalhos futuros, por exemplo, através da implantação das seguintes ideias:

- Desenvolver uma versão mobile;
- Desenvolver uma versão Web;
- Desenvolver relatórios da ordem de serviço;
- Realizar envios direto para o email/WhatsApp do cliente;
- Desenvolver um módulo financeiro.

## REFERÊNCIAS

BEZERRA, Jéssica Maria de Brito. **A importância do setor de serviços na economia brasileira: um Estudo da PAS/IBGE**. 2020.

BERNERS-LEE, Tim; FIELDING, Roy; FRYSTYK, Henrik. **Hypertext transfer protocol--HTTP/1.0**. 1996.

CANALTECH. **O que é DNS? | Trocá-lo pode ser a solução**. 2022. Disponível em <https://canaltech.com.br/internet/o-que-e-dns/>. Acesso em 05 Mar 2023.

EMBARCADERO TECHNOLOGIES INC. **RAD Studio**. 2022. Disponível em: <https://www.embarcadero.com/br/products/rad-studio/whats-new-in-10-4-sydney>. Acesso em 19.jun.2022.

GOYAL, Aayush; DYRESON, Curtis. Temporal JSON. In: **2019 IEEE 5th International Conference on Collaboration and Internet Computing (CIC)**. IEEE, 2019. p. 135-144.

GUEDES, Gilleanes T. A. **UML2: uma abordagem prática**. 2. ed. São Paulo: Novatec Editora, 2011.

GUEDES, Gilleanes TA. **UML 2–Guia Prático-2ª Edição**. Novatec Editora, 2014.

INDRUSIAK, Leandro Soares. **Linguagem java**. Grupo Java RS JUG Rio Grande do Sul, p. 19, 1996.

LUCIANO, Josué; ALVES, Wallison Joel Barberá. **Padrão de arquitetura MVC: Model-view-controller**. EPeQ Fafibe, v. 1, n. 3a, p. 102-107, 2017.

MEGIOLARO, Marcello Rodrigo de Oliveira. **Indicadores de manutenção industrial relacionados à eficiência global de equipamentos**. 2015. Trabalho de Conclusão de Curso. Universidade Tecnológica Federal do Paraná.

MILANI, André. **MySQL-guia do programador**. Novatec Editora, 2007.

MOZILLA. **HTTP**. 2023. Disponível em <https://developer.mozilla.org/pt-BR/docs/Web/HTTP>. Acesso em 10 Mar 2023.

OLIVEIRA, Pâmela Caroline Silva de. **Gestão de estoque e armazenamento de suprimentos e sua influência na manutenção de ativos: um caso aplicado em uma indústria de carcinicultura**. 2020. Dissertação de Mestrado. Universidade Federal do Rio Grande do Norte.

OMG Group. **OMG Unified Modeling Language (OMG UML)**. Superstructure. Versão 2.4.1.Ago. 2011.

ORACLE. **Oracle Communications Subscriber Database Server**. Provisioning Interface. Versão 8.5.Oct. 2020.

ORACLE. **O que é um Banco de Dados?**. 2023. Disponível em <https://www.oracle.com/br/database/what-is-database/>. Acesso em 01 Fev 2023.

RONDINA, Gustavo Garcia. Instrumentador de código para a linguagem Delphi. 2005.

SPACEPROGRAM. **Construindo um Diagrama de Classes da UML**. 2023. Disponível em <https://spaceprogrammer.com/uml/introducao-as-classes-associacoes-e-generalizacoes/>. Acesso em 20 Mar 2023.

STARUML. **StarUML**, 2022. Disponível em <https://staruml.io/>. Acesso em 21.jun.2022.

VAN DER LANS, Rick F. **SQL for MySQL Developers: A Comprehensive Tutorial and Reference: A Comprehensive Tutorial and Reference**. Pearson Education, 2007.

W3. **HTML Uniform Resource Locators**. 2023. Disponível em [https://www.w3schools.com/html/html\\_urlencode.asp](https://www.w3schools.com/html/html_urlencode.asp). Acesso em 02 Fev 2023.

WIKIPEDIA. **Diagrama de classes**. 2023. Disponível em [https://pt.wikipedia.org/wiki/Diagrama\\_de\\_classes#/media/Ficheiro:Diagrama\\_de\\_Classes\\_com\\_duas\\_classes.png](https://pt.wikipedia.org/wiki/Diagrama_de_classes#/media/Ficheiro:Diagrama_de_Classes_com_duas_classes.png). Acesso em 15 Mar 2023.

