

# Sistema Web de Suporte à Decisão para o Processo de Classificação e Certificação de Grãos

## Web-based Decision Support System for the Grain Classification and Certification Process

Maicon Aparecido Sartin  
mapsartin@unemat.br  
Universidade do Estado de Mato Grosso - UNEMAT

Leonardo Oliveira de Moraes  
[leonardo.moraes@unemat.br](mailto:leonardo.moraes@unemat.br)  
Universidade do Estado de Mato Grosso - UNEMAT

### ABSTRACT

Agriculture is a pillar of the Brazilian economy, solidifying the country as one of the world's largest food producers. In Mato Grosso, specifically, grain production is a central driver of development. The negotiation process between producers and buyers depends on the quality certification of the grains to validate delivery, a service currently performed by various specialized companies. This work aims to optimize this exact process. It proposes the development of a web system that uses dashboards and Key Performance Indicators (KPIs) for grain negotiation and classification. The platform allows for real-time data monitoring by certification companies, buyers, and producers, in addition to providing technical reports to validate transactions. Developed with modern web technologies, the system presents the data in a dynamic and responsive way, offering an intuitive interface that facilitates decision-making for everyone involved.

### CCS Concepts

- Software and its engineering → Development frameworks and environments;
- Information systems → Decision support systems → Data analytics.
- Applied computing → Agriculture;

### Keywords

Grain classification, Web system, Dashboards, Agribusiness, Decision making.

## 1. INTRODUÇÃO

O setor da agricultura é um dos principais impulsionadores do Produto Interno Bruto (PIB) nacional, consolidando o país como um dos maiores produtores de alimentos do mundo. Com o aumento da produção agrícola a cada ano, torna-se inevitável a existência de um grande déficit na capacidade de armazenamento de grãos no país e no estado de Mato Grosso [1][2][3]. O déficit de disponibilidade de armazenamento de grãos no país é uma realidade há muitos anos.

Diante do aumento da produtividade de grãos e do déficit de disponibilidade de armazenagem, faz-se necessária a presença de empresas de certificação de qualidade de grãos. Dependendo da qualidade do grão e das flutuações de preços, são necessários o armazenamento e uma logística adequada do produto. A qualidade do grão determina se o produto será destinado diretamente a um comprador ou se passará por um silo de secagem, por exemplo. Com isso, diversas empresas estão se especializando para a

prestação de serviços com o intuito de certificar a qualidade dos grãos.

A certificação de qualidade do grão é um documento formal gerado pela empresa certificadora. Tal empresa e laboratório associado necessitam de um credenciamento no MAPA para emitir laudos ou certificados de classificação oficial. As empresas certificadoras geralmente ficam em locais estratégicos e logisticamente viáveis em diversos pontos de embarque no país, como fazendas, silos, portos secos e terminais portuários. A logística no transporte de grãos é uma preocupação constante dos produtores para não perder a qualidade do produto, ou até mesmo, os contratos com os compradores dos grãos.

Desta forma, o controle de qualidade do grão e o logístico do transporte devem estar alinhados para o produtor não acarretar prejuízos financeiros [4]. Os sistemas de informação para classificadores de grãos trazem essa segurança ao produtor. A empresa certificadora junto à empresa de produção de grãos traz uma quantidade massiva de dados operacionais diários para serem utilizados em uma análise contínua para os produtores, a fim de gerar indicadores para tomadas de decisão. Os indicadores-Chave de Desempenho, chamados de KPI (Key Performance Indicator), auxiliam os produtores para melhorar a operação de logística e armazenagem, bem como tornam mais eficientes e automatizados os processos ao longo da cadeia produtiva visando o aumento de lucros.

Dado o contexto, esse trabalho visa apresentar um sistema web desenvolvido para gerenciamento de informações de empresas produtoras de grãos e empresas certificadoras da qualidade de grãos com indicadores-Chave de desempenho. As contribuições da proposta estão na otimização e automação dos processos, além de facilitar a tomada de decisão em tempo real. A interface gráfica (Frontend) é multiplataforma e responsiva com as informações pertinentes organizadas de forma simples e objetiva por meio de Dashboards interativos.

## 2. TRABALHOS RELACIONADOS

A alta demanda global por alimentos gerou uma vasta literatura sobre a cadeia de produção de grãos. No entanto, o cenário nacional possui restrições e características próprias que tornam parte dessa cadeia produtiva única e específica para a realidade brasileira. A análise dos trabalhos existentes revela um foco robusto em três pilares centrais: rastreabilidade, certificação e os sistemas de gestão necessários para implementá-los.

## 2.1 Rastreabilidade

Diversos estudos abordam a rastreabilidade como uma ferramenta fundamental na segurança alimentar e no controle de qualidade de alimentos de forma geral. A capacidade de rastrear grãos é complexa pelo envolvimento de diversos segmentos, pessoas e empresas ao longo da cadeia produtiva. Em um mundo ideal teríamos as informações exatas de rastreio desde a produção, transporte, armazenamento e industrialização final do produto.

Em [5], apresentam-se várias características necessárias para definir um limiar entre satisfação do consumidor e o custo para a empresa na rastreabilidade. Entre as diversas características envolvidas, citam-se a rotulagem e a relação com as marcas, padronização, certificação, qualidade com segurança, etc. A autora mostra a importância e as dificuldades da implantação a partir dos interesses do governo e dos diferentes agentes do sistema agroindustrial.

Esses interesses partem principalmente da regulamentação e das exigências internacionais na exportação de produtos agroindustriais. A ausência de consenso científico ou de estudos conclusivos sobre os efeitos de organismos geneticamente modificados na saúde humana impõe uma necessidade crescente de cautela em sua produção [6].

