UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO DE JOINVILLE
CURSO DE ENGENHARIA MECATRÔNICA

BERNARDO HOTT ROCHA

DESENVOLVIMENTO DE UM PLUGIN DE CONCATENAÇÃO DE SENTIMENTOS DISCENTES NO AMBIENTE VIRTUAL MOODLE

Joinville
2026

BERNARDO HOTT ROCHA

DESENVOLVIMENTO DE UM PLUGIN DE CONCATENAÇÃO DE SENTIMENTOS DISCENTES NO AMBIENTE VIRTUAL MOODLE

Trabalho apresentado como requisito parcial para obtenção do título de bacharel em Engenharia Mecatrônica, no Curso de Engenharia Mecatrônica, do Centro Tecnológico de Joinville, da Universidade Federal de Santa Catarina.

Orientador(a): Prof. Dr. Ricardo José Pfitscher

Joinville
2026

AGRADECIMENTOS

Agradeço, principalmente, aos meus pais, Célio e Adriana, e à minha irmã, Rafaela, que me deram suporte em todos os momentos, tanto financeiramente quanto como base na qual pude me apoiar para sobreviver aos dias longe de casa, lidando com as nuances do mundo.

Também agradeço aos meus demais familiares, que, em meus retornos para casa ao fim de cada semestre, me receberam de braços abertos, acolhendo minhas reclamações e lamentos e me mandando de volta recarregado de amor e carinho.

Ao meu orientador, professor Ricardo José Pfitscher, pela paciência, cobrança e compromisso com a minha formação e com a conclusão deste trabalho. Mesmo quando eu esperava um puxão de orelha, sempre fui recebido com preocupação e incentivo.

Aos meus amigos do Coluni, Topfílicos, que me permitiram chegar à graduação acreditando que o mundo pode ser um lugar melhor e que o amor contido em uma amizade transcende estados e sobrevive ao tempo.

Às equipes e entidades que me acolheram, dando-me um canto para descansar entre as aulas e fazendo com que toda essa caminhada tivesse muito mais graça e brilho. À BotCEM, por me dar base e experiência com tecnologia, além de me fazer perceber que os esquisitos não estão sozinhos. Ao Rotaract, por me preparar para liderar quando eu nunca achei que fosse capaz, e por confirmar que fazer o bem é vocação. Ao Diretório Acadêmico, por me mostrar que algumas lutas não podem ser evitadas e que a união faz a nossa força. À Bateria Nota Cem, por mostrar que mesmo alguém com meio ouvido, dois braços esquerdos e memória de peixe pode aprender um pouco de música, além de demonstrar que, dentro dos maiores profissionais, há sempre um noia.

Aos amigos que fiz em Joinville, que viveram cada reclamaçao, cada lágrima, cada sucesso e cada sorriso ao meu lado. Que me deram forças para lutar mais um dia e um ombro amigo para desabar sempre que precisei.

Ao George Manske e ao Rodrigo Knabach, por me darem a oportunidade de dar meus primeiros passos no mercado de trabalho e pelos ensinamentos que levarei comigo para minha vida profissional. E a Pedro Moura, pelo eterno lembrete de que estou ficando velho.

Por fim, agradeço a todas as histórias, sejam elas fofocas, aventuras de RPG, livros que li, séries e filmes a que assisti, jogos que joguei, histórias das pessoas que conheci e histórias de pessoas que não poderei mais conhecer. Foi por causa de vocês que sigo vendo beleza e sentido no amanhã.

RESUMO

A evasão no ensino superior brasileiro constitui um problema recorrente e demanda estratégias capazes de apoiar a identificação de dificuldades acadêmicas e emocionais dos estudantes. Nesse contexto, este trabalho propõe o desenvolvimento de um plugin local para a plataforma Moodle voltado à coleta, importação, classificação e síntese de sentimentos discentes. A ferramenta, implementada em PHP e integrada à arquitetura do Moodle, permite que estudantes respondam a formulários textuais, que bases externas previamente classificadas sejam importadas e que os resultados sejam apresentados ao docente por meio de um painel de análise. O sistema utiliza técnicas de Processamento de Linguagem Natural para pré-processamento textual, geração de nuvem de palavras e organização das respostas, além de integração com uma API externa de inteligência artificial para classificação de sentimentos e produção de resumos consolidados da turma. A proposta busca transformar respostas abertas em informações sintéticas e úteis ao professor, contribuindo para a compreensão do clima da disciplina e para o apoio a intervenções pedagógicas.

Keywords: Análise de Sentimentos; Moodle; Processamento de Linguagem Natural; Inteligência Artificial; Permanência Estudantil.

ABSTRACT

Student dropout in Brazilian higher education is a recurring problem and requires strategies capable of supporting the identification of academic and emotional difficulties faced by students. In this context, this work proposes the development of a local plugin for the Moodle platform aimed at collecting, importing, classifying and synthesizing students' sentiments. The tool, implemented in PHP and integrated into Moodle's architecture, allows students to answer textual forms, enables the import of previously classified external datasets and presents the results to teachers through an analysis dashboard. The system uses Natural Language Processing techniques for textual preprocessing, word cloud generation and response organization, in addition to integration with an external artificial intelligence API for sentiment classification and the production of consolidated class summaries. The proposal aims to transform open-ended responses into synthetic and useful information for teachers, contributing to the understanding of the course climate and supporting pedagogical interventions.

Keywords: Sentiment analysis; Moodle; Natural Language Processing; Artificial Intelligence; Student retention.

LISTA DE FIGURAS

Figura 1 – Fluxo metodológico proposto para o desenvolvimento do plugin. . . 18
Figura 2 – Visão geral da solução proposta. 22
Figura 3 – Diagrama de casos de uso do plugin. 24
Figura 4 – Diagrama de arquitetura da solução. 25
Figura 5 – Fluxo de estados das respostas no plugin. 26
Figura 6 – Modelo de dados simplificado do plugin. 28
Figura 7 – Interface de coleta de resposta discente. 30
Figura 8 – Tela de importação de bases externas. 31
Figura 9 – Fluxo de promoção dos registros importados. 32
Figura 10 – Fluxo de comunicação entre o plugin Moodle e a API externa de IA. 34
Figura 11 – Resumo consolidado apresentado ao professor. 36
Figura 12 – Nuvem de palavras gerada a partir das respostas discentes. 37
Figura 13 – Painel docente do plugin *analise_sentimentos* parte superior. 38
Figura 14 – Painel docente do plugin *analise_sentimentos* parte inferior. 39
Figura 15 – 540 Registros importados e utilizados como respostas no plugin. . . 44

LISTA DE QUADROS

Quadro 1 – Requisitos funcionais da solução. 23
Quadro 2 – Requisitos não funcionais da solução. 23
Quadro 3 – Estrutura simplificada de arquivos do plugin. 27
Quadro 4 – Síntese das tabelas do plugin. 28
Quadro 5 – Exemplo de configuração de colunas para a base *sentimentos540*. 31
Quadro 6 – Prompt utilizado para classificação de sentimentos. 33
Quadro 7 – Configuração conceitual da integração com IA. 34
Quadro 8 – Prompt utilizado para geração do resumo consolidado. 35
Quadro 9 – Síntese dos testes funcionais do plugin. 43
Quadro 10 – Estimativa de custo por ciclo para uso da API em cenário institucional. 46

LISTA DE ABREVIATURAS E SIGLAS

API	Application Programming Interface
CSV	Comma-Separated Values
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IA	Inteligência Artificial
JSON	JavaScript Object Notation
LLM	Large Language Model
LSTM	Long Short-Term Memory
Moodle	Modular Object-Oriented Dynamic Learning Environment
PLN	Processamento de Linguagem Natural
PHP	Hypertext Preprocessor
UML	Unified Modeling Language
URL	Uniform Resource Locator

SUMÁRIO

1	**INTRODUÇÃO**	**10**
1.1	OBJETIVOS	11
1.1.1	**Objetivo geral**	**11**
1.1.2	**Objetivos específicos**	**12**
2	**FUNDAMENTAÇÃO TEÓRICA**	**13**
2.1	MACHINE LEARNING	13
2.2	ANÁLISE DE SENTIMENTOS	14
2.3	PROCESSAMENTO DE LINGUAGEM NATURAL	14
2.4	GRANDES MODELOS DE LINGUAGEM E APIS DE IA	15
2.5	MOODLE	15
2.6	TRABALHOS RELACIONADOS	16
3	**METODOLOGIA**	**18**
3.1	USO DE INTELIGÊNCIA ARTIFICIAL COMO APOIO AO DESENVOLVIMENTO	19
4	**DESENVOLVIMENTO DO PLUGIN *ANALISE_SENTIMENTOS***	**21**
4.1	VISÃO GERAL DA SOLUÇÃO	21
4.2	REQUISITOS DA SOLUÇÃO	22
4.3	MODELAGEM FORMAL DA SOLUÇÃO	24
4.4	ARQUITETURA DO PLUGIN LOCAL	26
4.5	MODELO DE DADOS	27
4.6	COLETA DE RESPOSTAS DOS ESTUDANTES	29
4.7	IMPORTAÇÃO DE BASES EXTERNAS	30
4.8	PROMOÇÃO DE REGISTROS IMPORTADOS PARA RESPOSTAS	32
4.9	CLASSIFICAÇÃO DOS SENTIMENTOS COM IA	32
4.10	GERAÇÃO DO RESUMO CONSOLIDADO	35
4.11	GERAÇÃO DA NUVEM DE PALAVRAS	36
4.12	PAINEL DO DOCENTE	38
4.13	CONTROLE DE PERMISSÕES E VISIBILIDADE	39
4.14	AMBIENTE DE DESENVOLVIMENTO E IMPLANTAÇÃO	40
4.15	DISPONIBILIZAÇÃO DO CÓDIGO-FONTE	41
5	**RESULTADOS E DISCUSSÃO**	**42**
5.1	FUNCIONAMENTO GERAL DO PLUGIN	42
5.2	IMPORTAÇÃO E USO DAS BASES DE SENTIMENTOS	43
5.3	RESUMO GERADO POR INTELIGÊNCIA ARTIFICIAL	44
5.4	ESTIMATIVA DE CUSTOS DE USO DA API	45
5.5	NUVEM DE PALAVRAS	46

5.6 LIMITAÇÕES DA VERSÃO DESENVOLVIDA 47
6 CONCLUSÃO 48
REFERÊNCIAS 50

1 INTRODUÇÃO

A evasão no ensino superior brasileiro atingiu a marca de 57,2% ao ano em 2024, considerando instituições públicas e privadas, dentro das modalidades presencial e remota, conforme dados do Mapa do Ensino Superior no Brasil 2024, publicado pelo Instituto Semesp (Semesp, 2024). Compreender as razões por trás desses índices é essencial para desenvolver estratégias eficazes de retenção estudantil. Além dos impactos individuais na trajetória acadêmica e profissional dos discentes (Carvalho; Cunha, 2023), a evasão representa um custo socioeconômico relevante para o país, com perdas anuais que superam R$ 9 bilhões em investimentos públicos (Oliveira *et al.*, 2022).

Ao longo da graduação, os discentes podem enfrentar barreiras menos visíveis, como dificuldades econômicas, questões psicossociais, problemas familiares, insegurança acadêmica e desmotivação.(Carvalho; Cunha, 2023) Embora as intervenções pedagógicas possam contribuir para reverter parte desse cenário, a identificação precoce desses sinais é desafiadora, principalmente em turmas numerosas e heterogêneas, nas quais os docentes têm dificuldade em acompanhar individualmente a percepção dos estudantes sobre a disciplina.

Estratégias vêm sendo exploradas para compreender e antecipar os fatores que levam à evasão. Além de análises baseadas em histórico acadêmico e perfil socioeconômico, estudos têm incorporado variáveis como participação, engajamento e linguagem expressa pelos estudantes em ambientes virtuais (Becker *et al.*, 2017; Alencar *et al.*, 2021; Tazi *et al.*, 2024). Técnicas de mineração de dados, aprendizado de máquina e Processamento de Linguagem Natural (PLN) têm sido aplicadas para transformar dados educacionais em indicadores capazes de apoiar ações pedagógicas mais direcionadas (Pfitscher *et al.*, 2023).

Nesse contexto, a análise de sentimentos, área voltada à identificação de opiniões, emoções e avaliações subjetivas presentes em textos, tem sido aplicada como ferramenta de apoio à compreensão do comportamento discente em ambientes virtuais de aprendizagem (Atiq; Loui, 2022). Bóbó (Bóbó *et al.*, 2022) propõem o uso de dados textuais gerados por estudantes para avaliar o clima emocional da turma, destacando o potencial da abordagem para complementar métricas tradicionais de engajamento e apoiar práticas pedagógicas mais responsivas.

Apesar dos avanços na aplicação de técnicas de análise de sentimentos e modelos preditivos para apoiar a permanência estudantil, ainda há lacunas na integração dessas análises diretamente às plataformas de ensino utilizadas cotidianamente. No contexto de disciplinas de programação, Pfitscher *et al.* (2023) investigaram a coleta ativa de sentimentos discentes por meio de questionários abertos, combinando aná-

lise psicopedagógica e classificação automatizada baseada em Processamento de Linguagem Natural. Esse tipo de abordagem evidencia o potencial dos sentimentos como informação complementar ao acompanhamento docente, mas também reforça a necessidade de ferramentas integradas ao ambiente virtual de aprendizagem, capazes de apoiar o uso desses dados durante o andamento da disciplina.

Nesse contexto, este trabalho apresenta o desenvolvimento de um plugin local para a plataforma *Modular Object-Oriented Dynamic Learning Environment* (Moodle), um sistema de gestão de aprendizagem de código aberto (Dougiamas; Taylor, 2003). Plugins são componentes que adicionam funcionalidades a um sistema sem comprometer sua modularidade (Ribeiro; Silva, 2021). A ferramenta proposta, denominada *analise_sentimentos*, foi desenvolvida para coletar respostas textuais dos estudantes, importar bases externas de sentimentos discentes, classificar respostas com apoio de inteligência artificial e disponibilizar ao docente um painel com resumo consolidado e nuvem de palavras.

