



UNIVERSIDADE FEDERAL DA PARAÍBA  
CENTRO DE INFORMÁTICA

CIÊNCIA DA COMPUTAÇÃO  
MÉTODOS DE PROJETO DE SOFTWARE

# **Corus:**

## **Sistema de gerenciamento de tarefas**

**Aluno:**

Pedro Henrique Nogueira da Silva Santos	/20190023610
Ruanderson Gabriel Alves da Silva costa de Fontes	/20210024827
Yan Feitosa Cláudio	/20210094385
Dmitri Verdi Alves de Oliveira	/20220005979
Felipe Cavalcanti Apolinario	/20220070841

## Histórico de Revisões

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor(es)</b>
<dd/mm/aaaa>	<x.xx>	<detalhes>	<nome>

# Sumário

<b>Sumário.....</b>	<b>2</b>
<b>1. Introdução.....</b>	<b>2</b>
1.1. Propósito do documento.....	3
1.2. Visão geral do documento.....	4
<b>2. Descrição geral.....</b>	<b>5</b>
2.1. Motivação.....	5
2.2. Problemas identificados.....	6
2.3. Visão geral do sistema.....	6
2.4. Usuários do sistema.....	8
2.5. Suposições e Restrições Gerais.....	11
<b>3. Glossário.....</b>	<b>12</b>
<b>4. Elicitação de Requisitos.....</b>	<b>12</b>
<b>5. Análise de Requisitos.....</b>	<b>14</b>
5.1. Requisitos funcionais.....	14
5.2. Requisitos não funcionais.....	20
5.2.1 Usabilidade.....	20
5.2.2 Desempenho.....	22
5.2.3 Confiabilidade.....	23
5.2.4 Segurança.....	24
5.2.5 Portabilidade.....	24
5.2.6 Compliance.....	25
5.2.7 Internacionalização.....	25
<b>6. Especificação de Requisitos.....</b>	<b>26</b>
<Nome de subseção para agrupar casos de uso correlacionados>.....	26
<Nome de outra subseção para agrupar outros casos de uso correlacionados>.....	27
<b>7. Análise de casos de uso (diagrama de classes de análise).....</b>	<b>29</b>
<b>8. Descrição da interface com o usuário.....</b>	<b>30</b>
<b>9. Diagramas de Arquitetura.....</b>	<b>31</b>

# 1. Introdução

## 1.1. Propósito do documento

O presente documento descreve o projeto do sistema Corus, uma plataforma inteligente desenvolvida para aprimorar o aprendizado acadêmico. Esse sistema tem como objetivo central transformar materiais de estudo brutos, como PDFs, livros e resumos, em recursos pedagógicos avançados, incluindo resumos concisos, questões personalizadas e explicações passo a passo, permitindo que estudantes compreendam conteúdos complexos de maneira eficiente e aprofundada. A aplicação integra Inteligência Artificial e organização visual de conteúdos para criar um ecossistema de estudo adaptativo, reduzindo o tempo de preparação para provas e otimizando a retenção de conhecimento.

Este documento é elaborado conforme as práticas de Métodos de Projeto de Software, fornecendo informações detalhadas sobre os objetivos, requisitos, funcionalidades e estruturas do sistema. Ele serve como referência para:

- Desenvolvedores: Para compreensão clara dos requisitos e funcionalidades do sistema, permitindo uma implementação eficiente e alinhada às necessidades dos usuários;
- Testadores: Para elaboração de casos de teste para validar o correto funcionamento das funcionalidades propostas;
- Stakeholders: Para acompanhamento do progresso do projeto, entendimento do escopo e avaliação da importância do sistema para os usuários finais.

O documento também apresenta convenções, referências a outros materiais relevantes e está estruturado de maneira organizada, permitindo uma visão completa do projeto.

## 1.2. Visão geral do documento

Esta introdução fornece as informações necessárias para fazer um bom uso deste documento, explicitando seus objetivos e as convenções que foram adotadas no texto, além de conter uma lista de referências para outros documentos relacionados. As demais seções apresentam a especificação do sistema <Nome do sistema> e estão organizadas como descrito abaixo.

- Seção 2 – Descrição geral do sistema: apresenta uma visão geral do sistema, caracterizando qual é o seu escopo e descrevendo seus usuários.
- Seção 3 - Glossário do projeto
- Seção 4 - Descrição das metodologias aplicadas para coleta e validação de requisitos.
- Seção 5.1 – Requisitos funcionais (casos de uso): especifica todos os requisitos funcionais do sistema, descrevendo os fluxos de eventos, prioridades, atores, entradas e saídas de cada caso de uso a ser implementado.
- Seção 5.2 – Requisitos não funcionais: especifica todos os requisitos não funcionais do sistema, divididos em requisitos de usabilidade, confiabilidade, desempenho, segurança, distribuição, adequação a padrões e requisitos de hardware e software.

## **2. Descrição geral**

Corus é uma plataforma web inteligente projetada para revolucionar o aprendizado acadêmico. Seu propósito central é transformar materiais de estudo brutos (PDFs, livros, resumos) em recursos pedagógicos avançados — resumos concisos, questões personalizadas e explicações passo a passo —, permitindo que estudantes dominem conteúdos complexos com eficiência e profundidade. A aplicação integra tecnologia de IA e organização visual para criar um ecossistema de estudo adaptativo, reduzindo o tempo de preparação para provas e otimizando a retenção de conhecimento.

### **2.1. Motivação**

O aumento da quantidade de informações disponíveis têm transformado o modo como estudantes acessam e assimilam conhecimento. Apesar do acesso facilitado a livros, artigos e materiais digitais, muitos estudantes enfrentam dificuldades para organizar, compreender e revisar grandes volumes de conteúdo, o que compromete a eficiência do aprendizado e a preparação para avaliações acadêmicas.

