

CENTRO PAULA SOUZA
FATEC FERRAZ DE VASCONCELOS
6 ° SEMESTRE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

MELISSA MONTEIRO DA ROCHA

DRIVEINTEL

Software para precificar veículos

Ferraz de Vasconcelos

2025

CENTRO PAULA SOUZA
FATEC FERRAZ DE VASCONCELOS
6 ° SEMESTRE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

MELISSA MONTEIRO DA ROCHA

DRIVEINTEL

Software para precificar veículos

Trabalho apresentado à disciplina de
LABORATÓRIO DE ENGENHARIA DE
SOFTWARE, para obtenção de nota na
matéria do 6º Semestre do curso de
Análise e desenvolvimento de sistema,
pela Fatec Ferraz de Vasconcelos, com
orientação da Professora Patrícia
Mendes Sarno.

Ferraz de Vasconcelos – 2025

RESUMO

A precificação de veículos é um fator estratégico para empresas do setor automotivo, impactando diretamente a competitividade e a rentabilidade de concessionárias, locadoras e revendedores. Métodos tradicionais, como a Tabela FIPE, fornecem apenas um valor médio sem considerar fatores dinâmicos como oferta, demanda e sazonalidade. Essa limitação dificulta a definição de preços mais precisos e alinhados com a realidade do mercado.

Para solucionar esse problema, foi desenvolvido o DriveIntel, uma ferramenta de inteligência de mercado automotivo que utiliza automação, estatística e inteligência artificial para coletar, analisar e interpretar grandes volumes de dados de anúncios de veículos. O sistema emprega web scraping para extrair informações de diversas plataformas, processando os dados para gerar relatórios detalhados e identificar padrões de precificação. Além disso, um modelo de inteligência artificial prevê valores futuros de veículos, permitindo simulações baseadas em variáveis como marca, modelo, ano e quilometragem. O estudo teve como objetivo geral desenvolver uma solução capaz de auxiliar empresas na precificação estratégica, garantindo maior precisão na tomada de decisões sobre compra, venda e gestão de estoque. Os objetivos específicos incluíram a coleta automatizada de dados, a aplicação de modelos estatísticos e de IA para identificar padrões de precificação e a geração de insights estratégicos para otimizar a competitividade das empresas do setor automotivo. Os resultados demonstraram que o DriveIntel atende a essas necessidades ao fornecer um panorama atualizado do mercado, permitindo que empresas acompanhem oscilações de preços em tempo real e tomem decisões embasadas em dados. A solução proposta se mostrou eficaz na transformação de grandes volumes de informações brutas em insights acionáveis, possibilitando uma precificação mais dinâmica e ajustada às condições de mercado.

Conclui-se que o DriveIntel representa uma alternativa inovadora e complementar às tabelas de referência tradicionais, auxiliando empresas do setor automotivo a superar desafios na definição de preços estratégicos. Como

trabalho futuro, sugere-se a expansão da plataforma com modelos preditivos mais avançados e a integração de novas fontes de dados para aumentar ainda mais a precisão das análises.

Palavras-chave: Precificação de veículos, inteligência artificial, automação, análise de dados, inteligência de mercado.

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1 - Modelagem de Negócio Microserviço de RPA (API) | 18 |
| Figura 2 - Diagrama de Sequência Microserviço de RPA (API)..... | 24 |
| Figura 3 - Diagrama de Casos de Uso Específico Microserviço de RPA (API) | 24 |
| Figura 4 - Modelo de Entidade-Relacionamento Microserviço de RPA (API) . | 26 |
| Figura 5 - Diagrama de Classes Microserviço de RPA (API) | 27 |
| Figura 6 - Diagrama de Atividades do Negócio Microserviço de IA (API) | 28 |
| Figura 7 - Diagrama de Sequência - Microserviço de IA (API) | 33 |
| Figura 8 - Diagrama de Classes Microserviço de IA (API)..... | 34 |
| Figura 9 - Diagrama de Casos de Uso Microserviço de IA (API)..... | 35 |
| Figura 10 - Variáveis Microserviço de IA (API) | 37 |
| Figura 11 - Features Microserviço de IA (API) | 38 |
| Figura 12 - BPMN Projeto Django | 42 |
| Figura 13 - Diagrama de Sequência Projeto Django | 47 |
| Figura 14 - Diagrama de caso de Uso Geral Projeto Django | 48 |
| Figura 15 - Modelo Entidade-Relacionamento (ER) Projeto Django | 49 |

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 - Requisitos Funcionais - Microserviço de RPA..... | 19 |
| Tabela 2 - Requisitos Não Funcionais - Microserviço de RPA..... | 21 |
| Tabela 3 - Regras de Negócio - Microserviço de RPA..... | 22 |
| Tabela 4 - Matriz de Rastreabilidade - Microserviço de RPA..... | 23 |
| Tabela 5 - Especificação de Casos de Uso - Microserviço de RPA | 25 |
| Tabela 6 -Requisitos Funcionais – Microserviço de IA | 29 |
| Tabela 7 - Requisitos Não Funcionais – Microserviço de IA..... | 31 |
| Tabela 8 - Regras de Negócio – Microserviço de IA..... | 32 |
| Tabela 9 - Matriz de Rastreabilidade – Microserviço de IA..... | 32 |
| Tabela 10 - Especificações de Casos de Uso – Microserviço de IA | 36 |
| Tabela 11 - Requisitos Funcionais – Projeto Django | 43 |
| Tabela 12 - Requisitos Não Funcionais – Projeto Django | 45 |
| Tabela 13 - Regras de Negócio – Projeto Django | 46 |
| Tabela 14 - Matriz de Rastreabilidade – Projeto Django | 46 |

SUMÁRIO

| | |
|--|----|
| 1.INTRODUÇÃO | 10 |
| 2. OBJETIVOS | 12 |
| 2.1. Objetivo Geral | 12 |
| 2.2. Objetivos Específicos | 12 |
| 3. PROBLEMA | 13 |
| 3.1. O desafio do setor automotivo na precificação de veículos | 13 |
| 3.2. Dores do Cliente..... | 13 |
| 3.3. Solução oferecida pela DriveIntel..... | 14 |
| 4. A empresa | 16 |
| 4.1. O que fazemos? | 16 |
| 4.2. Nossa abordagem | 17 |
| 4.3. Por que escolher a DriveIntel? | 17 |
| 5. DESENVOLVIMENTO | 18 |
| 5.1 Microserviço de RPA (API) | 18 |
| 5.1.2 Modelagem de Negócio..... | 18 |
| 5.1.3 Engenharia de requisitos..... | 19 |
| 5.1.3.1Requisitos Funcionais | 19 |
| 5.1.3.2 Requisitos Não Funcionais..... | 20 |
| 5.1.3.3 Regras de Negócio..... | 21 |
| 5.1.3.4 Matriz Rastreabilidade..... | 23 |
| 5.1.4 PROJETO DA SOLUÇÃO | 24 |
| 5.1.4.1 Diagrama de Sequência..... | 24 |
| 5.1.4.2 Diagrama de Casos de Uso Específico | 24 |
| 5.1.4.3 Especificação de Casos de Uso..... | 25 |
| 5.1.4.4. Modelo Entidade-Relacionamento (ER) | 26 |
| 5.1.4.5. Diagrama de Classes | 27 |
| 5.2 Modelagem de Negócio – Microserviço de IA (API) | 28 |
| 5.2.1 Diagrama de Atividades do Negócio – BPMN | 28 |
| 5.2.2 Engenharia de Requisitos | 29 |
| 5.2.2.1. Requisitos Funcionais (RF) | 29 |

| | |
|---|----|
| 5.2.2.2. Requisitos Não Funcionais (RNF) | 30 |
| 5.2.2.3. Regras de Negócio..... | 31 |
| 5.2.2.4. Matriz de Rastreabilidade..... | 32 |
| 5.2.2.6. Diagrama de Sequência..... | 33 |
| 5.2.2.10. Diagrama de Classes | 34 |
| 5.2.2.11. Diagrama de Casos de Uso..... | 35 |
| 5.2.3.1. Variáveis..... | 37 |
| 5.2.3.1.2 Target (Variável Alvo)..... | 37 |
| 5.2.3.1.3 Feature (Recurso) | 37 |
| 5.2.3.1.3 Entendendo o tipo das features..... | 38 |
| 5.2.3.1.4 Features Categóricas | 39 |
| 5.2.3.1.5 Features Numéricas | 39 |
| 5.2.3.2 MÉTRICAS DE AVALIAÇÃO | 40 |
| 5.3 MODELAGEM DE NEGÓCIO – PROJETO DJANGO..... | 42 |
| 5.3.2.1 Diagrama de Atividades do Negócio -BPMN | 42 |
| 5.3.2.2 Requisitos Funcionais (RF) | 43 |
| 5.3.2.3 Requisitos Não Funcionais (RNF) | 44 |
| 5.3.2.4 Regras de Negócio..... | 46 |
| 5.3.2.5 Matriz de Rastreabilidade..... | 46 |
| 5.3.2.6 Diagrama de Sequência | 47 |
| 5.3.2.7 Diagrama de caso de Uso Geral | 48 |
| 5.3.2.8 Modelo Entidade-Relacionamento (ER) | 49 |
| 6. conclusão | 50 |
| 7. REFERÊNCIAS..... | 52 |

1.INTRODUÇÃO

A precificação de veículos é um fator fundamental para a dinâmica do setor automotivo, impactando concessionárias, locadoras e demais empresas que operam no mercado de compra e venda de automóveis. Tradicionalmente, tabelas de referência, como a Tabela FIPE, são amplamente utilizadas para estabelecer um valor médio de mercado. No entanto, essas tabelas apresentam um enfoque mais estático, sem considerar variações mais dinâmicas do mercado, como oscilações na oferta e demanda, sazonalidade e atributos específicos dos veículos.

Nesse contexto, a necessidade de ferramentas complementares que ofereçam uma visão analítica e estratégica se torna evidente. Com o avanço da inteligência artificial e da automação de dados, tornou-se possível desenvolver soluções que vão além das médias fixas, permitindo uma análise mais detalhada da precificação. Empresas do setor automotivo podem se beneficiar de insights personalizados, embasados em grandes volumes de dados coletados diretamente do mercado.

O DriveIntel surge como uma ferramenta de inteligência de mercado automotiva, focada na análise de precificação e tendências do setor. A plataforma utiliza web scraping para coletar e estruturar dados de anúncios de veículos, aplicando métodos estatísticos e modelos de inteligência artificial para identificar padrões de preço e simular cenários com base em variáveis específicas. Dessa forma, possibilita que concessionárias, locadoras e empresas do setor automotivo tenham acesso a informações estratégicas para tomada de decisão mais assertiva.

Diante desse cenário, este trabalho busca responder à seguinte questão: como a análise de grandes volumes de dados pode contribuir para a precificação estratégica de veículos no mercado automotivo? Para isso, serão exploradas metodologias de extração, processamento e análise de dados, além da aplicação da teoria do preço hedônico, conforme discutida por Brandão e Scarpel

(2009), que destaca o impacto de atributos como potência, luxo e marca na formação dos preços.