A formulação da versão desenvolvida concentra-se em uma solução funcional de apoio à interpretação docente. Assim, o plugin não substitui a avaliação pedagógica nem realiza diagnóstico individual de evasão. Seu objetivo é organizar e sintetizar respostas abertas, oferecendo indícios qualitativos sobre o clima da turma e possíveis pontos de atenção para o professor.

Os resultados são consolidados em um painel de análise acessível ao docente dentro do Moodle. Nesse painel, o professor pode visualizar classificações, resumo textual gerado por inteligência artificial, registros importados de bases externas e uma nuvem de palavras construída a partir dos termos mais recorrentes nas respostas. Essa abordagem busca reduzir o esforço de leitura manual de grandes volumes de comentários e transformar dados textuais dispersos em informação pedagógica sintética.

1.1 OBJETIVOS

Para apoiar o acompanhamento das percepções discentes em ambientes virtuais de aprendizagem, propõem-se os seguintes objetivos.

1.1.1 Objetivo geral

Desenvolver um plugin local para a plataforma Moodle capaz de coletar, importar, processar e sintetizar sentimentos expressos por estudantes em respostas textuais, fornecendo ao docente um painel de apoio à interpretação do clima da turma e de possíveis pontos de atenção pedagógica.

1.1.2 Objetivos específicos

- Implementar um formulário próprio no Moodle para coleta de respostas textuais dos estudantes;
- Desenvolver mecanismo de importação de bases externas de sentimentos discentes previamente classificadas;
- Integrar o plugin a uma API externa de inteligência artificial para classificação de respostas e geração de resumo consolidado;
- Disponibilizar ao professor um painel de análise contendo respostas, classificações, resumo textual e nuvem de palavras;

2 FUNDAMENTAÇÃO TEÓRICA

O avanço das tecnologias educacionais e a popularização dos ambientes virtuais de aprendizagem tornaram possível a coleta e análise de grandes volumes de dados gerados pelos estudantes, como mensagens, respostas abertas, entregas de atividades e interações em plataformas digitais (Becker *et al.*, 2017). No entanto, a simples disponibilidade desses dados não garante sua utilização pedagógica efetiva. Diante da elevada taxa de evasão no ensino superior brasileiro, torna-se relevante explorar abordagens que permitam aos docentes compreender o estado emocional e perceptivo de suas turmas de forma automatizada, contínua e integrada às plataformas já utilizadas (Oliveira *et al.*, 2022).

Em Simionescu *et al.* (Simionescu *et al.*, 2024), é demonstrado o potencial do Processamento de Linguagem Natural e da análise de sentimentos para transformar dados textuais não estruturados em informações significativas sobre o comportamento discente. Este trabalho fundamenta-se na interseção entre PLN, análise de sentimentos, inteligência artificial e educação, demonstrando a viabilidade técnica de transformar respostas textuais em indicadores úteis para acompanhamento pedagógico (Alencar *et al.*, 2021).

Para compreender a proposta deste trabalho, é necessário apresentar alguns conceitos fundamentais que sustentam o desenvolvimento do plugin. Inicialmente, discute-se o aprendizado de máquina, pois parte das abordagens de classificação automática de sentimentos depende da identificação de padrões em dados textuais. Em seguida, apresenta-se a análise de sentimentos, responsável por interpretar polaridades, opiniões e percepções expressas em linguagem natural. Também são abordados o Processamento de Linguagem Natural, empregado na limpeza, organização e representação dos textos; os grandes modelos de linguagem e APIs de IA, utilizados na classificação e na geração de resumos; e a plataforma Moodle, ambiente no qual a solução foi implementada. Por fim, são discutidos trabalhos relacionados que aproximam sentimentos discentes, ambientes virtuais de aprendizagem e permanência estudantil.

2.1 MACHINE LEARNING

Machine Learning, ou aprendizado de máquina, é uma subárea da inteligência artificial que permite que sistemas aprendam padrões a partir de dados, realizando previsões ou classificações sem serem explicitamente programados para cada situação específica. Essa abordagem é fundamental em diversas aplicações, como análise de sentimentos, em que o sistema pode detectar automaticamente polaridades ou categorias emocionais em textos (Alencar *et al.*, 2021). Pode-se subdividir o aprendizado

de máquina em quatro tipos principais: aprendizado supervisionado, aprendizado não supervisionado, aprendizado semissupervisionado e aprendizado por reforço (Alpaydin, 2020).

O aprendizado supervisionado utiliza pares de entrada e saída para gerar uma função que relacione ambos, ajustando os parâmetros do modelo com base na diferença entre a saída esperada e a obtida. Algoritmos como Naive Bayes, regressão logística e máquinas de vetor de suporte são exemplos amplamente utilizados. No contexto educacional, classificadores supervisionados podem ser treinados com sentimentos previamente rotulados, permitindo identificar padrões associados à motivação, dificuldade ou frustração dos estudantes.

O aprendizado não supervisionado analisa dados de entrada sem rótulos e busca padrões ou agrupamentos com base em características internas. Técnicas como *k-means*, análise de componentes principais e *autoencoders* são comumente empregadas (Alpaydin, 2020). No contexto da análise de sentimentos, esses métodos podem revelar grupos de respostas com vocabulários ou perfis emocionais semelhantes, mesmo sem uma definição prévia de classe.

2.2 ANÁLISE DE SENTIMENTOS

A análise de sentimentos destaca-se como uma vertente do Processamento de Linguagem Natural voltada à identificação de emoções, opiniões e avaliações subjetivas presentes em textos. Utilizando abordagens léxicas, estatísticas ou baseadas em modelos de linguagem, é possível classificar automaticamente textos como positivos, negativos, neutros ou ambíguos, além de identificar temas recorrentes associados a experiências dos usuários (Wankhade *et al.*, 2022).

No contexto educacional, a análise de sentimentos tem sido aplicada para interpretar respostas de estudantes, postagens em fóruns, comentários em ambientes virtuais e registros de interação com plataformas de aprendizagem. Essa leitura automatizada pode complementar informações tradicionais, como notas e frequência, oferecendo ao professor uma visão mais qualitativa sobre a experiência da turma (Tazi *et al.*, 2024; Baessa *et al.*, 2024).

2.3 PROCESSAMENTO DE LINGUAGEM NATURAL

O Processamento de Linguagem Natural é uma subárea da inteligência artificial voltada para o estudo e desenvolvimento de técnicas computacionais capazes de interpretar, compreender e gerar linguagem humana. O PLN busca construir sistemas que processem a linguagem de maneira estruturada, utilizando abordagens que vão desde modelos baseados em regras linguísticas até métodos estatísticos, redes

neurais profundas e grandes modelos de linguagem (Jurafsky; Martin, 2023; Caseli; Nunes, 2024).

Além disso, o PLN se destaca por transformar grandes volumes de dados textuais não estruturados em informações úteis. Isso é possível por meio de técnicas como tokenização, normalização, remoção de palavras irrelevantes, análise sintática e classificação semântica. No contexto deste trabalho, essas técnicas são utilizadas principalmente para limpeza das respostas textuais, cálculo de frequência de termos e apoio à geração de nuvem de palavras.

2.4 GRANDES MODELOS DE LINGUAGEM E APIS DE IA

Grandes modelos de linguagem, ou *Large Language Models* (LLMs), são modelos neurais treinados em grandes volumes de texto e capazes de realizar tarefas como classificação, resumo, geração textual, extração de informações e resposta a instruções em linguagem natural. A literatura recente descreve esses modelos como sistemas geralmente baseados na arquitetura *Transformer*, treinados em larga escala e posteriormente ajustados para melhorar sua capacidade de seguir instruções, alinhar respostas e executar tarefas em diferentes domínios (Zhao *et al.*, 2026).

Além do treinamento em larga escala, o uso prático desses modelos depende de estratégias de utilização, como elaboração de prompts, aprendizagem em contexto e integração com aplicações externas. Nesse cenário, o prompt assume papel relevante, pois define a tarefa, o formato esperado da resposta e as restrições que o modelo deve seguir. No contexto deste trabalho, os prompts foram empregados para orientar a classificação de sentimentos e a geração de resumos consolidados a partir das respostas discentes.

A adoção de uma API externa simplifica a implementação local, pois evita a necessidade de treinar ou hospedar modelos de linguagem no servidor Moodle. Por outro lado, essa escolha exige cuidados com privacidade, anonimização, custos de uso, tratamento de erros e dependência de conexão com o serviço externo.

2.5 MOODLE

Dentre os ambientes virtuais de aprendizagem mais utilizados, destaca-se o Moodle, um sistema de gerenciamento de cursos online de código aberto, amplamente adotado por instituições educacionais. O Moodle foi concebido com foco na criação de comunidades de aprendizagem e no apoio a práticas pedagógicas mediadas por tecnologia, oferecendo recursos para organização de conteúdos, comunicação, atividades avaliativas, acompanhamento de estudantes e gestão de cursos (Dougiamas; Taylor, 2003; Moodle, 2026). Por reunir interações, respostas, registros de participação e dados acadêmicos em um único ambiente, a plataforma também se torna

um espaço relevante para a aplicação de técnicas de análise de dados educacionais, Processamento de Linguagem Natural e análise de sentimentos.

Uma característica importante do Moodle é sua arquitetura modular, que permite ampliar suas funcionalidades por meio de plugins. Esses componentes adicionam recursos à plataforma sem exigir alterações diretas em seu núcleo, favorecendo a manutenção, a reutilização e a adaptação do ambiente a necessidades específicas de cada instituição ou projeto. (Moodle Developer Resources, 2026c).

O desenvolvimento de plugins no Moodle segue convenções próprias de organização. Em geral, cada componente possui uma estrutura de diretórios e arquivos padronizados, incluindo arquivos de versão, configurações, permissões, strings de idioma, scripts de instalação ou atualização do banco de dados e classes responsáveis por formulários e serviços internos (Moodle Developer Resources, 2026a). Além disso, o acesso ao banco de dados deve seguir as funções e rucursos fornecidas pela própria plataforma, de modo a preservar compatibilidade com diferentes sistemas gerenciadores de banco de dados e manter a integração com os mecanismos internos do Moodle (Moodle Developer Resources, 2026b).

Neste trabalho, optou-se pelo desenvolvimento de um plugin local, identificado pelo componente *analise_sentimentos*, por se tratar de uma solução que exige funcionalidades próprias e integradas ao ambiente Moodle. Essa escolha justifica-se porque a ferramenta não se limita à exibição de informações em uma região da interface, como ocorreria em um bloco, nem corresponde a uma atividade pedagógica convencional, como um fórum, questionário ou tarefa. O escopo do sistema envolve diferentes rotinas, incluindo páginas próprias de coleta, importação de bases externas, análise das respostas, configuração de parâmetros, processamento textual, comunicação com API externa e apresentação dos resultados em um painel docente. Dessa forma, o formato de plugin local permite centralizar essas funcionalidades em um único componente, mantendo a modularidade da plataforma e evitando alterações diretas no núcleo do Moodle.

2.6 TRABALHOS RELACIONADOS

A análise de sentimentos tem sido cada vez mais aplicada como estratégia para apoiar a permanência estudantil em cursos de graduação e ambientes virtuais de aprendizagem. Alencar (Alencar *et al.*, 2021) desenvolveram o framework SentiEduc, um sistema multiagente que realiza coleta, tradução e análise de sentimentos em textos discentes publicados no Moodle, incluindo fóruns, chats, mensagens e diários. A arquitetura do sistema integra tecnologias como SenticNet e JADE, além de fornecer ao professor gráficos e nuvens de palavras com o panorama emocional da turma.

De forma semelhante, Tazi (Tazi *et al.*, 2024) propuseram um modelo de predição de evasão em MOOCs baseado na análise dos sentimentos expressos por alunos

em fóruns de discussão. O estudo utilizou técnicas de PLN para pré-processamento textual e regressão logística para prever abandono, reforçando a importância do monitoramento das interações textuais para ações educacionais preventivas.

No cenário brasileiro, Pfitscher (Pfitscher *et al.*, 2023) desenvolveram uma abordagem baseada na coleta ativa de sentimentos por meio de questionários abertos aplicados em turmas de programação. Os textos foram classificados manualmente por especialistas da área psicopedagógica e também por modelos automatizados, indicando que a análise de sentimentos pode ser uma ferramenta complementar para apoiar a permanência estudantil.

Baessa (Baessa *et al.*, 2024) propuseram uma metodologia que combina sentimentos e variáveis de desempenho acadêmico, como notas, frequência e participação no Moodle, para a predição de evasão em turmas de programação. Os resultados demonstraram que a inclusão de sentimentos pode melhorar o desempenho de modelos preditivos.

Moreira (Moreira *et al.*, 2024) compararam diferentes ferramentas automatizadas de análise de sentimentos aplicadas a 540 respostas discentes coletadas por questionário. O estudo avaliou modelos baseados em LLMs, como LLaMA e Gemma, e o analisador léxico LeIA, evidenciando o potencial de modelos de linguagem para identificar classes negativas e ambíguas.

Diferentemente das propostas anteriores, este trabalho propõe o desenvolvimento de um plugin local para Moodle que reúne coleta, importação, classificação, síntese e visualização de sentimentos discentes em um painel docente. Outra diferença importante está na forma de coleta: enquanto trabalhos como os de Alencar *et al.* (2021) e Tazi *et al.* (2024) exploram textos produzidos em fóruns, chats e outros espaços de interação, a solução proposta adota um formulário de resposta ativa, no qual os estudantes são instigados a registrar explicitamente como se sentem em relação à disciplina. Essa escolha aproxima o plugin da metodologia de coleta ativa discutida por Pfitscher *et al.* (2023), ao mesmo tempo em que busca integrar esse processo diretamente ao Moodle. Em vez de priorizar a predição individual de evasão, a ferramenta busca apoiar a leitura coletiva da turma por meio de resumos e indicadores textuais interpretáveis.