## **APÊNDICE A - Descrição Textual para os Casos de Uso do Sistema**

Após a elaboração e análise do Diagrama de caso de Uso do sistema de ordem de serviço, foram realizadas as descrições textuais de cada caso de uso da tela de ordem de serviço a fim de descrever as funcionalidades presente. A seguir o Quadro 6 demonstra a descrição do caso de uso “Abrir Tela”, o Quadro 7 para “Sair tela”, o Quadro 8 para “Buscar”, o Quadro 9 para “Definir Modelo”, o Quadro 10 para “Iniciar”, o Quadro 11 para “Concluir”, o Quadro 12 para “Faturar”, o Quadro 13 para “Reprovar”, o Quadro 14 para “Aprovar”, o Quadro 15 para “Cadastrar Custo”, o Quadro 16 para “Cadastrar Item”, o Quadro 17 para “Cadastrar Produto”, o Quadro 18 para “Cadastrar custos de Funcionário”, Quadro 19 para “Analisar Custo” e o Quadro 20 para “Replicar”.

**Quadro 6 – Descrição Textual do caso de uso “Abrir Tela”**

<b>Caso de Uso:</b>	Abrir Tela	
<b>Descrição:</b>	Abrir tela de ordem de serviço	
<b>Precondição:</b>	Estar no menu principal do sistema	
<b>Curso de Eventos Básicos</b>		
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>	
1.Clicar no menu "Ordem de serviço"	2.O sistema abre a tela de ordem de serviço, com duas abas, sendo uma de consulta e outra de cadastro, por padrão abre na aba de consulta	
<b>Cursos Alternativos:</b>		
Usuário clica em outro menu		
<b>Pós-Condição:</b>	A tela de ordem de serviço é aberta mostrando todas as ordens cadastradas	

Fonte: Autoria Própria (2023)

**Quadro 7 – Descrição Textual do caso de uso “Sair Tela”**

<b>Caso de Uso:</b>	Sair tela	
<b>Descrição:</b>	Sair da tela de ordem de serviço	
<b>Precondição:</b>	Estar na tela de ordem de serviço	
<b>Curso de Eventos Básicos</b>		
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>	
1.Usuário clica no botão "Sair"	2.O sistema direciona para o menu principal do sistema	
<b>Cursos Alternativos:</b>		
Usuário clica em outro botão		
<b>Pós-Condição:</b>	A tela de menu principal é mostrada	

Fonte: Autoria Própria (2023)

Quadro 8 – Descrição Textual do caso de uso “Buscar”

<b>Caso de Uso:</b>	Buscar
<b>Descrição:</b>	Trazer os dados das outras ordens de serviços já cadastradas no sistema de acordo com os filtros disponíveis em tela
<b>Precondição:</b>	Estar na tela de ordem de serviço na aba de consulta
<b>Curso de Eventos Básicos</b>	
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>
1.Usuário informa os filtros de busca desejado e clicar em "Consultar"	2.O sistema realiza uma requisição do tipo GET para o <i>Web Service</i>
	3.O sistema recebe a resposta do <i>Web Service</i> a converte em um Data Source
	4. O <i>Web Service</i> busca as informações no banco de dados de acordo com os filtros definidos pelo usuário
	5.O sistema carrega no grid os dados de resposta
<b>Cursos Alternativos:</b>	
Usuário clica em outro botão	
<b>Pós-Condição:</b>	Os dados disponíveis em tela são atualizados de acordo com a consulta requisitada pelo usuário

Fonte: Autoria Própria (2023)

Quadro 9 – Descrição Textual do caso de uso “Definir Modelo”

<b>Caso de Uso:</b>	Definir Modelo
<b>Descrição:</b>	Alterar a situação da ordem de serviço para "MODELO"
<b>Precondição:</b>	Estar com uma ordem de serviço selecionada e a situação da mesma deve ser "ORÇAMENTO"
<b>Curso de Eventos Básicos</b>	
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>
1.Usuário clica no botão "Modelo"	2.O sistema mostra uma confirmação para alterar a situação da ordem para "MODELO"
3.O Usuário confirma a alteração da situação	4.O sistema verifica se existe um registro selecionado
	5.O sistema verifica se a condição da ordem selecionada é igual a "ORÇAMENTO"
	6.O sistema realiza uma requisição do tipo PUT solicitando a alteração da situação da ordem para "MODELO"
	7.O <i>Web Service</i> valida as informações recebidas e caso estejam corretas atualiza no banco de dados
	8.O <i>Web Service</i> retorna referente a alteração da situação para "MODELO"
	9. Os dados alterados são mostrados em tela
<b>Cursos Alternativos:</b>	
Usuário não confirma a alteração de situação	

A registro selecionado está com a situação diferente de "ORÇAMENTO"	
Usuário clica em outro botão	
<b>Pós-Condição:</b>	A situação é alterada no banco de dados

Fonte: Autoria Própria (2023)

Quadro 10 – Descrição Textual do caso de uso “Iniciar”

