



ANTONIO MARANGUAPE

ANDRY ALBUQUERQUE DA SILVA

CLEANDA SALOMÃO DA SILVA ALVES

JUAN PABLO INHAMUS DA SILVA

MARCIO FERREIRA DE CARVALHO JUNIOR

TAINÁ CRUZ DA COSTA

TIAGO FIGUEIRA DO NASCIMENTO

NAVISLOG

MANAUS/AM

2025

ANTONIO MARANGUAPE

ANDRY ALBUQUERQUE

CLEANDA SALOMÃO DA SILVA ALVES

JUAN PABLO INHAMUS DA SILVA

MARCIO FERREIRA DE CARVALHO JUNIOR

TAINÁ CRUZ DA COSTA

TIAGO FIGUEIRA DO NASCIMENTO

NAVISLOG

Trabalho apresentado à Faculdade
Metropolitana de Manaus (FAMETRO) como
requisito parcial para a obtenção de nota na
disciplina de Gerencia de configuração

Orientadores: Paulo Rodrigo.

MANAUS/AM

2025

1. Introdução	4
2. Plano de Projeto	5
2.1. Registro de Alterações	5
2.2. Identificação do Projeto	5
2.3. Objetivo do Projeto	5
2.4. Escopo	5
2.5. Cronograma Geral (Macro)	7
2.6. Metodologia de Desenvolvimento	5
2.7. Papéis e Responsabilidades	7
2.8. Recursos Necessários	6
2.9. Riscos e Estratégias de Mitigação	6
2.10. Critérios de Sucesso	8
2.11. Comunicação	7
3. Documento de Requisitos	8
3.1. Objetivo	8
3.2. Escopo do Sistema	8
3.3. Definições, Acrônimos e Abreviações	9
3.4. Visão Geral do Sistema	9
3.5. Requisitos Funcionais	9
3.6. Requisitos Não Funcionais	10
3.7. Regras de Negócio	10
3.8. Casos de Uso (Resumo)	11
3.9. Protótipos / Wireframes	11
4. Especificação de Documento de Software	12
4.1. Introdução	12
4.2. Arquitetura Geral do Sistema	12
4.3. Casos de Uso Detalhados	12
4.4. Modelos UML	14
4.5. Diagrama de Casos de Uso (Textual)	14
4.6. Diagrama de Classes (Descrição)	16
5. Estrutura do Banco de Dados	18
6. Conclusão	19
7. Referências	20
8. Anexos	21

1 INTRODUÇÃO

O setor portuário desempenha um papel essencial no comércio nacional e internacional, sendo responsável por grande parte do transporte e movimentação de cargas no país. No entanto, apesar de sua relevância estratégica, ainda enfrenta diversos desafios relacionados à falta de digitalização, integração e automação dos processos operacionais.

Entre os principais problemas enfrentados estão atrasos frequentes nas operações de carga e descarga, baixa rastreabilidade durante a movimentação, processos burocráticos e pouco automatizados que elevam os custos operacionais, além da ausência de integração entre operadores, transportadores e autoridades portuárias.

Neste contexto, o projeto NavisLog surge como uma solução digital para otimizar a logística portuária, oferecendo funcionalidades de agendamento, rastreabilidade e monitoramento. Seu objetivo principal é aumentar a eficiência das operações, reduzir falhas humanas e integrar todos os agentes do processo.

O principal objetivo do sistema é aumentar a eficiência das operações portuárias, reduzindo falhas humanas, agilizando a comunicação entre os diferentes agentes e promovendo uma integração centralizada e segura dos dados. Dessa forma, o sistema busca contribuir para a modernização do setor e para o fortalecimento da competitividade logística brasileira.