3 METODOLOGIA

A presente pesquisa teve como objetivo desenvolver e avaliar funcionalmente um plugin local para a plataforma Moodle capaz de realizar a coleta, importação, classificação e síntese de sentimentos expressos por estudantes em respostas textuais. Para isso, foram combinadas técnicas de Processamento de Linguagem Natural, integração com uma API externa de inteligência artificial e recursos nativos da arquitetura Moodle, como páginas locais, banco de dados, permissões e configurações administrativas.

Como ponto de partida, foram utilizadas bases de dados associadas aos trabalhos de Pfitscher (Pfitscher *et al.*, 2023) e Moreira (Moreira *et al.*, 2024), compostas por respostas discentes a perguntas abertas em contexto educacional. Essas bases contêm classificações humanas e classificações automatizadas produzidas por ferramentas como Gemma, LLaMA e LeIA. A escolha dessas bases se justifica pela proximidade com o contexto deste trabalho, pois os dados são oriundos de estudantes em disciplinas de programação e foram organizados para investigar sentimentos relacionados à permanência estudantil.

A implementação foi conduzida em ambiente local de desenvolvimento Moodle, utilizando Docker para execução dos serviços web e banco de dados. O plugin foi desenvolvido em PHP como um componente local, com o nome *analise_sentimentos*. Sua estrutura inclui páginas para resposta discente, painel do professor, importação de bases externas, promoção de registros importados para respostas analisáveis, geração de resumo por IA e atualização da nuvem de palavras.

O fluxo metodológico adotado pode ser dividido em seis etapas: 1º levantamento teórico, 2º definição da arquitetura do plugin, 3º implementação da coleta e armazenamento das respostas, 4º desenvolvimento da importação de bases externas, 5º integração com serviço de inteligência artificial e 6º construção das visualizações docentes. A Figura 1 apresenta esse fluxo de forma esquemática.

Figura 1 – Fluxo metodológico proposto para o desenvolvimento do plugin.



Fonte: Elaborado pelo autor com apoio de ferramenta de IA Gemini.

A coleta de dados no plugin ocorre por meio de formulário próprio, no qual o estudante registra uma resposta textual sobre sua percepção em relação à disciplina. Além disso, o sistema permite importar bases externas em formato tabular, associando colunas de texto e classificações previamente existentes. Após importados, os registros podem ser promovidos para a tabela principal de respostas, possibilitando seu uso nos mesmos processos aplicados às respostas coletadas diretamente no Moodle.

O pré-processamento textual inclui normalização para minúsculas, remoção de pontuação, tratamento de acentos e filtragem de palavras pouco informativas. Esse processamento é utilizado principalmente para geração de frequências e construção da nuvem de palavras. A classificação dos sentimentos e a geração dos resumos são realizadas por meio de uma API externa de inteligência artificial, chamada pelo servidor Moodle a partir de uma chave configurada pelo administrador do sistema.

A validação nesta etapa possui caráter funcional e exploratório. Para isso, verificou-se o funcionamento das páginas do plugin, a gravação dos dados no banco, a importação de bases externas, a geração de classificações, a produção de resumo textual e a atualização da nuvem de palavras. Assim, a avaliação concentra-se na viabilidade técnica da ferramenta e na coerência dos artefatos produzidos, deixando uma validação pedagógica com docentes e estudantes como etapa futura.

3.1 USO DE INTELIGÊNCIA ARTIFICIAL COMO APOIO AO DESENVOLVIMENTO

Durante a elaboração deste trabalho, foram utilizadas ferramentas de inteligência artificial generativa como apoio auxiliar ao processo de desenvolvimento do código, escrita e revisão acadêmica. As principais ferramentas empregadas foram o ChatGPT, na versão/modelo GPT-5.5 Thinking, e o Google Gemini, na versão/modelo Gemini 3 Flash. O uso dessas ferramentas ocorreu em atividades de revisão textual, identificação de inconsistências argumentativas, correção gramatical, aprimoramento da clareza do texto, geração de imagens e apoio na interpretação de erros.

O chat GPT-5.5 foi utilizada como ferramenta empregada para auxiliar na interpretação de erros durante o uso do Docker, na análise de problemas de inicialização do Moodle e na organização de possíveis soluções para falhas encontradas no ambiente local de desenvolvimento.

O Google Gemini foi utilizado como ferramenta auxiliar para geração de algumas imagens e diagramas inseridos no trabalho, especialmente representações visuais relacionadas ao fluxo metodológico, arquitetura, modelagem e comunicação entre componentes. As imagens geradas foram revisadas e selecionadas pelo autor, considerando sua aderência ao conteúdo técnico descrito no texto.

Ressalta-se que as ferramentas de inteligência artificial não substituíram a autoria, a implementação do plugin, a definição do escopo, a análise dos resultados ou a responsabilidade técnica sobre o trabalho. As sugestões geradas foram avaliadas,

filtradas e adaptadas pelo autor, cabendo a este a decisão final sobre o conteúdo incluído no texto. Portanto, a IA foi utilizada como instrumento de apoio ao desenvolvimento, à revisão e à organização do trabalho, e não como fonte primária de dados, validação experimental ou tomada autônoma de decisões metodológicas.

4 DESENVOLVIMENTO DO PLUGIN *ANALISE_SENTIMENTOS*

Este capítulo apresenta o desenvolvimento do plugin *analise_sentimentos*, descrevendo sua concepção, requisitos, modelagem, arquitetura, estrutura de dados e principais funcionalidades. São detalhadas as etapas de coleta de respostas, importação de bases externas, promoção de registros, classificação de sentimentos com apoio de IA, geração de resumo consolidado, construção da nuvem de palavras, painel docente, controle de permissões e ambiente de implantação. O objetivo é demonstrar como a proposta foi transformada em uma solução funcional integrada ao Moodle.

4.1 VISÃO GERAL DA SOLUÇÃO

O plugin desenvolvido tem como finalidade apoiar o docente na compreensão das percepções e sentimentos manifestados pela turma em relação a uma disciplina específica. Essa proposta se apoia na aplicação da análise de sentimentos em contextos educacionais, área que busca interpretar opiniões, emoções e avaliações subjetivas presentes em textos produzidos por estudantes (Atiq; Loui, 2022; Wankhade *et al.*, 2022). Para isso, a solução permite a coleta de relatos textuais, a classificação individual dessas respostas por meio de uma API externa de inteligência artificial e a geração de um resumo consolidado, capaz de sintetizar os principais aspectos percebidos nas respostas.

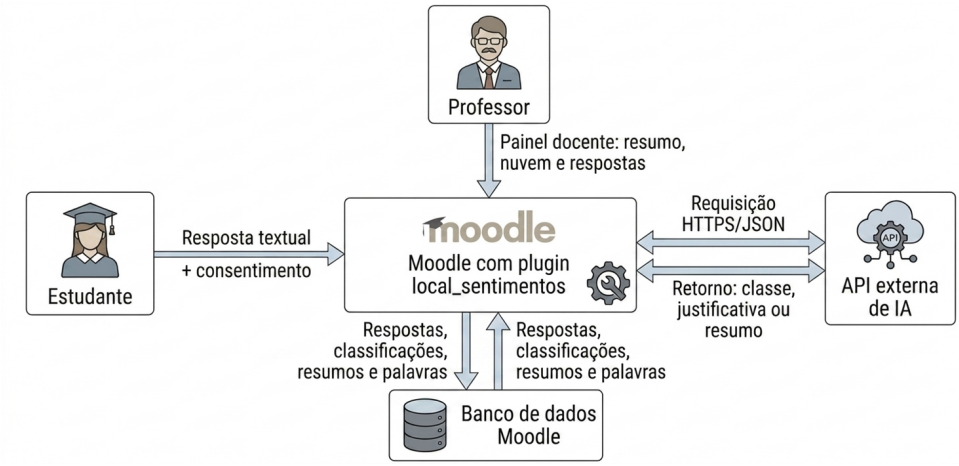
Além do resumo textual, a ferramenta disponibiliza uma nuvem de palavras, construída a partir dos termos mais recorrentes, oferecendo uma visualização complementar dos temas predominantes. Recursos desse tipo já foram explorados em trabalhos voltados à análise de sentimentos em ambientes virtuais de aprendizagem, como forma de apresentar ao professor uma visão geral do clima emocional da turma (Alencar *et al.*, 2021).

Com o uso recorrente ao longo do semestre, o professor passa a ter a possibilidade de acompanhar mudanças na percepção da turma em relação à disciplina, aos conteúdos ministrados, às avaliações e a outros fatores que possam impactar a experiência acadêmica dos estudantes. Dessa forma, o plugin busca transformar respostas abertas, muitas vezes dispersas e difíceis de analisar manualmente, em informações organizadas e de fácil interpretação. Essa abordagem dialoga com a coleta ativa de sentimentos em disciplinas de programação, explorada por Pfitscher *et al.* (2023), na qual respostas abertas foram utilizadas para compreender percepções discentes relacionadas à permanência estudantil.

A organização geral da solução proposta está representada na Figura 2, que ilustra a relação entre estudante, professor, Moodle, banco de dados e API externa de

IA.

Figura 2 – Visão geral da solução proposta.



Fonte: Elaborado pelo autor com apoio de ferramenta de IA Gemini.

É importante destacar que a ferramenta não tem como objetivo substituir a avaliação pedagógica do docente nem realizar diagnósticos individuais sobre evasão ou desempenho acadêmico. Seu propósito é facilitar a comunicação entre estudantes e professor, fornecendo indícios qualitativos sobre o sentimento geral da turma e possíveis pontos de atenção. Assim, cabe ao docente interpretar os resultados e decidir se alterações metodológicas, intervenções pedagógicas ou novas estratégias de acompanhamento são necessárias.

4.2 REQUISITOS DA SOLUÇÃO

A definição dos requisitos da solução teve como objetivo organizar as funcionalidades esperadas do plugin e estabelecer os limites técnicos da versão desenvolvida. Essa etapa é importante para relacionar o problema identificado, os objetivos do trabalho e os recursos efetivamente implementados no ambiente Moodle. Considerando a natureza modular da plataforma, os requisitos foram divididos em requisitos funcionais, associados às ações executadas pelo sistema, e requisitos não funcionais, relacionados a aspectos de segurança, integração, usabilidade e manutenção (Dougiamas; Taylor, 2003; Ribeiro; Silva, 2021).

Os requisitos funcionais representam as operações essenciais do plugin, abrangendo desde a coleta de respostas discentes até a apresentação dos resultados ao docente. Já os requisitos não funcionais estabelecem condições necessárias para que a ferramenta opere de forma adequada no ambiente virtual de aprendizagem, respeitando permissões, anonimização, organização dos dados e dependência de serviços externos.

O Quadro 1 apresenta os principais requisitos funcionais definidos para a solução.

Quadro 1 – Requisitos funcionais da solução.

Código	**Descrição**
RF01	Permitir que estudantes registrem respostas textuais por meio de formulário próprio no Moodle.
RF02	Permitir que o docente visualize respostas e classificações em um painel específico da disciplina.
RF03	Permitir a promoção de registros importados para a tabela principal de respostas analisáveis.
RF04	Classificar respostas textuais com apoio de uma API externa de inteligência artificial.
RF05	Gerar um resumo consolidado da turma a partir do conjunto de respostas disponíveis.
RF06	Gerar uma nuvem de palavras com os termos mais recorrentes nas respostas.
RF07	Permitir ao docente controlar a visibilidade do formulário e da nuvem de palavras.
RF08	Armazenar resultados processados, como classificação, justificativa, modelo utilizado, resumo e palavras da nuvem.

Fonte: Elaborado pelo autor.

O Quadro 2 apresenta os principais requisitos não funcionais considerados durante o desenvolvimento.

Quadro 2 – Requisitos não funcionais da solução.

Código	**Descrição**
RNF01	O plugin deve ser executado como componente local do Moodle, respeitando sua estrutura de diretórios, permissões e banco de dados.
RNF02	O acesso ao painel de análise deve ser restrito a usuários com perfil docente, gestor ou equivalente.
RNF03	A chave da API deve ser configurada em área administrativa, sem ser inserida diretamente no código-fonte.
RNF04	O sistema deve permitir anonimização do usuário associado às respostas, quando configurado pelo administrador.
RNF05	A comunicação com a API externa deve ocorrer por requisições HTTP, preferencialmente sobre HTTPS.
RNF06	O sistema deve registrar a origem das respostas, diferenciando registros coletados pelo formulário e registros importados.
RNF07	A solução deve apresentar tratamento básico para falhas de comunicação com a API externa.
RNF08	As visualizações apresentadas ao docente devem atuar como apoio interpretativo, sem substituir a avaliação pedagógica.

Fonte: Elaborado pelo autor.

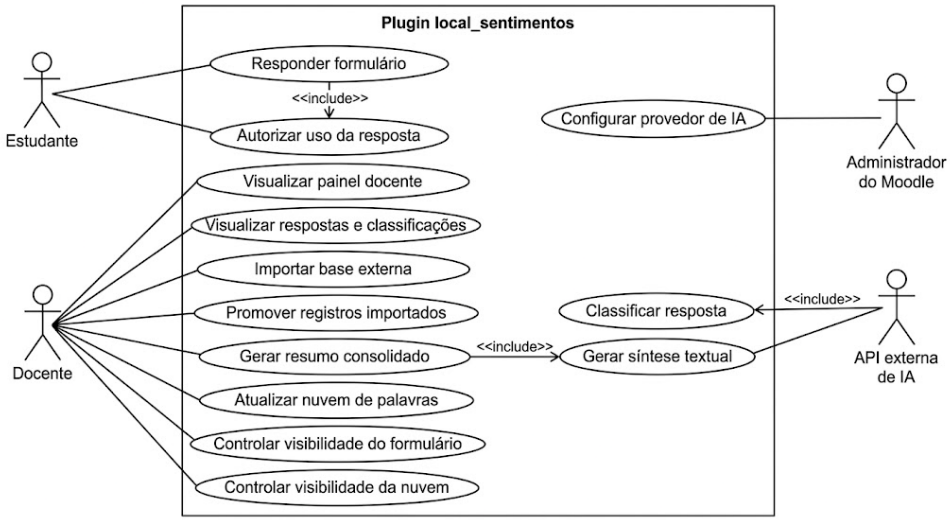
4.3 MODELAGEM FORMAL DA SOLUÇÃO

Além da descrição textual das funcionalidades, a solução foi modelada a partir de três representações principais: diagrama de casos de uso, diagrama de arquitetura e fluxo de estados da resposta. Essas representações auxiliam na compreensão das interações entre usuários, sistema Moodle, banco de dados e API externa de inteligência artificial.