Assim, o objetivo geral deste estudo é desenvolver uma ferramenta de inteligência de mercado que auxilie empresas automotivas na análise de precificação. Como objetivos específicos, busca-se automatizar a coleta de dados, empregar técnicas de inteligência artificial para identificação de padrões de precificação e fornecer insights estratégicos para a tomada de decisão empresarial.

2. OBJETIVOS

2.1. Objetivo Geral

Desenvolver **uma ferramenta de inteligência de mercado** para o setor automotivo, capaz de coletar, analisar e interpretar dados de precificação de veículos por meio de **automação, estatística e inteligência artificial**, auxiliando empresas na **tomada de decisões estratégicas** sobre compra, venda e precificação.

2.2. Objetivos Específicos

1. **Coletar e organizar grandes volumes de dados** provenientes de anúncios de veículos por meio de *web scraping*, consolidando informações relevantes para a análise de preços e tendências do mercado automotivo.
2. **Aplicar modelos estatísticos e técnicas de inteligência artificial** para identificar padrões de precificação e permitir simulações baseadas em diferentes cenários, considerando variáveis como modelo, ano e características do veículo.
3. **Fornecer insights estratégicos para empresas do setor automotivo**, permitindo que concessionárias, locadoras e gestores tomem decisões baseadas em dados, otimizando investimentos e estratégias comerciais.

3. PROBLEMA

3.1. O desafio do setor automotivo na precificação de veículos

Empresas do setor automotivo, como concessionárias, locadoras e grandes revendedores, enfrentam desafios constantes na definição de preços estratégicos para veículos. A precificação influencia diretamente a competitividade no mercado, impactando a margem de lucro, o giro do estoque e a capacidade de tomada de decisão.

3.2. Dores do Cliente

1. Falta de dados detalhados sobre o mercado em tempo hábil

- a. Muitas empresas baseiam suas decisões apenas em tabelas de referência fixas, como a Tabela FIPE, que fornece apenas um valor médio sem considerar fatores específicos do mercado.
- b. Não há visibilidade sobre tendências, variações regionais, sazonalidade e outros atributos que afetam a precificação real

2. Dificuldade em acompanhar a volatilidade do mercado

- a. O mercado automotivo é dinâmico, com preços variando conforme oferta, demanda, mudanças econômicas e características dos veículos.
- b. Empresas que não possuem ferramentas para acompanhar essas flutuações acabam adotando preços que podem ser defasados ou imprecisos, levando a perdas financeiras.

3. Falta de previsibilidade na precificação de veículos

- a. Empresas precisam estimar o valor futuro de um veículo, seja para planejar compras e vendas, definir investimentos ou ajustar estoques.

- b. Métodos tradicionais não oferecem previsões baseadas em dados históricos e tendências de mercado, dificultando o planejamento estratégico.

3.3. Solução oferecida pela DriveIntel

A DriveIntel resolve essas dores por meio da automação, inteligência artificial e transformação de dados em insights estratégicos, permitindo que as empresas tenham um panorama completo e atualizado do mercado.

Como fazemos isso?

1. Automação da coleta e análise de dados

- a. Utilizamos *web scraping* para coletar diariamente grandes volumes de dados de anúncios de veículos em diversas plataformas.
- b. Esses dados são processados automaticamente para calcular médias, identificar padrões de preço e gerar relatórios detalhados.

2. Previsão de preços com Inteligência Artificial

- a. O DriveIntel utiliza modelos de IA treinados com dados históricos para prever valores de veículos em diferentes cenários.
- b. A previsão considera marca, modelo, ano, quilometragem, oferta e demanda, gerando estimativas mais precisas para o mercado.

3. Geração de insights estratégicos

- a. A plataforma transforma dados brutos em insights acionáveis, permitindo que empresas tomem decisões informadas sobre preços, compras e estratégias de vendas.

- b. Relatórios detalhados e gráficos interativos fornecem uma visão completa e contextualizada do mercado automotivo.

4. A EMPRESA

A DriveIntel é uma empresa de tecnologia especializada em inteligência de mercado para o setor automotivo, oferecendo soluções inovadoras para análise de preços, monitoramento de tendências e previsões de mercado. Nosso propósito é transformar grandes volumes de dados em insights estratégicos, permitindo que concessionárias, lojistas e profissionais do setor tomem decisões mais precisas, ágeis e embasadas em dados concretos.

4.1. O que fazemos?

Nosso ecossistema coleta, estrutura e analisa dados de anúncios de veículos, proporcionando uma visão detalhada das variações de preços e do comportamento do mercado. Combinamos automação, ciência de dados e inteligência artificial para transformar informações brutas em conhecimento estratégico, permitindo que nossos clientes:

- Acompanhem tendências de preços e compreendam as dinâmicas do mercado.
- Prevejam oscilações futuras nos valores dos veículos com maior precisão.
- Tomem decisões estratégicas sobre compra, venda e gestão de estoque.
- Reduzam o tempo de análise, contando com relatórios automatizados e atualizados periodicamente.

4.2. Nossa abordagem

Sabemos que a precificação de veículos é um fator essencial para o setor automotivo e que ferramentas como a Tabela FIPE são amplamente utilizadas como referência. No entanto, o mercado é dinâmico, influenciado por fatores como localização, quilometragem, oferta e demanda, e sazonalidade. A DriveIntel vai além das referências tradicionais, fornecendo uma análise granular e contextualizada, permitindo uma precificação mais estratégica e alinhada à realidade do mercado.

4.3. Por que escolher a DriveIntel?

- Dados confiáveis e atualizados – Extraímos e analisamos informações de múltiplas fontes.
- Tecnologia avançada – Utilizamos inteligência artificial para gerar previsões precisas de preços.
- Automação e eficiência – Relatórios detalhados são gerados automaticamente, otimizando processos.
- Inteligência para decisões estratégicas – Nossa plataforma permite uma precificação mais precisa e competitiva.

Na DriveIntel, acreditamos que a informação é a chave para um mercado automotivo mais eficiente e transparente. Nosso compromisso é fornecer ferramentas inteligentes e inovadoras, garantindo que nossos clientes tenham mais controle, segurança e assertividade em suas operações.

5. DESENVOLVIMENTO

5.1 Microserviço de RPA (API)

5.1.2 Modelagem de Negócio

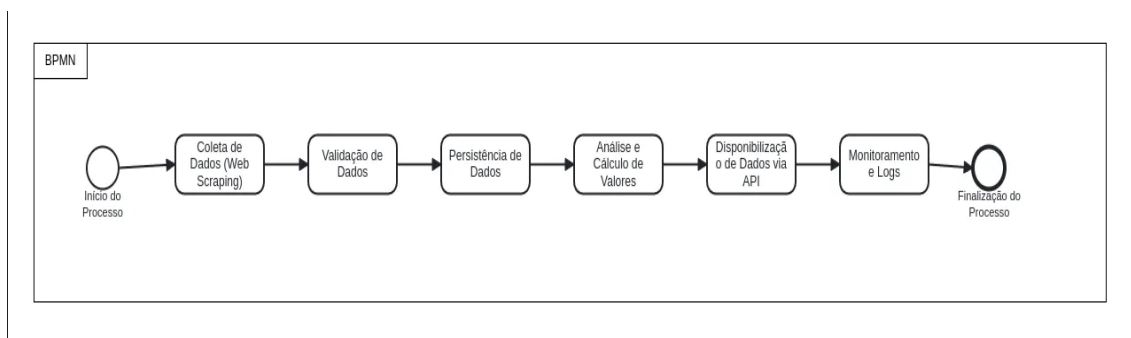


Figura 1 - Modelagem de Negócio Microserviço de RPA (API)

5.1.3 Engenharia de requisitos

5.1.3.1 Requisitos Funcionais

| Requisitos Funcionais | | | |
|-----------------------|--|---|---|
| Versão do projeto | | v2025.4.19-sprint-13 | |
| Código | Requisito | Descrição | Referência Implementação |
| RF001 | Executar Processo ETL | O sistema deve realizar o processo de web scraping nos sites pré-definidos, coletando as URLs dos anúncios de veículos disponíveis. Os dados extraídos devem passar por um processo de sanitização para remover inconsistências e garantir sua integridade. Em seguida, devem ser validados conforme as regras de negócio e armazenados no banco de dados para uso posterior. | apps/development/routes Rota /run/crawler/{machine}/{site} |
| RF002 | Cálculo média de Mercado | O sistema deve calcular a média de preço dos veículos com base nos anúncios coletados, considerando critérios de agrupamento definidos nas regras de negócio. | apps/development/routes Rota /assign/code_model/{machine} /calculate/averages/{machine} |
| RF003 | Gerar Relatórios | A aplicação deve gerar relatórios contendo insights e visualizações de dados baseados nos anúncios coletados em um determinado período. Os relatórios devem incluir métricas estatísticas relevantes para análise de preços e tendências do mercado de veículos. | apps/calcs/routes Rota /report/insights/task/{machine} /{year_reference}/{month_reference} |
| RF004 | Gerenciar Anúncios Coletados | O sistema deve permitir operações de CRUD nos dados coletados. | apps/records/routes Rota /update/{machine}, /duplicate/task/{machine}, /deactivate/task/{machine}, /list/task/{machine} |
| RF005 | Exportar Anúncios | O sistema deve permitir o download dos anúncios coletados em formato XLSX, proporcionando aos usuários a possibilidade de analisar os dados em ferramentas externas, como planilhas eletrônicas. | apps/core/routes - Rota /download/excel/{task_id}/{machine} |
| RF006 | Cálculo KPIs do mercado | O sistema deve fornecer indicadores-chave de desempenho (KPIs) do mercado automotivo com base nos dados coletados. | apps/core/routes - Rota /general_kpi/{machine} /{year_reference}/{month_reference} |
| RF007 | Cálculo Análises Geográficas do mercado | O sistema deve calcular a distribuição geográfica dos preços de veículos, apresentando a média de preços por estado e cidade. Além disso, deve medir a correlação entre preço e quantidade de anúncios em cada região, permitindo a visualização da concentração de ofertas e variações de preço no mercado. | apps/core/routes - Rota /geographic_pricing_analysis/ {machine}/{year_reference} /{month_reference} |
| RF008 | Gerar Insights | O sistema deve consumir os dados brutos coletados e transformar os mesmos em informações acionáveis que apoiem o usuário na tomada de decisão estratégica para precificação de veículos. | apps/calcs/routes Rota /analyze/insights/market /{machine} /{year_reference}/{month_reference} |
| RF009 | Identificar marcas relevantes | O sistema deve com base em cálculos estatísticos identificar marcas relevantes para o setor. | apps/calcs/db_manager Método describe_market_by_reference_date |
| RF010 | Identificar padrões de preços para as marcas | O sistema deve identificar padrões de preços para as marcas, permitindo o entendimento de média de preços, preços acima da média bem como preços abaixo da média para a marca. | apps/calcs/db_manager Método get_top_10_brands_prices |

