FACULDADE DE TECNOLOGIA SENAI DE DESENVOLVIMENTO GERENCIAL (FATESG)

ENGENHARIA DE SOFTWARE

DANIELLE BRAGANÇA MACHADO ALVES

GUILHERME FELIPE ALVES CARDOSO

MICHELLY LIMA DE OLIVEIRA

RYAN RODRIGUES DOS SANTOS

JÚLLYA DE ANDRADE CARVALHO

SISTEMA COMPUTACIONAL PARA GERENCIAMENTO FINANCEIRO DA FROTA DA GYNLOG

PROJETO INTEGRADOR APLICADO À MODELAGEM, REQUISITOS E CONSTRUÇÃO DE SOFTWARE ORIENTADO A OBJETOS

GOIÂNIA

2025

DANIELLE BRAGANÇA MACHADO ALVES

GUILHERME FELIPE ALVES CARDOSO

MICHELLY LIMA DE OLIVEIRA

RYAN RODRIGUES DOS SANTOS

JÚLLYA DE ANDRADE CARVALHO

SISTEMA COMPUTACIONAL PARA GERENCIAMENTO FINANCEIRO DA FROTA DA GYNLOG

PROJETO INTEGRADOR APLICADO À MODELAGEM, REQUISITOS E CONSTRUÇÃO DE SOFTWARE ORIENTADO A OBJETOS

Trabalho apresentado a Faculdade Senai Fatesg como conclusão do segundo período, desenvolver um sistema em Java para registrar, controlar e gerar relatórios sobre os gastos da frota veicular da empresa GynLog.

Orientadores: Prof. Eugênio Júlio Messala C.

Prof. Plinio Marcos Mendes Carneiro

Prof. Amós Souza Fernandes

GOIÂNIA

2025

SUMÁRIO

1. INTRODUÇÃO

O presente Projeto Integrador aborda o desafio de Controle de gastos da frota veicular para a empresa GynLog, atuante no setor de transporte de cargas. A gestão eficiente de frotas é crucial para a saúde financeira de empresas desse ramo, e a GynLog identificou a necessidade de contratar o desenvolvimento de um sistema que permita o registro e o controle detalhado dos gastos financeiros associados a cada um dos veículos que compõem sua frota.

Atualmente, a empresa enfrenta dificuldades na organização e monitoramento dessas informações, uma vez que o controle dos veículos, das despesas e das movimentações financeiras não é realizado de forma centralizada. Esse cenário gera riscos como perda de dados, inconsistência entre registros, dificuldades na geração de relatórios gerenciais e pouca visibilidade sobre custos críticos — como combustível, manutenção, IPVA e multas. Além disso, a ausência de um processo padronizado dificulta a identificação de veículos inativos, o acompanhamento das despesas mensais e a análise estratégica dos gastos operacionais. Diante dessa realidade, tornou-se evidente a necessidade de uma solução tecnológica capaz de estruturar essas informações e automatizar o processo de gestão de gastos da frota.

Dessa forma, o objetivo central deste projeto foi o planejamento e a construção de um software para gerenciar esses gastos. A solução desenvolvida consiste em uma aplicação em Java, utilizando a programação orientada a objetos. O sistema foi projetado para permitir o registro de veículos, categorização de despesas e transações, além de gerar relatórios gerenciais essenciais.

Como parte do Curso Superior de Engenharia de Software, 2º Período (2025), este Projeto Integrador permitiu colocar em prática vários conhecimentos desenvolvidos ao longo do semestre. Durante sua realização, foi possível aplicar técnicas de elicitação e especificação de requisitos, desenvolver soluções utilizando princípios da construção de software orientado a objetos e modelar processos fundamentais da engenharia de software.

Mais do que cumprir uma demanda acadêmica, o projeto serviu como oportunidade para transformar teoria em prática e aprofundar habilidades essenciais para a formação profissional.

2. DESENVOLVIMENTO

2.1 Programação Orientada a Objeto (POO)

A Programação Orientada a Objetos (POO) é o pilar principal utilizado no desenvolvimento do sistema de controle de veículos, despesas e movimentações. Essa abordagem permite representar cada elemento do sistema como um objeto do mundo real, como: Veículo, Despesa, Movimentação, facilitando a organização lógica do software.