2 PLANO DE PROJETO

2.1. Registro de Alterações

VERSÃO	ALTERAÇÕES
1.0	Plano de Projeto
1.3	Documentação de Requisitos
1.2	Especificação de documento de Software

2.2. Identificação do Projeto

- Nome: NavisLog
- Data de Início: 10/09/2025
- Data Estimada de Conclusão:
- Gerente de Projeto / Scrum Master: Taina Cruz
- Cliente / Product Owner: Tiago Figueira

Equipe de Desenvolvimento: Andry, Antônio, Cléo, Juan, Márcio, Tainá e Tiago

2.3 Objetivo do Projeto

Desenvolver uma plataforma digital acessível via web e dispositivos móveis, para aumentar a eficiência das operações portuárias, reduzindo falhas humanas, agilizando a comunicação entre os diferentes agentes e promovendo uma integração centralizada e segura dos dados para otimizar a logística portuária, com foco em cargas.

Módulos Principais:

- Módulo 1: Cadastro e gestão de embarcações e cargas, Agendamento de atracação e desatracação.
- Módulo 2: Rastreamento de cargas, Relatórios operacionais e estratégicos e Controle de usuários com permissões específicas.

2.4 Escopo

Incluído no escopo:

- Cadastro de embarcações e cargas.
- Agendamento de operações portuárias.
- Rastreamento de cargas.
- Relatórios gerenciais.

- Perfis de usuários (gerente, administrador, operador, transportador).
- Interface responsiva (web - mobile).
- Autenticação com dois fatores (2FA).
- Dashboard com indicadores e alertas operacionais.
- Histórico de movimentações portuárias.
- Sistema de controle de versões.

Fora do escopo:

- Controle financeiro.
- Gestão de tripulação.
- Integração com sistemas aduaneiros.
- Controle financeiro (faturamento, pagamentos, emissão de notas fiscais)
- Gestão de tripulação (contratos, escalas, informações pessoais de marítimos).
- Integração direta com sistemas aduaneiros ou da Receita Federal.
- Processamento de imagens ou leitura por OCR de documentos físicos.
- Chatbot ou atendimento virtual automatizado.
- Integração com sensores IoT (temperatura, umidade, peso das cargas).
- Monitoramento via câmeras de vigilância em tempo real.
- Gestão de contratos ou acordos logísticos entre parceiros.
- Previsão climática integrada para planejamento de atracação.
- Cálculo automatizado de tarifas portuárias ou taxas logísticas.
- Suporte multilíngue para uso internacional.
- Reconhecimento facial para acesso ao sistema.
- Aplicativo nativo offline para ambientes sem conexão com internet.

2.5. Cronograma Geral (Macro)

Semana	Entregas Previstas
1–2	Planejamento, setup do ambiente, levantamento de requisitos e design inicial
2–3	Desenvolvimento dos módulos de cadastro.
3–4	Implementação do rastreamento de cargas e relatórios
4–5	Integração de módulos, testes finais e documentação.

2.6 Metodologia de Desenvolvimento

O desenvolvimento será conduzido utilizando a metodologia ágil Scrum, dividida em quatro sprints quinzenais. Cada sprint contará com planejamento, execução, revisão e retrospectiva. Serão utilizadas ferramentas como Trello para gerenciamento de tarefas, Figma para design de interface e Python 3.14 no backend.

2.7. Papéis e Responsabilidades

Papel	<u>Responsável</u>	Atividades principais
Product Owner	Tiago Figueira	Priorização do backlog, validação das entregas
Scrum Master	Tainá Cruz	Remoção de impedimentos, condução das cerimônias
Dev Team	Tiago Figueira, Cleo, Marcio Tainá, Juan, Andry, Antonio	Codificação, testes, documentação técnica

2.8 Recursos Necessários

- Ferramentas de Desenvolvimento: Figma, SQLITE3, MySQL-WorkBench, Visual Studio Code, Draw.io.
- Gerenciamento: Trello
- Versionamento e Repositórios: Git e Git Hub
- Testes: Vercel, Flask, Render.
- Infraestrutura: Hospedagem local ou em nuvem (Flask, Heroku, Vercel, Render, Hostinger.)