## 2.2 Certificação de Processos e Unidades

Uma parte importante no processo de rastreabilidade é a certificação do produto "da lavoura à indústria". A certificação é o mecanismo formal de validação da qualidade do produto, neste caso, o grão. Existem diversas vertentes de estudos com esse tema, bem como a indicação da complexidade e a relação com os múltiplos pontos de controle.

As duas principais certificações estão relacionadas ao produto e às unidades armazenadoras. Por exemplo, nas unidades armazenadoras diversos requisitos e registros internacionais devem ser levados em consideração para se tornar uma unidade certificada [7][8]. De forma semelhante, a empresa certificadora de produtos para alcançar o registro internacional para o controle de qualidade do produto. No produto, a empresa certificadora emite laudos certificando a qualidade do produto com diversas características e informações em relação aos grãos coletados e à carga a ser certificada.

Essas ferramentas necessitam de diversas informações [9], a classificação de grãos é regulamentada pelo Ministério da Agricultura (MAPA), conforme normas técnicas definidas pelo governo. A Lei nº 9.972/2000 torna a classificação obrigatória para produtos vegetais, subprodutos e resíduos de valor econômico, geralmente relacionados ao consumo humano.

A existência de soluções de mercado, como a plataforma [10], evidencia a demanda prática por ferramentas que digitalizem e gerenciem laudos e classificações de grãos. A ferramenta permite gerenciar a produção dos classificadores e gerar laudos técnicos detalhados de forma rápida e segura. Existem estudos de visão computacional para automatizar a análise e classificação dos grãos, que atualmente são feitas de forma manual [11][12].

## 2.3 Sistemas de Gestão Agroindustrial

A agroindústria utiliza diversos Sistemas de Gestão (SGA), cada um focado em segmentos ou pontos específicos da cadeia produtiva. Os SGAs mais comuns são aplicados na gestão de fazendas, otimização logística, controle de produtividade,

rastreabilidade e processos de certificação. Para que a rastreabilidade e a certificação sejam efetivas, elas dependem fundamentalmente desses SGAs e da tecnologia da informação (TI) para o registro e a manutenção dos dados históricos.

Em [13], os autores definem um modelo de Sistema Integrado de Gestão da Qualidade (SIGQ) para unidades de beneficiamento de sementes. A metodologia proposta orienta a gestão dos principais processos, demonstrando como o SIGQ permite identificar e minimizar gargalos ao analisar diversos fatores, como tecnologia, recursos humanos, gerenciamento, produtividade e logística.

## 3. DESENVOLVIMENTO DO SISTEMA WEB

### 3.1 Estrutura Geral

A proposta visa a integração de informações de produtores rurais com as empresas certificadoras de grãos, trazendo indicadores KPI e dashboards interativos para facilitar a tomada de decisão do produtor ou das empresas certificadoras. As certificadoras prestam serviços aos compradores dos grãos e fazem o intermédio com os produtores. Cada negociação entre os compradores e os produtores de grãos é regida por meio de um contrato. Esse contrato contém diversas características e requisitos mínimos da negociação como a qualidade do produto, quantidade, preço, prazo, entre outros. Dessa forma, a prestadora de serviço, responsável pela classificação e certificação dos grãos, emite laudos que avaliam a qualidade do produto durante sua trajetória até o destino final. Esses documentos atestam que a qualidade negociada se mantém inalterada ou atende ao padrão de classificação definido em contrato.

As prestadoras de serviço em conjunto com os produtores contêm um fluxo de volume de dados muito alto, ou seja, são muitos laudos de certificação emitidos diariamente. Os laudos são emitidos em vários pontos de embarque diferentes (locais onde são retirados os produtos (grãos)) ou unidades de armazenamento de grãos. A quantidade de negociações entre compradores e produtores é alta e tem impacto no volume de dados, devido à emissão de laudos estar relacionada à negociação e por várias prestadoras de serviço (certificadoras) distintas. Além disso, existe outra exigência importante no sistema para disponibilização dos dados, devido à necessidade das empresas de acompanhar diariamente as informações em tempo real da operação para tomadas de decisão.

As informações do sistema estão baseadas em Gestão de Relacionamento com o Cliente (Customer Relationship Management (CRM)). O gerenciamento das informações dos prestadores de serviços e produtores está organizado com as tecnologias atuais, frameworks, bibliotecas, etc.; algumas podem ser visualizadas no diagrama de componentes da Figura 1. Na figura observa-se uma visão geral do sistema de informação para classificação de grãos.

Os principais componentes são o Frontend, Backend, Camada de dados e fontes externas. Nota-se a existência de duas fontes de dados, um interno e outro externo ao sistema, além de atualizações por *migrations*. A padronização dos dados com JSON e a persistência dos dados são fatores importantes da proposta. Além disso, as permissões de usuários ou empresas são características importantes, devido à relação dos KPIs e dashboards serem

dinâmicos e totalmente relacionados ao usuário logado e suas devidas permissões.

### 3.2 Backend

O Backend do sistema consiste em dois componentes principais: a coleta de dados automática e a API de serviços FastAPI.

A coleta de dados automática, quando habilitada, tem a função de execução de tarefas assíncronas (*AutoDataCollector*) para coletas periódicas, consultando as fontes externas (PostgreSQL/APIs) e persistindo os dados processados. Inclusive, faz a verificação periódica das configurações (*DataSourceConfig*) para identificar quais fontes precisam ser atualizadas e aciona o *DataCollectorService* para buscar e persistir os novos dados. O uso de *asyncio* e locks distribuídos (via Redis) garante a execução da

principais serviços são definir os *endpoints* da API REST, gerenciar as requisições e as respostas, integrar todos os outros componentes como serviços, o ORM e as tarefas em segundo plano.