Tabela 1 - Requisitos Funcionais - Microserviço de RPA

5.1.3.2 Requisitos Não Funcionais

| Requisitos não funcionais | | | |
|---------------------------|---------------------------------|---|---|
| Versão do projeto | | v2025.4.19-sprint-13 | |
| Código | Requisito | Descrição | Referência Implementação |
| RNF001 | Escalabilidade | O sistema deve ser escalável, permitindo o aumento da capacidade de processamento conforme a demanda de novos sites e maior volume de dados. | O docker compose está estabelecendo diversos serviços de forma que permitir o escalonamento da aplicação. |
| RNF002 | Modularidade | A arquitetura deve ser estruturada de forma que seja fácil adicionar novos sites ao processo de web scraping sem impactar os sites já existentes. | Estamos seguindo os princípios de clean architecture. |
| RNF003 | Tolerância a Falhas | O processo de automação (RPA) deve ser robusto e implementar tratativas de erro para lidar com falhas transitórias, como perda de conexão com a internet, timeout de resposta dos sites e páginas inacessíveis. | O código está envolvido em blocos try except para conferir tal requisito. |
| RNF004 | Processamento Assíncrono | O sistema deve permitir a execução assíncrona das tarefas para garantir melhor aproveitamento dos recursos e evitar bloqueios durante a coleta e processamento dos dados. | Bibliotecas como celery estão sendo utilizadas para tal |
| RNF005 | Baixo Acoplamento | Os módulos do sistema devem ser desenvolvidos de forma desacoplada, permitindo que funcionalidades possam ser alteradas ou substituídas sem impactar outras partes do sistema. | A arquitetura está seguindo o princípio de separação de responsabilidades. |
| RNF006 | Observabilidade e Monitoramento | O sistema deve possuir logs detalhados das execuções do robô e cálculos estatísticos, possibilitando auditoria e rastreabilidade dos processos. | Ferramentas como Logging Frameworks estão sendo utilizadas para captura e armazenamento de logs. |
| RNF007 | Performance | O sistema deve ser capaz de executar tarefas que envolvem operações de entrada e saída (I/O-bound), como chamadas a APIs e acesso a banco de dados, sem impedir a execução eficiente de tarefas que exigem alto uso de CPU (CPU-bound), como cálculos estatísticos e processamento de grandes volumes de dados. | O processamento assíncrono está sendo gerenciado pelo Celery com o pool padrão (prefork), garantindo boa eficiência para tarefas CPU-bound. |
| RNF007 | Documentação da API | O sistema deve possuir uma documentação automática e atualizada no Swagger, refletindo todas as rotas disponíveis e seus respectivos parâmetros. Além disso, o acesso à documentação deve ser protegido por um token de autenticação. | A geração automática da documentação está sendo realizada via Swagger/OpenAPI, e a proteção está sendo aplicada por meio de autenticação via token. |
| RNF008 | Testes Automatizados | O sistema deve possuir uma suíte de testes automatizados para validar as funcionalidades críticas, garantindo a confiabilidade do código antes da implantação. | Os testes estão sendo desenvolvidos com PyTest, cobrindo validação de dados, execução de tarefas assíncronas e APIs. |
| RNF009 | Segurança das Credenciais | O sistema deve armazenar credenciais de acesso a sites e serviços externos de forma segura, utilizando variáveis de ambiente ou um gerenciador de segredos, evitando exposição em código-fonte. | .env está sendo utilizado. |

Tabela 2 - Requisitos Não Funcionais - Microserviço de RPA

5.1.3.3 Regras de Negócio

| Regras de Negócio | | | | |
|-------------------|-------|--|--|---|
| Versão do projeto | | | v2025.4.19-sprint-13 | |
| Código | RF | Regra | Descrição | Referência Implementação |
| RN001 | RF001 | Descarte de Anúncios Inválidos | Anúncios sem informações essenciais (preço, modelo, ano modelo) são descartados. | apps/core/data_processing Classe BusinessRules método should_discard |
| RN002 | RF001 | Corte de Preço | Veículos com preço abaixo de R\$5.000 são descartados. | apps/core/data_processing Classe BusinessRules método price_cutoff_value |
| RN003 | RF001 | Inferir dados essenciais ausentes | Antes de descartar o anúncio que possui algum dado essencial faltante o sistema deve tentar inferir o valor do mesmo buscando o em alguma parte do anúncio, tal qual o título, descrição etc. | apps/core/data_processing Classe BusinessRules método infer_missing_year, infer_missing_model e infer_missing_price |
| RN004 | RF002 | Padronização de Atributos do Veículo | O sistema deve padronizar alguns atributos do veículo, garantindo que os valores estejam alinhados às nomenclaturas consideradas válidas pelo sistema. | apps/core/data_processing - Classe BusinessRules, métodos standardize_fuel, standardize_gear e standardize_bodywork |
| RN005 | RF003 | Agrupamento para Cálculo de Médias | As médias de preço devem ser calculadas agrupando os veículos pelo modelo e ano-modelo. | apps/core/db_manager - Classe DatabaseManager Método assing_code_model e apps/calcs/da_manager - Classe DatabaseManagerAverage Método get_machine_average |
| RN006 | RF004 | Período de Referência para Geração de Relatórios | Todos os insights, métricas e análises estatísticas do relatório devem ser calculados com base nos anúncios coletados em um determinado ano e mês de referência. Apenas os dados desse período devem ser considerados no processamento das informações do relatório. | apps/calcs/utils - Função generate_report_and_insights |
| RN007 | RF004 | Atualização de Anúncios | Os anúncios coletados podem ser editados apenas nos seguintes campos: preço, descrição e título. Além disso, as atualizações devem seguir as regras de negócio estabelecidas, garantindo que nenhum dado inválido ou inconsistente seja salvo no sistema. Caso os novos dados não atendam às validações, a atualização não será permitida. | apps/core/db_manager - Classe DatabaseManager Método save_to_database |

| | | | | |
|--------|-------|---|--|--|
| RN008 | RF004 | Duplicação de Anúncios | A criação de anúncios só pode ocorrer por meio da duplicação de um anúncio já existente. O anúncio duplicado deve passar pelas mesmas validações de regras de negócio aplicadas aos anúncios coletados, impedindo a inserção de dados que não estejam de acordo com os critérios do sistema. | apps/records/tasks - Função <code>duplicate_records_task</code> |
| RN009 | RF005 | Reprocessamento das Médias | Sempre que um anúncio for excluído ou duplicado, o sistema deve recalcular as médias de mercado para garantir a consistência das informações. No caso de atualização de um anúncio, não há necessidade de reprocessamento das médias. | apps/records/routes - <code>/duplicate/task/{machine}</code> , <code>/deactivate/task/{machine}</code> |
| RN0010 | RF006 | Filtragem de dados exportados via excel | Deve ser possível aplicar filtros de datas de referência ao solicitar o export dos anúncios | apps/core/routes - Rota <code>/download/excel/{task_id}/{machine}</code> |
| RN0011 | RF007 | Frequência dos dados analisados nas KPIs | Os KPIs devem incluir um ranking com os 10 modelos de veículos mais anunciados no período analisado. | apps/core/db_manager - Classe <code>DatabaseManager</code> Método <code>general_kpi</code> |
| RN0012 | RF008 | Frequência dos dados das análises geográficas | A análise geográfica deve considerar apenas os 10 estados com maior volume de anúncios ativos no período de referência. Para cada um desses estados, o sistema deve calcular o total de anúncios e a média de preços dos veículos listados. | apps/core/db_manager - Classe <code>DatabaseManager</code> Método <code>geographic_pricing_analysis</code> |
| RN0013 | RF009 | Conteúdo dos insights gerados | As informações acionáveis geradas pelo sistema devem ser baseadas em um determinado período de referência ano e mês, e devem trazer a relação das marcas que se destacaram positivamente e que se mostram como uma boa oportunidade de investimentos. | apps/calcs/db_manager Método <code>market_insights_by_reference_date</code> |
| RN0014 | RF010 | Critérios de marcas relevantes | As marcas relevantes são as 10 marcas que possuem maior oferta para o determinado período de referência (mês/ano) | apps/calcs/db_manager Método <code>describe_market_by_reference_date</code> |
| RN0015 | RF001 | Critérios de marcas selecionadas para identificação de padrões de preço | Os padrões de precificação (média de preço e outliers) devem ser identificados e calculados para as marcas relevantes no determinado período de referência (mês/ano) | apps/calcs/db_manager Método <code>get_top_10_brands_prices</code> |

Tabela 3 - Regras de Negócio - Microsserviço de RPA

5.1.3.4 Matriz Rastreabilidade

| Matriz Rastreabilidade | | | | | | | | | | | | | | | |
|--|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| Versão do projeto v2025.4.19-sprint-13 | | | | | | | | | | | | | | | |
| Requi sito Funci onal | | | | | | | | | | | | | | | |
| | RN 001 | RN 002 | RN 003 | RN 004 | RN 005 | RN 006 | RN 007 | RN 008 | RN 009 | RN 010 | RN 011 | RN 012 | RN 013 | RN 014 | RN 015 |
| RF0 01 | | | | | | | | | | | | | | | |
| RF0 02 | | | | | | | | | | | | | | | |
| RF0 03 | | | | | | | | | | | | | | | |
| RF0 04 | | | | | | | | | | | | | | | |
| RF0 05 | | | | | | | | | | | | | | | |
| RF0 06 | | | | | | | | | | | | | | | |
| RF0 07 | | | | | | | | | | | | | | | |
| RF0 08 | | | | | | | | | | | | | | | |
| RF0 09 | | | | | | | | | | | | | | | |
| RF0 10 | | | | | | | | | | | | | | | |

Tabela 4 - Matriz de Rastreabilidade - Microserviço de RPA

5.1.4 PROJETO DA SOLUÇÃO

5.1.4.1 Diagrama de Sequência

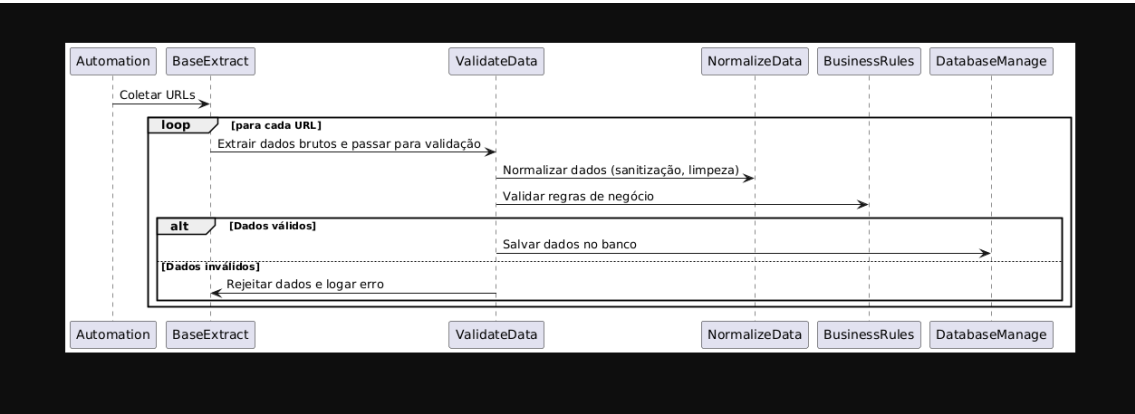


Figura 2 - Diagrama de Sequência Microserviço de RPA (API)