O diagrama de casos de uso tem como objetivo representar os principais atores envolvidos no sistema e as funcionalidades acessadas por cada um. Na versão desenvolvida, os atores principais são o estudante, o docente, o administrador do Moodle e a API externa de IA. O estudante interage com o formulário de resposta, enquanto o docente acessa o painel, gera análises, importa bases, promove registros e controla a visibilidade dos recursos. O administrador é responsável pela configuração da API e das opções globais do plugin.

Os casos de uso podem ser organizados de acordo com os atores envolvidos. O estudante utiliza o sistema para responder ao formulário e autorizar o uso de sua resposta. O docente acessa o painel de análise, visualiza respostas e classificações, importa bases externas, promove registros importados, gera o resumo consolidado, atualiza a nuvem de palavras e controla a visibilidade do formulário e da nuvem para os estudantes. O administrador do Moodle configura o provedor de IA, incluindo chave de API, modelo e URL compatível. Por fim, a API externa de IA participa dos casos de uso relacionados à classificação de respostas e à geração de sínteses textuais.A Figura 3 apresenta o diagrama de casos de uso da solução.

Figura 3 – Diagrama de casos de uso do plugin.



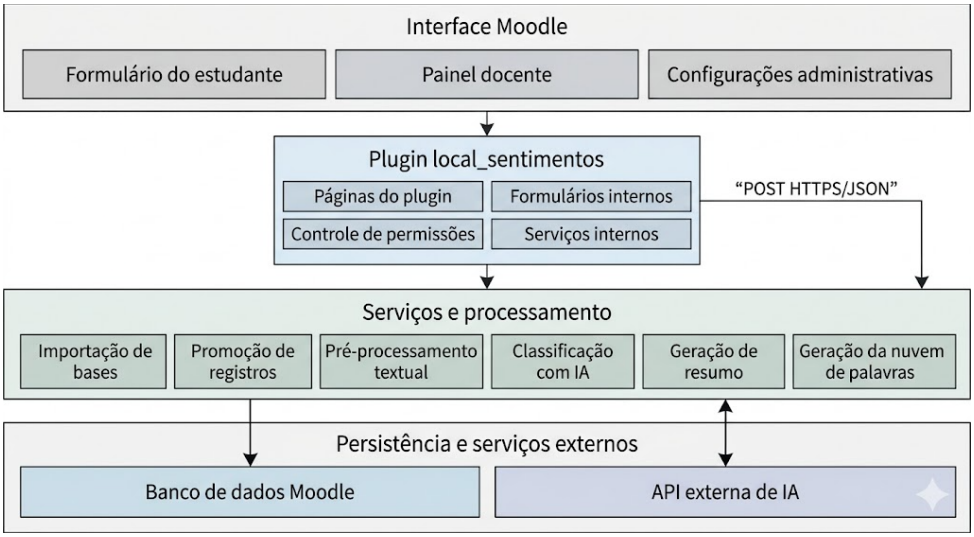
Fonte: Elaborado pelo autor com apoio de ferramenta de IA Gemini.

O diagrama de arquitetura representa a organização geral dos componentes

do sistema. A solução pode ser compreendida em quatro camadas principais: interface Moodle, serviços internos do plugin, banco de dados Moodle e serviço externo de IA. A interface Moodle reúne as páginas acessadas por estudantes, docentes e administradores. Os serviços internos concentram as regras de negócio, como processamento textual, classificação, geração de resumo, importação de bases e geração de nuvem de palavras. O banco de dados armazena respostas, bases importadas, configurações, resumos e palavras processadas. A API externa de IA é utilizada para tarefas de classificação e síntese textual.

A Figura 4 apresenta o diagrama de arquitetura da solução.

Figura 4 – Diagrama de arquitetura da solução.

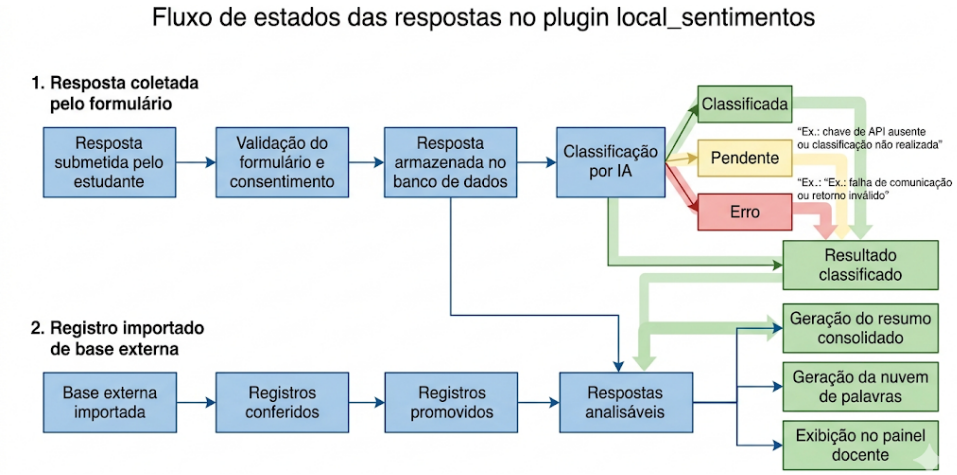


Fonte: Elaborado pelo autor com apoio de ferramenta de IA Gemini.

O fluxo de estados descreve o ciclo de vida de uma resposta dentro do plugin. Uma resposta criada pelo formulário inicia no estado de registro submetido. Após a validação do formulário e do consentimento, ela é armazenada no banco de dados. Em seguida, pode ser classificada pela API de IA, assumindo estados como classificada, pendente ou erro, conforme o retorno obtido. Posteriormente, a resposta pode ser utilizada na geração do resumo consolidado e na atualização da nuvem de palavras. Para registros importados, o fluxo começa na base externa, passa por conferência e promoção, e então segue o mesmo ciclo das respostas coletadas pelo formulário.

A Figura 5 apresenta o fluxo de estados da resposta.

Figura 5 – Fluxo de estados das respostas no plugin.



Fonte: Elaborado pelo autor com apoio de ferramenta de IA Gemini.

4.4 ARQUITETURA DO PLUGIN LOCAL

Considerando a arquitetura modular do Moodle e o escopo do trabalho, optou-se pelo desenvolvimento de um plugin local. O Moodle é um ambiente virtual de aprendizagem de código aberto projetado para apoiar comunidades de aprendizagem e permitir a expansão de suas funcionalidades por meio de componentes adicionais (Dougiamas; Taylor, 2003). A utilização de plugins é compatível com essa proposta modular, pois permite adicionar funcionalidades específicas sem alterar diretamente o núcleo da plataforma (Ribeiro; Silva, 2021).

A escolha por um plugin local também se justifica pela necessidade de concentrar funcionalidades administrativas, páginas próprias e serviços internos em um único componente. O plugin local permite criar rotinas mais amplas de apoio à plataforma. Essa característica é adequada ao escopo deste trabalho, pois a solução envolve coleta, importação, processamento, integração com API externa, armazenamento e visualização em painel próprio.

Esse tipo de componente permite criar páginas próprias, serviços internos, formulários, configurações administrativas e rotinas de acesso ao banco de dados sem a necessidade de implementar uma atividade específica ou um bloco separado. Essa escolha tornou possível centralizar as funcionalidades propostas em um único componente, identificado pelo nome *analise_sentimentos*.

A estrutura do plugin foi organizada em diferentes partes. A página de configurações permite ao administrador informar o provedor de IA, a chave da API, o modelo de linguagem utilizado e a URL da API compatível. A página de resposta discente disponibiliza o formulário pelo qual os estudantes podem registrar seus sentimentos em relação à disciplina. O painel do docente reúne as principais funcionalidades da

ferramenta, como visualização de respostas, importação de bases externas, promoção de registros importados, geração de resumo consolidado e atualização da nuvem de palavras.

O Quadro 3 resume sua estrutura principal de diretórios e arquivos.

Quadro 3 – Estrutura simplificada de arquivos do plugin.

```
local/sentimentos/
classes/form/
classes/service/
db/access.php
db/install.xml
db/upgrade.php
view.php
dashboard.php
importar_base.php
promover_base.php
atualizar_nuvem.php
settings.php
```

Fonte: Elaborado pelo autor.

Além dessas páginas principais, foram implementadas rotinas auxiliares para separar responsabilidades específicas do sistema. Entre elas, destacam-se os serviços de comunicação com a API de IA, processamento textual, importação de bases externas, geração da nuvem de palavras e controle de configurações por disciplina. Essa separação contribui para uma organização mais clara do código e facilita futuras manutenções ou expansões da ferramenta.

4.5 MODELO DE DADOS

O modelo de dados do plugin foi estruturado para separar as diferentes origens e finalidades das informações utilizadas. As respostas coletadas diretamente pelo formulário do Moodle são armazenadas em uma tabela própria, contendo informações como disciplina, usuário, resposta textual, classificação atribuída pela IA, justificativa da classificação, modelo utilizado e data de criação do registro. Essa organização é coerente com a necessidade de transformar dados educacionais textuais em registros estruturados para posterior análise, visualização e interpretação (Jurafsky; Martin, 2023).

As bases externas importadas são armazenadas inicialmente em uma tabela separada. Essa decisão permite que os registros sejam conferidos antes de serem utilizados nas análises do plugin, evitando que dados incorretos ou colunas mal associadas afetem diretamente o resumo consolidado e a nuvem de palavras. Após a verificação, esses registros podem ser promovidos para a tabela principal de respos-

tas, passando a participar dos mesmos processos aplicados às respostas coletadas pelo formulário.

O Quadro 4 apresenta uma síntese das principais tabelas utilizadas pelo plugin. Para melhorar a legibilidade, o Quadro 4 apresenta apenas o sufixo das tabelas, omitindo o prefixo comum `local_sentimentos_`.

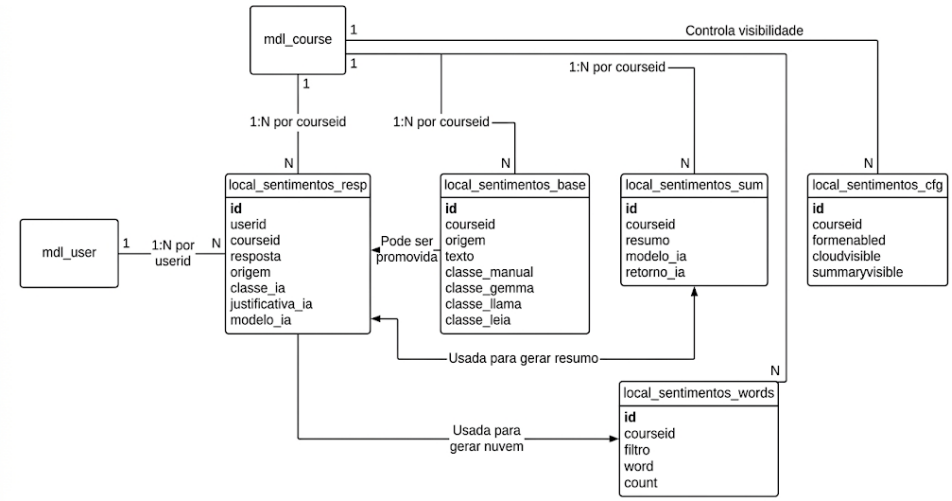
Quadro 4 – Síntese das tabelas do plugin.

Tabela	**Finalidade**	**Campos principais**
`resp`	Armazena as respostas, sejam elas coletadas pelo formulário ou promovidas de bases externas.	`id`, `userid`, `courseid`, `resposta`, `origem`, `classe_ia`, `justificativa_ia`, `modelo_ia`.
`base`	Armazena registros importados antes de sua promoção para a tabela principal de respostas.	`id`, `courseid`, `origem`, `texto`, `classe_manual`, `classe_gemma`, `classe_llama`, `classe_leia`.
`sum`	Armazena o resumo consolidado gerado por IA para cada disciplina.	`id`, `courseid`, `resumo`, `modelo_ia`, `retorno_ia`.
`cfg`	Armazena configurações específicas da disciplina, como liberação do formulário, resumo e nuvem.	`id`, `courseid`, `formenabled`, `cloudvisible`, `summaryvisible`.
`words`	Armazena os termos processados para composição da nuvem de palavras.	`id`, `courseid`, `filtro`, `word`, `count`.

Fonte: Elaborado pelo autor.

A Figura 6 apresenta uma representação simplificada do modelo de dados adotado no plugin.

Figura 6 – Modelo de dados simplificado do plugin.



Fonte: Elaborado pelo autor com apoio de ferramenta de IA Gemini.

Também foram previstas tabelas auxiliares para armazenar os resumos gerados por IA, as palavras selecionadas para compor a nuvem de palavras e as configurações específicas de cada disciplina. Dessa forma, o plugin consegue manter

informações distintas, como visibilidade da nuvem, liberação do formulário e resultados processados, sem misturá-las com as respostas originais dos estudantes.

4.6 COLETA DE RESPOSTAS DOS ESTUDANTES

Inicialmente, foi considerada a possibilidade de extrair sentimentos discentes a partir de fóruns e de outros espaços do Moodle que permitem manifestações textuais mais espontâneas. Essa alternativa encontra respaldo em trabalhos que analisam textos produzidos em fóruns, chats e outros espaços de interação em ambientes virtuais de aprendizagem (Alencar *et al.*, 2021; Tazi *et al.*, 2024). Entretanto, durante o desenvolvimento, observou-se que essa abordagem exigiria maior complexidade de integração com diferentes módulos da plataforma.