O Corus surge como uma solução para esses desafios, oferecendo uma plataforma inteligente capaz de processar materiais de estudo brutos e transformá-los em recursos pedagógicos claros e estruturados. Por meio da integração de LLMs (Large Language Models) e ferramentas de organização visual, o sistema permite que os usuários:

- Economizem tempo na preparação e revisão de conteúdos;
- Recebam explicações detalhadas e personalizadas, adequadas ao seu nível de conhecimento;
- Transformem documentos extensos em resumos concisos e exercícios direcionados;
- Tenham uma experiência de estudo adaptativa, que prioriza a retenção e

compreensão de conceitos-chave.

A motivação principal do projeto é, portanto, potencializar o aprendizado acadêmico, tornando o estudo mais eficiente, organizado e acessível, beneficiando estudantes de diferentes níveis e áreas de conhecimento, além de contribuir para a inovação na forma como o conteúdo educacional é consumido e estruturado.

## 2.2. Problemas identificados

Problema	Impacto no Usuário	Solução Corus
Síntese manual de conteúdos	Perda de 60-70% do tempo de estudo	Resumos automáticos com IA
Questões descontextualizadas	Baixa eficácia na preparação para provas	Geração de questões baseadas no material
Fragmentação de recursos	Dificuldade de revisão e perda de foco	Organização visual em "esferas"

## 2.3. Visão geral do sistema

O Corus é um sistema web que auxilia usuários na **organização de materiais** e na **geração de conteúdos de estudo** de forma ágil. Ele permite que o usuário crie estruturas visuais chamadas **esferas** (e sub-esferas), ou alternativamente pastas e subpastas, onde poderá **anexar arquivos** (PDFs ou imagens). A partir desses arquivos, o sistema é capaz de realizar extração de texto (com suporte a OCR) e, em

seguida, gerar automaticamente **resumos, questões e explicações** com base no material enviado.

O sistema é **acessível em qualquer dispositivo via navegador**, bastando o login do usuário para recuperar sua conta e dados armazenados na nuvem. Dessa forma, garante portabilidade e continuidade de uso.

### Funcionalidades principais

- Criação e gerenciamento de **esferas e sub-esferas**, bem como visualização alternativa por **pastas/subpastas**.
- Upload e armazenamento de arquivos (PDFs e imagens).
- **Extração de texto** de PDFs digitais e uso de **OCR** para PDFs escaneados ou imagens.
- Geração de **resumos automáticos, questões e explicações** a partir de um arquivo ou de todos os arquivos associados a uma esfera.
- Possibilidade de exportar conteúdos gerados (ex.: PDF, TXT) ou compartilhá-los por link.
- Gerenciamento de **assinaturas e limites de uso** de acordo com o plano escolhido.

### Escopo negativo (não contemplado nesta versão)

- **Modo offline:** não será suportado, pois o sistema depende de processamento em nuvem.
- **Integração com plataformas de ensino (LMS):** não está no escopo inicial; poderá ser adicionada futuramente.
- **Treinamento de modelos próprios:** o sistema utilizará apenas APIs de IA comerciais já treinadas.
- **Colaboração em tempo real (multiusuário simultâneo):** não está prevista na primeira versão.



- **Processamento massivo ilimitado:** limitado pelos planos de assinatura para viabilidade de custos.

## Interações com outros sistemas

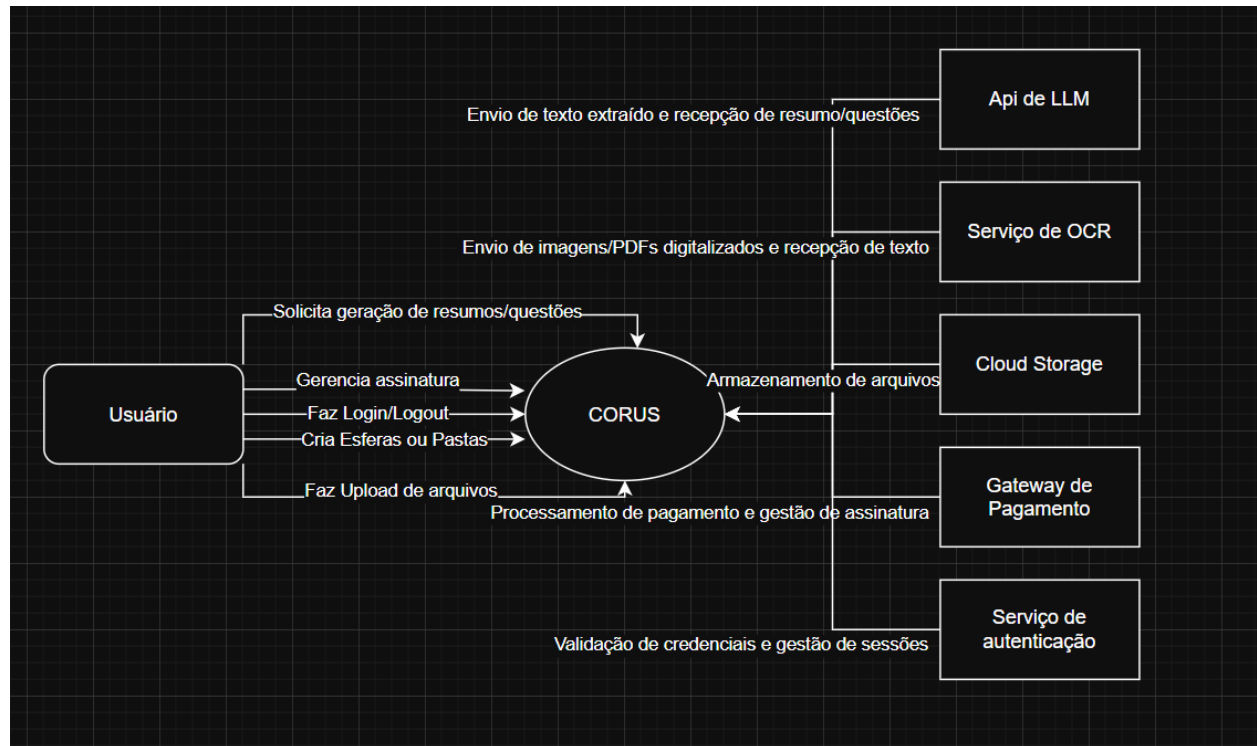
Embora o Corus seja em grande parte auto-contido, ele depende de integrações com serviços externos para algumas funcionalidades específicas:

- **LLM (ex.: OpenAI GPT-4o):** responsável pela geração de resumos, questões e explicações. Comunicação via API REST segura.
- **OCR (ex.: Google Vision API ou Tesseract):** utilizado para extrair texto de imagens e PDFs digitalizados.
- **Armazenamento em nuvem (ex.: AWS S3 ou Firebase Storage):** guarda os arquivos originais e conteúdos gerados.
- **Gateway de pagamento (ex.: Stripe, Mercado Pago):** processa cobranças e gerencia assinaturas.
- **Serviço de autenticação (ex.: Firebase Auth ou JWT customizado):** gerencia logins e sessões seguras.

## Componentes internos

- **Frontend (React/Tailwind):** interface para organização em esferas/pastas, upload de arquivos e solicitação de geração de conteúdos.
- **Backend (API REST):** responsável por autenticação, orquestração de uploads, integração com OCR, LLM e pagamentos.
- **Banco de dados relacional (PostgreSQL):** armazena informações de usuários, arquivos, esferas e conteúdos gerados.
- **Serviço de fila de jobs:** processa de forma assíncrona as operações de maior custo (OCR e geração via LLM), garantindo melhor experiência ao usuário.

## Diagrama de contexto



## 2.4. Usuários do sistema

O Corus foi projetado para atender diferentes perfis de usuários que compartilham uma necessidade comum: **organizar seus materiais e transformar documentos complexos em conteúdos resumidos, claros e objetivos**. Esses usuários enfrentam dificuldades em administrar seu tempo de estudo ou trabalho, processar grandes volumes de informação e revisar conteúdos de forma eficiente.

Características gerais dos usuários

- **Buscam produtividade e otimização do tempo:** precisam acessar conteúdos relevantes rapidamente, muitas vezes com prazos curtos.
- **Possuem diferentes níveis de familiaridade com tecnologia:** alguns usuários podem estar acostumados com ferramentas digitais de estudo, enquanto outros necessitam de uma interface mais simples e intuitiva.

- **Têm perfis variados de uso:** desde estudantes que precisam revisar matérias até profissionais que organizam projetos e documentos técnicos.  
**Utilizam múltiplos dispositivos:** esperam poder acessar o sistema em qualquer lugar, seja no computador, tablet ou celular.
- **Preferências de interação:** alguns preferem uma interface visual e interativa (organização em esferas), enquanto outros se sentem mais confortáveis com estruturas tradicionais de pastas e subpastas.

## Perfis principais de usuários

### 1. Estudantes (Ensino Médio, Graduação, Pós-graduação)

- **Necessidades:** resumos rápidos de textos extensos, criação de questionários para prática, explicações acessíveis.
- **Problemas enfrentados:** sobrecarga de leitura, falta de tempo para revisar todo o material antes de provas, dificuldade em organizar anotações e materiais digitais.

### 2. Profissionais e Pesquisadores

- **Necessidades:** organização de documentos técnicos, geração de resumos objetivos, extração de pontos-chave de artigos e relatórios.
- **Problemas enfrentados:** excesso de informação em múltiplos PDFs, dificuldade de manter organização em grandes projetos, pouco tempo para análise detalhada de todos os documentos.

### 3. Usuários Autônomos/Curiosos

- **Necessidades:** organizar projetos pessoais (ex.: cursos online, hobbies, estudos independentes) e acessar informações de forma simplificada.
- **Problemas enfrentados:** falta de ferramentas flexíveis que unam

organização e geração automática de conteúdos, necessidade de algo prático e acessível sem curva de aprendizado complexa.

Desafios que o sistema busca resolver

- Reduzir o tempo necessário para absorção de grandes volumes de conteúdo.
- Eliminar a dificuldade em centralizar e organizar arquivos de diferentes origens.
- Oferecer formas de revisão mais eficazes (resumos, questões e explicações).
- Disponibilizar acesso rápido e contínuo em qualquer dispositivo conectado.

Características gerais

- Foco principal: Estudantes de ensino superior graduação/pós) e pré-vestibulandos.
- Habilidades Técnicas: Básicas em navegação web; familiaridade com apps de estudo (Anki, Notion).
- Necessidades comuns:
  - Redução do tempo de síntese de conteúdos.
  - Preparação eficaz para provas com alto volume de matérias.
  - Centralização de recursos dispersos(PDFs, anotações, simulados)

## 2.5. Suposições e Restrições Gerais

- **Conexão de Internet Estável:** pressupõe-se que os usuários possuam acesso contínuo e estável à internet, condição necessária para upload de arquivos, processamento via servidores remotos e sincronização dos dados entre dispositivos.
- **Idioma dos Materiais:** na versão inicial (MVP), o sistema será otimizado para materiais em português. A compatibilidade com outros idiomas será considerada em versões futuras, alinhada ao requisito de internacionalização.
- **Qualidade de Arquivos para OCR:** documentos digitalizados devem ter

resolução mínima de **200 DPI**, garantindo legibilidade adequada para reconhecimento óptico de caracteres (OCR).