<b>Caso de Uso:</b>	Iniciar
<b>Descrição:</b>	Alterar a situação da ordem de serviço para "EXECUTANDO"
<b>Precondição:</b>	Estar com uma ordem de serviço selecionada e a situação da mesma deve ser "APROVADO"
<b>Curso de Eventos Básicos</b>	
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>
1.Usuário clica no botão "Executar"	2.O sistema mostra uma confirmação para alterar a situação da ordem para "EXECUTANDO"
3.O usuário confirma a alteração da situação	4.O sistema verifica se existe um registro selecionado
	5.O sistema verifica se a condição da ordem selecionada é igual a "APROVADO"
	6.O sistema realiza uma requisição do tipo PUT solicitando a alteração da situação da ordem para "EXECUTANDO"
	7.O <i>Web Service</i> valida as informações recebidas e caso estejam corretas atualiza no banco de dados
	8.O <i>Web Service</i> retorna referente a alteração da situação para "EXECUTANDO"
	9. Os dados alterados são mostrados em tela
<b>Cursos Alternativos:</b>	
Usuário não confirma a alteração de situação	
A registro selecionado está com a situação diferente de "APROVADO"	
Usuário clica em outro botão	
<b>Pós-Condição:</b>	A situação é alterada no banco de dados

Fonte: Autoria Própria (2023)

Quadro 11 – Descrição Textual do caso de uso “Concluir”

<b>Caso de Uso:</b>	Concluir
<b>Descrição:</b>	Alterar a situação da ordem de serviço para "CONCLUIDO"
<b>Precondição:</b>	Estar com uma ordem de serviço selecionada e a situação da mesma deve ser "EXECUTANDO"
<b>Curso de Eventos Básicos</b>	
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>
1.Usuário clica no botão "Concluir"	2.O sistema mostra uma confirmação para alterar a situação da ordem para "CONCLUIDO"
3.O usuário confirma a alteração da situação	4.O sistema verifica se existe um registro selecionado

	5.O sistema verifica se a condição da ordem selecionada é igual a "EXECUTANDO"
	6.O sistema realiza uma requisição do tipo PUT solicitando a alteração da situação da ordem para "CONCLUIDO"
	7.O <i>Web Service</i> valida as informações recebidas e caso estejam corretas atualiza no banco de dados
	8.O <i>Web Service</i> retorna referente a alteração da situação para "CONCLUIDO"
	9. Os dados alterados são mostrados em tela
<b>Cursos Alternativos:</b>	
Usuário não confirma a alteração de situação	
A registro selecionado está com a situação diferente de "EXECUTANDO"	
Usuário clica em outro botão	
<b>Pós-Condição:</b>	A situação é alterada no banco de dados

Fonte: Autoria Própria (2023)

Quadro 12 – Descrição Textual do caso de uso “Faturar”

<b>Caso de Uso:</b>	Faturar
<b>Descrição:</b>	Alterar a situação da ordem de serviço para "FATURADO"
<b>Precondição:</b>	Estar com uma ordem de serviço selecionada e a situação da mesma deve ser "CONCLUIDO"
<b>Curso de Eventos Básicos</b>	
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>
1.Usuário clica no botão "Faturar"	2.O sistema mostra uma confirmação para alterar a situação da ordem para "FATURADO"
3.O usuário confirma a alteração da situação	4.O sistema verifica se existe um registro selecionado
	5.O sistema verifica se a condição da ordem selecionada é igual a "CONCLUIDO"
	6.O sistema realiza uma requisição do tipo PUT solicitando a alteração da situação da ordem para "FATURADO"
	7.O <i>Web Service</i> valida as informações recebidas e caso estejam corretas atualiza no banco de dados
	8.O <i>Web Service</i> retorna referente a alteração da situação para "FATURADO"
	9. Os dados alterados são mostrados em tela
<b>Cursos Alternativos:</b>	
Usuário não confirma a alteração de situação	
A registro selecionado está com a situação diferente de "CONCLUIDO"	
Usuário clica em outro botão	
<b>Pós-Condição:</b>	A situação é alterada no banco de dados

Fonte: Autoria Própria (2023)

Quadro 13 – Descrição Textual do caso de uso “Reprovar”

<b>Caso de Uso:</b>	Reprovar
<b>Descrição:</b>	Alterar a situação da ordem de serviço para "REPROVADO"
<b>Precondição:</b>	Estar com uma ordem de serviço selecionada e a situação da mesma deve ser "ORÇAMENTO"
<b>Curso de Eventos Básicos</b>	
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>
1.Usuário clica no botão "Reprovar"	2.O sistema mostra uma confirmação para alterar a situação da ordem para "REPROVADO"
3.O usuário confirma a alteração da situação	4.O sistema verifica se existe um registro selecionado
	5.O sistema verifica se a condição da ordem selecionada é igual a "ORÇAMENTO"
	6.O sistema realiza uma requisição do tipo PUT solicitando a alteração da situação da ordem para "REPROVADO"
	7.O <i>Web Service</i> valida as informações recebidas e caso estejam corretas atualiza no banco de dados
	8.O <i>Web Service</i> retorna referente a alteração da situação para "REPROVADO"
	9. Os dados alterados são mostrados em tela
<b>Cursos Alternativos:</b>	
Usuário não confirma a alteração de situação	
A registro selecionado está com a situação diferente de "ORÇAMENTO"	
Usuário clica em outro botão	
<b>Pós-Condição:</b>	A situação é alterada no banco de dados