Por esse motivo, optou-se pela criação de um formulário próprio, diretamente associado às funcionalidades do plugin. Essa escolha aproxima-se da estratégia de coleta ativa de sentimentos adotada em trabalhos anteriores, nos quais os estudantes respondem a perguntas abertas para expressar suas percepções sobre a disciplina (Pfitscher *et al.*, 2023).

O formulário apresenta uma pergunta aberta ao estudante: “Como você está se sentindo em relação à disciplina?”. A resposta é registrada em um campo textual e pode ser utilizada para classificação, resumo e geração da nuvem de palavras. Além disso, foi incluído um campo de consentimento, de modo que a resposta só seja enviada caso o estudante autorize seu uso para análise dentro da ferramenta.

A anonimização das respostas foi implementada no momento do armazenamento dos registros no banco de dados. Para isso, o plugin não preserva o identificador real do estudante na tabela principal de respostas, registrando o campo `userid` com valor igual a zero. Dessa forma, a resposta permanece associada à disciplina por meio do campo `courseid`, mas deixa de manter vínculo direto com o usuário que a enviou. Essa decisão foi adotada para reduzir a exposição de dados pessoais e reforçar o caráter coletivo da análise, uma vez que o objetivo da ferramenta é apoiar a leitura do clima geral da turma, e não acompanhar individualmente estudantes. Na versão desenvolvida, a anonimização é tratada como condição obrigatória de funcionamento, não sendo prevista a desativação desse comportamento como opção de uso comum do plugin.

A interface de coleta disponibilizada ao estudante é ilustrada na Figura 7.

Figura 7 – Interface de coleta de resposta discente.



Fonte: Captura de tela do ambiente Moodle local, elaborada pelo autor.

As respostas coletadas pelo formulário são armazenadas na tabela principal do plugin e diferenciadas dos registros oriundos de bases externas. Essa separação permite identificar a origem dos dados e possibilita que o sistema trate respostas coletadas diretamente no Moodle e respostas importadas de maneira organizada.

4.7 IMPORTAÇÃO DE BASES EXTERNAS

A funcionalidade de importação de bases externas foi implementada tanto para apoiar os testes do plugin quanto para permitir o uso de respostas obtidas por outras formas de coleta. Por meio dessa função, o usuário pode inserir uma base tabular contendo respostas textuais e classificações previamente existentes, como classificações humanas ou produzidas por ferramentas automatizadas.

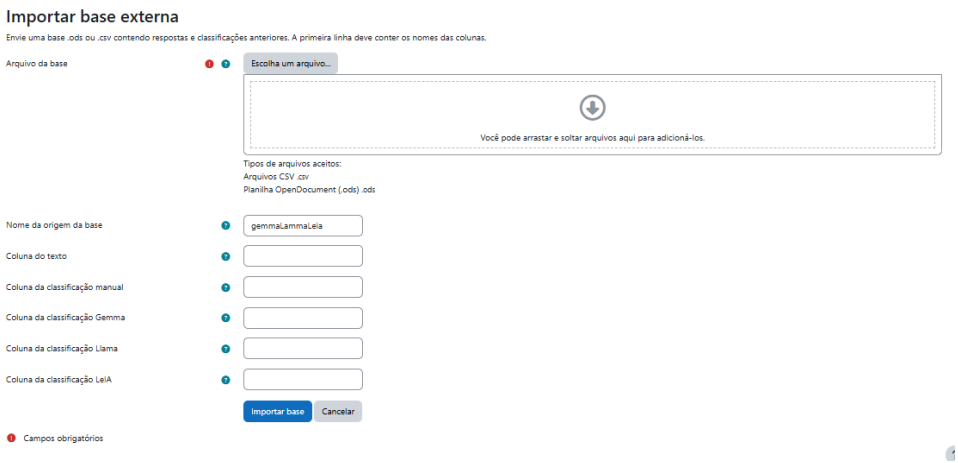
A utilização de bases externas é relevante porque permite testar a ferramenta com dados reais previamente organizados em pesquisas sobre sentimentos discentes. Neste trabalho, essa funcionalidade foi pensada a partir de bases associadas a estudos anteriores no contexto educacional, especialmente aqueles que utilizam respostas abertas classificadas por especialistas ou por diferentes ferramentas de análise de sentimentos (Pfitscher *et al.*, 2023; Moreira *et al.*, 2024).

A base utilizada nos testes foi disponibilizada em repositório público no Zenodo, permitindo sua identificação, acesso e reutilização em pesquisas relacionadas à análise de sentimentos discentes (Moreira *et al.*, 2025). No contexto deste trabalho, essa base foi empregada para testar o fluxo de importação, promoção de registros, geração de resumo e construção da nuvem de palavras.

A tela de importação permite o envio do arquivo da base e o preenchimento dos campos necessários para associar as colunas do arquivo aos campos esperados pelo plugin. Entre essas informações estão o nome da origem da base, a coluna de

texto, a coluna de classificação manual, a coluna de classificação Gemma, a coluna de classificação LLaMA e a coluna de classificação LeIA. Outras bases com menos campos, ou outros campos, também podem ser utilizadas. O não preenchimento correto faz com que o próprio código estabeleça qual coluna é referente ao tipo de informação. A Figura 8 apresenta a tela utilizada para essa etapa.

Figura 8 – Tela de importação de bases externas.



Fonte: Captura de tela do ambiente Moodle local, elaborada pelo autor.

Essa etapa é necessária porque diferentes bases podem utilizar nomes distintos para colunas semanticamente equivalentes. O Quadro 5 apresenta um exemplo de configuração utilizada para a base *sentimentos540*.

Quadro 5 – Exemplo de configuração de colunas para a base *sentimentos540*.

```
Coluna do texto: Sentimento
Classificacao manual: Humanos
Classificacao Gemma: Gemma
Classificacao LLaMA: Lamma
Classificacao LeIA: LeIA
```

Fonte: Elaborado pelo autor.

A implementação foi influenciada pelas bases utilizadas nos trabalhos relacionados, especialmente aquelas associadas à comparação entre classificações humanas, LLaMA, Gemma e LeIA (Moreira *et al.*, 2024). Dessa forma, o plugin pode aproveitar dados previamente classificados e utilizá-los como base de teste para geração de resumo, nuvem de palavras e validação funcional da ferramenta. Quando a base já contém classificações, a API de IA pode ser utilizada principalmente para a geração do resumo consolidado.

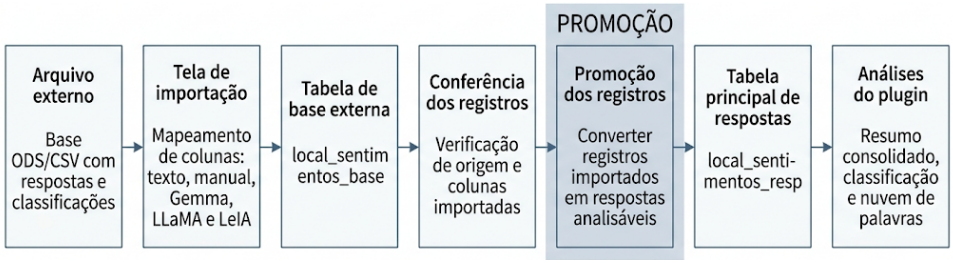
4.8 PROMOÇÃO DE REGISTROS IMPORTADOS PARA RESPOSTAS

Como mencionado anteriormente, os registros provenientes de bases externas são armazenados separadamente das respostas coletadas pelo formulário do plugin. Essa divisão foi adotada para permitir maior controle sobre os dados importados, possibilitando que a base seja conferida antes de ser utilizada nos processos de análise.

Para que esses registros externos passem a compor o conjunto principal de respostas, foi implementada a funcionalidade de promoção de registros importados. Essa função copia ou converte os dados selecionados da tabela de bases externas para a tabela principal de respostas, tornando-os disponíveis para as demais funcionalidades do plugin, como a geração de resumo consolidado e a construção da nuvem de palavras.

O fluxo de promoção dos registros importados é apresentado na Figura 9.

Figura 9 – Fluxo de promoção dos registros importados.



Fonte: Elaborado pelo autor com apoio de ferramenta de IA Gemini.

Essa etapa é especialmente útil durante os testes, pois permite utilizar bases já existentes como se fossem respostas registradas diretamente no Moodle. Ao mesmo tempo, mantém-se a distinção entre dados originalmente coletados pelo formulário e dados provenientes de bases externas, preservando a organização e a rastreabilidade das informações. Essa estratégia também favorece o reaproveitamento de dados previamente classificados em pesquisas anteriores, como as bases utilizadas em estudos sobre sentimentos discentes em disciplinas de programação (Pfitscher *et al.*, 2023; Moreira *et al.*, 2024).

4.9 CLASSIFICAÇÃO DOS SENTIMENTOS COM IA

As respostas adquiridas por meio do formulário próprio podem ser enviadas para a API de IA configurada no plugin, com o objetivo de classificá-las em categorias de sentimento. Na versão desenvolvida, as classes consideradas foram positivo, negativo e ambos, sendo esta última utilizada para representar respostas ambíguas, mistas ou que contenham simultaneamente aspectos favoráveis e desfavoráveis. A

identificação de polaridades e ambiguidades em textos é um desafio recorrente em análise de sentimentos, especialmente em contextos educacionais, nos quais uma mesma resposta pode conter elogios, dificuldades e críticas (Wankhade *et al.*, 2022; Moreira *et al.*, 2024).

Para realizar essa classificação, o plugin envia à API um prompt específico juntamente com a resposta textual do estudante. O modelo de linguagem retorna a classe atribuída, uma justificativa resumida e o retorno bruto da requisição. Essa estrutura permite que o docente visualize não apenas a classificação final, mas também uma breve explicação sobre o motivo da atribuição daquela classe.

O Quadro 6 apresenta o prompt utilizado para orientar o modelo na tarefa de classificação.

Quadro 6 – Prompt utilizado para classificação de sentimentos.

```
Voce e um classificador de sentimentos no contexto educacional.
Classifique a resposta de um estudante sobre uma disciplina.
Use apenas uma das classes: POSITIVO, NEGATIVO ou AMBOS.
POSITIVO indica satisfacao, motivacao, confianca, interesse ou tranquilidade.
NEGATIVO indica dificuldade, medo, ansiedade, desmotivacao, frustracao,
sobrecarga ou risco de desistir.
AMBOS indica presenca simultanea de aspectos positivos e negativos.
Responda apenas com JSON valido no formato:
{"classe":"POSITIVO|NEGATIVO|AMBOS","justificativa":"frase curta"}
```

Fonte: Elaborado pelo autor com apoio de ferramenta de IA Gemini.

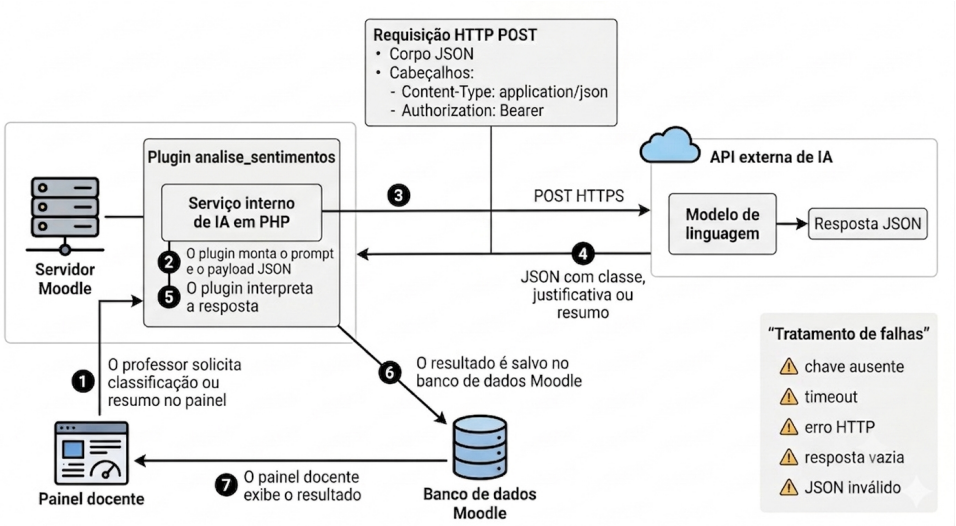
Do ponto de vista da comunicação entre sistemas, o Moodle atua como cliente HTTP e a API externa de IA atua como servidor. A requisição é realizada pelo serviço interno do plugin por meio de uma chamada HTTP do tipo POST, contendo um corpo em formato JSON. O cabeçalho da requisição informa o tipo de conteúdo, por meio de `Content-Type: application/json`, e envia a chave de autenticação no formato `Authorization: Bearer`. Nos testes deste trabalho, utilizou-se a API da OpenAI, cuja documentação descreve o uso de autenticação por chave de API e requisições HTTP para interação com modelos de linguagem (OpenAI, 2026).

A URL padrão configurada utiliza o protocolo HTTPS, o que permite que os dados trafeguem por um canal criptografado entre o servidor Moodle e o provedor externo de IA. Assim, a modelagem do canal de comunicação pode ser compreendida como uma troca síncrona entre Moodle e API externa. Conforme representado na Figura 10, o fluxo inicia quando o professor solicita uma classificação ou resumo no painel docente. Em seguida, o plugin monta o prompt e o payload em formato JSON, envia uma requisição HTTP do tipo POST por HTTPS para a API externa, aguarda o retorno do modelo, interpreta a resposta recebida, armazena o resultado no banco de dados Moodle e, por fim, apresenta a informação processada no painel docente. A mesma figura também indica os principais pontos de falha tratados pelo sistema,

como chave ausente, tempo limite de requisição, erro HTTP, resposta vazia e retorno em JSON inválido.

A comunicação com a API externa também exige o tratamento de possíveis falhas durante o envio e o recebimento das requisições. Entre as situações previstas estão a ausência de chave de API configurada, falhas de conexão, tempo limite de resposta, códigos HTTP fora da faixa de sucesso, retorno vazio e respostas em formato incompatível com o JSON esperado. Nesses casos, o plugin evita interpretar a falha como uma classificação válida, registrando o problema ou mantendo a resposta em estado pendente, conforme o tipo de ocorrência. Esse tratamento é necessário porque a integração depende de um serviço externo ao Moodle e, portanto, está sujeita a indisponibilidade, erros de autenticação, instabilidade de rede ou mudanças no formato de resposta fornecido pelo provedor.