2.2 Estrutura e Organização

O sistema foi dividido em camadas e classes específicas:

Modelos (Model): Veículos, Tipos de Despesas, Movimentações

Controle (Controller): VeiculoControle, DespesaControle, MovimentacaoControle

Persistência (DAO): arquivos .txt com leitura/escrita

Interface Gráfica (CRUD): telas Swing como TelaCadastroVeiculos, TelaDespesas, TelaMovimentações.

Essa estrutura orientada a objetos facilita: manutenção do sistema,reutilização de código e evolução futura sem quebrar partes já existentes.

2.3 Encapsulamento

As classes possuem atributos privados, como:

idVeiculo, placa, modelo, anoFabricacao, status, descricao, idDespesa, idMovimentacao, data, valor, entre outros

E métodos públicos de acesso (getters e setters).

Isso impede que outro trecho do sistema altere informações diretamente no arquivo texto ou dentro do objeto.

Todas as alterações passam pelas regras do Controle, garantindo a integridade dos dados.

2.4 Polimorfismo e Interfaces

O projeto utiliza interfaces — como IVeiculosCRUD, ITiposDespesasCRUD, IMovimentacaoCRUD, para garantir um padrão de métodos:

salvar();

atualizar();

listar();

remover();

buscarPorId();

Dessa forma aumenta a flexibilidade do sistema e também gera facilidade de substituição do DAO no futuro, como: trocar o arquivo txt por um banco de dados.

2.5. Tratamento de Exceções

O sistema utiliza tratamento de erros de forma obrigatória, garantindo que o usuário seja avisado quando:

Tenta editar trocando o ID;

Insere placa que já existe;

Cadastrar com o mesmo id;

Não é permitido remover o id;

Erro na Leitura;

Placa não pode ser nula;

O id não pode ser nulo;

O ano de Fabricação não pode ser nulo;

Despesa for nula;

Data for nula;

Se o veículo estiver inativo;

Valor inválido;

Movimentação sem um veículo vinculado.

A POO foi fundamental para o desenvolvimento do sistema pois foi possível organizar o código em camadas e classes bem definidas, garantir a segurança dos dados via encapsulamento, permitiu foco apenas nas informações relevantes, padronizou operações com interfaces e utilizou exceções para evitar erros no cadastro, edição e remoção.

3. Modelagem de Processos de Negócio (BPMN)

Para representar de forma clara e estruturada os processos de negócio da GynLog que serão informatizados, foi utilizado o Business Process Model and Notation (BPMN). Essa notação gráfica facilita a visualização das etapas envolvidas nas rotinas da empresa e torna o entendimento acessível tanto para a equipe de desenvolvimento quanto para os responsáveis do setor operacional.

Importância:

O BPMN oferece uma linguagem padronizada e de fácil interpretação, permitindo que todos os envolvidos, técnicos e não técnicos, compreendam como as atividades se conectam. Isso contribui para alinhamento, redução de ambiguidades e melhor tomada de decisão durante o desenvolvimento do sistema.

Aplicação no Projeto:

No contexto deste Projeto Integrador, o BPMN foi empregado para:

- Modelar o fluxo de registro dos veículos, incluindo dados como placa e status (ativo/ inativo).

- Representa o processo de categorização das despesas.

- Detalhar o fluxo de registro das transações financeiras.

- Estruturar os processos de geração dos relatórios gerenciais.

Em anexo segue o diretório para o Github:

https://github.com/RyanRodr1gues1/Projeto-Integrador_Gynlog-Eng2_Fluxos.git

4. Sistemas Operacionais

O sistema foi projetado para funcionar tanto em ambientes Linux quanto Windows, garantindo maior flexibilidade de uso dentro da GynLog. Essa compatibilidade reforça a escolha da linguagem Java, já que essa linguagem permite que o software seja executado de forma consistente em diferentes sistemas operacionais, sem necessidade de adaptações específicas.

Além disso, o sistema oferecerá a possibilidade de gerar arquivos em pdf, permitindo que o usuário visualize os registros de maneira mais prática. Esse recurso torna o controle financeiro mais transparente e possibilita análises externas de forma organizada.