2.9. Riscos e Estratégias de Mitigação

Risco	<u>Probabilidade</u>	<u>Impacto</u>	Mitigação
Falta de tempo dos membros	<u>Alta</u>	<u>Alto</u>	Reuniões curtas diárias para controle
Dificuldades técnicas com Spring Boot	Média	<u>Médio</u>	Mentoria ou uso de tutoriais / pares
Atrasos na entrega de requisitos	Média	<u>Alto</u>	Reuniões frequentes com o Product Owner
Perda de dados	<u>Baixa</u>	<u>Alto</u>	Backups automáticos do banco de dados

2.10 Critérios de Sucesso

- Funcionalidades entregues conforme escopo
- Relatórios gerados corretamente
- Sistema com segurança básica (login, JWT, werzekeg security, Flask login)
- Interface intuitiva e responsiva
- Documentação completa (README, banco)

2.11 Comunicação

- Reuniões semanais: Segunda - Terça-feira.
- Daily Scrum: 1 hora e 30 minutos por dia, via Discord ou presencial
- Ferramentas de comunicação: WhatsApp, Discord
- Repositório: GitHub (privado ou público com controle de branches)

3 DOCUMENTO DE REQUISITOS

3.1 Objetivo

Fornecer uma solução web que permita definir as funcionalidades de cadastro de embarcações e cargas, agendamento de operações portuárias, rastreamento de cargas, emitir relatórios, garantindo clareza na implementação.

3.2 Escopo do Sistema

O sistema web terá os seguintes módulos principais:

- Gerente: gestão completa de usuários, cargas e embarcações, emitir relatórios e assinaturas, permitir edições e cancelamentos de operações.
- Administrador: cadastro de usuários, cargas, embarcações, emitir relatórios
- Operador Portuário: agendamento e execução de operações de carga/descarga.
- Transportador: acompanhamento e rastreio das cargas.

3.3 Definições, Acrônimos e Abreviações

Termo	Definição
PO	Product Owner
CRUD	Create, Read, Update, Delete
API	Interface de Programação de Aplicações
JWT	JSON Web Token (token de autenticação)
RF	Requisito Funcional
RNF	Requisito Não Funcional

3.4 Visão Geral do Sistema

O sistema será acessado por diferentes perfis de usuários e permitirá interação centralizada para agilizar operações portuárias.

- Gerente
- Administrador
- Operador Portuário
- Transportador

3.5 Requisitos Funcionais (RF)

ID	Requisito Funcional
RF01	O sistema deve permitir o cadastro e login de usuários, com autenticação em duas etapas.
RF02	O sistema deve disponibilizar uma interface web responsiva, acessível via navegadores em dispositivos móveis e desktops.
RF03	O sistema de permitir cadastro de embarcações e cargas.
RF04	O sistema deve permitir agendamento de atracação e desatracação.
RF05	O sistema deve emitir relatórios operacionais e estratégicos.
RF06	O sistema deve permitir diferentes níveis de permissão por tipo de usuário.
RF07	O sistema deve armazenar de forma segura e eficiente e segura todos os tipos de dados pertinentes às operações aduaneiras para fins de inspeção e controle de cargas.
RF08	O sistema deve reconhecer categoricamente as informações preenchidas, e analisar os dados automaticamente.
RF09	O sistema deve ter mecanismos de defesa contra invasores e sistemas de alertas de invasões.
RF10	Cada nova implementação que o sistema irá receber deverá ser primordialmente documentada, testada, analisada e posteriormente implementada de forma segura.
RF11	O sistema deve permitir a recuperação de senha por e-mail ou SMS.
RF12	O sistema deve manter um log de auditoria com todas as ações críticas realizadas pelos usuários.
RF13	O sistema deve notificar automaticamente o operador ou transportador sobre alterações em agendamentos.
RF14	O sistema deve permitir a exportação de relatórios em formatos PDF e Excel.

RF15	O sistema deve validar campos obrigatórios antes da submissão de formulários.
RF16	O sistema deve permitir a associação de múltiplas cargas a uma única embarcação.
RF17	O sistema deve oferecer um painel de controle para o administrador acompanhar o status geral das operações.
RF18	O sistema deve permitir a importação de dados via planilhas CSV para migração inicial.
RF19	O sistema deve exibir alertas visuais em caso de conflito de agendamentos.
RF20	O sistema deve armazenar localmente e na nuvem os dados críticos para redundância.