Figura 10 – Fluxo de comunicação entre o plugin Moodle e a API externa de IA.



Fonte: Elaborado pelo autor com apoio de ferramenta de IA Gemini.

A configuração conceitual dessa integração é apresentada no Quadro 7.

Quadro 7 – Configuração conceitual da integração com IA.

```
Provedor de IA: OpenAI
Modelo: gpt-4o-mini
Endpoint: https://api.openai.com/v1/chat/completions
Entrada: resposta textual do estudante
Saida esperada: classe de sentimento, justificativa e retorno bruto
```

Fonte: Elaborado pelo autor.

O código também prevê tratamentos básicos para falhas na integração. Caso a chave de API não esteja configurada, a resposta recebe o estado `PENDENTE`, indicando que a classificação não foi realizada. A chamada à API é envolvida por trata-

mento de exceções, retornando a classe `ERRO` quando ocorre falha durante a comunicação. O serviço também verifica erro de inicialização da biblioteca de requisições, falhas na execução da chamada HTTP, códigos de resposta fora da faixa de sucesso e tempo máximo de requisição. Além disso, quando a IA não retorna um JSON estruturado, o sistema tenta normalizar a classe a partir do texto recebido, reduzindo a chance de perda total da resposta.

Apesar desses tratamentos, a integração com provedores externos ainda representa um ponto sensível da solução. Em versões futuras, podem ser aprimorados mecanismos de retentativa, limitação de tamanho das respostas enviadas, controle mais refinado de tokens, fila de processamento assíncrona e validação mais rígida do formato retornado pela IA.

4.10 GERAÇÃO DO RESUMO CONSOLIDADO

A geração do resumo consolidado segue uma lógica semelhante à classificação de sentimentos, porém utiliza o conjunto de respostas da turma como entrada. Nesse processo, o plugin reúne as respostas disponíveis e envia o conteúdo para a API de IA, acompanhado de um prompt que orienta o modelo a produzir uma síntese textual do clima geral da turma. A produção de sínteses automáticas é uma das aplicações do Processamento de Linguagem Natural e dos grandes modelos de linguagem, especialmente quando há necessidade de transformar grandes volumes de texto em informações mais acessíveis (Caseli; Nunes, 2024).

O objetivo do resumo não é listar individualmente cada resposta, mas identificar padrões recorrentes, dificuldades frequentes, percepções positivas e possíveis pontos de atenção. Dessa forma, o docente recebe uma visão geral mais acessível, reduzindo o esforço necessário para interpretar manualmente um grande volume de comentários. Essa proposta dialoga com estudos que buscam utilizar sentimentos e informações textuais para apoiar a tomada de decisão pedagógica e a permanência estudantil (Pfitscher *et al.*, 2023; Baessa *et al.*, 2024).

O Quadro 8 apresenta o prompt utilizado para orientar a geração do resumo consolidado.

Quadro 8 – Prompt utilizado para geração do resumo consolidado.

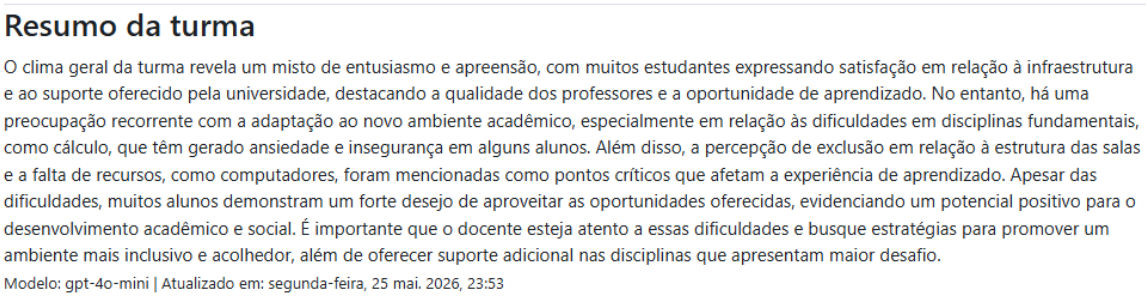
```
Voce e um assistente de analise psicopedagogica para docentes.
A partir de respostas abertas de estudantes sobre uma disciplina, gere um unico
paragrafo em portugues do Brasil.
O paragrafo deve resumir o sentimento geral da turma, destacar dificuldades
recorrentes, sinais positivos e possiveis pontos de atencao.
Nao cite nomes de alunos. Nao invente dados. Nao faca diagnostico clinico.
Escreva em tom academico e util ao professor.
```

Fonte: Elaborado pelo autor.

Na versão desenvolvida, o prompt apresentado no Quadro 8 está definido diretamente no código do plugin. Portanto, sua alteração não pode ser realizada pela interface administrativa do Moodle, exigindo modificação no código-fonte. Essa decisão simplificou a implementação inicial e garantiu maior controle sobre o formato do resumo gerado, mas reduz a flexibilidade para docentes ou administradores ajustarem as instruções conforme diferentes contextos pedagógicos.

A Figura 11 apresenta a captura do painel com o resumo consolidado gerado pela IA.

Figura 11 – Resumo consolidado apresentado ao professor.



Fonte: Captura de tela do ambiente Moodle local, elaborada pelo autor.

Durante os testes, observou-se que a IA foi capaz de gerar parágrafos sintéticos e coerentes mesmo com uma quantidade de respostas de quinhentos e quarenta. Ainda assim, o prompt utilizado deve orientar o modelo a evitar respostas excessivamente longas, repetições ou interpretações não sustentadas pelos dados fornecidos. O resumo, portanto, atua como uma ferramenta de apoio à leitura docente, e não como uma conclusão automática sobre a turma.

Quando não há respostas suficientes registradas, mínimo de duas respostas, o sistema retorna uma mensagem informando que não existem dados para geração do resumo. Caso a API não retorne conteúdo textual, o plugin também registra uma mensagem indicando ausência de retorno textual. Esses comportamentos contribuem para evitar que o docente interprete como resultado válido uma resposta inexistente ou incompleta.

4.11 GERAÇÃO DA NUVEM DE PALAVRAS

Para a geração da nuvem de palavras, inicialmente foi considerada a integração com o plugin *Wordcloud* do Moodle, identificado como *mod_wordcloud*. Esse plugin permite coletar termos com os participantes do curso e exibi-los graficamente de acordo com sua frequência, funcionando como uma atividade colaborativa de visualização textual (Moodle Plugins Directory, 2026). Embora essa alternativa permitisse

a inserção de palavras, frequências e cores, sua utilização deixava parte da funcionalidade fora do controle direto do plugin desenvolvido. Por esse motivo, optou-se por implementar uma solução própria de nuvem de palavras dentro do *analise_sentimentos*.

A aplicação utiliza um pré-processamento textual para remover termos pouco informativos ou recorrentes demais para a análise, como artigos, preposições, pronomes, conectivos, formas de tratamento, termos acadêmicos genéricos e algumas variações escritas sem acento. Etapas como normalização, tokenização e filtragem de palavras são procedimentos comuns em tarefas de Processamento de Linguagem Natural, especialmente quando se busca transformar textos não estruturados em representações analisáveis (Jurafsky; Martin, 2023). Na versão atual, a lista de palavras filtradas, assim como as regras de pré-processamento estão definidas diretamente no código do plugin, não sendo possível editar essa filtragem pela interface do Moodle.

Após o pré-processamento, as palavras são contabilizadas de acordo com sua frequência nas respostas. Cada termo é associado a uma contagem, que aumenta conforme novas ocorrências são identificadas, inclusive quando a repetição ocorre dentro da mesma resposta. Para a versão desenvolvida, foi adotado o limite de vinte palavras exibidas, considerado suficiente para representar os temas mais recorrentes sem tornar a visualização excessivamente poluída.

A visualização resultante da nuvem de palavras é exemplificada na Figura 12.

Figura 12 – Nuvem de palavras gerada a partir das respostas discentes.



Fonte: Captura de tela do ambiente Moodle local, elaborada pelo autor.

Visualmente, as palavras mais frequentes são exibidas com tamanho maior, enquanto as menos frequentes aparecem em tamanho reduzido. Além disso, cada palavra pode ser associada a uma cor distinta, facilitando a leitura visual do conjunto. Ao posicionar o cursor sobre uma palavra, é possível visualizar sua frequência total, combinando uma leitura qualitativa da nuvem com uma informação quantitativa de apoio. A utilização de nuvens de palavras como recurso de visualização já aparece em propostas de análise de sentimentos em ambientes virtuais, funcionando como uma forma de apresentar rapidamente temas predominantes ao professor (Alencar *et al.*, 2021).

4.12 PAINEL DO DOCENTE

O painel de sentimentos é a principal interface de uso do plugin pelo professor. Nele, são reunidas as funcionalidades de consulta e processamento das respostas, permitindo que o docente acesse os registros coletados, gere o resumo consolidado, atualize a nuvem de palavras, importe bases externas e promova registros importados para a tabela principal de respostas. A centralização dessas funcionalidades em um painel próprio está alinhada à lógica de plugins em ambientes virtuais de aprendizagem, nos quais funcionalidades específicas podem ser agregadas à plataforma para apoiar práticas pedagógicas (Ribeiro; Silva, 2021).

O painel também oferece acesso ao formulário de resposta, possibilitando que o professor teste o funcionamento da ferramenta antes de liberá-la aos estudantes. Essa opção deve ser utilizada apenas para verificação funcional, uma vez que respostas de teste podem interferir nos resultados caso sejam mantidas junto aos dados reais da turma.

A geração do resumo por IA ocorre mediante ação do professor, pois é recomendável que haja um conjunto mínimo de respostas antes da solicitação. Essa escolha evita o uso desnecessário de requisições à API e contribui para que o resumo gerado seja mais representativo. De modo semelhante, a nuvem de palavras também é atualizada manualmente, permitindo que o docente decida quando deseja reconstruí-la com base nas respostas disponíveis.

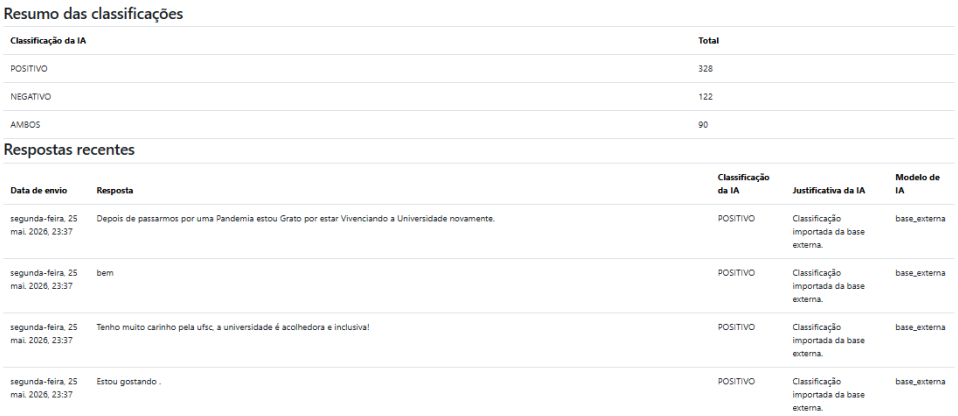
A interface geral do painel docente é apresentada nas Figura 13 e Figura 14.

Figura 13 – Painel docente do plugin *analise_sentimentos* parte superior.



Fonte: Captura de tela do ambiente Moodle local, elaborada pelo autor.

Figura 14 – Painel docente do plugin *analise_sentimentos* parte inferior.



Resumo das classificações

Classificação da IA	Total
POSITIVO	328
NEGATIVO	122
AMBOS	90

Respostas recentes

Data de envio	Resposta	Classificação da IA	Justificativa da IA	Modelo de IA
segunda-feira, 25 mai. 2026, 23:37	Depois de passarmos por uma Pandemia estou Grato por estar Vivenciando a Universidade novamente.	POSITIVO	Classificação importada da base externa.	base_externa
segunda-feira, 25 mai. 2026, 23:37	bem	POSITIVO	Classificação importada da base externa.	base_externa
segunda-feira, 25 mai. 2026, 23:37	Tenho muito carinho pela ufsc, a universidade é acolhedora e inclusiva!	POSITIVO	Classificação importada da base externa.	base_externa
segunda-feira, 25 mai. 2026, 23:37	Estou gostando .	POSITIVO	Classificação importada da base externa.	base_externa

Fonte: Captura de tela do ambiente Moodle local, elaborada pelo autor.

As funcionalidades relacionadas às bases externas foram organizadas em botões específicos. O primeiro direciona o usuário para a tela de importação da base, na qual são informadas as colunas correspondentes ao texto e às classificações. O segundo permite utilizar os registros importados como respostas analisáveis pelo plugin, etapa que requer confirmação para evitar acionamentos acidentais.

4.13 CONTROLE DE PERMISSÕES E VISIBILIDADE

O plugin também contempla mecanismos de controle de permissões e visibilidade. Esses recursos permitem diferenciar as ações disponíveis para docentes e estudantes, respeitando a lógica de uso do Moodle e evitando que funcionalidades administrativas sejam acessadas por usuários sem autorização. A existência de controle de papéis e permissões é coerente com a arquitetura de ambientes virtuais de aprendizagem, nos quais diferentes usuários acessam funcionalidades distintas conforme suas atribuições (Dougiamas; Taylor, 2003).

No plugin, o acesso às páginas é protegido por autenticação no curso e por capacidades específicas do Moodle. A funcionalidade de resposta é associada à permissão de responder ao formulário, enquanto o painel, a importação, a análise e a configuração são restritos a docentes, docentes editores ou gestores. Além disso, ações sensíveis, como a geração de resumo e a atualização de recursos do painel, utilizam verificação de chave de sessão, reduzindo o risco de acionamentos indevidos por requisições externas.