Fonte: Autoria Própria (2023)

Quadro 14 – Descrição Textual do caso de uso “Aprovar”

<b>Caso de Uso:</b>	Aprovar
<b>Descrição:</b>	Alterar a situação da ordem de serviço para "APROVADO"
<b>Precondição:</b>	Estar com uma ordem de serviço selecionada e a situação da mesma deve ser "ORÇAMENTO"
<b>Curso de Eventos Básicos</b>	
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>
1.Usuário clica no botão "Aprovar"	2.O sistema mostra uma confirmação para alterar a situação da ordem para "APROVADO"
3.O usuário confirma a alteração da situação	4.O sistema verifica se existe um registro selecionado
	5.O sistema verifica se a condição da ordem selecionada é igual a "ORÇAMENTO"
	6.O sistema realiza uma requisição do tipo PUT solicitando a alteração da situação da ordem para "APROVADO"

	7.O <i>Web Service</i> valida as informações recebidas e caso estejam corretas atualiza no banco de dados
	8.O <i>Web Service</i> retorna referente a alteração da situação para "APROVADO"
	9. Os dados alterados são mostrados em tela
<b>Cursos Alternativos:</b>	
Usuário não confirma a alteração de situação	
A registro selecionado está com a situação diferente de "ORÇAMENTO"	
Usuário clica em outro botão	
<b>Pós-Condição:</b>	A situação é alterada no banco de dados

Fonte: Autoria Própria (2023)

Quadro 15 – Descrição Textual do caso de uso “Cadastrar Custo”

<b>Caso de Uso:</b>	Cadastrar Custo	
<b>Descrição:</b>	Cadastrar/Alterar/Inativar um custo da ordem de serviço selecionada	
<b>Precondição:</b>	O usuário deve estar cadastrando ou editando uma ordem de serviço, selecionado na aba de custos gerais, clicar em "Adicionar" ou "Remover" ou clicar duas vezes em um item na tabela de custos.	
<b>Curso de Eventos Básicos</b>		
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>	
1.Usuário clica no botão "Cadastrar"	2.O sistema abre uma nova tela para inserção dos dados	
3.O usuário informa os dados do Custo		
4.O usuário clica no botão "Confirmar"	5.O sistema faz uma requisição para o <i>Web Service</i> do tipo POST	
	6.O <i>Web Service</i> valida os dados e caso os dados informados estejam corretos o <i>Web Service</i> insere no banco de dados	
	7.O <i>Web Service</i> envia a resposta ao sistema	
	8.O sistema recebe a resposta do <i>Web Service</i>	
	9.O sistema atualiza a tabela de custos e habilita novamente para inserir um novo custo	
10. Após cadastrar todos os custos o usuário clicar em "Fechar"	11.O sistema fecha a tela de custos	
12. O usuário clica duas vezes no custo que deseja alterar	13.O sistema verifica se está com um custo selecionado	
	14.Sistema abre uma nova tela com todas as informações do custo já carregadas em seus devidos campos	
15.O usuário altera os dados do custo		
16.O usuário clica no botão "Confirmar"	17.O sistema faz uma requisição para o <i>Web Service</i> do tipo PUT	

	18.O <i>Web Service</i> valida os dados e caso os dados informados estejam corretos, atualiza no banco de dados
	19.O <i>Web Service</i> envia a resposta ao sistema
	20.O sistema recebe a resposta do <i>Web Service</i>
	21.O sistema atualiza os dados na tabela de custo e habilita a inserção de um novo custo
22.O usuário clica no botão "Fechar"	23.O sistema fecha a tela de custos
24.O usuário clica no botão "Remover"	25.O sistema faz uma requisição para o <i>Web Service</i> do tipo DELETE
	26.O <i>Web Service</i> valida os dados e caso os dados informados estejam corretos, inativa no banco de dados
	27.O <i>Web Service</i> envia a resposta ao sistema
	28.O sistema recebe a resposta do <i>Web Service</i>
	29.O sistema oculta o registro na tabela de custos
<b>Cursos Alternativos:</b>	
Usuário clica no botão "Cancelar"	
No caso de alteração/inativação não havia nenhum registro selecionado	
O <i>Web Service</i> localiza uma divergência nos dados informados e retorna um erro para o sistema	
<b>Pós-Condição:</b>	As informações são persistidas no banco de dados