A biblioteca *Pydantic* tem a função de validação e parse de dados em Python, baseada em type hints, além de auxiliar a FastAPI na validação dos dados. Com a *Pydantic*, o desenvolvedor pode definir a forma dos dados (esquemas) usando classes Python, com a intenção de garantir que os dados recebidos (ex.: de uma requisição JSON) ou enviados fiquem padronizados ao esquema definido. Por exemplo, o DTO (*Data Transfer Objects*) *DataSourceConfigCriacaoDTO* define exatamente quais campos são esperados ao se criar uma nova fonte de dados, seus tipos de dados e a obrigatoriedade. A FastAPI usa esses modelos para validar automaticamente os dados de entrada, retornando erros

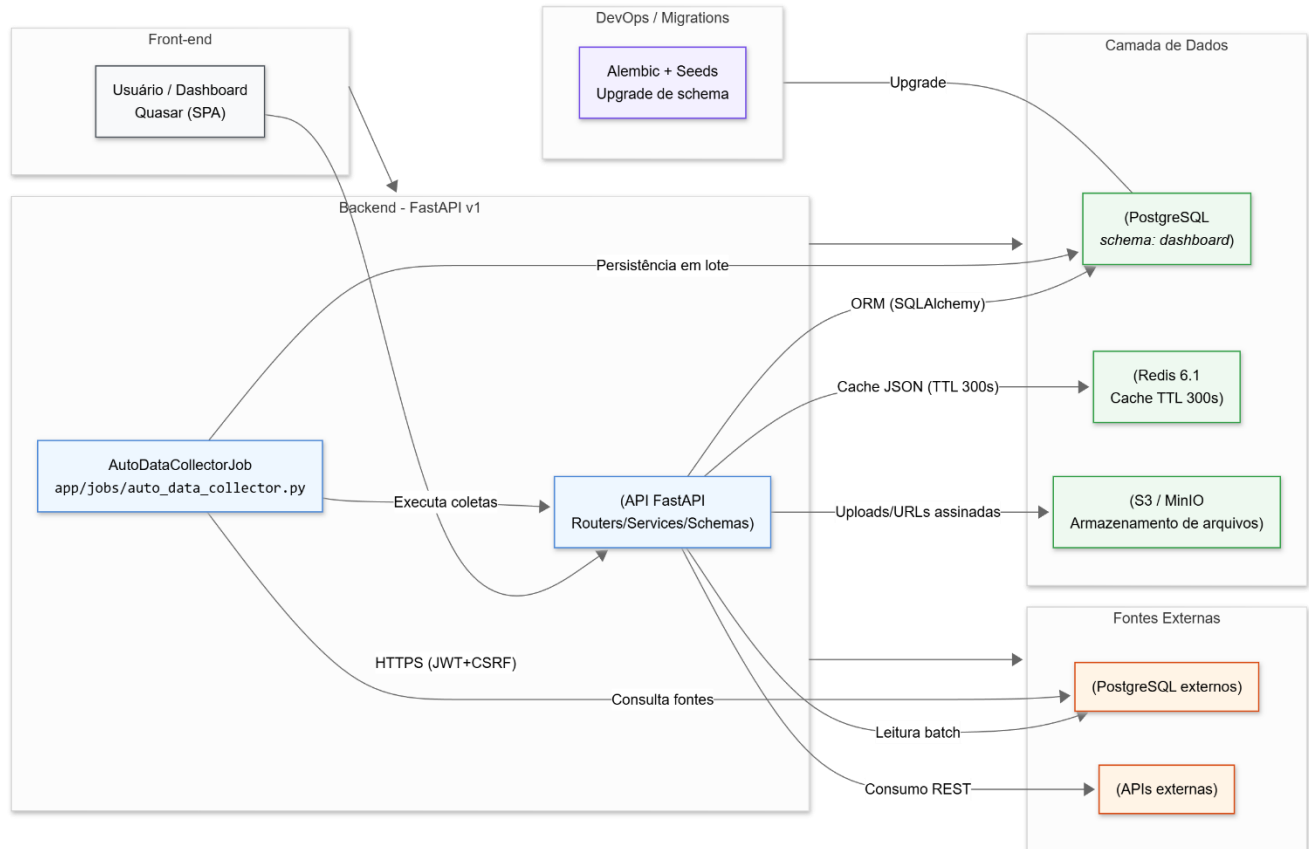


Figura 1. Diagrama de Componentes.

aplicação de forma eficiente e segura, sem travar e sem risco de execuções duplicadas. As tarefas reutilizam os serviços de domínio da API como o cancelamento seguro e a própria flag de habilitação. As principais tarefas da coleta de dados automática como a execução de coletas, persistência em lote e consulta segura nas fontes de dado estão especificadas na Figura 1.

O FastAPI é um framework web moderno de alto desempenho para a criação de APIs em Python. Suas principais características são o desempenho (comparável a NodeJS e Go), a documentação automática e interativa (via OpenAPI e Swagger UI) e o uso de *type hints* (dicas de tipo) do Python para validação de dados e o aumento da produtividade do desenvolvimento. Este componente representa a arquitetura central (core) de todo o sistema de Backend. Os

claros ao cliente caso os dados sejam inválidos, e também para serializar os dados de saída, garantindo que a API seja previsível e bem documentada.

Na organização da estrutura do código do Backend, pode-se destacar outras características relevantes como:

- **main.py:** lifespan, migrações Alembic, seeds padrão, montagem do subapp /api/v1, health-check;
- **appcore:** config (JWT, S3, Redis, CORS, rate limit, flags), security (JWT/cookies/CSRF), redis\_cliente;
- **app/api:** routers v1; deps (sessão DB, CSRF, autenticação por cookies/perfil);

- **app/services:** auth, usuários, empresas, menus, cards, PowerBI, filtros, conexões externas, fontes de dados, coleta/persistência, storage;
- **app/models:** entidades ORM; mixins de auditoria;
- **app/jobs/auto\_data\_collector.py:** job assíncrono cancelável, condicionado por flag;
- **alembic/versions:** evolução do schema (tabelas base, filtros, campos dinâmicos, dashboard\_data\_row).