No painel docente, há opções para controlar a liberação do formulário e da nuvem de palavras aos estudantes. O formulário pode ser habilitado ou desabilitado conforme a organização da disciplina, o momento pedagógico ou a necessidade de restringir temporariamente novas respostas. Essa funcionalidade permite ao professor decidir quando deseja coletar percepções da turma.

A nuvem de palavras também pode ser compartilhada ou ocultada dos estudantes. Quando liberada, ela pode atuar como um recurso visual de retorno coletivo, permitindo que a turma observe os termos mais recorrentes nas respostas. Quando mantida restrita, permanece como ferramenta de análise exclusiva do docente. Assim, o plugin oferece flexibilidade para diferentes formas de uso pedagógico.

Do ponto de vista de privacidade, a versão desenvolvida prevê campo de consentimento no formulário e configuração de anonimização das respostas. Quando a anonimização está habilitada, o identificador do usuário é armazenado como zero, reduzindo a associação direta entre resposta textual e estudante. Entretanto, como o sistema pode enviar respostas a uma API externa de IA, é necessário que a instituição e o docente estejam atentos às políticas de uso, privacidade e proteção de dados aplicáveis. Em versões futuras, recomenda-se ampliar os mecanismos de governança dos dados, incluindo logs administrativos, política de retenção, revisão institucional do uso de IA externa e eventuais adequações à Lei Geral de Proteção de Dados.

4.14 AMBIENTE DE DESENVOLVIMENTO E IMPLANTAÇÃO

Para a realização deste trabalho, foi necessário configurar um ambiente local de desenvolvimento com o Moodle, plataforma virtual de aprendizagem de código aberto. Também foi utilizado o pacote de idioma Português-Brasil, considerando o contexto nacional da proposta e a necessidade de disponibilizar a interface do plugin em português. O Moodle foi escolhido por sua natureza aberta, modular e amplamente utilizada em contextos educacionais (Dougiamas; Taylor, 2003).

Esse processo de implantação foi realizado em ambiente local, hospedado no computador pessoal do desenvolvedor, utilizando containers Docker para executar os serviços necessários ao funcionamento do Moodle. Nessa configuração, o Docker foi responsável por isolar e organizar os componentes da aplicação, incluindo o servidor web, o interpretador PHP e o banco de dados, simulando uma infraestrutura semelhante à de um servidor Moodle real.

A implantação do plugin envolveu a cópia dos arquivos desenvolvidos para o diretório `local/sentimentos` da instalação Moodle executada no container, o acesso ao painel administrativo da plataforma para detecção do novo componente, a criação ou atualização das tabelas definidas no arquivo `install.xml` e a configuração dos parâmetros necessários ao funcionamento da API de IA. Dessa forma, o ambiente local hospedado via Docker permitiu testar o processo básico de instalação, preservar o isolamento dos serviços e facilitar a depuração de erros durante o desenvolvimento, sem interferir em um servidor institucional.

A linguagem utilizada no desenvolvimento foi PHP, que é a linguagem base da plataforma Moodle e responsável pela criação das páginas, formulários, serviços e rotinas do plugin. Durante o processo, foi necessário compreender a hierarquia de

arquivos, o sistema de permissões, as tabelas de banco de dados e as convenções de desenvolvimento da plataforma. Esses elementos representaram desafios importantes, mas também foram fundamentais para integrar o plugin ao funcionamento nativo do Moodle.

4.15 DISPONIBILIZAÇÃO DO CÓDIGO-FONTE

O código-fonte do plugin *analise_sentimentos* foi disponibilizado em um repositório público no GitHub, com o objetivo de permitir consulta, reprodução, análise e futura continuidade do desenvolvimento. O repositório contém os arquivos principais do plugin, incluindo páginas de interface, serviços internos, formulários, arquivos de configuração, estrutura de banco de dados e rotinas relacionadas à importação, classificação, resumo e geração da nuvem de palavras.

O acesso ao repositório pode ser realizado pelo endereço apresentado em Rocha (2025). A disponibilização do código contribui para a transparência do trabalho e permite que a solução seja examinada, adaptada ou expandida em versões futuras.

5 RESULTADOS E DISCUSSÃO

Este capítulo apresenta os resultados obtidos a partir da validação funcional do plugin *analise_sentimentos* em ambiente Moodle local. São discutidos o funcionamento geral da ferramenta, a importação e utilização das bases de sentimentos, a geração de resumos por inteligência artificial, a estimativa de custos da API, a nuvem de palavras e as principais limitações identificadas. A análise possui caráter técnico e exploratório, concentrando-se na verificação das funcionalidades implementadas e na coerência dos artefatos produzidos, sem pretender realizar validação pedagógica definitiva em turmas reais.

5.1 FUNCIONAMENTO GERAL DO PLUGIN

Em relação às proposições iniciais do projeto, foram implementadas as funcionalidades centrais de coleta de respostas, classificação de sentimentos com apoio de IA, geração de resumo consolidado e exibição de uma nuvem de palavras com os termos mais recorrentes. Além disso, o plugin passou a contar com mecanismos de importação de bases externas e promoção desses registros para uso nas análises.

As funcionalidades foram testadas em ambiente Moodle local, utilizando tanto respostas simuladas quanto registros importados de bases externas. Os testes permitiram verificar o funcionamento das páginas principais, a gravação das respostas no banco de dados, a comunicação com a API de IA e a geração dos artefatos apresentados ao docente. Essa etapa caracteriza uma validação funcional da ferramenta, voltada a verificar sua viabilidade técnica antes de uma avaliação pedagógica em contexto real.

O Quadro 9 apresenta a síntese dos testes funcionais realizados.

Quadro 9 – Síntese dos testes funcionais do plugin.

Funcionalidade	**Critério de teste**	**Resultado esperado**	**Status**
Coleta de resposta	Enviar resposta válida pelo formulário discente.	Registro salvo na tabela principal de respostas.	APROVADO
Classificação por IA	Submeter resposta textual para classificação.	Retorno de classe, modelo e justificativa.	APROVADO
Importação de base	Importar planilha com respostas e classificações.	Registros armazenados na tabela de bases externas.	APROVADO
Promoção de registros	Promover base importada para respostas analisáveis.	Registros disponíveis na tabela principal de respostas.	APROVADO
Resumo consolidado	Gerar resumo a partir das respostas disponíveis.	Resumo textual salvo e exibido no painel docente.	APROVADO
Nuvem de palavras	Atualizar nuvem a partir das respostas disponíveis.	Lista visual com os termos mais recorrentes.	APROVADO
Controle de visibilidade	Alterar liberação do formulário e da nuvem.	Estado atualizado para a disciplina selecionada.	APROVADO

Fonte: Elaborado pelo autor.

Embora a ferramenta ainda não tenha sido validada em uma turma real ao longo de um semestre, os testes funcionais indicaram que a estrutura desenvolvida é capaz de organizar respostas textuais e transformá-las em informações de apoio à interpretação docente. Dessa forma, a versão atual atende ao objetivo de demonstrar a viabilidade técnica da solução proposta, aproximando-se de trabalhos que buscam utilizar dados textuais e sentimentos discentes como apoio à permanência estudantil (Pfitscher *et al.*, 2023; Baessa *et al.*, 2024).

5.2 IMPORTAÇÃO E USO DAS BASES DE SENTIMENTOS

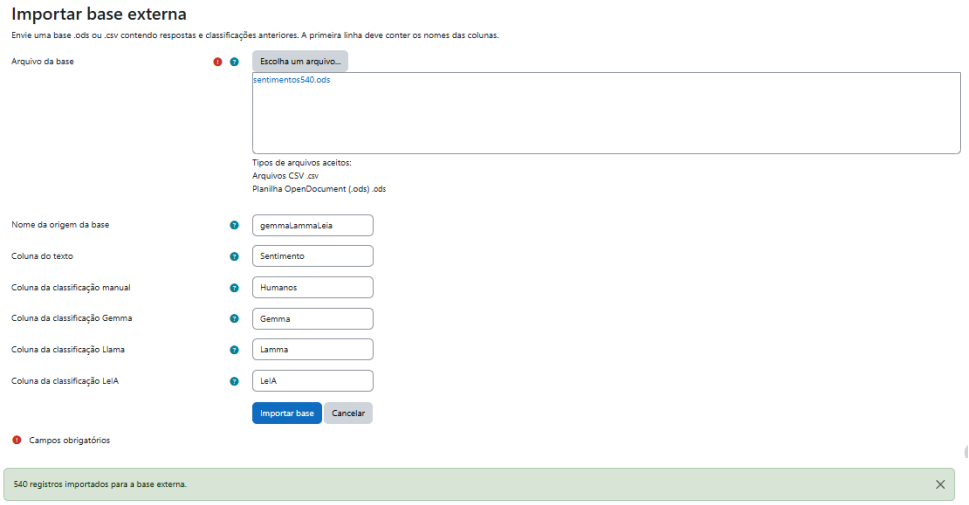
Para os testes com bases externas, foi utilizada a base *sentimentos540.ods*, composta por respostas discentes previamente classificadas. Essa base contém classificações humanas e classificações geradas por diferentes ferramentas, incluindo os modelos Gemma e LLaMA, além do analisador léxico LeIA. O uso dessa base se justifica por sua relação direta com estudos que compararam ferramentas automatizadas de análise de sentimentos aplicadas a respostas discentes no contexto educacional (Moreira *et al.*, 2024).

O uso dessa base foi relevante por permitir testar o plugin com um volume maior de respostas e com dados associados a pesquisas anteriores na área de análise de sentimentos em contexto educacional. Além disso, a presença de diferentes

classificações possibilita observar como dados já rotulados podem ser reaproveitados pelo plugin para fins de análise, visualização e geração de resumo.

A importação exigiu o pareamento entre os nomes das colunas da planilha e os campos esperados pelo sistema. Após esse processo, os registros puderam ser armazenados na tabela de bases externas e posteriormente promovidos para a tabela principal de respostas, passando a ser utilizados nas funcionalidades de resumo e nuvem de palavras.

Figura 15 – 540 Registros importados e utilizados como respostas no plugin.



Fonte: Captura de tela do ambiente Moodle local, elaborada pelo autor.

5.3 RESUMO GERADO POR INTELIGÊNCIA ARTIFICIAL

Nos testes realizados, os resumos gerados pela inteligência artificial apresentaram coerência com o conteúdo das respostas utilizadas. O modelo foi capaz de sintetizar críticas, dificuldades, percepções positivas e sugestões presentes nos relatos, sem transformar o resultado em uma simples listagem de comentários individuais. Esse comportamento é compatível com o uso de modelos de linguagem para tarefas de síntese textual e extração de informações em linguagem natural (Jurafsky; Martin, 2023; OpenAI, 2026).

Também foi observado que o resumo não tentou suavizar excessivamente as críticas presentes nas respostas, mantendo pontos de atenção relevantes para o professor. Ao mesmo tempo, os aspectos positivos mencionados pelos estudantes também foram preservados, resultando em uma síntese equilibrada sobre o clima geral da turma.

Esse resultado indica que o uso de uma API de IA pode contribuir para reduzir o esforço de leitura manual de grandes volumes de respostas abertas. Entretanto, o resumo deve ser compreendido como apoio interpretativo, e não como conclusão

definitiva. A análise final permanece sob responsabilidade do docente, que deve considerar o contexto da turma e da disciplina.

Como o resumo é produzido a partir de uma API externa de IA, seus resultados dependem diretamente do conjunto de respostas enviado e das instruções presentes no prompt. Por esse motivo, a análise do docente permanece indispensável. A ferramenta deve ser compreendida como mecanismo de síntese e apoio interpretativo, não como avaliação automática definitiva da disciplina ou dos estudantes.

5.4 ESTIMATIVA DE CUSTOS DE USO DA API

Embora o plugin tenha sido estruturado de forma genérica para permitir a configuração de diferentes provedores de IA compatíveis com requisições HTTP, os testes deste trabalho foram realizados utilizando a API da OpenAI, com o modelo *gpt-4o-mini*. A escolha desse modelo ocorreu em função de seu baixo custo por token, sua compatibilidade com tarefas de classificação e síntese textual e sua disponibilidade por meio de endpoint compatível com o formato utilizado pelo plugin.

Segundo a política de preços divulgada pela OpenAI para o modelo *gpt-4o-mini*, o custo é de US$ 0,15 por milhão de tokens de entrada e US$ 0,60 por milhão de tokens de saída (OpenAI, 2024). Como as tarefas realizadas pelo plugin envolvem textos curtos, especialmente respostas abertas de estudantes e retornos sintéticos em formato de classe, justificativa ou resumo, o custo observado durante os testes foi muito reduzido. Mesmo com o uso de bases com 540 e 720 registros, além de envios pontuais de aproximadamente 20 a 30 respostas, o valor cobrado no ambiente de testes não atingiu US$ 0,01.

Para avaliar a viabilidade em um cenário institucional, foi elaborada uma estimativa hipotética considerando um campus com 40 professores, cada um responsável por 3 disciplinas, e turmas com 40 estudantes por disciplina. Nesse cenário, haveria 120 turmas-disciplina e 4.800 respostas textuais por ciclo de coleta. Para a estimativa, considerou-se uma margem conservadora de 300 tokens de entrada e 80 tokens de saída por classificação individual, além de 3.500 tokens de entrada e 500 tokens de saída para a geração de um resumo consolidado por turma-disciplina.

O Quadro 10 apresenta a estimativa de consumo e custo para esse cenário.

Quadro 10 – Estimativa de custo por ciclo para uso da API em cenário institucional.

Operação	**Entradas**	**Saídas**	**Custo estimado**	**Critério adotado**
Classificação individual	1.440.000 tokens	384.000 tokens	US$ 0,4464	4.800 respostas, com 300 tokens de entrada e 80 tokens de saída por resposta.
Resumo consolidado	420.000 tokens	60.000 tokens	US$ 0,0990	120 turmas-disciplina, com 3.500 tokens de entrada e 500 tokens de saída por resumo.
Total estimado	**1.860.000 tokens**	**444.000 tokens**	**US$ 0,5454**	Custo total para um ciclo de coleta em 120 turmas-disciplina.

Fonte: Elaborado pelo autor com base nos valores divulgados pela OpenAI para o modelo *gpt-4o-mini*.