- **Dependência de Serviços Externos:** o sistema depende de APIs de terceiros (ex.: OpenAI GPT, OCR Service, sistema de pagamento), estando sujeito a alterações de custo, disponibilidade e políticas dessas plataformas.
- **Plataforma de Execução:** o sistema será desenvolvido como aplicação **web**, compatível com navegadores modernos (Chrome, Firefox, Edge, Safari). Não haverá suporte oficial a navegadores obsoletos ou dispositivos sem capacidade mínima de processamento.
- **Armazenamento em Nuvem:** todos os dados serão armazenados em serviços de nuvem confiáveis (ex.: AWS, Firebase), ficando sujeitos às políticas de armazenamento e limites desses provedores.
- **Escopo de Segurança:** o sistema implementará autenticação segura e proteção de dados conforme LGPD, mas a segurança da máquina do usuário (ex.: antivírus, firewall pessoal) não está sob responsabilidade do Corus.

### 3. Glossário

- **Usuário:** Entidade que representa qualquer pessoa cadastrada na plataforma Corus, capaz de interagir com todas as funcionalidades do sistema mediante autenticação. Engloba estudantes, professores e outros perfis de usuários que utilizam a plataforma para organização e geração de conteúdos de estudo.
  - **Atributos:**
    - id: Único, obrigatório, Identificador único do usuário no sistema
    - Nome Completo: até 100 caracteres, obrigatório - Nome civil completo do usuário
    - Email: String Formato válido, único, obrigatório - E-mail utilizado para login e comunicação
    - Senha: String Hash bcrypt, obrigatório - Senha criptografada

- Data de criação: Timestamp, Automático, obrigatório - Data e hora do cadastro no sistema
- Último login: Timestamp - Data e hora do último acesso
- Plano assinatura: Enum {free, premium, enterprise}
- Status da conta: Enum {ativa, inativa, suspensa}

○ **Exemplo prático:**

"id\_usuario": "usr\_5f8g3h2j1k",

"nome\_completo": "Marina Silva Oliveira",

"email": "marina.oliveira@email.com",

"senha": "M@rinaEstud@2024!"//Hash seguro: \$2b\$12\$T1y4G8sHqJkLmNpO2vC5De

"data\_criacao": "2024-09-15T10:30:00Z",

"ultimo\_login": "2024-10-18T14:22:18Z",

"plano\_assinatura": "premium",

"status\_conta": "ativa"

- **Documento original:** Arquivo digital enviado pelo usuário através da plataforma Corus, servindo como fonte primária para processamento e geração de conteúdos. Representa a matéria-prima inicial do sistema.

○ **Atributos:**

- id: UUID Único, obrigatório - Identificador único do documento
- Título original: até 100 caracteres - Nome original do arquivo
- Formato do documento: Enum{PDF, JPG, PNG, JPEG}
- Tamanho: Máx. 25 MB
- Usuário id: UUID, Obrigatório - ID do usuário proprietário
- Data upload: Timestamp, Automático - Data/hora do upload
- Hash do arquivo: String, SHA-256 - Hash para verificação de integridade
- Status upload: Enum {completo, falha, pendente}

- Caminho armazenamento: String - Local no cloud storage

- **Exemplo prático:**

```
"id_documento": "doc_orig_7f8g9h0i1j2k3l4m",
"titulo_original": "apostila-neuroanatomia-cap5.pdf",
"formato_midia": "PDF",
"tamanho": 18.5,
"usuario_id": "usr_a1b2c3d4e5f6",
"data_upload": "2024-10-18T14:25:33.120Z",
"hash_arquivo":
"e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855",
"status_upload": "completo",
"caminho_armazenamento":
"usuarios/usr_a1b2c3d4e5f6/documentos/originais/apostila-neuroanatomia-cap5.pdf",
"metadados": {
  "autor_original": "Prof. Silva",
  "data_criacao_original": "2023-08-15",
  "quantidade_paginas": 24
}
```

- **Documento gerado:** Conteúdo criado automaticamente pelo Corus através de processamento de IA, derivado do documento original. Inclui resumos, questões, explicações e outros materiais de estudo gerados automaticamente.

- **Atributos:**

- id conteudo gerado: UUID, Único, obrigatório
- Documento de origem id: UUID, Obrigatório
- Tipo do conteúdo: Enum {resumo, questao, explicacao, mapa\_mental} - Tipo de conteúdo gerado
- Formato de saída: Enum {texto, pdf, html, json} - Formato de output

- Conteúdo Texto - Conteúdo gerado (texto completo)
- Data geração: Timestamp, Automático - Data/hora da geração
- Modelo de ia utilizado: String - Versão do modelo de IA usado
- Versão de geração: Integer - Versão do conteúdo (para updates)
- Idioma: String, ISO 639-1 - Idioma do conteúdo gerado

- **Exemplo prático:**

```
"id_conteudo_gerado": "gen_5r6s7t8u9v0w1x2y",
"documento_origem_id": "doc_orig_7f8g9h0i1j2k3l4m",
"tipo_conteudo": "resumo",
"formato_saida": "texto",
"conteudo": "RESUMO - NEUROANATOMIA CAPÍTULO 5\n\nO sistema nervoso
autônomo divide-se em simpático e parassimpático...\n\n• Simpático: resposta de luta
ou fuga\n• Parassimpático: resposta de repouso e digestão\n\nPrincipais
neurotransmissores: noradrenalina (simpático) e acetilcolina (parassimpático).",
"data_geracao": "2024-10-18T14:35:47.220Z",
"modelo_ia_utilizado": "GPT-4o-2024-08-06",
"versao_geracao": 1,
"idioma": "pt",
"metricas_geracao": {
    "tempo_processamento": "45.2s",
    "tokens_utilizados": 1257,
    "confianca_geracao": 0.92
}
```

- **Questão:** Item avaliativo gerado automaticamente a partir do conteúdo de um documento, utilizado para revisão e teste de conhecimento.