5.1.4.2 Diagrama de Casos de Uso Específico

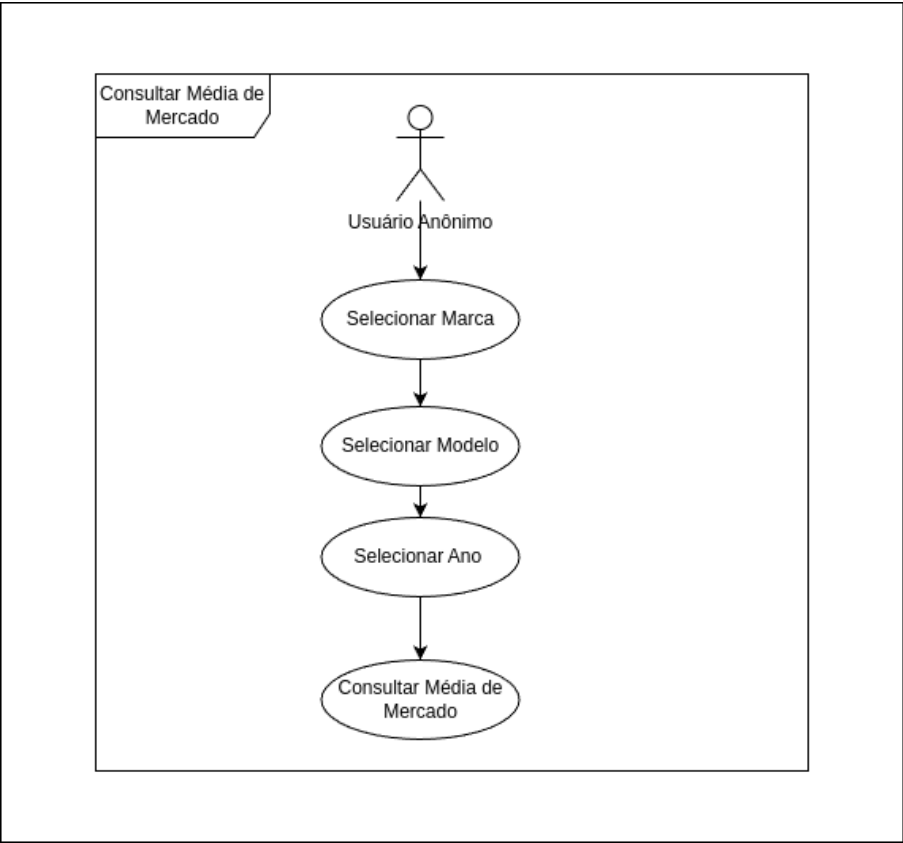


Figura 3 - Diagrama de Casos de Uso Específico Microserviço de RPA (API)

5.1.4.3 Especificação de Casos de Uso

| | |
|---|--|
| Nome do Caso de Uso | UC01 - Consultar Médias de Mercado |
| Caso de Uso Geral | Permite que o usuário consulte as médias de preços de veículos a partir da seleção de uma marca, modelo e ano. |
| Ator Principal | Usuário anônimo |
| Atores Secundários | |
| Resumo | |
| Pré-condições | 1. O sistema deve ter uma base de dados preenchida com marcas, modelos e anos. Portanto o processo ETL deve ter ocorrido previamente. Ou o backup do banco deve ter sido restaurado. |
| Pós-condições | |
| Cenário Principal | |
| Ações do Ator | Ações do Sistema |
| O Usuário acessa a interface da aplicação, na funcionalidade de consulta de médias. | |
| | O sistema apresenta uma lista de marcas disponíveis. |
| O Usuário seleciona uma marca. | |
| | O sistema apresenta uma lista de modelos associados à marca selecionada. |
| O Usuário seleciona um modelo. | |
| | O sistema apresenta uma lista de anos associados ao modelo selecionado. |
| O Usuário seleciona um ano. | |
| | O sistema retorna a média de preços para o veículo com os critérios escolhidos. |
| Restrições Validações | |
| Cenário Alternativo | |
| Ações do Ator | Ações do Sistema |
| | |
| | |
| | |

Tabela 5 - Especificação de Casos de Uso - Microserviço de RPA

5.1.4.4. Modelo Entidade-Relacionamento (ER)

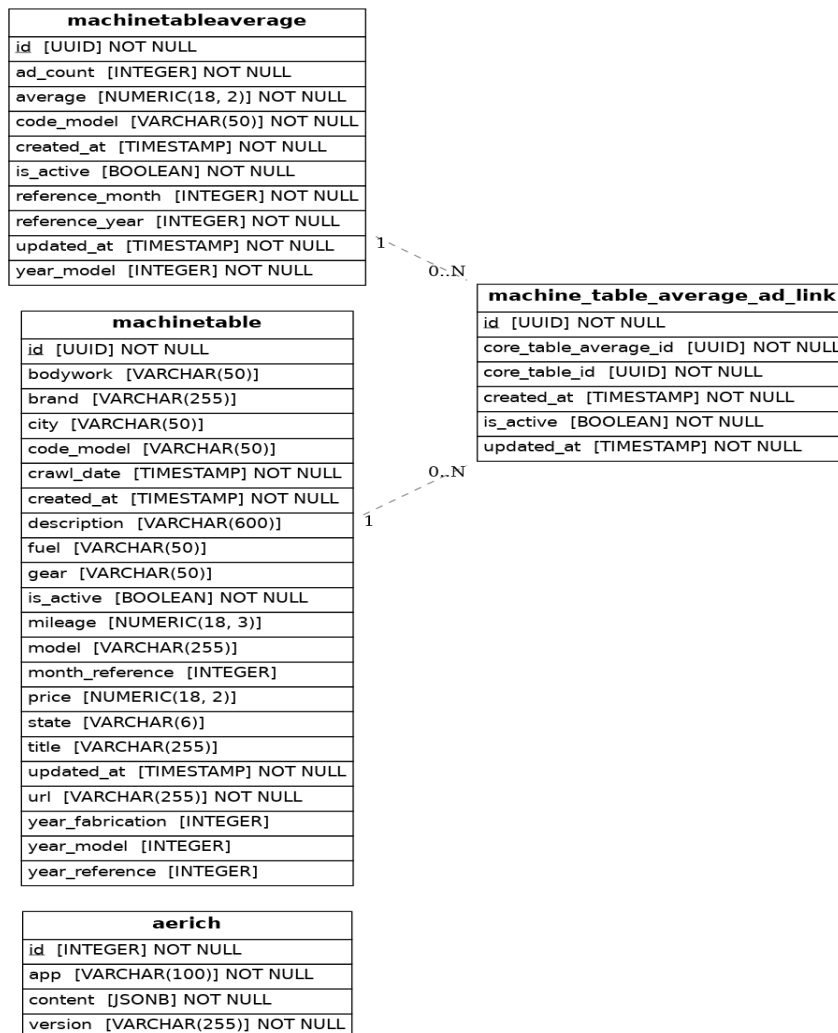


Figura 4 - Modelo de Entidade-Relacionamento Microserviço de RPA (API)

5.1.4.5. Diagrama de Classes



Figura 5 - Diagrama de Classes Microserviço de RPA (API)

5.2 Modelagem de Negócio – Microserviço de IA (API)

5.2.1 Diagrama de Atividades do Negócio – BPMN

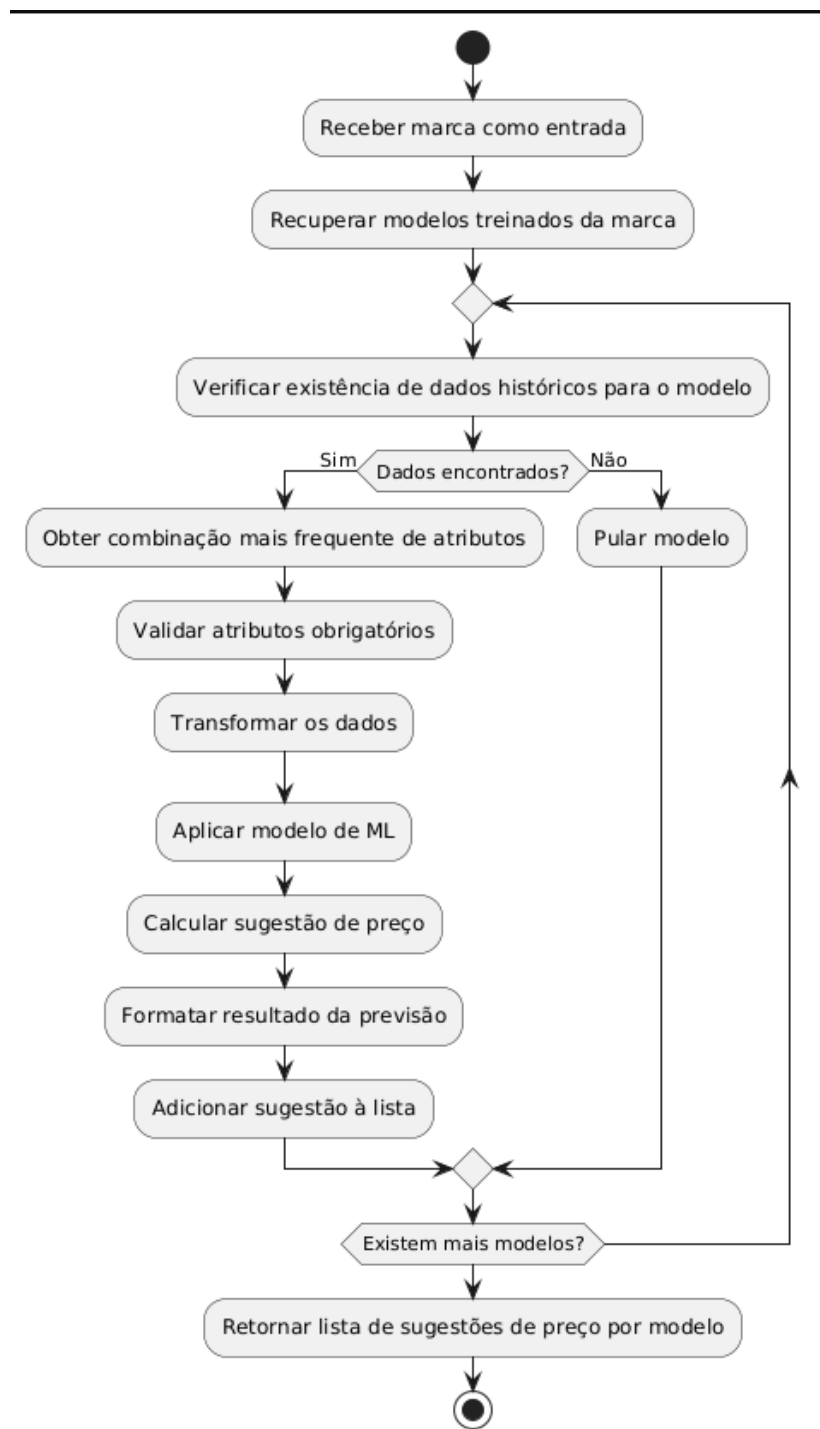


Figura 6 - Diagrama de Atividades do Negócio Microserviço de IA (API)