O diagrama da Figura 2 detalha o processo de leitura de dados de um card, que utiliza uma estratégia de cache. O Cliente de Aplicação de Página Única (SPA-U) faz uma requisição 'GET' ao Router (R) para obter os dados de um card específico. O Router primeiro consulta o *DashboardCardService* (C) para buscar as configurações desse card no banco de dados (PostgreSQL (DB)). A partir das configurações, o Router solicita os dados do *DashboardDataService* (D), que verifica a existência dos dados no Cache (Redis). Caso existam, os dados são apresentados, senão o serviço busca os dados diretamente no DB e salva no Cache (Redis) com um tempo de expiração (TTL). O Router (R) envia a resposta final (HTTP 200) em formato JSON para o Cliente SPA (U). Nessa interação, a sequência de acessos, requisições e funcionalidades está vinculada ao cliente, às rotas, às informações de cartões e dados do dashboard, às configurações de núcleo (Redis) e ao acesso ao sistema gerenciado de banco de dados (PostgreSQL).

### 3.3 Camada de Dados, Migrações e Fontes Externas

Dentre as principais funções da camada de dados estariam o caching e exclusão mútua com o Redis, mapeamento objeto-relacional e migrações com SQLAlchemy e Alembic, fontes externas e persistência dos dados com o PostgreSQL e armazenamento de dados com S3/MinIO. As relações entre as funcionalidades podem ser observadas na Figura 1.

extremamente rápido e versátil, que armazena os dados em memória RAM. Seu funcionamento é baseado em uma estrutura de chave-valor, mas suporta diversos tipos de dados complexos como listas, *hashes* e *sets*. O Redis é utilizado para duas funções críticas:

- **Caching:** tem a função de reduzir o número de consultas ou requisições ao banco de dados principal com o intuito de melhorar o tempo de resposta do Frontend. Geralmente, utiliza-se o cache para dados acessados com maior frequência, como os resultados de coletas externas. Tais dados são armazenados temporariamente no Redis com um TTL (Time-To-Live, ou tempo de vida) configurável (padrão de 5 minutos).
- **Locks Distribuídos:** Em um ambiente onde a aplicação pode ser executada em múltiplas instâncias (escalonamento horizontal), o Redis é usado para implementar um sistema de "travas" (locks). Isso garante que tarefas sensíveis, como a coleta automática de dados (*AutoDataCollectorJob*), sejam executadas por apenas uma instância por vez, prevenindo condições de corrida (exclusão mútua), duplicação de processamento e inconsistências nos dados.

O Alembic é uma ferramenta de migração de banco de dados leve e robusta, criada especificamente para ser usada com o SQLAlchemy. A ferramenta permite o gerenciamento e o versionamento nas alterações do esquema do banco de dados (como criar tabelas, adicionar colunas, criar índices) de forma pragmática e controlada. Cada alteração é um "script de migração" que pode ser aplicado (upgrade) ou revertido (downgrade). O Alembic é fundamental para a manutenção e evolução do banco de dados, bem como o monitoramento das mudanças nos modelos de dados do *SQLAlchemy* (*target\_metadata*). Assim, o Alembic pode gerar automaticamente um script de migração contendo as operações DDL (Data Definition Language) necessárias. Isso garante que a

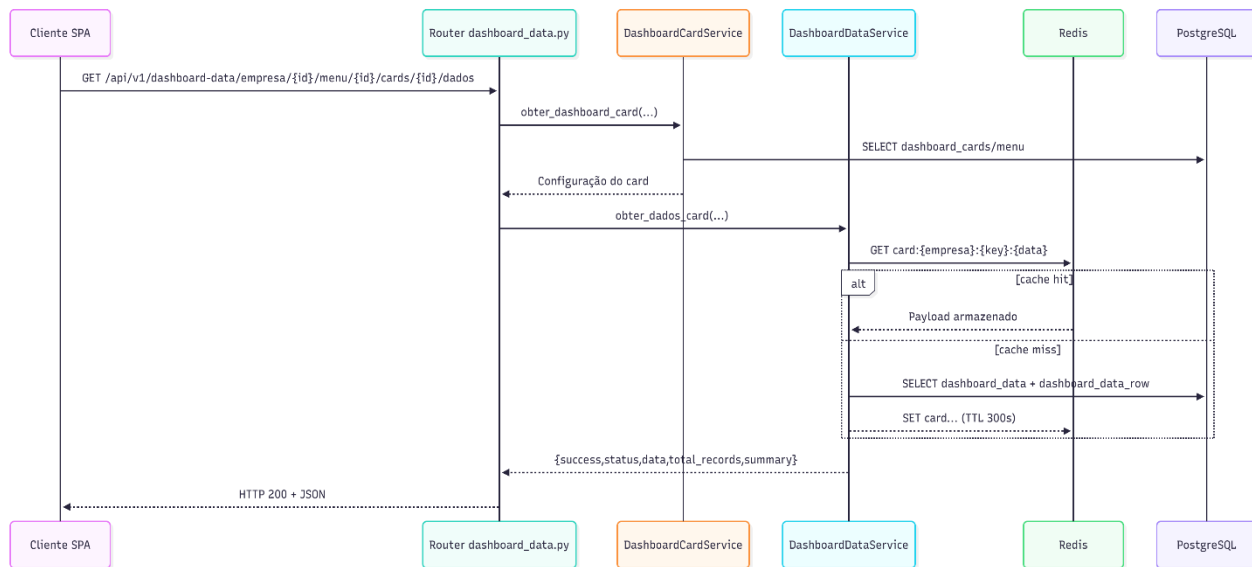


Figura 2. Diagrama de sequência.