- **Atributos:**

- Enunciado: texto da questão (até 500 caracteres)



- Tipo: domínio {múltipla escolha, verdadeiro/falso, dissertativa}
- Dificuldade: domínio {fácil, médio, difícil}
- Alternativas: 4 opções para questões de múltipla escolha
- Resposta Correta: identificador da alternativa correta ou texto explicativo
- Documento de Origem: relação com documento fonte

○ **Exemplo prático:**

Enunciado: "Qual o principal neurotransmissor envolvido na depressão?"

Tipo: múltipla escolha

Dificuldade: médio

Alternativas: ["Dopamina", "Serotonina", "Acetilcolina", "GABA"]

Resposta Correta: "Serotonina"

- **Resumo:** Síntese concisa gerada automaticamente a partir do conteúdo de um ou mais documentos.

○ **Atributos:**

- Conteúdo: texto resumido (até 8000 caracteres)
- Nível de Detalhe: domínio {básico, intermediário, avançado}
- Formato: domínio {tópicos, narrativo}
- Palavras-chave: até 40 termos principais
- Documentos Originais: relação com documentos fonte

○ **Exemplo prático:**

Conteúdo: "A depressão é um transtorno mental caracterizado por... Principais tratamentos incluem..."

Nível: intermediário

Formato: tópicos

Palavras-chave: ["depressão", "neurotransmissores", "SSRI"]

- **Esfera:** Estrutura visual principal de organização de conteúdos, representada como nó interativo.

- **Atributos:**

- Nome: identificador (até 50 caracteres)
- Cor: código hexadecimal (ex: #FF5733)
- Ícone: identificador do ícone visual
- Data de Criação: timestamp
- Usuário Proprietário: relação com usuário criador

- **Exemplo prático:**

Nome: "Neurociência"

Cor: #4A7B9D

Ícone: "brain"

Data de Criação: 2024-10-18 14:30:00

- **Sub-Esfera:** Esfera secundária hierarquicamente vinculada a uma esfera pai.

- **Atributos:**

- Herda todos os atributos de Esfera
- Esfera Pai: relação com esfera superior
- Nível de Profundidade: numérico (1-3)

- **Exemplo prático:**

Nome: "Neurotransmissores"

Esfera Pai: "Neurociência"

Nível: 2

- **Assinatura:** Plano de acesso às funcionalidades do sistema com diferentes limites de uso.

- **Atributos:**

- Tipo: domínio {free, premium, enterprise}
- Limite de Processamentos: número mensal de OCR/IA

- Limite de Armazenamento: em MB/GB
- Data de Início: timestamp
- Data de Término: timestamp
- Status: domínio {ativa, cancelada, expirada}
- **Exemplo prático:**

Tipo: premium

Limite Processamentos: 100/mês

Limite Armazenamento: 10GB

Status: ativa

- **Pagamento:** Transação financeira relacionada à assinatura do serviço.
  - **Atributos:**
    - Valor: numérico com duas casas decimais
    - Método: domínio {cartão, pix, boleto}
    - Status: domínio {pendente, processado, falha}
    - Data de Processamento: timestamp
    - ID da Transação: código do gateway de pagamento
  - **Exemplo prático:**