3.6 Requisitos Não Funcionais (RNF)

ID	Requisito Não Funcional
RNF01	O sistema deve garantir confidencialidade e integridade dos dados, utilizando criptografia via SSL/TLS, Werzek Security, Flask login.
RNF02	O sistema deve implementar autenticação de dois fatores para acesso seguro dos usuários.
RNF03	O sistema deve ter alta disponibilidade e desempenho, suportando múltiplos acessos simultâneos.
RNF04	O sistema deve seguir boas práticas de design responsivo para uso em smartphones e tablets.
RNF05	O sistema deve ser compatível com navegadores modernos (Chrome, Firefox, Edge, Firefox etc.)
RNF06	O tempo de resposta das requisições não deve ultrapassar 4 segundos em 95% das requisições.
RNF07	O sistema deve manter compatibilidade com as principais versões dos navegadores nos últimos 2 anos.
RNF08	O sistema deve permitir a realização de backups automáticos diários.
RNF09	O sistema deve registrar logs de falhas com nível de severidade e timestamp.
RNF10	Toda comunicação entre cliente e servidor deve ser criptografada (HTTPS/SSL).
RNF11	A aplicação deverá ter tolerância a falhas, permitindo continuidade parcial das funções em caso de falha em módulo secundário.

3.7 Regras de Negócio

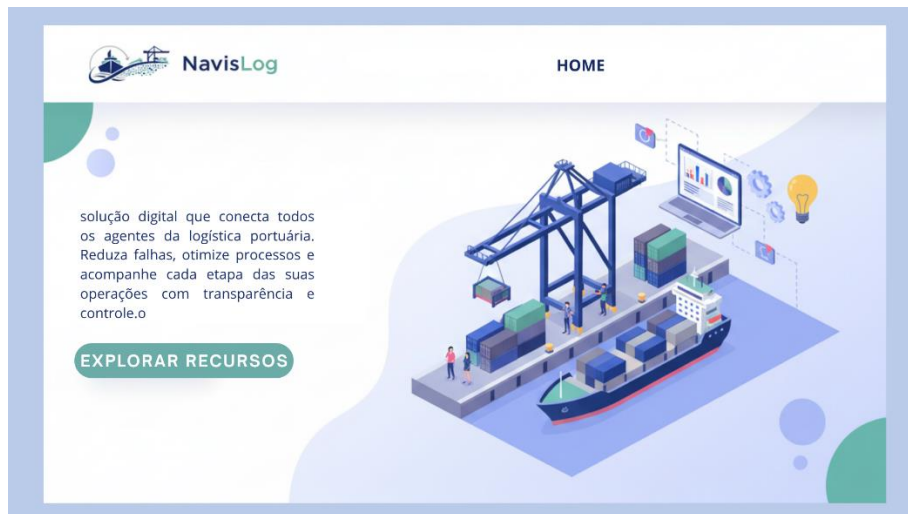
ID	Regra de Negócio
RN01	Um agendamento só poderá ser confirmado se houver disponibilidade da embarcação.
RN02	Alterações em operações devem ser registradas com data e hora.
RN03	Apenas usuários autorizados podem acessar relatórios estratégicos.
RN04	O tempo de recuperação após falha (RTO) não deve ultrapassar 4 horas.
RN05	O banco de dados deverá suportar no mínimo 100.000 registros sem perda de desempenho.
RN06	O sistema deverá possuir suporte técnico básico com SLA de resposta de até 48h úteis.
RN07	O sistema deve bloquear o acesso após 5 tentativas de login mal-sucedidas.
RN08	O sistema deverá ter controle de sessões via token de acessos com tempo específicos de permanência do usuário.
RN09	Todos os registros de acesso ao sistema (login, logout e falhas de autenticação) deverão ser armazenados os IPs de origem, IDs e usuário responsável.
RN10	Em caso de inatividade por mais de 30 minutos, a sessão do usuário deverá ser encerrada automaticamente por segurança.