Fonte: Autoria Própria (2023)

Quadro 16 – Descrição Textual do caso de uso “Cadastrar Item”

<b>Caso de Uso:</b>	Cadastrar Item	
<b>Descrição:</b>	Cadastrar/Alterar/Inativar um item da ordem de serviço selecionada	
<b>Precondição:</b>	O usuário deve estar cadastrando ou editando uma ordem de serviço, selecionado na aba de itens, clicar em "Adicionar" ou "Remover" ou clicar duas vezes em um item na tabela de itens.	
<b>Curso de Eventos Básicos</b>		
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>	
1.Usuário clica no botão "Cadastrar"	2.O sistema abre uma nova tela para inserção dos dados	
3.O usuário informa os dados do item		
4.O usuário clica no botão "Confirmar"	5.O sistema faz uma requisição para o <i>Web Service</i> do tipo POST	
	6.O <i>Web Service</i> valida os dados recebidos e caso os dados informados estejam corretos o <i>Web Service</i> insere no banco de dados	
	7.O <i>Web Service</i> envia a resposta ao sistema	
	8.O sistema recebe a resposta do <i>Web Service</i>	

	9.O sistema atualiza a tabela de item e habilita novamente para inserir um novo item
10. Após cadastrar todos os itens o usuário clicar em "Fechar"	11.O sistema fecha a tela de itens
12. O usuário clica duas vezes em cima no item que deseja alterar	13.O sistema verifica se está com um item selecionado
	14.Sistema abre uma nova tela com todas as informações do item já carregadas em seus devidos campos
15.O usuário altera os dados do item	
16.O usuário clica no botão "Confirmar"	17.O sistema faz uma requisição para o <i>Web Service</i> do tipo PUT
	18.O <i>Web Service</i> valida os dados e caso os dados informados estejam corretos, atualiza no banco de dados
	19.O <i>Web Service</i> envia a resposta ao sistema
	20.O sistema recebe a resposta do <i>Web Service</i>
	21.O sistema atualiza os dados na tabela de itens e habilita a inserção de um novo item
22.O usuário clica no botão "Fechar"	23.O sistema fecha a tela de itens
24.O usuário clica no botão "Remover"	25.O sistema faz uma requisição para o <i>Web Service</i> do tipo DELETE
	26.O <i>Web Service</i> valida os dados e caso os dados informados estejam corretos, inativa no banco de dados
	27.O <i>Web Service</i> envia a resposta ao sistema
	28.O sistema recebe a resposta do <i>Web Service</i>
	29.O sistema oculta o registro na tabela de item
<b>Cursos Alternativos:</b>	
Usuário clica no botão "Cancelar"	
No caso de alteração/inativação não havia nenhum registro selecionado	
O <i>Web Service</i> localiza uma divergência nos dados informados e retorna um erro para o sistema	
<b>Pós-Condição:</b>	As informações são persistidas no banco de dados

Fonte: Autoria Própria (2023)

Quadro 17 – Descrição Textual do caso de uso “Cadastrar Produto”

<b>Caso de Uso:</b>	Cadastrar Produto
<b>Descrição:</b>	Cadastrar/Alterar/Inativar um produto da ordem de serviço selecionada
<b>Precondição:</b>	O usuário deve estar cadastrando ou editando uma ordem de serviço, selecionado na aba de produtos, clicar em "Adicionar" ou "Remover" ou clicar duas vezes em um item na tabela de produtos.
<b>Curso de Eventos Básicos</b>	

<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>
1.Usuário clica no botão "Cadastrar"	2.O sistema abre uma nova tela para inserção dos dados
3.O usuário informa os dados do produto	
4.O usuário clica no botão "Confirmar"	5.O sistema faz uma requisição para o <i>Web Service</i> do tipo POST
	6.O <i>Web Service</i> valida os dados recebidos e caso os dados informados estejam corretos o <i>Web Service</i> insere no banco de dados
	7.O <i>Web Service</i> envia a resposta ao sistema
	8.O sistema recebe a resposta do <i>Web Service</i>
	9.O sistema atualiza a tabela de produto e habilita novamente para inserir um novo produto
10. Após cadastrar todos os produtos o usuário clicar em "Fechar"	11.O sistema fecha a tela de produtos
12. O usuário clica duas vezes em cima do produto que deseja alterar	13.O sistema verifica se está com um produto selecionado
	14.Sistema abre uma nova tela com todas as informações do produto já carregadas em seus devidos campos
15.O usuário altera os dados do produto	
16.O usuário clica no botão "Confirmar"	17.O sistema faz uma requisição para o <i>Web Service</i> do tipo PUT
	18.O <i>Web Service</i> valida os dados e caso os dados informados estejam corretos, atualiza no banco de dados
	19.O <i>Web Service</i> envia a resposta ao sistema
	20.O sistema recebe a resposta do <i>Web Service</i>
	21.O sistema atualiza os dados na tabela de produtos e habilita a inserção de um novo produto
22.O usuário clica no botão "Fechar"	23.O sistema fecha a tela de produtos
24.O usuário clica no botão "Remover"	25.O sistema faz uma requisição para o <i>Web Service</i> do tipo DELETE
	26.O <i>Web Service</i> valida os dados e caso os dados informados estejam corretos, inativa no banco de dados
	27.O <i>Web Service</i> envia a resposta ao sistema
	28.O sistema recebe a resposta do <i>Web Service</i>
	29.O sistema oculta o registro na tabela de produtos
<b>Cursos Alternativos:</b>	
Usuário clica no botão "Cancelar"	