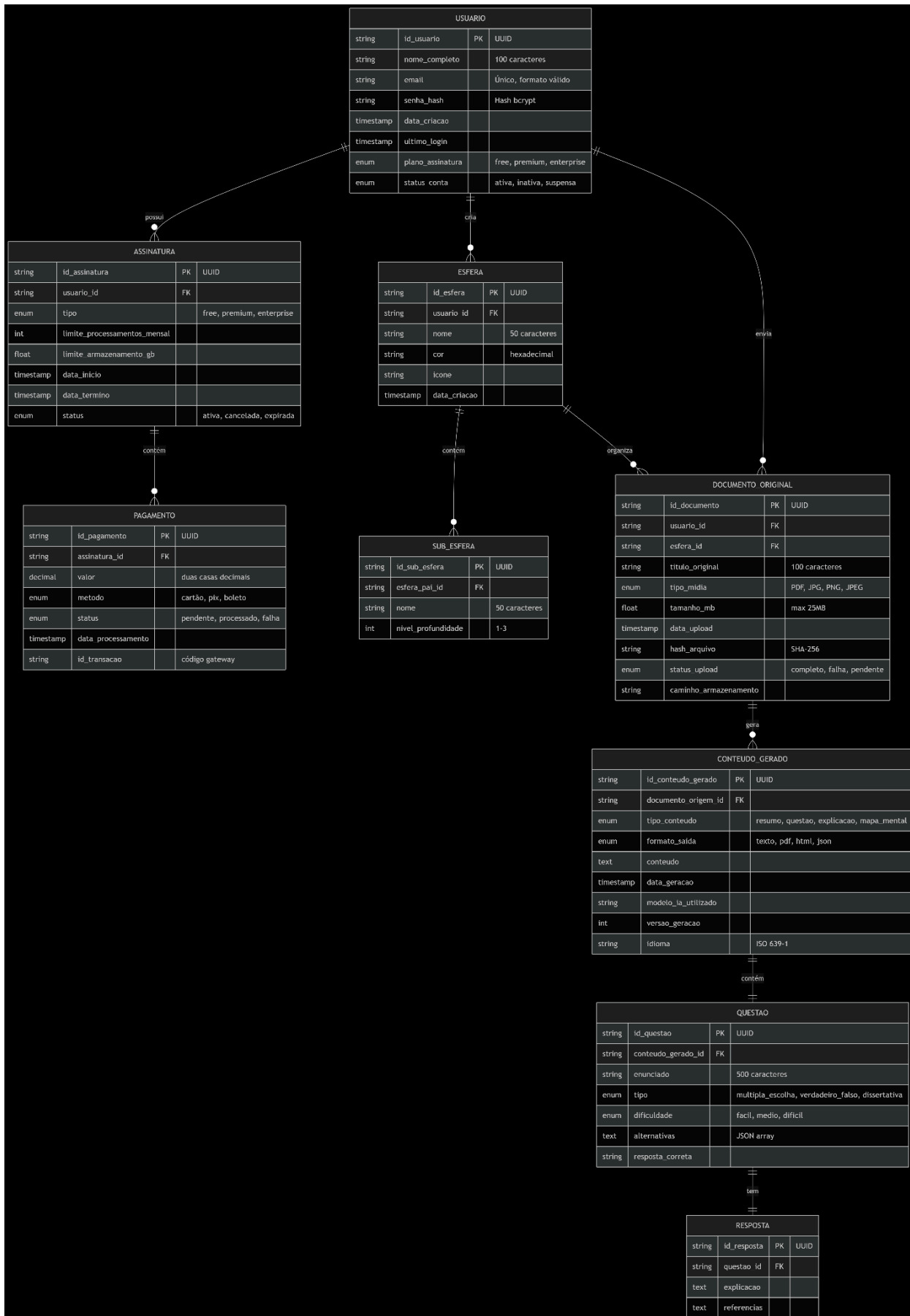
Valor: 29.90

Método: cartão

Status: processado

ID Transação: "ch\_1JABCDEFGHII"

## Diagrama Entidade-Relacionamento



## **4. Elicitação de Requisitos**

### **4.1. Técnica 1, sessão de Brainstorm**

#### **Justificativa:**

O brainstorming foi selecionado por sua eficiência em gerar ideias inovadoras em equipes coesas com conhecimento do domínio. Como alunos e usuários finais potenciais do sistema, a técnica permitiu explorar soluções criativas baseadas em nossas experiências reais com ferramentas de estudo.

#### **Detalhamento da Sessão:**

Data e Horário: 16 de Agosto de 2025, 19:00 às 21:30 (duração: 2h30)

Plataforma: Google Meet (videoconferência)

#### **Participantes:**

- Yan Feitosa
- Ruanderson Gabriel
- Felipe Apolinário
- Dmitri Verdi
- Pedro Henrique N.

Dinâmica da Sessão: Realizou-se a apresentação da ideia geral do Corus, seguida de uma discussão sobre possíveis abordagens de desenvolvimento e requisitos a serem incorporados ao sistema. A partir desse processo, definiu-se um escopo inicial para o projeto, bem como o público-alvo e os principais requisitos funcionais e não funcionais.

#### **Requisitos Definidos na Sessão:**

- **Requisitos Funcionais:**

1. O sistema deve permitir o upload de arquivos em PDF, DOCX e imagens escaneadas.
2. O sistema deve extrair automaticamente textos de imagens e PDFs escaneados via OCR.
3. O sistema deve gerar resumos automáticos a partir dos materiais carregados.
4. O sistema deve criar questões personalizadas de múltipla escolha e discursivas.
5. O sistema deve fornecer explicações passo a passo sobre os conteúdos estudados.
6. O sistema deve disponibilizar um painel de acompanhamento do progresso de estudo.
7. O sistema deve permitir exportar resumos e listas de questões em PDF.

- **Requisitos Não Funcionais:**

1. O sistema deve apresentar interface intuitiva e responsiva (usabilidade).
2. O sistema deve garantir a privacidade dos arquivos enviados (segurança).
3. O processamento de resumos e questões deve ocorrer em tempo razoável (desempenho).
4. A plataforma deve estar disponível via web e adaptada para dispositivos móveis.
5. O sistema deve ser escalável, suportando múltiplos usuários simultaneamente.

## **4.2. Técnica 2, aplicação de Formulário Online (Obs: Simulado)**

### **Justificativa:**

A técnica de formulário foi escolhida por sua praticidade e alcance, permitindo coletar dados de forma estruturada e organizada junto ao público-alvo. Diferente de reuniões síncronas, o formulário possibilitou que os participantes respondessem no seu próprio ritmo, gerando registros objetivos que facilitaram a análise posterior.

### **Detalhamento da Sessão:**

- Data e Horário de Disponibilização: 16 de Agosto de 2025, das 18h00 às 23h59
- Plataforma: Google Forms
- Participantes:
  - Yan Feitosa
  - Ruanderson Gabriel
  - Felipe Apolinário
  - Dmitri Verdi
  - Pedro Henrique Nogueira

### **Estrutura do Formulário:**

O formulário foi dividido em três seções principais:

1. Perfil do Usuário (ex.: frequência de estudos, ferramentas que já utiliza).
2. Necessidades e Dificuldades (ex.: principais obstáculos no processo de estudo).
3. Expectativas em relação ao Corus (ex.: funcionalidades desejadas, nível de importância de cada requisito).

### **Resultados Simulados:**

- Perfil: 80% dos participantes afirmaram estudar diariamente usando materiais em PDF; 60% utilizam resumos prontos encontrados online.

- **Necessidades/Dificuldades:** 100% relataram dificuldade em organizar o tempo de estudo; 80% destacaram a sobrecarga de conteúdo como principal problema.
- **Funcionalidades mais votadas:**
  - Geração automática de resumos → 5 votos (prioridade máxima)
  - Criação de questões personalizadas → 4 votos
  - Explicações passo a passo → 3 votos
  - Dashboard com progresso do estudo → 3 votos
  - OCR para leitura de PDFs escaneados → 2 votos

### **Conclusão:**

A coleta via formulário possibilitou identificar claramente as prioridades dos usuários potenciais. A análise dos resultados permitiu consolidar um escopo inicial do Corus, onde funcionalidades como geração de resumos e criação de questões foram priorizadas, enquanto outras, como OCR e dashboards de progresso, foram registradas como melhorias futuras.