O Redis é uma parte importante do Backend pelo desempenho agregado à aplicação, principalmente em operações de leitura e escrita, tornando-o ideal para aplicações de baixa latência. O Redis corresponde a um banco de dados NoSQL de código aberto,

estrutura do banco de dados evolua de forma consistente junto com o código da aplicação, facilitando a implantação em diferentes ambientes (desenvolvimento, homologação, produção).

A biblioteca *SQLAlchemy* de ORM em Python fornece um conjunto de ferramentas de alto nível para se comunicar com bancos de dados relacionais. A principal vantagem ao desenvolvedor é a interação com o banco de dados usando objetos Python (modelos) em vez de escrever SQL puro, com a intenção de abstrair as diferenças entre os dialetos de SQL e o aumento da produtividade. A biblioteca no sistema é utilizada para definir o modelo de dados da aplicação (tabelas definidas como classes em Python) e realizar as consultas internas ao banco de dados. Assim, a biblioteca traduz as operações em objetos Python para comandos SQL, facilitando as operações de CRUD.

O banco de dados interno do sistema é construído com o sistema gerenciador PostgreSQL. Na Figura 3 observa-se o diagrama de entidade e relacionamento do sistema com as entidades internas do sistema e os dados coletados de outras aplicações e registrados no banco do sistema. As entidades definidas no banco de dados são: PERFIS, USUARIOS, USUARIOS\_X\_EMPRESAS, EMPRESAS, MENUS, DASHBOARD\_CARDS, FILTROS\_DASHBOARD, CONTEUDOS\_POWERBI, DASHBOARD\_DATA, DASHBOARD\_DATA\_ROW, DATABASE\_CONNECTIONS, DATA\_SOURCE\_CONFIGS. As entidades vinculadas a dados externos são principalmente as quatro últimas.

O sistema possui uma arquitetura flexível para consumir dados de fontes externas, configuradas através das entidades *DatabaseConnection* e *DataSourceConfig*. O sistema pode se conectar a distintos tipos de fontes externas como instâncias de conexões diretas ao PostgreSQL e APIs HTTP (Hypertext Transfer Protocol). As instâncias do PostgreSQL podem executar consultas com o *PostgresCollector* por meio da biblioteca *psycpg2*. Já as APIs HTTP com o *APIExternaCollector* por meio da biblioteca *requests* realizam requisições HTTP (GET/POST) a *endpoints* de APIs de terceiros.

O acesso aos dados externos contém uma preocupação com a segurança, assim, todas as credenciais são armazenadas de forma criptografada e as queries SQL passam por um validador (*sql\_validator*) para prevenir ataques de injeção de SQL. A arquitetura suporta também diversos tipos de autenticação (como Bearer Token, ApiKey, Basic Auth), com os tokens e segredos armazenados de forma criptografada. Isso permite que o sistema se integre com uma vasta gama de serviços externos para enriquecer os dados dos dashboards.

Amazon S3 é um serviço de armazenamento de objetos em nuvem da AWS, conhecido por sua escalabilidade, durabilidade e segurança. MinIO é uma alternativa de código aberto compatível com a API do S3, que pode ser auto-hospedada (on-premises ou em qualquer nuvem). Ambos permitem armazenar e recuperar grandes volumes de dados não estruturados, como imagens, vídeos, backups e logs. O sistema utiliza uma abstração (via boto3) que se comunica com a API do S3 e permite flexibilidade na implantação em nuvem da AWS e em uma infraestrutura própria com MinIO. Sua função é servir como um repositório central para arquivos e objetos, como laudos técnicos, imagens de empresas ou outros documentos. O serviço de armazenamento do sistema gerencia a criação de buckets, aplica políticas de acesso, valida os tipos de arquivo permitidos, gera nomes únicos para evitar colisões e oferece funcionalidades para upload, exclusão e geração de URLs temporárias e seguras (presigned URLs) para acesso controlado aos arquivos.

### 3.4 Frontend

No Frontend utiliza-se diversas tecnologias para facilitar a interação e a experiência do usuário final por meio de menus, cartões, dashboards, gráficos, responsividade e etc. Além disso, o principal objetivo é reunir dados padronizados de vários locais em apenas um único local por meio das conexões com fontes externas.

As fontes externas contêm diversos tipos de aplicações como: legadas, planilhas, PowerBI, Dephi, nuvem, etc. Outra informação importante está relacionada à segurança no acesso dos usuários no sistema para verificação de perfil ou acesso por rota. O sistema adota uma autenticação via JSON Web Token (JWT) junto com cookies HTTPOnly, bem como a proteção de Falsificação de Solicitação Entre Sites (em inglês, Cross-Site Request Forgery (CSRF)), políticas CORS (Cross-Origin Resource Sharing) e senhas com hash (bcrypt).

Algumas tecnologias e frameworks atuais foram utilizados no sistema para garantir uma interface de usuário (UI) amigável e responsiva.

Dentre as várias tecnologias, frameworks e bibliotecas utilizadas e citadas anteriormente, pode-se destacar o Vue.js 3, Quasar e o Pinia.