5.2.2 Engenharia de Requisitos

5.2.2.1. Requisitos Funcionais (RF)

| Regras de Negócio | | | | |
|-------------------|--------------|---|--|---|
| Versão do projeto | | | v2025.3.9-sprint-3 | |
| Código | RF | Regra | Descrição | Referência Implementação |
| RN001 | RF001, RF002 | Marca válida | A previsão de preço só pode ser realizada para marcas que estavam presentes no conjunto de treinamento do modelo. Caso contrário, o sistema deve rejeitar a entrada. | settings - Classe Config Método load_valid_brands |
| RN002 | RF001, RF002 | Formatar saída no padrão monetário brasileiro | A previsão do preço do carro deve ser retornada no formato "XX.XXX,XX", garantindo legibilidade para o usuário. | apps/car/routes - Rota /predict |
| RN003 | RF001, RF002 | Cálculo dinâmico de médias de preço | O sistema deve calcular preços médios dinâmicos para marca, estado e cidade, usando os dados do conjunto original de treinamento. | apps/car/data_processing - Função transform_data |
| RN004 | RF001, RF002 | Cálculo do desvio de preço | O desvio de preço deve ser calculado com base na média de preços do modelo e ano do veículo. | apps/car/data_processing - Função transform_data, calculate_price_deviation |
| RN005 | RF001, RF002 | Identificação de marcas de luxo | Marcas premium (AUDI, BMW, MERCEDES, PORSCHE) devem ser identificadas como uma feature extra no modelo. | apps/car/data_processing - Função transform_data |
| RN006 | RF002 | Previsão Padronizada por Marca | A previsão de preços para todos os modelos de uma marca deve utilizar os mesmos parâmetros de entrada, incluindo quilometragem, tipo de combustível, câmbio, carroceria, cidade e estado. Além disso, a predição sempre deve considerar o próximo ano vigente como ano-modelo. | apps/car/routes - Rota /brand_predict/{brand} |

Tabela 6 - Requisitos Funcionais – Microserviço de IA

5.2.2.2. Requisitos Não Funcionais (RNF)

| Versão do projeto | | | |
|-------------------|--|--|---|
| Código | Requisito | Descrição | Referência Implementação |
| RNF001 | Validação via Pydantic | A validação da marca ocorre no nível do Pydantic, impedindo que requisições inválidas sejam processadas pelo sistema. | apps/car/schema Classe Car |
| RNF002 | Formato de entrada | Os dados de entrada devem ser recebidos em formato JSON seguindo um esquema padronizado. | apps/car/schema Classe Car |
| RNF003 | Carregamento único de artefatos | Os artefatos de IA (modelo, normalizador, transformer, datasets auxiliares) são carregados uma única vez durante a inicialização do servidor e armazenados em memória. | main Função startup_event |
| RNF004 | Gerenciamento de estado do aplicativo | O estado do aplicativo (app.state) é utilizado para armazenar os artefatos de IA, permitindo acesso rápido sem necessidade de reprocessamento. | main Função startup_event |
| RNF005 | Persistência dos artefatos | Os arquivos .pkl e .csv necessários para a IA ficam armazenados em diretórios específicos (artifacts/ e data/), garantindo que o sistema possa reiniciar sem perder informações importantes. | artifacts e data |
| RNF006 | Ambiente de desenvolvimento interativo | O sistema deve permitir experimentação e validação de modelos de IA em notebooks Jupyter. | notebooks |
| RNF007 | Reprodutibilidade dos experimentos | Os notebooks devem conter código e documentação suficientes para permitir a reprodução dos experimentos realizados na fase de desenvolvimento. | notebooks |
| RNF007 | Documentação do Processo de Modelagem | Os notebooks devem conter uma descrição clara de todas as etapas do desenvolvimento do modelo, desde a preparação dos dados até a obtenção do modelo final. | notebooks/car_prices_brazil.ipnyb |
| RNF008 | Validação do Modelo com Métricas Matemáticas | O modelo de previsão de preços deve ser validado utilizando métricas quantitativas, incluindo R ² , MSE, RMSE e MAE. O valor de R ² deve ser superior a 0,75, garantindo que o modelo explique a maior parte da variação dos dados e que os erros estejam dentro de uma faixa aceitável. | notebooks/car_prices_brazil.ipnyb - Seção Avaliação do modelo |
| RNF009 | Containerização do Sistema | O sistema deve ser executável em um ambiente Docker, garantindo portabilidade, facilidade de implantação e consistência entre diferentes ambientes. | docker-compose.yml |
| RNF010 | Disponibilização de Documentação Interativa | A API deve disponibilizar uma interface interativa via Swagger UI, permitindo a visualização estruturada dos endpoints, envio de requisições de teste e consulta de respostas em tempo real. Essa | apps/docs |

| | | | |
|--|--|---|--|
| | | documentação deve ser gerada automaticamente a partir das definições dos endpoints, garantindo sempre a atualização conforme mudanças na API. | |
|--|--|---|--|

Tabela 7 - Requisitos Não Funcionais – Microserviço de IA

5.2.2.3. Regras de Negócio

| Regras de Negócio | | | | |
|-------------------|-------|---|--|---|
| Versão do projeto | | | v2025.3.9-sprint-3 | |
| Código | RF | Regra | Descrição | Referência Implementação |
| RN001 | RF001 | Marca válida | A sugestão de preço só pode ser realizada para marcas que estavam presentes no conjunto de treinamento do modelo. Caso contrário, o sistema deve rejeitar a entrada. | settings - Classe Config Método load_valid_brands |
| RN002 | RF001 | Formatar saída no padrão monetário brasileiro | A sugestão do preço do carro deve ser retornada nos formatos "XX.XXX,XX", "XX.XXXX,XX" garantindo legibilidade para o usuário. | apps/car/routes - Rota /predict |
| RN003 | RF001 | Cálculo dinâmico de médias de preço | O sistema deve calcular preços médios dinâmicos para marca, estado e cidade, usando os dados do conjunto original e de testes no momento em que o modelo foi treinado. | apps/car/data_processing - Função transform_data |
| RN004 | RF001 | Cálculo do desvio de preço | O desvio de preço deve ser calculado com base na média de preços do modelo e ano do veículo. | apps/car/data_processing - Função transform_data, calculate_price_deviation |
| RN005 | RF001 | Identificação de marcas de luxo | Marcas premium (AUDI, BMW, MERCEDES, PORSCHE) devem ser identificadas como uma feature extra no modelo. | apps/car/data_processing - Função transform_data |
| RN006 | RF001 | Sugestão por marca | A geração de sugestões de preço deve ser realizada com base em todos os modelos de veículos pertencentes à marca selecionada, considerando apenas aqueles que estiverem presentes nos dados utilizados para o treinamento do modelo de IA. | apps/car/routes - Rota /brand_predict/{brand} |
| RN007 | RF001 | Uso de variáveis independentes | A sugestão de preço deve ser calculada com base em variáveis independentes características do veículo, tais como: ano modelo, quilometragem, tipo de câmbio, tipo de combustível, carroceria, cidade e estado. Esses atributos são | apps/car/routes - Rota /brand_predict/{brand} |

| | | | | |
|-------|-------|--|--|---|
| | | | processados e utilizados como entrada no modelo de previsão, cujo alvo é o valor estimado para o anúncio. | |
| RN008 | RF001 | Uso da combinação mais frequente para sugestão | Para cada modelo da marca selecionada, a sugestão de preço é baseada na combinação mais frequente das variáveis independentes encontrada nos dados históricos de anúncios. Essa abordagem prioriza a configuração mais popular e provável do veículo, garantindo que a sugestão esteja alinhada com o comportamento real do mercado. | apps/car/routes - Rota /brand_predict/{brand} |

Tabela 8 - Regras de Negócio – Microserviço de IA

5.2.2.4. Matriz de Rastreabilidade

| Matriz Rastreabilidade | | | | | | | | |
|---------------------------------------|------------------|-------|-------|-------|-------|-------|-------|-------|
| Versão do projeto v2025.4.16-sprint-5 | | | | | | | | |
| Requisito Funcional | Regra de negócio | | | | | | | |
| | RN001 | RN002 | RN003 | RN004 | RN005 | RN006 | RN007 | RN008 |
| RF001 | | | | | | | | |

Tabela 9 - Matriz de Rastreabilidade – Microserviço de IA

5.2.2.6. Diagrama de Sequência

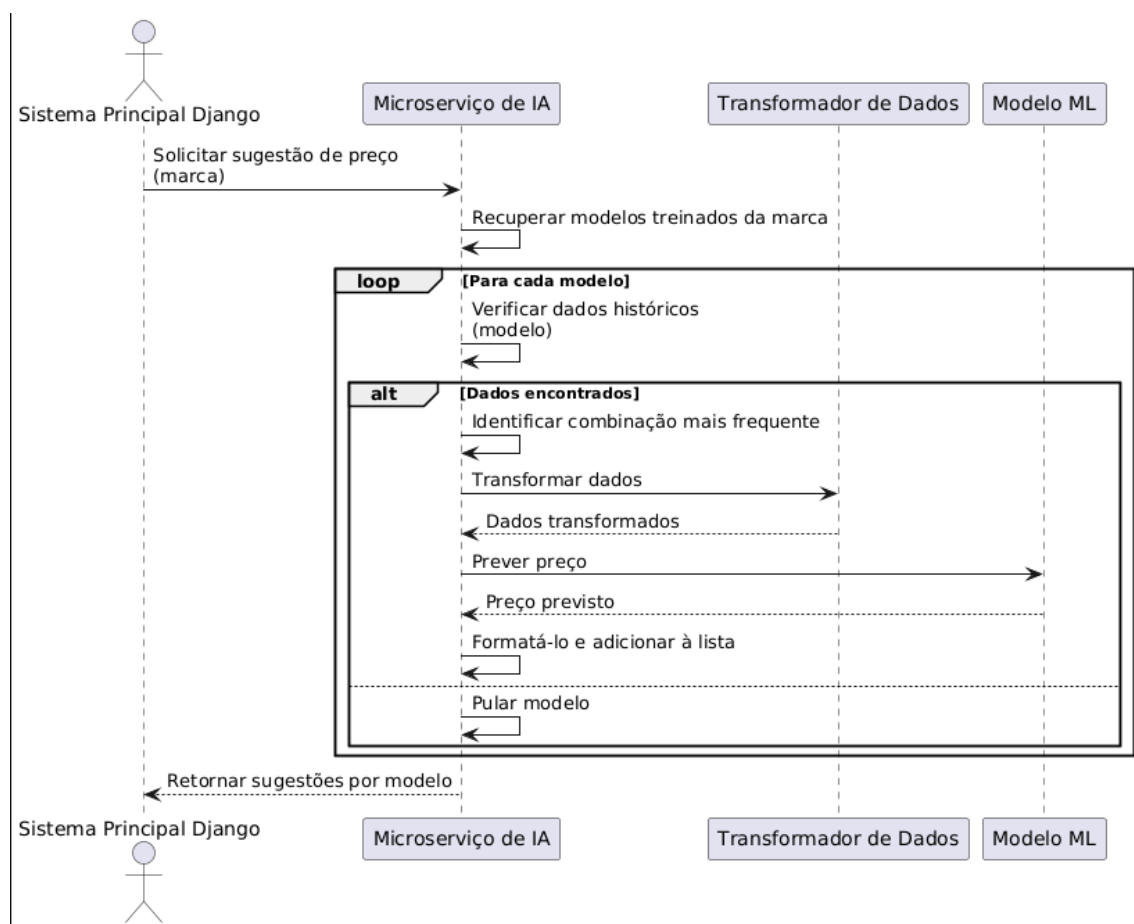


Figura 7 - Diagrama de Sequência - Microserviço de IA (API)