3.8 Casos de Uso (resumo)

Caso de Uso	Atores envolvidos
Cadastrar embarcação	Administrador
Cadastrar carga	Administrador, Operador
Agendar atracação/ desatracação	Administrador, Operador
Rastrear carga	Administrador, Operador, transportador
Emitir relatórios	Administrador

3.9 Protótipo / Wireframe (didático)

- Tela Inicial



4 ESPECIFICAÇÃO DE DOCUMENTO DE SOFTWARE

ESPECIFICAÇÃO DE REQUISITOS DE SOFTWARE (ERS)

4.1 Introdução

Este documento detalha os requisitos descritos anteriormente, especificando comportamentos esperados, fluxos de uso, validações, estrutura de dados e modelos UML (descritos em texto). Serve como base técnica para desenvolvedores e testadores.

4.2 Arquitetura Geral do Sistema

Arquitetura de Camadas

O sistema será dividido em:

- Camada de Apresentação (Front-end): Interface para interação com usuário (html e css.)
- Camada de Aplicação (Back-end): Regras de negócio e controle de fluxos (Flask, python 3.14)
- Camada de Persistência: Banco de dados relacional (MySQL, SQL, SQLite3)
- Camada de Segurança: (werkzeug security) para hash de senhas e (flask_login) para autenticação da sessão do usuário

4.3 Casos de Uso Detalhados

UC01 – Cadastrar Embarcação

- Ator Primário: Gerente/Administrador

Fluxo Principal:

1. Administrador acessa o módulo de cadastro de embarcações.
2. Informa dados obrigatórios: nome da embarcação, identificação IMO, capacidade, país de origem e status.
3. O sistema valida os campos obrigatórios.
4. O sistema grava os dados no banco e exibe a mensagem de sucesso.

Fluxo Alternativo:

Caso a embarcação já esteja cadastrada, o sistema exibe: “Embarcação já registrada.”

UC02 – Cadastrar Carga

- Ator Primário: Administrador / Operador Portuário

Fluxo Principal:

1. O ator acessa o módulo “Cadastro de Cargas”.
2. Informa tipo da carga, peso, destino, origem e embarcação associada.
3. O sistema valida os campos obrigatórios.
4. O sistema registra a carga e associa à embarcação selecionada.

Fluxo Alternativo:

Caso a embarcação não exista, o sistema notifica: “Embarcação não encontrada.”

UC03 – Agendar Atracação/Desatracação

- Ator Primário: Administrador / Operador Portuário

Fluxo Principal:

1. O ator seleciona o menu “Agendamentos”.
2. Escolhe o tipo de operação (atracação ou desatracação).
3. Define data, hora e embarcação.
4. O sistema verifica disponibilidade de berço e conflitos de horário.
5. Caso disponível, o sistema confirma e registra o agendamento.

Fluxo Alternativo:

Caso exista conflito de horário, o sistema exibe alerta: “Conflito de agendamento detectado.”

UC04 – Rastrear Carga

- Atores Primários: Administrador, Operador Portuário, Transportador

Fluxo Principal:

1. O ator acessa o módulo “Rastreamento”.
2. Informa o número da carga ou embarcação.
3. O sistema busca as informações no banco de dados.
4. O sistema exibe status, histórico de movimentações e localização atual.

Fluxo Alternativo:

Caso o identificador não exista, o sistema exibe: “Carga não localizada.”

UC05 – Emitir Relatórios Operacionais

- Ator Primário: Administrador / Gerente

Fluxo Principal:

1. O ator acessa o módulo “Relatórios”.
2. Seleciona o tipo de relatório (embarque, carga, agendamento, usuários).
3. Define filtros de data, embarcação ou status.
4. O sistema gera o relatório com base nos dados filtrados.
5. O usuário pode exportar em PDF ou Excel.

Fluxo Alternativo:

Caso não existam dados no período solicitado, o sistema exibe: “Sem registros para o filtro aplicado.”