No caso de alteração/inativação não havia nenhum registro selecionado	
O <i>Web Service</i> localiza uma divergência nos dados informados e retorna um erro para o sistema	
<b>Pós-Condição:</b>	As informações são persistidas no banco de dados

Fonte: Autoria Própria (2023)

Quadro 18 – Descrição Textual do caso de uso “Cadastrar Custos de Funcionários”

<b>Caso de Uso:</b>	Cadastrar Custos de Funcionários	
<b>Descrição:</b>	Cadastrar/Alterar/Inativar um custo com funcionário da ordem de serviço selecionada	
<b>Precondição:</b>	O usuário deve estar cadastrando ou editando uma ordem de serviço, selecionado na aba de custos com funcionários, clicar em "Adicionar" ou "Remover" ou clicar duas vezes em um custo com funcionário na tabela de custos com funcionários.	
<b>Curso de Eventos Básicos</b>		
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>	
1.Usuário clica no botão "Cadastrar"	2.O sistema abre uma nova tela para inserção dos dados	
3.O usuário informa os dados do custo com funcionário		
4.O usuário clica no botão "Confirmar"	5.O sistema faz uma requisição para o <i>Web Service</i> do tipo POST	
	6.O <i>Web Service</i> valida os dados recebidos e caso os dados informados estejam corretos o <i>Web Service</i> insere no banco de dados	
	7.O <i>Web Service</i> envia a resposta ao sistema	
	8.O sistema recebe a resposta do <i>Web Service</i>	
	9.O sistema atualiza a tabela de custos com funcionário e habilita novamente para inserir um novo custo com funcionário	
10. Após cadastrar todos os custos com funcionários o usuário clicar em "Fechar"	11.O sistema fecha a tela de custos com funcionários	
12. O usuário seleciona na tabela de custo com funcionário qual deseja alterar e clica duas vezes em cima do registro	13.O sistema verifica se está com um custo com funcionário selecionado	
	14.Sistema abre uma nova tela com todas as informações do custo com funcionário já carregadas em seus devidos campos	
15.O usuário altera os dados do custo com funcionário		
16.O usuário clica no botão "Confirmar"	17.O sistema faz uma requisição para o <i>Web Service</i> do tipo PUT	
	18.O <i>Web Service</i> valida os dados e caso os dados informados estejam corretos, atualiza no banco de dados	

	19.O <i>Web Service</i> envia a resposta ao sistema
	20.O sistema recebe a resposta do <i>Web Service</i>
	21.O sistema atualiza os dados na tabela de custo com funcionário e habilita a inserção de um novo custo com funcionário
22.O usuário clica no botão "Fechar"	23.O sistema fecha a tela de custo com funcionário
24.O usuário clica no botão "Remover"	25.O sistema faz uma requisição para o <i>Web Service</i> do tipo DELETE
	26.O <i>Web Service</i> valida os dados e caso os dados informados estejam corretos, inativa no banco de dados
	27.O <i>Web Service</i> envia a resposta ao sistema
	28.O sistema recebe a resposta do <i>Web Service</i>
	29.O sistema oculta o registro na tabela de custos com funcionário
<b>Cursos Alternativos:</b>	
Usuário clica no botão "Cancelar"	
No caso de alteração/inativação não havia nenhum registro selecionado	
O <i>Web Service</i> localiza uma divergência nos dados informados e retorna um erro para o sistema	
<b>Pós-Condição:</b>	As informações são persistidas no banco de dados

Fonte: Autoria Própria (2023)

Quadro 19 – Descrição Textual do caso de uso “Analisar Custo”

<b>Caso de Uso:</b>	Analisar Custo
<b>Descrição:</b>	Selecionar a aba “Análise de Custo”
<b>Precondição:</b>	
<b>Curso de Eventos Básicos</b>	
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>
1.Clicar na aba "Análise de custo"	2.O sistema mostra uma análise agrupada de todos os produtos, itens, custos com funcionários e custos gerais, trazendo seus valores e ao fim mostrando o resultado final da ordem de serviço (Lucro/Prejuízo)
<b>Cursos Alternativos:</b>	
<b>Pós-Condição:</b>	O usuário visualizar o lucro/prejuízo da ordem e verifica se há necessidade de alterar os valores de produtos ou itens e também realizar corte de custos