O cálculo foi realizado conforme as Equações 1 e 2, considerando separadamente os tokens enviados ao modelo e os tokens retornados pela API.

$$C_{\text{entrada}} = \frac{T_{\text{entrada}}}{10^6} \cdot 0{,}15 \tag{1}$$

$$C_{\text{saida}} = \frac{T_{\text{saida}}}{10^6} \cdot 0{,}60 \tag{2}$$

Em que T_{entrada} representa a quantidade de tokens enviados à API e T_{saida} representa a quantidade de tokens retornados pelo modelo. No cenário estimado, o custo total para um ciclo de coleta envolvendo 40 professores, 3 disciplinas por professor e 40 estudantes por disciplina seria inferior a US$ 1,00. Mesmo considerando múltiplas coletas ao longo do semestre, o custo permaneceria baixo em relação ao potencial de uso pedagógico da ferramenta.

É importante ressaltar que essa estimativa considera apenas o custo da API de IA e não inclui despesas relacionadas à infraestrutura do Moodle, manutenção do servidor, conexão com a internet, armazenamento ou eventuais variações de câmbio. Além disso, os valores podem mudar conforme o provedor, o modelo utilizado, o tamanho médio das respostas e a política de preços vigente no momento da implantação. Ainda assim, os testes realizados e a estimativa apresentada indicam que o uso de modelos leves para classificação e síntese textual pode ser economicamente viável para aplicações institucionais de acompanhamento discente.

5.5 NUVEM DE PALAVRAS

Inicialmente, foi testada a possibilidade de utilizar um plugin de nuvem de palavras já disponível na biblioteca do Moodle. Essa alternativa permitia inserir palavras, frequências e cores, mas sua integração mantinha parte da funcionalidade fora da estrutura do plugin desenvolvido, dificultando ajustes e personalizações.

A solução adotada foi implementar uma nuvem de palavras própria, limitada a vinte termos principais. Essa abordagem permitiu controlar diretamente o pré-processamento textual, a seleção das palavras, os tamanhos exibidos e as cores associadas. Com isso, a nuvem tornou-se uma visualização complementar ao resumo gerado por IA, oferecendo ao professor uma forma rápida de identificar recorrências nas respostas. A utilização de técnicas de pré-processamento textual é fundamental para que representações baseadas em frequência não sejam dominadas por termos de baixo valor informativo (Jurafsky; Martin, 2023).

Durante os testes, observou-se que as respostas dos estudantes nem sempre seguem a norma formal da língua portuguesa. Foram identificadas abreviações, gírias, palavras sem acentuação e erros de digitação. Para lidar com esse problema, foi adotada uma filtragem mais robusta de termos, incluindo variações comuns de escrita e palavras pouco informativas, como pronomes, verbos de ligação, o nome da instituição entre outros, de modo a melhorar a relevância dos termos exibidos.

5.6 LIMITAÇÕES DA VERSÃO DESENVOLVIDA

Uma das principais limitações da versão desenvolvida é a ausência de validação em uma turma real ao longo de um semestre. Embora os testes funcionais tenham demonstrado a viabilidade técnica do plugin, seria necessário aplicá-lo em contexto real para avaliar sua utilidade pedagógica, sua aceitação pelos docentes e sua capacidade de representar adequadamente o clima da turma. Estudos voltados à permanência estudantil indicam que sentimentos podem contribuir para análises educacionais mais amplas, especialmente quando combinados com outros indicadores acadêmicos (Baessa *et al.*, 2024; Tazi *et al.*, 2024).

Outra limitação está relacionada à dependência de uma API externa de inteligência artificial. Essa escolha simplifica a implementação e permite utilizar modelos avançados de linguagem, mas também introduz fatores como custo de uso, necessidade de conexão com a internet, tratamento de erros externos e cuidados com privacidade. Em versões futuras, poderia ser avaliada a integração com modelos locais, APIs gratuitas ou serviços institucionais de IA.

Também foram identificadas possibilidades de aprimoramento na interface e nas funcionalidades do plugin. Entre elas, destacam-se a adaptação mais ampla para diferentes provedores de LLM, a melhoria visual das telas, a visualização completa das respostas, inclusão do nivel de satisfação por escala e a integração com outros dados do Moodle, como frequência, notas e participação. Esses dados poderiam complementar a análise textual e oferecer ao docente uma visão mais ampla sobre o acompanhamento da turma e a permanência estudantil, conforme sugerido por abordagens que combinam sentimentos com variáveis acadêmicas (Baessa *et al.*, 2024).

6 CONCLUSÃO

Este trabalho apresentou o desenvolvimento de um plugin local para a plataforma Moodle voltado à coleta, importação, classificação e síntese de sentimentos discentes. A proposta teve como motivação a necessidade de oferecer ao professor uma ferramenta de apoio à interpretação de respostas abertas, permitindo transformar relatos textuais em informações organizadas sobre o clima da turma. Essa motivação está alinhada a estudos que investigam a análise de sentimentos como recurso de apoio ao acompanhamento discente em ambientes educacionais (Alencar *et al.*, 2021; Pfitscher *et al.*, 2023).

A solução desenvolvida, denominada *analise_sentimentos*, implementou um formulário próprio para coleta de respostas dos estudantes, mecanismos de importação de bases externas, promoção de registros importados para a tabela principal, classificação de sentimentos com apoio de uma API externa de inteligência artificial, geração de resumo consolidado e construção de uma nuvem de palavras. Essas funcionalidades foram reunidas em um painel docente, permitindo que o professor acompanhe os dados processados em um único ambiente.

Os testes realizados em ambiente Moodle local indicaram que o plugin é funcional e capaz de operar com respostas coletadas diretamente ou importadas de bases externas. O resumo gerado por inteligência artificial mostrou-se coerente com o conjunto de respostas analisadas, apresentando uma síntese útil para a leitura docente. A nuvem de palavras, por sua vez, atuou como recurso complementar, destacando termos recorrentes e auxiliando na identificação visual de temas frequentes.

Apesar dos resultados positivos, a versão desenvolvida ainda possui limitações. A validação foi realizada em ambiente local, sem aplicação contínua em uma turma real. Dessa forma, a avaliação pedagógica da ferramenta permanece como etapa futura.

Do ponto de vista técnico, o trabalho também evidenciou a importância da modelagem do fluxo de dados e da comunicação entre sistemas. A solução desenvolvida depende da articulação entre interface Moodle, banco de dados, serviços internos do plugin e API externa de IA. Essa integração demonstrou ser viável, mas também destacou pontos de atenção relacionados a tratamento de erros, privacidade, dependência de serviços externos e necessidade de validação das classificações geradas.

Como trabalhos futuros, recomenda-se a aplicação do plugin em disciplinas reais, com acompanhamento ao longo do semestre e coleta de feedback de docentes e estudantes, contando com uma interface para o envio do termo de compromisso. Também podem ser exploradas melhorias na interface, integração com outros indicadores do Moodle, assim como a validação do Moodle para distribuição oficial do plugin,

adaptação para diferentes provedores de IA e desenvolvimento de modelos próprios ou institucionais para análise de sentimentos. Ainda assim, a versão implementada demonstra a viabilidade de integrar técnicas de análise textual e inteligência artificial ao Moodle, oferecendo uma ferramenta prática de apoio à permanência estudantil e à gestão pedagógica.

REFERÊNCIAS

SANTOS ALENCAR, M. A. dos; MAGALHãES NETTO, J. F. de; MORAIS, F. de. A sentiment analysis framework for virtual learning environment. **Applied Artificial Intelligence**, Taylor & Francis, v. 35, n. 7, p. 520–536, 2021. Disponível em: https://doi.org/10.1080/08839514.2021.1904594. Acesso em: 26 maio 2025.

ALPAYDIN, E. ***Introduction to Machine Learning***. 4. ed. Cambridge, MA: The MIT Press, 2020. ISBN 9780262043793.

ATIQ, A.; LOUI, R. P. Sentiment analysis in educational settings: Techniques and applications. **Journal of Educational Data Mining**, v. 14, n. 3, p. 45–68, 2022.

BAESSA, R. N. F. *et al.* Predição de desistência em turmas de programação utilizando sentimentos. **Revista Novas Tecnologias na Educação (RENOTE)**, v. 22, n. 3, p. 375–386, 2024.

BECKER, L.; MOREIRA, M.; SANTOS, R. Moodle predicta: Um módulo de visualização de dados baseado no comportamento dos estudantes. *In*: SBC. **Anais do Workshop de Informática na Escola**. Sociedade Brasileira de Computação, 2017. p. 77–86. Disponível em: https://sol.sbc.org.br/index.php/wie/article/view/2488. Acesso em: 26 maio 2025.

BóBó, M.; CAMPOS, F.; STRöELE, V.; OLIVEIRA, J. Identificação do perfil emocional do aluno através de análise de sentimento: combatendo a evasão escolar. *In*: **Anais do Simpósio Brasileiro de Informática na Educação (SBIE)**. SBC, 2022. p. 1–10. Disponível em: https://www.researchgate.net/publication/337410173.

CARVALHO, R.; CUNHA, M. ***Evasão universitária e desigualdade social no Brasil***. São Paulo: EdUSP, 2023.

MEDEIROS CASELI, H. de; GRAçAS VOLPE NUNES, M. das. ***Processamento de Linguagem Natural: Conceitos, Técnicas e Aplicações em Português***. 3. ed. BPLN, 2024. Acesso em: 16 junho 2025. Disponível em: https://brasileiraspln.com/livro-pln/3a-edicao/.

DOUGIAMAS, M.; TAYLOR, P. Moodle: Using learning communities to create an open source course management system. *In*: **Proceedings of the EDMEDIA 2003 Conference**. Association for the Advancement of Computing in Education (AACE), 2003. p. 171–178. Disponível em: https://dougiamas.com/archives/edmedia2003/.

JURAFSKY, D.; MARTIN, J. H. ***Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition***. 3. ed. Pearson, 2023. Disponível em: https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf.

Moodle. **Moodle.org: Home**. 2026. Acesso em: 9 jun. 2026. Disponível em: https://moodle.org/.

Moodle Developer Resources. **Common files**. 2026. Acesso em: 9 jun. 2026. Disponível em: https://moodledev.io/docs/4.5/apis/commonfiles.

Moodle Developer Resources. **Data manipulation API**. 2026. Acesso em: 9 jun. 2026. Disponível em: https://moodledev.io/docs/4.5/apis/core/dml.

Moodle Developer Resources. **Plugin types**. 2026. Acesso em: 9 jun. 2026. Disponível em: https://moodledev.io/docs/5.1/apis/plugintypes.

Moodle Plugins Directory. **Wordcloud**. 2026. Acesso em: 2 jun. 2026. Disponível em: https://moodle.org/plugins/mod_wordcloud.

MOREIRA, B. G.; CAMARGO, L. C.; PFITSCHER, R. J.; GARCIA, T. R. Comparação de ferramentas para análise de sentimentos aplicada no contexto educacional. *In*: **Anais do Simpósio Brasileiro de Informática na Educação (SBIE)**. [S.l.: s.n.], 2024. p. 1–10.

MOREIRA, B. G.; GARCIA, T. R.; PFITSCHER, R. J.; CAMARGO, L. C. **Sentimentos classificados - Humanos e IA**. Zenodo, 2025. Acesso em: 2 jun. 2026. Disponível em: https://zenodo.org/records/15521409.

OLIVEIRA, J.; SILVA, A.; SANTOS, B. O custo econômico da evasão no ensino superior. **Revista Brasileira de Políticas Educacionais**, v. 15, n. 2, p. 45–67, 2022.

OPENAI. **GPT-4o mini: advancing cost-efficient intelligence**. 2024. Acesso em: 31 maio 2026. Disponível em: https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/.

OPENAI. **OpenAI API Reference**. 2026. https://platform.openai.com/docs/api-reference. Acesso em: 26 maio 2026.

PFITSCHER, R. J. *et al.* Análise de sentimentos em turmas de programação com vistas ao apoio à permanência estudantil. **Anais do Congresso Brasileiro de Informática na Educação (CBIE)**, p. 1–10, 2023.

RIBEIRO, A.; SILVA, L. Desenvolvimento de plugins para ambientes virtuais de aprendizagem: uma abordagem modular. *In*: **Anais do Congresso Brasileiro de Informática na Educação**. [S.l.]: SBC, 2021. p. 45–52. Acesso em: 26 maio 2025.

ROCHA, B. H. **Plugin local_sentimentos para Moodle**. 2025. Repositório GitHub. Acesso em: 2 jun. 2026. Disponível em: https://github.com/Bhott21/Plugin-Sentimentos.

SEMESP, I. **Mapa do Ensino Superior no Brasil 2024**. 2024. https://www.semesp.org.br/wp-content/uploads/2024/04/mapa-do-ensino-superior-no-brasil-2024.pdf. Acesso em: 26 maio 2025.

SIMIONESCU, C.; MARCU, D.; MACIUCA, M. S. Toward better education quality through students' sentiment analysis using automl. *In*: **BRAIN. Broad Research in Artificial Intelligence and Neuroscience**. LUMEN Publishing, 2024. v. 15, n. 2, p. 320–343. Disponível em: https://doi.org/10.18662/brain/15.2/578. Acesso em: 26 maio 2025.

TAZI, K.; AZZOUZI, S.; SRAIDI, S.; CHARAF, M. E. H. Mooc drop-out prediction based on learners sentiment analysis. *In*: **Proceedings of the 10th International Conference on Optimization and Applications (ICOA)**. IEEE, 2024. p. 1–8.

Disponível em: https://ieeexplore.ieee.org/document/10754144. Acesso em: 26 maio 2025.

WANKHADE, K.; PATIL, A.; BHOSALE, S. A survey on sentiment analysis methods, applications, and challenges. **International Journal of Computer Applications**, v. 175, n. 13, p. 1–7, 2022. Disponível em: https://www.researchgate.net/publication/358421868.

ZHAO, W. X. *et al.* A survey of large language models. **Frontiers of Computer Science**, v. 20, n. 12, p. 2012627, 2026.