O Vue.js é um framework progressivo e de código aberto para a construção de interfaces de usuário (UI) em JavaScript. É a base de toda a interface do usuário e a *Composition API* é utilizada para estruturar a lógica de cada componente e view, como no *DashboardPageComFiltros.vue*, onde `ref`, `computed`, etc. São usados para gerenciar o estado, reatividade e ciclo de vida do componente. Outro framework de UI é o Quasar com a proposta de escrever o código uma vez e implantar em todos os lugares. Ele oferece um extenso conjunto de componentes de UI pré-construídos e prontos para uso (seguindo o padrão Material Design) e ferramentas de CLI que permitem gerar SPAs (Single Page Apps), PWAs (Progressive Web Apps), aplicações mobile (via Cordova ou Capacitor) e aplicações desktop (via Electron) a partir de uma única base de código. O Quasar é responsável pela aparência visual e pela experiência do usuário. Componentes como `<q-btn>`, `<q-badge>`, `<q-tooltip>`, `<q-toggle>`, `<q-spinner-dots>` e o sistema de grid responsivo são amplamente utilizados para construir uma interface rica, interativa e responsiva, além de acelerar drasticamente o desenvolvimento.

A biblioteca Pinia serve para o gerenciamento de estado e oferece um armazenamento (store) centralizado para guardar dados a serem compartilhados entre múltiplos componentes, de uma forma mais simples e intuitiva que seu predecessor, o Vuex. O Pinia é utilizado para gerenciar o estado global da aplicação. Por exemplo, *useDashboardCardsStore*, *useMenuStore* e *useFiltroDashboardStore* são stores que centralizam o estado e a lógica relacionados aos cards do dashboard, aos menus e ao sistema de filtros, respectivamente. Isso torna o estado da aplicação mais previsível, fácil de depurar e desacopla a lógica de negócio dos componentes de UI.

Com um menu definido e um registro de empresa com sua respectiva conexão, é possível construir diversos tipos de cards agregados em painéis completos. Cada Paineis pode conter KPIs, gráficos, mapas e tabelas para a análise e acompanhamento operacional de qualquer ramo de negócio e por segmentos do negócio, nesse caso para a classificação de grãos.





Figura 4. Cadastrar ou editar um componente visual.

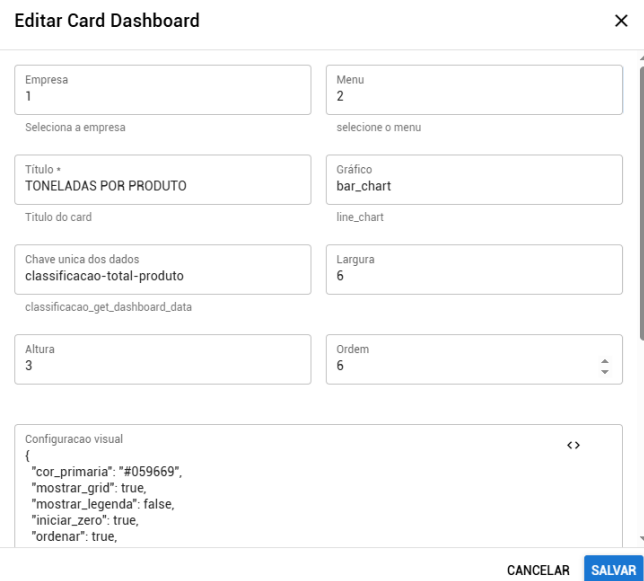


Figura 5. Cadastrar ou editar um componente visual (continuação).

Essas informações podem determinar diferentes tomadas de decisão como a quantidade de colaboradores a serem alocados para determinado ponto de embarque, dependendo do volume em determinado(s) dia(s). Além de embutir a despesa vinculada a esses colaboradores na negociação. Já nas informações dos laudos, da geolocalização, e da rastreabilidade em tempo real podem garantir a presença do colaborador no ponto de embarque. Outra característica importante dos laudos emitidos é a verificação de padronização, com isso, é possível comunicar ao cliente em tempo real para tomada de decisão. Nesse caso, por exemplo, se essa carga for refugada ou negociada a outro preço, ou ainda encaminhada para repadronização em empresas especializadas nesse quesito.

Essa tomada de decisão tem uma característica extremamente importante devido à sua responsabilidade financeira com a negociação da carga.

Com todas as informações em dispositivo móvel, possibilita uma negociação mais transparente a todos os envolvidos. Assim, o sistema visa facilitar a visualização, o gerenciamento e a análise desses dados em tempo real por meio de um Frontend SPA e uma API REST no Backend. O objetivo é oferecer uma experiência intuitiva, responsiva e profissional, com dashboards, gráficos e KPIs e para o cliente conseguir administrar de forma eficiente sua operação no dia a dia, além de auxiliar na tomada de decisões com informações precisas e atualizadas.

#### 4. RESULTADOS E DISCUSSÃO

A “Blind” Sistemas é a empresa detentora do projeto na qual foi implementada a solução para entregar aos seus clientes um sistema que integra outras soluções específicas da própria empresa ou de terceiros. Neste trabalho foi explorado o Sistema para Certificadoras de Grãos, uma ferramenta estratégica para alavancar os negócios dos clientes. Atualmente, a empresa contém dezenas de parceiros no segmento, além das principais certificadoras de grãos do país utilizando o sistema. Os clientes estão atuando em quase todos os estados do país, com volume maior no Mato Grosso, Goiás, Pará e Paraná. Com equipes em diversos pontos de embarque e desembarque, como armazéns, fazendas e portos de exportação.

Anteriormente a essa proposta, os relatórios dos dados eram manuais por meio de planilhas ou em diferentes sistemas e com dados separados, dificultando a análise dos dados e a tomada de decisão. Com a inserção do sistema web de Dashboard e KPIs nas certificadoras de grãos, os fluxos automatizados e otimizados tiveram melhorias significativas em diversas etapas no processo, como na coleta, no gerenciamento, no armazenamento, na segurança e na visualização de dados.