Fonte: Autoria Própria (2023)

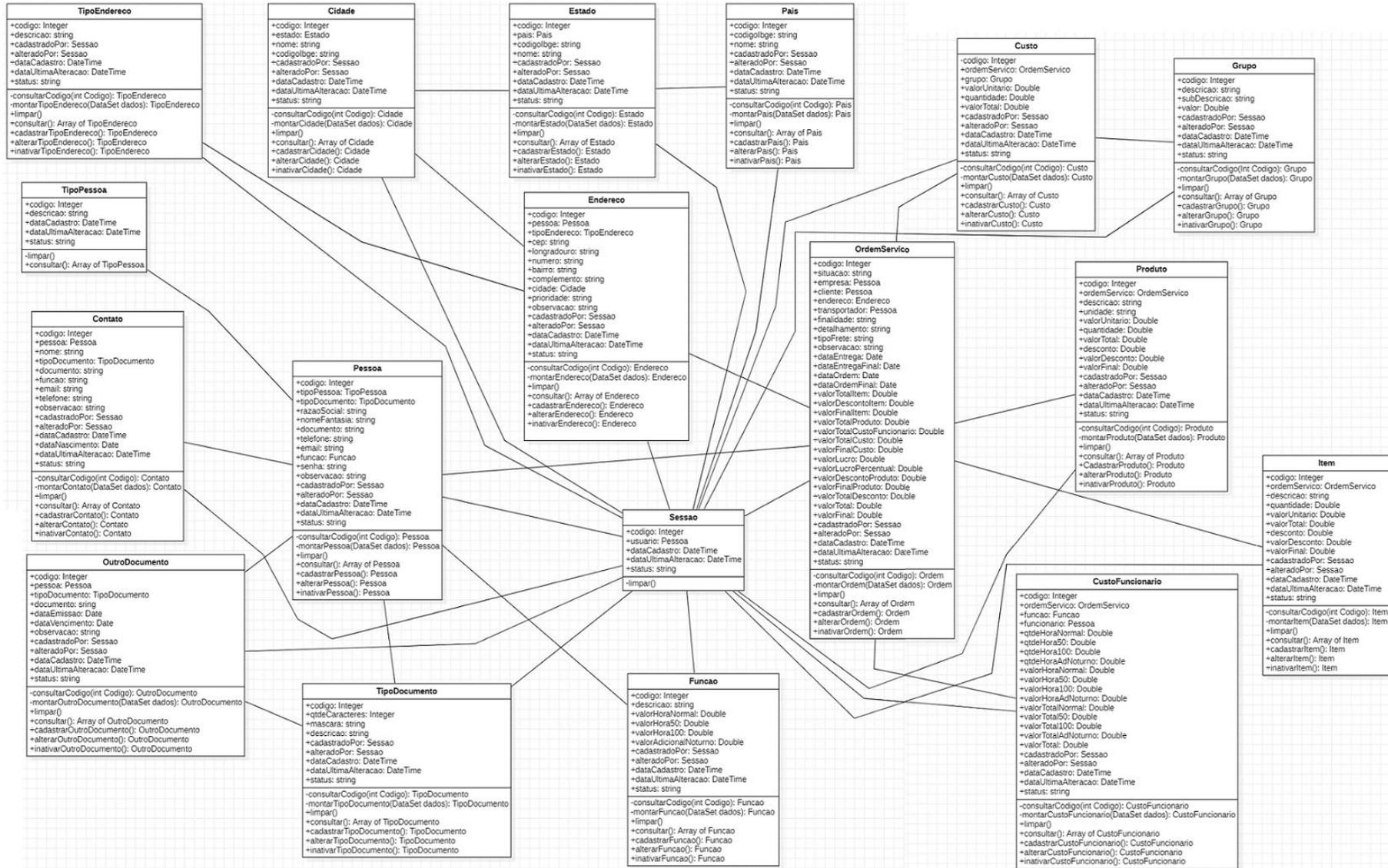
Quadro 20 – Descrição Textual do caso de uso “Replicar”