UC06 – Gerenciar Usuários

- Ator Primário: Gerente / Administrador

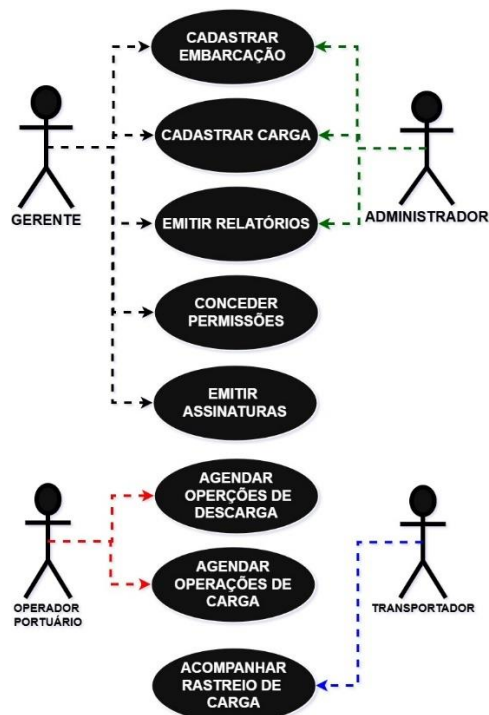
Fluxo Principal:

1. O ator acessa o painel “Gerenciamento de Usuários”.
2. Visualiza lista de usuários cadastrados.
3. Pode criar, editar ou desativar usuários.
4. O sistema valida o nível de permissão e registra as alterações.

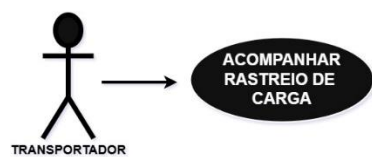
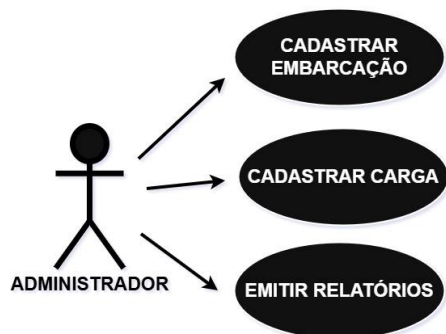
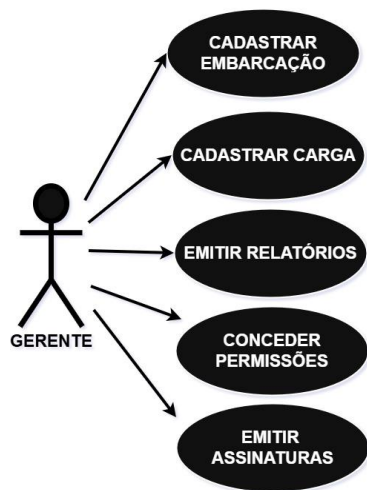
Fluxo Alternativo:

Caso o ator não tenha permissão suficiente, o sistema exibe: “Acesso negado.”