O sistema web fornece diversas melhorias no acesso instantâneo a indicadores-chave de desempenho (KPIs), dashboards interativos com dados sempre atualizados, ainda permite aos usuários tomarem decisões mais rápidas e bem informadas. Outra melhoria está na rastreabilidade dos laudos e da qualidade dos grãos, trazendo aos usuários o aumento da transparência e da confiança nas negociações. A implementação das validações de segurança para consultas SQL, banco de dados estruturados, persistência nos dados e etc; garantem a consistência dos dados em único local seguro, criptografado e com acesso restrito por perfis de usuário, confirmando as diversas melhorias em vários aspectos.

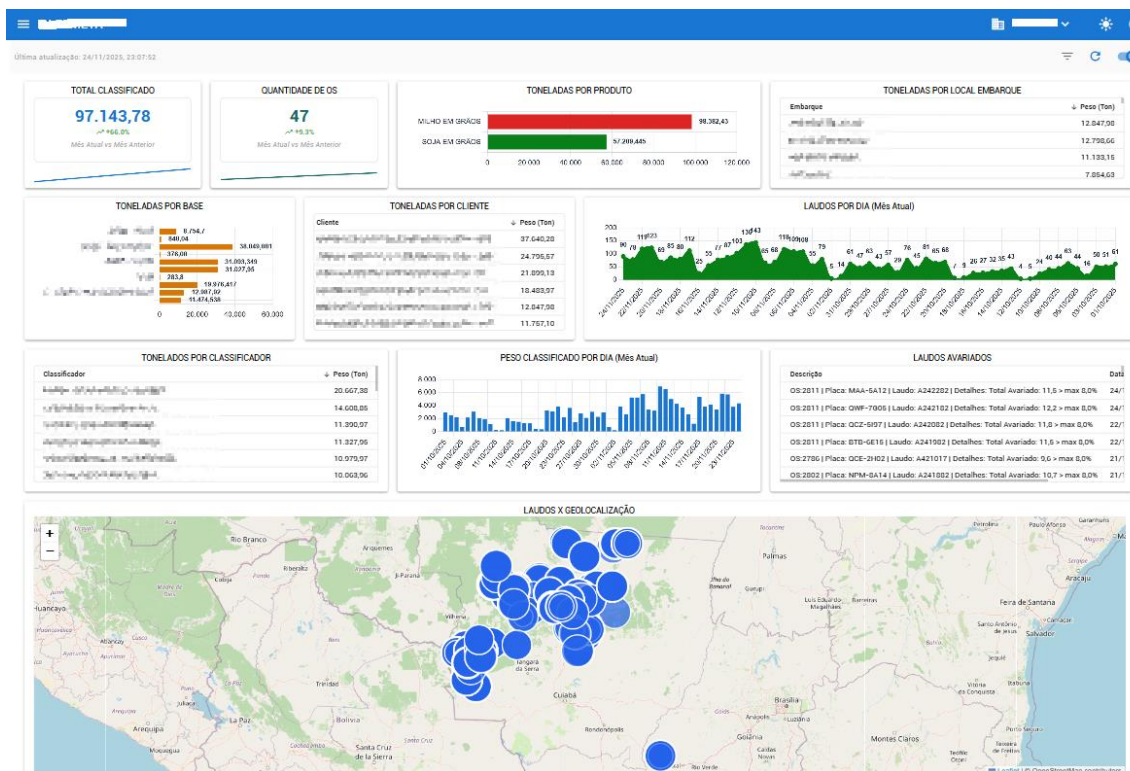


Figura 6. Dashboard.

Ademais, essa aplicação web se diferencia por ser um sistema de gerenciamento de dados que permite a construção dinâmica de dashboards e KPIs, suportando múltiplas empresas e usuários, e a equipe técnica da “Blind” Sistemas (acesso gerencial) tenha total liberdade para construir painéis completos com diversas opções de gráficos dinamicamente dentro do site segundo a visão estratégica da empresa parceira da “Blind”.

Além das melhorias, uma grande variedade de otimizações pode ser elencada:

- Otimização de Processos: automatizar tarefas como a coleta de dados, solicitando relatórios, planilhas e análise de terceiros para tomadas de decisões operacionais.
- Otimização de Tempo: Reduziu drasticamente o tempo necessário para obter e analisar dados críticos, graças à coleta automática, ao *caching* de dados e à apresentação em dashboards em tempo real.
- Otimização de Recursos: Permite uma melhor alocação de recursos humanos e materiais, fornecendo insights claros e acionáveis a partir dos KPIs e dashboards.
- Otimização de Custos: Contribuiu para a redução de perdas financeiras em negociações de grãos, resultado de uma gestão mais eficaz da qualidade e da logística.
- Otimização de Desempenho Técnico: Utiliza *caching* com Redis para diminuir a carga sobre o banco de dados e acelerar a entrega de informações. Implementa persistência em lote para processar e armazenar grandes volumes de dados de forma eficiente.

Dentre os diversos benefícios da implementação do sistema pode-se citar uma subdivisão em três partes. Para a Empresa (Desenvolvedora/Operadora):

- Integra vários sistemas e serviços da “Blind”, centralizando as informações, com a disponibilidade de dados em qualquer lugar e em tempo real, além de agregar valor ao cliente e fortalecer a parceria entre as partes.
- Escalabilidade: A arquitetura robusta, baseada em tecnologias como FastAPI, Redis, S3/MinIO e Alembic, oferece escalabilidade para suportar o crescimento da demanda e a integração com diversas fontes de dados.
- Vantagem Competitiva: Posiciona a empresa com uma solução moderna e eficaz para um nicho de mercado específico, mas relevante na agricultura.

Para as Empresas Usuárias (Certificadoras, Produtores):

- Visão Abrangente e Centralizada: Dashboards com KPIs que consolidam todas as informações críticas sobre a classificação e negociação de grãos em um único lugar.
- Redução de Perdas e Riscos: Um controle logístico e de qualidade dos grãos, visando evitar prejuízos financeiros decorrentes de uma gestão errônea ou decisões atrasadas.
- Apoio a Decisões Estratégicas: Suporte robusto para a tomada de decisões sobre alocação de recursos, negociação de preços e identificação e resolução de gargalos operacionais.