5.2.2.10. Diagrama de Classes

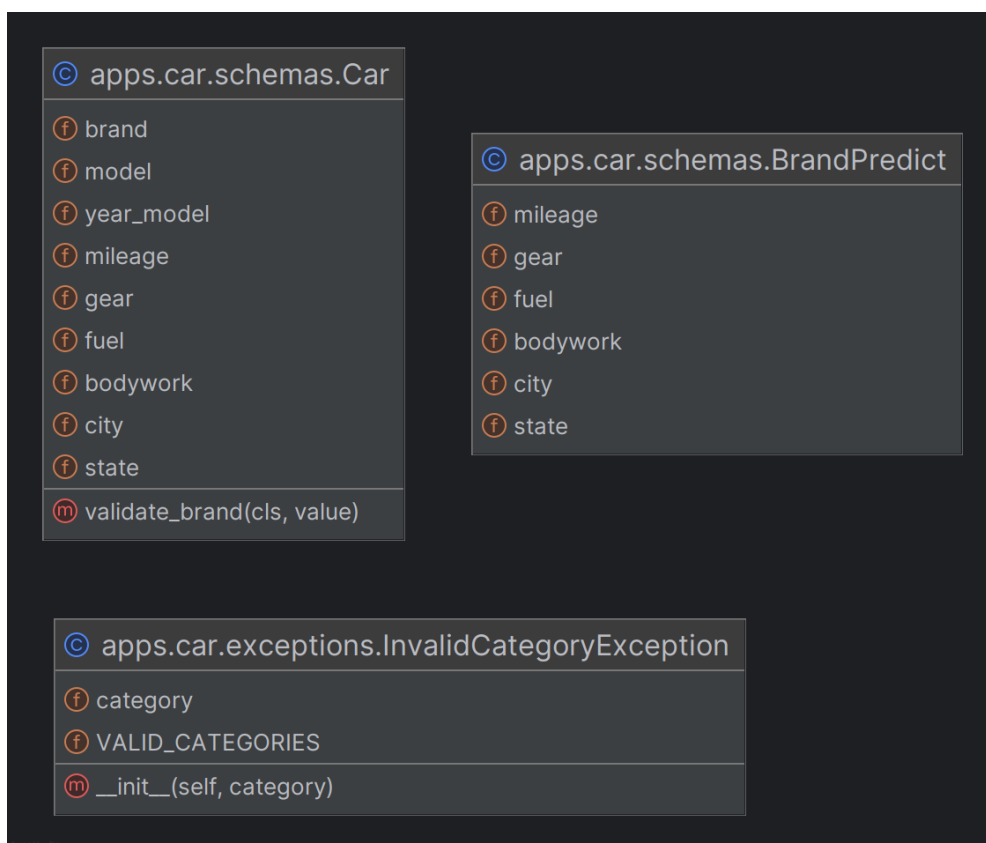


Figura 8 - Diagrama de Classes Microserviço de IA (API)

5.2.2.11. Diagrama de Casos de Uso

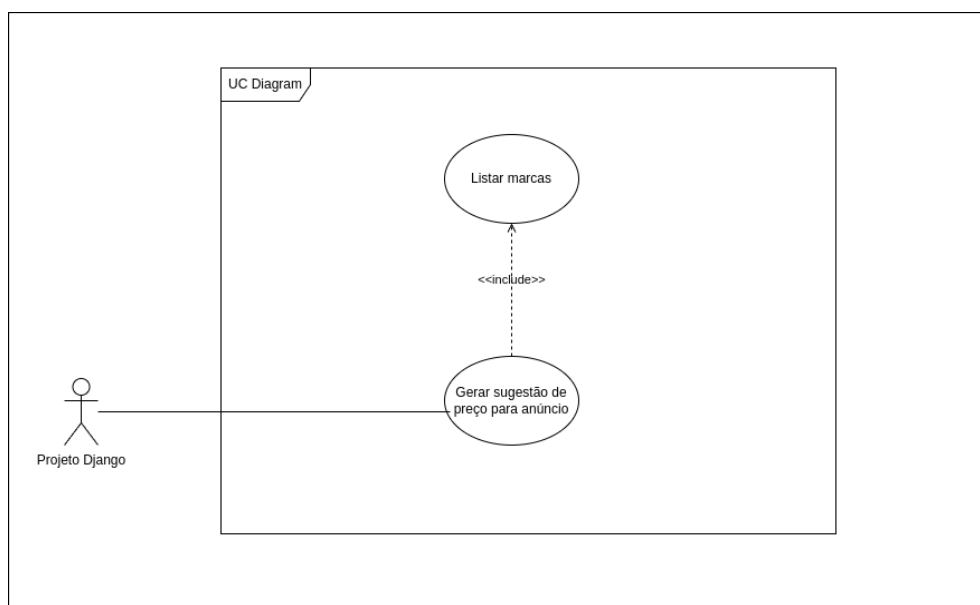


Figura 9 - Diagrama de Casos de Uso Microserviço de IA (API)

| Nome do Caso de Uso | UC01 - Gerar sugestão de preço de anúncio |
|--|--|
| | 36 |
| Caso de Uso Geral | Retorna sugestões de preço com IA para todos os modelos de determinada marca, vistos no treinamento do modelo preditivo de Inteligência artificial. |
| Ator Principal | Sistema django, projeto principal. |
| Atores Secundários | |
| Resumo | |
| Pré-condições | 1. Um usuário autenticado e com as devidas permissões de acesso deve estar logado no sistema django; 2. O usuário cumpridor da pré condição 1 deve ter acessado o caso de uso Listar marcas previamente. |
| Pós-condições | |
| Cenário Principal | |
| Ações do Ator | Ações do Sistema |
| O sistema django faz uma requisição a api de IA para sugerir preços de anúncios. | |
| | O sistema recuperar todos os modelos da marca vistos no treinamento. |
| | O sistema efetua o cálculo de previsão e retorna a resposta para o projeto principal. |
| Restrições Validações | |
| | |
| Cenário Alternativo | |
| Ações do Ator | Ações do Sistema |
| | |
| | |
| | |

5.2.3.1. Variáveis

Hyundai HB20
1.0 COMFORT PLUS TURBO Á/G AUTOMÁTICO

Feature = Variável independente

R\$ 89.290,00 Ver parcelas

Nome: Target = Variável Alvo, dependente
E-mail:
Telefone:

Chamar no Whatsapp

Enviar mensagem

| | | | |
|---------------------|---------------------------|--------------------|----------------------|
| ANO 2023/2024 | QUILOMETRAGEM 48.443km | COMBUSTÍVEL Á/G | CÂMBIO Automático |
| CARROCERIA Hatch | CÓDIGO 5230274 | COR Prata | CIDADE Betim - MG |

Features = Variáveis independentes

OPCIONAIS
Alarme, Ar Condicionado, Freio ABS, Limp e/ou Desemb, Trava Elétrica, Direção Elétrica, Controle eletrônico de estabilidade

Figura 10 - Variáveis Microserviço de IA (API)

5.2.3.1.2 Target (Variável Alvo)

A variável alvo representa o valor que se deseja prever ou explicar dentro de um modelo de aprendizado de máquina. Em problemas de classificação, o target corresponde a categorias específicas para as quais os dados de entrada são mapeados, enquanto em problemas de regressão, trata-se de um valor contínuo. No contexto de modelagem preditiva, os valores da variável alvo no conjunto de treinamento servem como referência para que o modelo aprenda os padrões existentes nos dados e faça previsões futuras (MEDIUM, 2017).

5.2.3.1.3 Feature (Recurso)

As features são as variáveis independentes que servem como entrada para o modelo preditivo. Elas fornecem as informações necessárias para que a máquina possa inferir padrões e realizar previsões. Cada feature representa uma característica específica do conjunto de dados e pode ser transformada ou combinada por meio de técnicas de engenharia de recursos para melhorar o desempenho do modelo. O número total de features em um conjunto de dados é frequentemente denominado dimensionalidade, sendo um fator determinante na complexidade do modelo de aprendizado (MEDIUM, 2017).

No caso específico do projeto, o objetivo é prever o preço, e as demais features (marca, modelo, ano etc), são variáveis existentes que podemos usar para essa previsão.

5.2.3.1.3 Entendendo o tipo das features

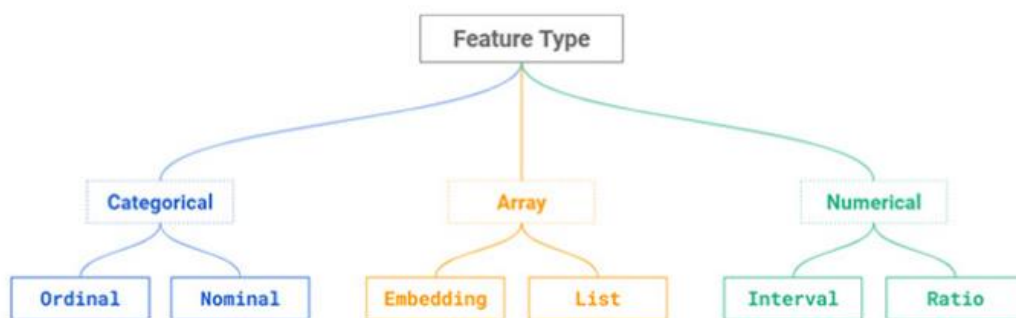


Figura 11 - Features Microserviço de IA (API)

As features podem ser classificadas em **categóricas** e **numéricas**, sendo fundamentais para a construção de modelos de aprendizado de máquina.