4.4 Diagrama de Casos de Uso (geral)

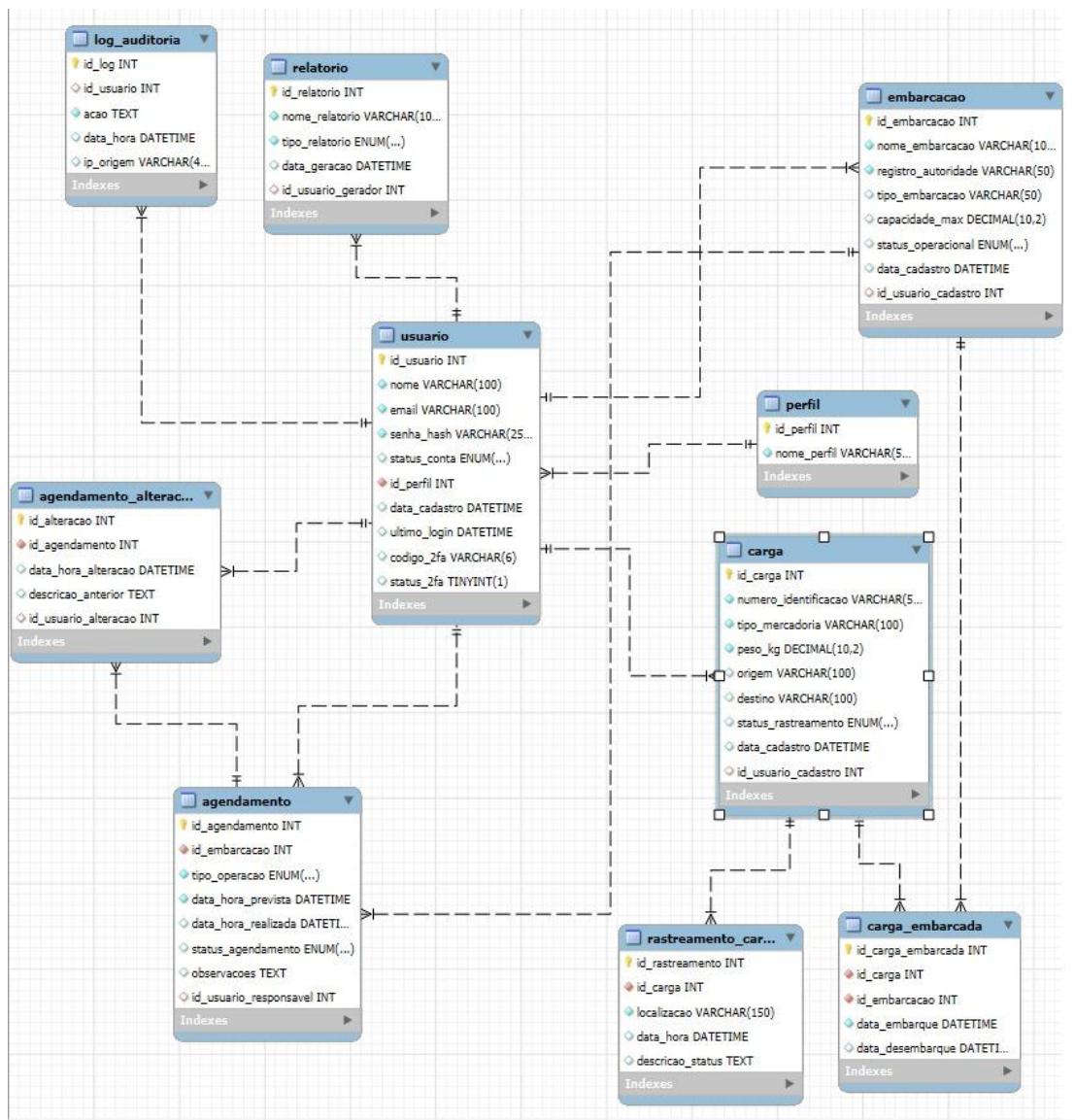


4.5 Diagrama de Casos de Uso (simplificado – textual)



4.6 Diagrama Entidade Relacionamento

5. Diagrama de Classes - SQL



CONCLUSÃO

O projeto NavisLog foi idealizado como uma solução digital voltada para melhorar a comunicação e a integração entre todos os agentes da logística portuária. A proposta busca reduzir falhas, otimizar processos e permitir o acompanhamento de cada etapa das operações com mais transparência e controle. Durante o desenvolvimento da documentação e do planejamento do sistema, foi possível compreender melhor as necessidades do setor e estruturar um projeto que futuramente poderá contribuir para a modernização das atividades portuárias. O trabalho também proporcionou à equipe uma vivência prática de como planejar e documentar um sistema real, aplicando conceitos de engenharia de software, organização e trabalho em equipe.

REFERÊNCIA

TURBAN, Efraim; POLLARD, Carol; WOOD, Gregory. Tecnologia da Informação para Gestão: em busca do desempenho estratégico e da produtividade. 10. ed. Rio de Janeiro: LTC, 2018.

<https://www.gov.br/antag>

<https://www.portodesantos.com.br>