Para os Clientes Finais (Trading de grãos):

- **Transparência e Confiança:** Acesso facilitado a informações precisas sobre a qualidade e classificação dos grãos, aumentando a confiança em todo o processo de negociação.
- **Agilidade nos Processos:** A disponibilidade de dados em tempo real agiliza os processos de compra e venda.

## 5. CONCLUSÃO

Neste trabalho, apresentou-se um sistema web para otimização de processos no auxílio de tomada de decisões estratégicas com base na análise de dados com dashboard e KPIs, para atender à necessidade crítica de informações gerenciais em tempo real para empresas certificadoras de qualidade de grãos.

O sistema foi desenvolvido com diversas tecnologias e técnicas de programação atuais, permitindo que vários usuários e múltiplas empresas tenham acesso a menus e dashboards distintos. A principal característica é a configuração dinâmica para possibilitar a construção de cards e dashboards livremente dentro do sistema pela equipe técnica. As informações são disponibilizadas por meio de dados padronizados em JSON para gerar os componentes como Charts.js, além de tabelas, gráficos, mapas e cards de KPI com diferentes tipos de visualizações de forma moderna e design minimalista.

A aplicação segue uma arquitetura moderna com Frontend com SPA, desenvolvido com Vue.js 3 (*Composition API*) e o framework de UI Quasar, para garantir uma interface responsiva e intuitiva que facilita a experiência do usuário. O Backend baseado em FastAPI (python) oferece alto desempenho, utilizando PostgreSQL para persistência de dados, modelados via SQLAlchemy e Alembic para migrações. O sistema demonstrou uma grande variedade de melhorias, otimizações e benefícios em sua implementação. As melhorias foram elencadas em diversos pontos como na coleta, no gerenciamento, no armazenamento, na segurança e na visualização de dados. As otimizações foram classificadas em: Processos, Tempo, Recursos, Custos e Desempenho Técnico. Por fim, os benefícios estão relacionados a todos os participantes do sistema como Empresa (Desenvolvedora/Operadora), Empresas Usuárias (Certificadoras, Produtores) e Clientes Finais (Produtores/Trading de Grãos).

## 6. REFERÊNCIAS

- [1] de Souza, M. F. and Bastos, G. 2025. REFLEXÕES SOBRE O DÉFICIT DE ARMAZENAGEM DE GRÃOS. FGV Agro. Obtido em 14 de novembro de 2025 de <https://agro.fgv.br/noticia/reflexoes-sobre-o-deficit-de-armazenagem-de-graos>.
- [2] Tomazela, J. M. 2023. Maior safra da história: Falta de armazém para 124 mi de t de grãos. Estadão. Obtido em 7 de novembro de 2025 de <https://www.estadao.com.br/economia/maior-safra-da-historia-falta-de-armazem-para-124-mi-de-t-de-graos/>.
- [3] Magri, C. 2023. Armazenagem de grãos no brasil demanda r\$ 102 bi em investimentos. Bloomberg Línea. Obtido em 15 de novembro de 2025 de <https://www.bloomberglinea.com.br/agro/armazenagem-de-graos-no-brasil-demanda-r-102-bi-em-investimentos/>.
- [4] Serviço Nacional de Aprendizagem Rural (SENAR). 2018. Grãos: armazenamento de milho, soja, feijão e café. Number 216 in Coleção SENAR. SENAR, Brasília, DF.
- [5] Machado, R. T. M. 2005. Sinais de qualidade e rastreabilidade de alimentos: Uma visão sistêmica. Organizações Rurais e Agroindustriais/Rural and Agro-Industrial Organizations. 1511-2016-131188, 11.
- [6] Myszczyk, A. P. and Wandscheer, C. B. and Glitz, F. 2010. Segurança alimentar e consumo: Rastreabilidade e certificação de grãos gm e non-gm. Revista Cesumar – Ciências Humanas e Sociais Aplicadas. 15, 1.
- [7] de Caldas Rabelo, J., dos Santos, N., de Caldas Rabelo, K. C., Furquim, M. G. D., and de Sousa Júnior, J. C. 2022. Análise comparativa entre os requisitos técnicos de certificação de unidades armazenadoras de grãos com a ciência do direito: uma abordagem bibliográfica. Research, Society and Development. 11, 1, e37111125031–e37111125031.
- [8] Correia, F. G. 2021. CERTIFICAÇÃO DE UNIDADE ARMAZENADORA DE GRÃOS NO ESTADO DO MARANHÃO – ESTUDO DE CASO. Trabalho de Conclusão de Curso (Graduação em Engenharia de Alimentos). Universidade Federal da Paraíba, João Pessoa.
- [9] Rasêra, I. T. 2007. Classificação de soja. Trabalhador na classificação de produtos de origem vegetal. SENAR-PR, Curitiba.
- [10] Classific. 2025. Classific: Classificação e Negociação de Grãos. Obtido em 7 de novembro de 2025 de <https://classific.com.br/>.
- [11] Casadei, A. 2023. Corngrain: base de dados e classificação automática de grãos de milho. Monografia (MBA em Inteligência Artificial e Big Data). Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos.
- [12] Ribeiro, A. V. 2015. Desenvolvimento de um sistema para classificação de grãos de milho em dispositivos móveis. Dissertação de Mestrado. Instituto de Informática, Universidade Federal de Goiás, Goiânia.
- [13] Tillmann, C. A. 2006. Modelo de sistema integrado de gestão da qualidade para a implantação nas unidades de beneficiamento de sementes. Trabalho de Pós-Graduação. Universidade Federal de Pelotas.