5.2.3.1.4 Features Categóricas

As features categóricas representam variáveis qualitativas, ou seja, dados discretos que não possuem um significado numérico direto. Elas podem ser divididas em dois tipos principais:

- **Nominais:** Não possuem uma ordem natural. Exemplos incluem a cor de um veículo ou a marca de um carro. Para utilizá-las em modelos de aprendizado de máquina, pode-se aplicar técnicas como **One-Hot Encoding (OHE)**, que transforma cada categoria em uma nova coluna binária, indicando a presença ou ausência daquela categoria.
- **Ordinais:** Possuem uma ordem natural, como classificações de satisfação (ruim, médio, bom) ou faixas de quilometragem de um veículo (0-10 mil km, 10-50 mil km, etc.). Nesse caso, a técnica **Label Encoding (LE)** pode ser aplicada, atribuindo valores numéricos ordenados a cada categoria.

A transformação correta das variáveis categóricas é essencial para evitar que modelos interpretem relações inexistentes entre categorias nominais ou desconsiderem a ordem presente nas variáveis ordinais.

5.2.3.1.5 Features Numéricas

As features numéricas representam dados quantitativos e permitem operações matemáticas diretas, como soma e média. Elas podem ser subdivididas em dois tipos:

- **Variáveis de Intervalo:** Não possuem um zero absoluto. Um exemplo seria a temperatura em Celsius, onde zero não representa ausência de temperatura.
- **Variáveis de Razão:** Possuem um zero absoluto, como a idade de um carro ou a quilometragem rodada.

Para otimizar o desempenho de modelos de aprendizado de máquina, as features numéricas podem ser transformadas utilizando técnicas de

escalonamento. Um exemplo comum é a normalização ou padronização por meio de escalonadores como **MinMaxScaler** ou **StandardScaler**, que ajustam os valores para uma faixa específica ou distribuem os dados de acordo com a média e o desvio padrão (HOPSWORK, 2022).

5.2.3.2 MÉTRICAS DE AVALIAÇÃO

A avaliação de modelos de regressão é um passo fundamental para garantir sua precisão e confiabilidade. Para isso, diversas métricas podem ser utilizadas, cada uma com características específicas que auxiliam na interpretação dos resultados.

R-quadrado (R^2) O R^2 mede a proporção da variância da variável dependente que é explicada pelas variáveis independentes do modelo. Quanto mais próximo de 1, maior a capacidade explicativa do modelo sobre os dados. Essa métrica é útil para avaliar a qualidade do ajuste do modelo e sua capacidade de representar a variabilidade dos dados analisados (Medium, 2023).

R-quadrado Ajustado O R^2 ajustado é uma versão do R^2 que penaliza a inclusão excessiva de variáveis preditoras. Ele considera o número de variáveis utilizadas no modelo, evitando o overfitting e fornecendo um indicativo mais preciso da qualidade do ajuste. Essa métrica é especialmente útil ao comparar modelos com diferentes quantidades de variáveis explicativas (Medium, 2023).

Erro Quadrático Médio (MSE) O MSE calcula a média dos erros quadráticos entre os valores reais e os valores previstos pelo modelo. Quanto menor o MSE, melhor o desempenho do modelo. No entanto, por ser uma métrica que eleva os erros ao quadrado, ela é sensível a outliers, o que pode distorcer a avaliação do modelo (Medium, 2023).

Raiz do Erro Quadrático Médio (RMSE) O RMSE é obtido a partir da raiz quadrada do MSE, tornando a interpretação da métrica mais intuitiva, pois os erros passam a ser expressos na mesma unidade da variável dependente. Assim como o MSE, é sensível a valores discrepantes, sendo uma boa escolha para

avaliar modelos quando há necessidade de interpretar os erros na mesma escala da variável alvo (Medium, 2023).

Erro Absoluto Médio (MAE) O MAE mede a diferença absoluta média entre os valores previstos e os valores reais. Diferente do MSE e do RMSE, essa métrica não penaliza tanto os outliers, sendo mais robusta em relação a erros grandes. Quanto menor o MAE, mais precisas são as previsões do modelo. Essa métrica é indicada quando há interesse em avaliar a magnitude dos erros de previsão sem amplificar a influência de outliers (Medium, 2023).

A escolha da melhor métrica para avaliar um modelo de regressão depende do contexto da análise e dos objetivos do estudo. O R^2 e o R^2 ajustado são ideais para medir a qualidade do ajuste do modelo, enquanto o MSE e o RMSE são apropriados para casos em que a penalização de grandes erros é desejada. O MAE, por sua vez, é uma excelente alternativa quando a robustez a outliers é um fator crítico na avaliação do modelo.

5.3 MODELAGEM DE NEGÓCIO – PROJETO DJANGO

5.3.2.1 Diagrama de Atividades do Negócio -BPMN

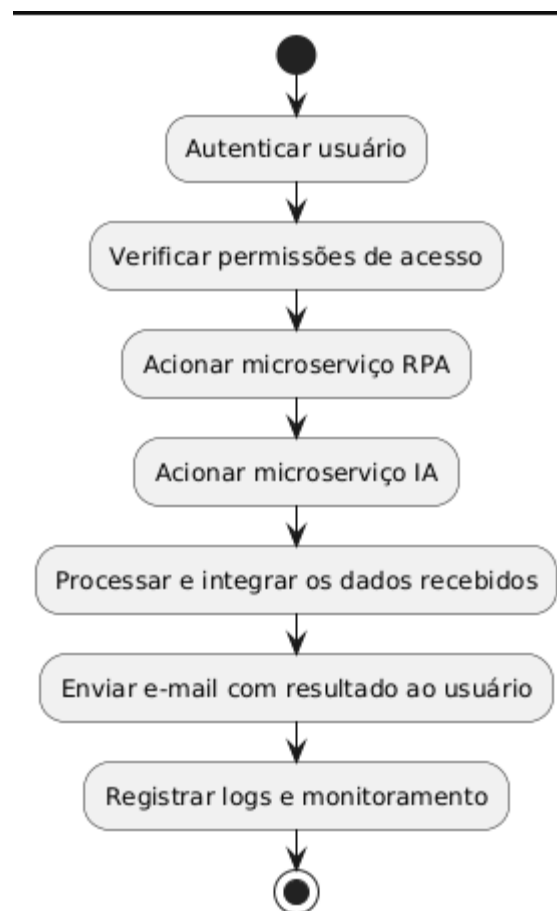


Figura 12 - BPMN Projeto Django

5.3.2.2 Requisitos Funcionais (RF)

| Requisitos Funcionais | | | |
|-----------------------|--|---|--|
| Versão do projeto | v2025.3.9-sprint-6 | | |
| Código | Requisito | Descrição | Referência Implementação |
| RF001 | Autenticação de Usuário | O sistema deve permitir que os usuários realizem o processo de autenticação para acessar funcionalidades restritas. Após a autenticação bem-sucedida, o usuário será autorizado a acessar as funcionalidades que lhe são permitidas de acordo com seu nível de permissão. | apps/user/views - Classe MyTokenObtainPairView |
| RF002 | Envio de E-mails Automáticos após execução do robô | O sistema deve enviar e-mails automáticos para os usuários após a conclusão do processo ETL no microserviço de RPA | apps/crawler/tasks - Tarefa run_all_process_from_crawler_api |

Tabela 11 - Requisitos Funcionais – Projeto Django

5.3.2.3 Requisitos Não Funcionais (RNF)

| Requisitos Não Funcionais | | | |
|---------------------------|--|--|--|
| Versão do projeto | | v2025.3.9-sprint-6 | |
| Código | Requisito | Descrição | Referência Implementação |
| RNF001 | Autenticação | O sistema deve utilizar JWT (JSON Web Token) para autenticação. O usuário deve realizar login para obter um token de acesso, que será necessário para consumir as APIs do sistema. | apps/user/views - Classe MyTokenObtainPairView |
| RNF002 | Validação e Expiração de Tokens | O sistema deve validar tokens JWT em todas as requisições autenticadas e garantir que tokens expirados não sejam aceitos. | A validação está sendo feita em cada classe do arquivo views de cada diretório, que possui a permissão IsAuthenticated |
| RNF003 | Controle de Acesso e Permissões | O sistema deve armazenar e gerenciar permissões específicas para cada usuário, determinando quais operações podem ser realizadas. O controle de acesso deve ser implementado utilizando Permission Classes, garantindo que apenas usuários autorizados possam acessar determinadas funcionalidades. | apps/core/permissions |
| RNF004 | Comunicação entre Microserviços | A comunicação entre o projeto Django e os microserviços de RPA e IA deve ser realizada exclusivamente por meio do Django, impedindo acessos diretos externos aos microserviços. O Django deve atuar como um intermediário central, garantindo que todas as requisições sejam autenticadas e autorizadas antes de serem encaminhadas aos microserviços. Essa abordagem assegura controle, segurança e padronização na troca de informações dentro do ecossistema. | Foi implementada uma integração de backends, o frontend apenas consome a aplicação django com o backend integrado. |
| RNF005 | Segurança e Proteção de Dados | As credenciais de acesso às APIs dos microserviços e outras informações sensíveis devem ser armazenadas de forma segura, sem exposição direta no código-fonte. | Bibliotecas como decouple estão sendo usadas para evitar a exposição de informações sensíveis. |
| RNF006 | Containerização e Padronização do Ambiente | O sistema deve ser executado em containers Docker para garantir a padronização do ambiente de desenvolvimento, teste e produção. O projeto deve possuir arquivos de configuração adequados, permitindo a escalabilidade e a fácil replicação do ambiente. O uso de Docker Compose deve ser adotado para a orquestração dos serviços internos, facilitando a gestão das dependências. | docker-compose.yml |
| RNF007 | Testes Automatizados | O sistema deve possuir testes automatizados unitários, garantindo a validação do funcionamento das principais funcionalidades. A cobertura dos testes deve ser mensurada, permitindo identificar áreas do código não testadas e aprimorar a confiabilidade do sistema. | Testes unitários implementados com PyTest. Cobertura dos testes mensurada com Coverage. |
| RNF008 | Padrões de Estilo de Código | O sistema deve seguir os padrões de estilo de código definidos pela PEP8, garantindo a legibilidade e a consistência do código-fonte. Ferramentas como o | A biblioteca autopep8 está instalada e seu uso é descrito por meio de |

| | | | |
|--------|------------------------------------|--|--|
| | | autopep8 devem ser utilizadas para aplicar as correções automaticamente, mantendo o código alinhado aos padrões recomendados. | documentação no README do projeto. |
| RNF009 | Manutenção da Documentação Swagger | A documentação da API, gerada pelo Swagger, deve ser mantida sempre atualizada e funcional. O sistema deve garantir que quaisquer alterações na API sejam refletidas automaticamente na documentação, assegurando que ela esteja sempre sincronizada com os endpoints existentes e suas funcionalidades. A documentação deve ser acessível de forma clara e objetiva, permitindo que desenvolvedores e mantenedores do sistema possam consultar as rotas e entender como interagir com a API de forma eficiente. | config/settings - Lista THIRD_APPS, definição da lib drf_yasg |