<b>Caso de Uso:</b>	Replicar	
<b>Descrição:</b>	Gerar uma nova ordem de serviço igual a selecionada	
<b>Precondição:</b>	A situação da ordem de serviço deve ser diferente de "REPROVADO", ter um registro selecionado	
<b>Curso de Eventos Básicos</b>		
<b>Ação do Ator:</b>	<b>Resposta do Sistema:</b>	
1.Clicar no botão "Replicar"	2.O sistema verifica se tem um registro selecionado	
	3.O sistema mostra uma mensagem de confirmação para replicar o registro selecionado	
	4.O sistema gerar uma nova ordem de serviço igual a selecionada	
	5.O sistema localiza a nova ordem de serviço e automaticamente altera a mesma, assim facilitando o trabalho do usuário	
<b>Cursos Alternativos:</b>		
O usuário não confirma a replicação da ordem de serviço		
O usuário seleciona uma ordem de serviço com situação "REPROVADO"		
<b>Pós-Condição:</b>	O usuário edita a nova ordem de serviço cadastrada	

Fonte: Autoria Própria (2023)

## **APÊNDICE B – Diagrama de Classe *Web Service***

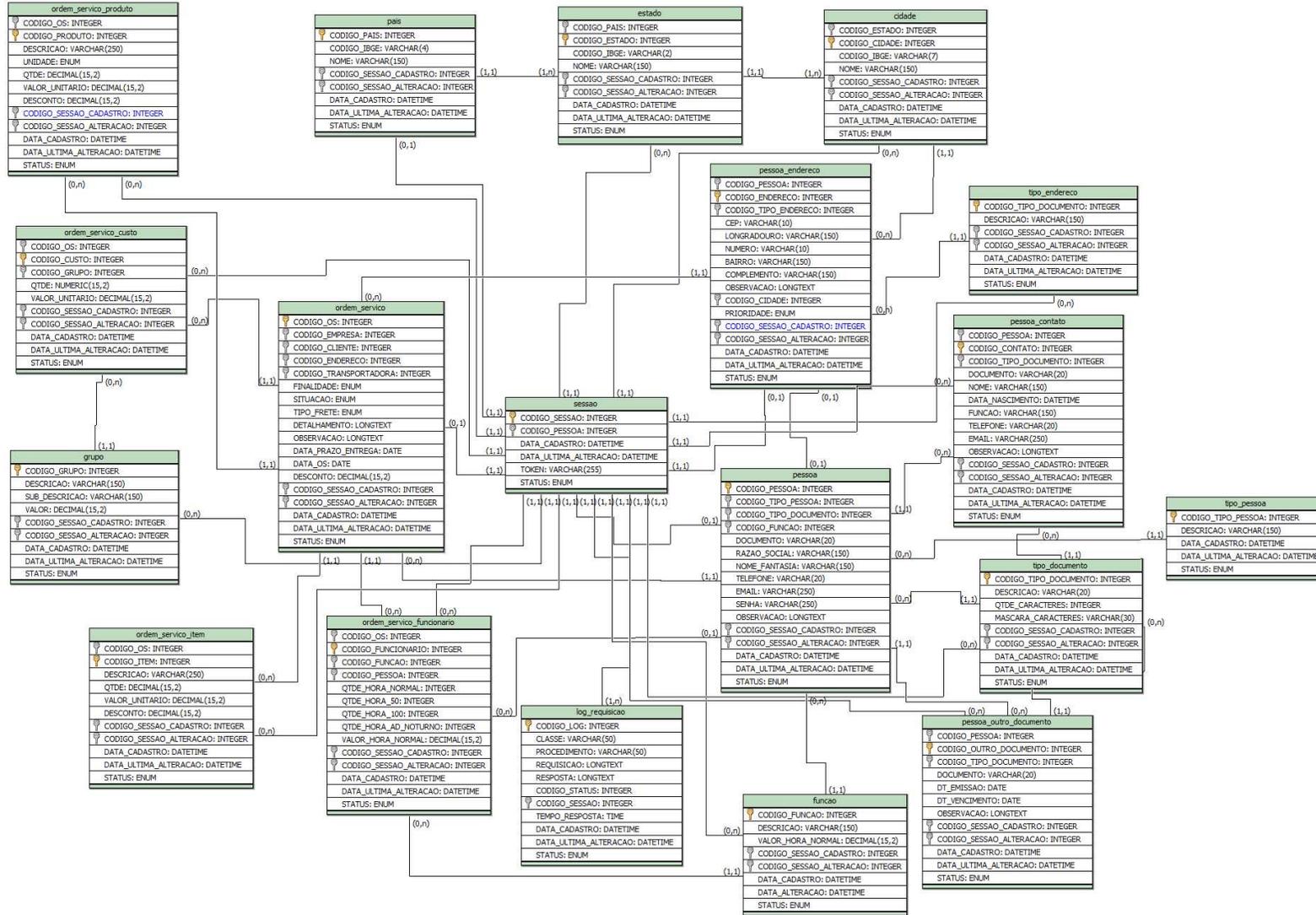
Figura 17 – Diagrama de Classe Web Service



Fonte: Autoria Própria (2023)

## **APÊNDICE C – Representação do Modelo Lógico**

Figura 18 – Representação do Modelo Lógico



Fonte: Autoria Própria (2023)