## **Considerações finais**

A partir da aplicação das duas técnicas de elicitação - brainstorming estruturado e formulário online - foram identificados e validados os requisitos essenciais que orientarão o desenvolvimento do sistema Corus. Com base nas necessidades dos usuários e nas capacidades técnicas da equipe, estabelecemos que:

O sistema Corus deve incluir:

- Funcionalidades Principais:
- Upload e armazenamento de arquivos nos formatos PDF, DOCX e imagens
- Extração de texto via OCR para materiais digitalizados
- Geração automática de resumos concisos a partir dos conteúdos estudados
- Criação de questões personalizadas (múltipla escolha e discursivas)
- Explicações passo a passo sobre os conteúdos gerados
- Sistema de organização por esferas visuais e estrutura tradicional de pastas
- Painel de acompanhamento do progresso de estudo
- Exportação de conteúdos em formatos PDF e texto



- Processamento de pagamentos com múltiplos métodos e gestão de assinaturas

A interface deve ter/ou ser:

- Intuitiva e de fácil navegação
- Responsiva e adaptada para dispositivos móveis
- Visualmente organizada através do sistema de esferas coloridas
- Acessível com adequação aos padrões WCAG 2.1
- Personalizável em temas e preferências de exibição

O sistema deve oferecer/gerar/enviar:

- Resumos automáticos com três níveis de detalhamento
- Questões contextualizadas com diferentes níveis de dificuldade
- Explicações detalhadas para cada resposta gerada
- Recibos digitais e comprovantes de pagamento
- Notificações sobre status de processamento e atividades
- Relatórios de progresso e desempenho de estudo
- Links compartilháveis com controle de acesso e validade

Requisitos não funcionais assegurados:

- Segurança na armazenagem de dados e privacidade dos arquivos
- Desempenho adequado no processamento de conteúdos
- Escalabilidade para suportar múltiplos usuários simultâneos
- Confiabilidade no processamento de pagamentos
- Disponibilidade contínua durante períodos de alta demanda
- Manutenibilidade com documentação abrangente
- Conformidade com a LGPD para tratamento de dados pessoais

Estes requisitos, consolidados a partir das técnicas de elicitação aplicadas, formam a base sólida para o desenvolvimento da primeira versão do Corus, garantindo que o sistema atenda tanto às necessidades imediatas dos usuários quanto permita expansões futuras.

## **5. Análise de Requisitos**

### **5.1. Requisitos funcionais**

[RF 01] Criar Conta

Descrição:

Permitir cadastro de novos usuários via formulário com validação em tempo real.

Casos de uso relacionados:

-

Prioridade: Essencial

[RF 02] Login

Descrição:

Sistema deve verificar credenciais (e-mail/senha) contra banco de dados.

Casos de uso relacionados:

-

Prioridade: Essencial

[RF 03] Atualização de Perfil

Descrição:

Permitir alteração de: nome, foto de perfil e senha (com confirmação da senha atual).

Notificar por e-mail em caso de mudanças críticas.

Casos de uso relacionados:

-

Prioridade: Desejável

[RF 04] Gestão de Assinaturas

Descrição:

Interface para:

- Selecionar planos (Free/Premium)
- Visualizar benefícios de cada plano

- Upgrade/Downgrade
- Cancelamento com retenção de dados

Casos de uso relacionados:

-

Prioridade: Essencial

## [RF 05] Processamento de Pagamentos

Descrição:

Integração com APIs de pagamento (Stripe/Mercado Pago) para:

- Armazenamento seguro de cartões
- Cobrança recorrente
- Retentativas automáticas
- Geração de recibos digitais

Casos de uso relacionados:

-

Prioridade: Essencial

### [RF 05.01] Integração com Gateway de Pagamento

Descrição:

Implementar conexão segura com APIs de pagamento (Stripe/Mercado Pago) utilizando chaves de API criptografadas e seguindo padrões PCI DSS.

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 05.02] Armazenamento Seguro de Cartões

##### Descrição:

Utilizar tokenização para armazenamento seguro de informações de cartão de crédito, sem reter dados sensíveis no sistema local.

##### Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 05.03] Cobrança Recorrente Automática

##### Descrição:

Implementar sistema de cobrança automática mensal/anual conforme plano selecionado, com notificação prévia ao usuário.

##### Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 05.04] Sistema de Retentativas

##### Descrição:

Implementar lógica de retentativas automáticas para cobranças recusadas, com:

- 3 tentativas em intervalos de 72h
- Suspensão automática após falhas consecutivas
- Notificação ao usuário sobre falhas

##### Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 05.05] Geração de Recibos Digitais

Descrição: Emitir recibos em formato PDF após cada transação bem-sucedida, contendo:

- Dados da transação
- Informações do serviço
- Dados fiscais obrigatórios

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 05.06] Gestão de Reembolsos

Descrição:

Implementar fluxo para processamento de reembolsos parciais ou totais conforme política de cancelamento.

Casos de uso relacionados:

-

Prioridade: Importante

#### [RF 05.07] Notificações de Transação

Descrição:

Enviar notificações por e-mail para:

- Confirmação de pagamento
- Falha na transação
- Atualização de método de pagamento

Casos de uso relacionados:

-

Prioridade: Importante

#### [RF 05.08] Painel de Faturas

Descrição:

Disponibilizar histórico completo de transações e faturas para consulta do usuário.

Casos de uso relacionados:

-

Prioridade: Importante

#### [RF 05.09] Validação de Dados de Pagamento

Descrição:

Implementar validação em tempo real de:

- Número de cartão válido
- Data de validade
- Código de segurança

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 05.10] Suporte a Múltiplos Métodos

Descrição:

- Implementar suporte a:
- Cartão de crédito/débito

- PIX
- Boleto bancário

Casos de uso relacionados:

-

Prioridade: Desejável

#### [RF 06] Criação de Esferas Base

Descrição:

Permitir criação de esferas-mãe com: nome, ícone e descrição. Limite inicial: 5 esferas/usuário.

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 07] Gestão de Sub-Esferas

Descrição:

Criar espaços hierárquicos (máx. 3 níveis) com arrastar/soltar. Opção de herdar propriedades da esfera-pai.