Tabela 12 - Requisitos Não Funcionais – Projeto Django

5.3.2.4 Regras de Negócio

| Regras de Negócio | | | | |
|-------------------|-------|--|---|--|
| Versão do projeto | | v2025.3.9-sprint-6 | | |
| Código | RF | Regra | Descrição | Referência Implementação |
| RN001 | RF001 | Dados para autenticação | A autenticação do usuário deve ser realizada exclusivamente por meio de username e senha. | apps/user/views - Classe MyTokenObtainPairView |
| RN002 | RF002 | Filtrar usuários a receber o email após execução do robô | O sistema deve enviar e-mails automáticos apenas para usuários corporativos que possuam a permissão CanFinalReport e que estejam ativos. O e-mail será enviado após a conclusão do processo ETL no microserviço de RPA e conterá o relatório final gerado durante o processo. Somente usuários com a permissão apropriada e status ativo poderão receber esses e-mails, garantindo que o acesso aos relatórios seja restrito de acordo com as permissões estabelecidas. | apps/send_email/utils - Função send_report_email |

Tabela 13 - Regras de Negócio – Projeto Django

5.3.2.5 Matriz de Rastreabilidade

| Matriz Rastreabilidade | | |
|------------------------|--|--------------------|
| Versão do projeto | | v2025.3.9-sprint-6 |
| Requisito Funcional | | Regra de negócio |
| | | RN001 RN002 |
| RF001 | | |
| RF002 | | |

Tabela 14 - Matriz de Rastreabilidade – Projeto Django

5.3.2.6 Diagrama de Sequência

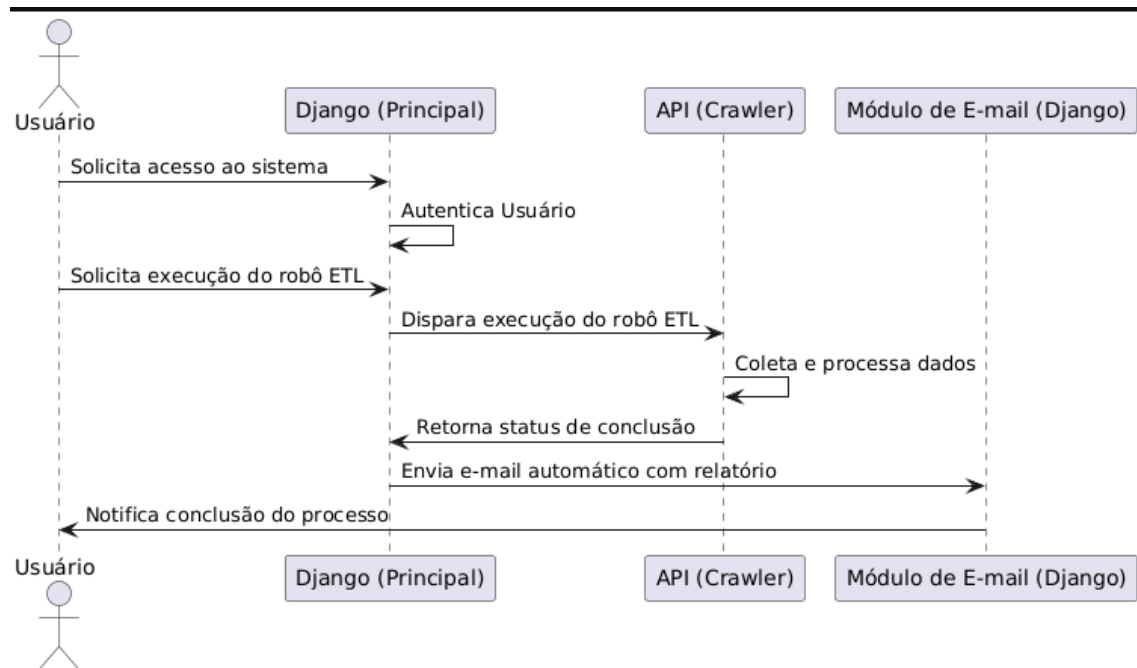


Figura 13 - Diagrama de Sequência Projeto Django

5.3.2.8 Modelo Entidade-Relacionamento (ER)

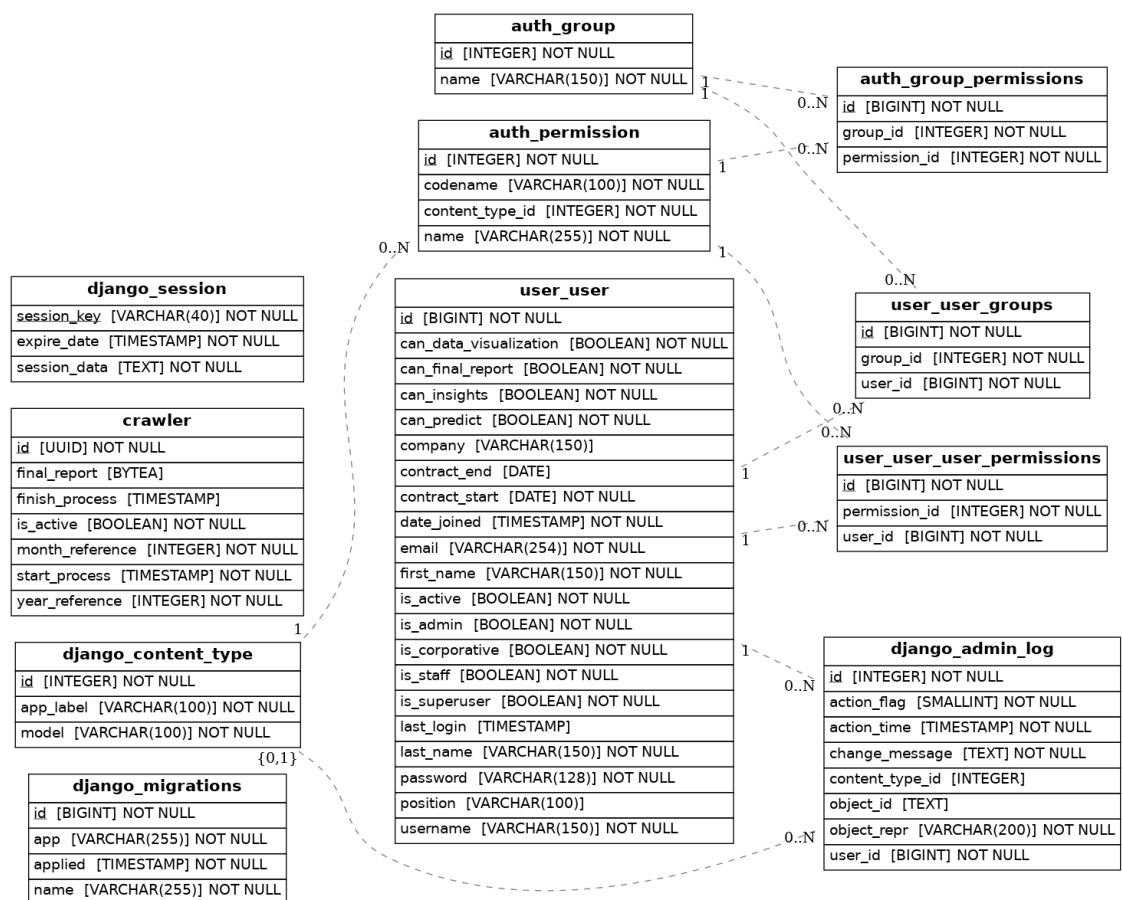


Figura 15 - Modelo Entidade-Relacionamento (ER) Projeto Django

6. CONCLUSÃO

A precificação estratégica de veículos é um desafio fundamental para empresas do setor automotivo, impactando diretamente a competitividade, a margem de lucro e a eficiência operacional de concessionárias, locadoras e grandes revendedores. Diante da limitação das tabelas de referência tradicionais, como a Tabela FIPE, que fornecem apenas um valor médio sem considerar variações de mercado, torna-se essencial a adoção de soluções baseadas em análise de dados e inteligência artificial.

Neste contexto, o DriveIntel se consolida como uma ferramenta inovadora de inteligência de mercado automotivo, atendendo à necessidade de empresas que buscam maior precisão na definição de preços e na identificação de tendências. A plataforma cumpre seu objetivo geral ao desenvolver um sistema capaz de coletar, analisar e interpretar grandes volumes de dados de precificação de veículos por meio de automação, estatística e inteligência artificial.

Os objetivos específicos também foram atingidos:

1. A coleta e organização de dados foram viabilizadas através de web scraping, permitindo estruturar informações provenientes de anúncios de veículos para análise de preços e tendências.
2. A aplicação de modelos estatísticos e inteligência artificial possibilitou a identificação de padrões de precificação e a simulação de diferentes cenários de mercado, considerando variáveis como modelo, ano e características dos veículos.
3. A geração de insights estratégicos permitiu que empresas do setor automotivo tomassem decisões embasadas em dados, otimizando a gestão de preços, compras e investimentos.

A solução proposta demonstrou sua relevância ao fornecer uma abordagem dinâmica e adaptável às oscilações do mercado, transformando grandes volumes de informações brutas em relatórios detalhados e acionáveis. Dessa forma, o DriveIntel se estabelece como um complemento essencial às

ferramentas tradicionais, permitindo que empresas do setor automotivo tomem decisões mais precisas, maximizando sua competitividade e lucratividade.

Como perspectiva futura, a solução poderá ser aprimorada com novos modelos preditivos e a integração de fontes de dados adicionais, ampliando ainda mais a capacidade de análise e previsão de tendências do mercado automotivo.

7. REFERÊNCIAS

BRANDÃO, Camilo; SCARPEL, Rodrigo Arnaldo. **Importância das características na precificação de veículos nacionais**: Importance of characteristics in pricing national vehicles. Scielo, [S. l.], p. 1-1, 9 jun. 2009. Disponível em: <https://www.scielo.br/j/prod/a/S7GLKxR4hykmW9RZsXZHX8m/>. Acesso em: 18 nov. 2024.

HOPSWORK. **Feature Types for Machine Learning**. [S. l.], 2022. Disponível em: <https://www.hopsworks.ai/post/feature-types-for-machine-learning>. Acesso em: 19 mar. 2025.

MEDIUM. **Some Key Machine Learning Definitions**. [S. l.], 2017. Disponível em: <https://medium.com/technology-nineleaps/some-key-machine-learning-definitions-b524eb6cb48#:~:text=If%20the%20training%20set%20is,the%20input%20in%20your%20system>. Acesso em: 17 mar. 2025.

MEDIUM. **Understanding Common Regression Evaluation Metrics: MAE, MSE, RMSE, R2, and Adjusted R2**. [S. l.], 2023. Disponível em: <https://ai.plainenglish.io/understanding-common-regression-evaluation-metrics-mae-mse-rmse-r2-and-adjusted-r2-6c5709e614c4>. Acesso em: 19 mar. 2025.