Critério	**Linux**	**Windows**	**macOS**	**Unix**
Desenvolvedor	Comunidade global + empresas (ex.:	Microsoft	Apple	Empresas diversas (ex.: IBM, HP, Oracle)

	Red Hat, Canonical)			
Código-fonte	Aberto (open source)	Fechado (proprietário)	Parcialmente fechado (baseado em BSD, mas com camadas proprietárias)	Geralmente fechado, mas baseado em padrões POSIX
Licença	GPL, LGPL, MIT, entre outras	Licença comercial Microsoft	Licença proprietária Apple	Licenças proprietárias variadas (System V, BSD etc.)
Interface gráfica	Diversas opções: GNOME, KDE, XFCE etc.	Interface Windows (Explorer)	Interface Aqua	Geralmente minimalista; algumas distribuições têm GUI própria
Interface em linha de comando	Shells como Bash, Zsh, Fish; poderosa e amplamente utilizada	CMD e PowerShell	Terminal (Zsh/Bash)	Muito forte, tradicionalmente focado em CLI
Núcleo (Kernel)	Linux Kernel (monolítico modular)	Windows NT Kernel	XNU (híbrido, baseado em Mach + BSD)	Variantes de kernels Unix (System V, BSD etc.)
Suporte a hardware	Excelente, mas depende da distribuição; pode ter limitações em hardware muito novo	Muito amplo; excelente compatibilidade	Limitado a hardware Apple	Depende da implementação (AIX, HP-UX etc.); mais restrito

Segurança	Alta; forte arquitetura de permissões; muito menos alvo de malware	Boa, porém o mais visado por ataques por ser o mais usado	Alta; sistema mais fechado e controlado	Muito alta; usado em ambientes críticos e servidores
Estabilidade e desempenho	Muito estável, excelente em servidores e multitarefa	Bom desempenho; pode degradar em longos usos	Muito estável e otimizado para hardware Apple	Extremamente estável e usado em sistemas corporativos
Suporte a software/aplicações	Grande variedade, principalmente open source; alguns softwares comerciais não têm versão nativa	Maior compatibilidade com softwares comerciais e jogos	Bom suporte, mas limitado ao ecossistema Apple	Voltado para aplicações corporativas e industriais

5. Conclusão

O desenvolvimento deste Projeto Integrador permitiu solucionar de forma eficiente o problema enfrentado pela GynLog no controle financeiro de sua frota veicular. A criação do sistema em Java, utilizando Programação Orientada a Objetos, possibilitou organizar dados de forma estruturada, garantir integridade das informações e automatizar processos que antes eram feitos manualmente.

Com a modelagem BPMN, foi possível compreender e representar com clareza os fluxos operacionais da empresa, facilitando a transformação desses processos em funcionalidades reais no software. A implementação dos requisitos funcionais e não funcionais proporcionou uma solução robusta, capaz de atender às demandas atuais e permitir evolução futura.

O projeto não apenas cumpriu com os objetivos acadêmicos do curso, mas também proporcionou experiência prática em engenharia de software, modelagem, desenvolvimento orientado a objetos e organização de processos. Como resultado, o sistema entrega mais precisão, segurança e agilidade ao gerenciamento financeiro da frota, contribuindo diretamente para a eficiência operacional e tomada de decisão da GynLog.

6. Referências

Java Enums Explained in 6 Minutes. YouTube, Disponível em: https://youtu.be/wq9SJb8VeyM?si=Q96SnxEHw623Co0O.

JavaGram - Gerando um documento PDF com a biblioteca itextPDF. YouTube, Disponível em: https://youtu.be/VREjjRFKR3A?si=nMbVcf7PG1Mt0OOd.

JavaGram - Gerando uma lista com imagens do banco em um documento PDF - Parte 2/2. YouTube, Disponível em: https://youtu.be/CzblpOxikH4?si=wtAbK7QlNPnjRAWA.

Java Swing UI Design - Login And Register. YouTube, Disponível em: https://youtu.be/ZUGkQHtLS4U?si=pHoWSnJ9DnKYy9iC.