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 08] Modo de Organização Tradicional

Descrição:

Interface alternativa com pastas/subpastas e visualização em lista. Sincronização

bidirecional com estrutura de esferas.

Casos de uso relacionados:

-

Prioridade: Importante

#### [RF 09] Upload de Materiais de Estudo

Descrição:

O sistema deve permitir upload de arquivos nos formatos PDF, JPG e PNG através de arrastar-e-soltar ou seleção manual. Deve validar tipo/tamanho(máx. 25MB) e exibir pré-visualização para imagens.

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 10] Processamento OCR Inteligente

Descrição:

Para arquivos de imagem e PDFs não-textuais, o sistema deve aplicar OCR com correção automática de erros (ex: confusão entre "o" e "0"), preservação de estruturas complexas (tabelas, fórmulas).

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 11] Geração de Resumos Concisos

Descrição:



O sistema deve gerar resumos, mantendo conceitos-chave, relações causais e terminologia técnica. Usuário poderá ajustar nível de detalhe (básico/avançado) e receber versão em formato de tópicos ou narrativo.

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 12] Criação de Questões Personalizadas

Descrição:

O sistema deve gerar questões por material com as seguintes características:

- 5-15 questões/documento
- Tipos: múltipla escolha, V/F, dissertativas
- Dificuldade ajustável
- Tags temáticas automáticas

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 13] Explicações Detalhadas

Descrição:

Fornecer para cada questão:

- Resposta correta fundamentada
- Explicação passo a passo
- Referências ao texto original

Casos de uso relacionados:

-

Prioridade: Essencial

[RF 14] Compilação de Conteúdo por Esfera

Descrição:

Gerar relatórios unificados contendo:

- Resumo agregado dos materiais
- Banco de questões consolidado
- Mapa conceitual da esfera

Casos de uso relacionados:

-

Prioridade: Essencial

[RF 15] Biblioteca de Estudos

Descrição:

Exibir materiais organizados por:

- Tipo (PDF, imagem, texto)
- Data de upload
- Esfera associada
- Status de processamento

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RF 16] Acesso a Conteúdos Gerados

##### Descrição:

Visualização unificada de: resumos, questões e explicações vinculados a cada material/esfera, com histórico de versões.

##### Casos de uso relacionados:

- 

Prioridade: Essencial

#### [RF 17] Exclusão de Arquivos

##### Descrição:

##### Remoção segura com:

- Confirmação em 2 etapas
- Exclusão em cascata (conteúdos gerados)
- Período de recuperação (7 dias)

##### Casos de uso relacionados:

- 

Prioridade: Importante

#### [RF 18] Exportação de Conteúdo

##### Descrição:

##### Download em formatos:

- PDF (com formatação preservada)
- TXT (texto puro)
- CSV (para questões)

##### Casos de uso relacionados:

-

Prioridade: Desejável

## **5.2. Requisitos não funcionais**

### **5.2.1 Usabilidade**

[RNF 01] Interface intuitiva

Métrica:  $\geq 80\%$  das tarefas concluídas sem ajuda em teste de usabilidade. O usuário deve ser capaz de realizar as operações básicas do sistema (criar tarefa, editar tarefa, concluir tarefa) sem treinamento em até 5 minutos de uso.

Critério de Aceitação:  $\geq 80\%$ .

Casos de uso relacionados:

-

Prioridade: Essencial

[RNF 02] Feedback de ações

Métrica: Todas as interações devem ter feedback visual ou sonoro imediato para os eventos de carregamento, sucesso, erro e confirmação.

Casos de uso relacionados:

-

Prioridade: Desejável

[RNF 03] Consistência da interface

Métrica: Manter padrão de cores, tipografia e layout em todas as telas para facilitar navegação.

Casos de uso relacionados:

-

Prioridade: Importante

[RNF 04] Acessibilidade

Métrica: Conformidade com diretrizes WCAG 2.1 nível AA, garantindo acessibilidade a usuários com deficiência visual ou motora.

Casos de uso relacionados:

-

Prioridade: Essencial

### **5.2.2 Desempenho**

[RNF 05] Tempo de resposta da interface

Métrica: Resposta  $\leq 2s$  para operações comuns (CRUD, abrir/salvar tarefas);  $\leq 15s$  para geração via LLM.

Critério:  $\geq 95\%$  dos casos atendidos.

Casos de uso relacionados:

-

Prioridade: Essencial

[RNF 06] Escalabilidade

Métrica: O sistema deve suportar até 10.000 usuários simultâneos sem degradação perceptível de desempenho.

Critério: Testes de carga confirmam.

Casos de uso relacionados:

-

Prioridade: Importante

[RNF 07] Processamento em lote

Métrica: Importação de até 5.000 tarefas em planilha deve ser concluída em  $\leq 60$  segundos.

Casos de uso relacionados:

-

Prioridade: Essencial

### **5.2.3 Confiabilidade**

[RNF 08] Disponibilidade

Métrica: Sistema disponível  $\geq 99,5\%$  do tempo mensal, exceto manutenções programadas.

Critério: Monitoramento confirmar disponibilidade

Casos de uso relacionados:

-

Prioridade: Essencial

[RNF 09] Recuperação de falhas

Métrica: Recuperação em até 5 minutos após restabelecimento do serviço em caso de falha inesperada.

Casos de uso relacionados:

-

Prioridade: Essencial

[RNF 10] Backup automático

Métrica: Backups automáticos diários de dados críticos do banco e storage.

Critério: Recuperação bem-sucedida em falhas

Casos de uso relacionados:

-

Prioridade: Essencial

#### **5.2.4 Segurança**

[RNF 11] Autenticação

Métrica: Uso de JWT, hash seguro (bcrypt), senhas criptografadas e suporte a autenticação de dois fatores (2FA).

Critério: Sem armazenamento de senha em texto puro

Casos de uso relacionados:

-

Prioridade: Essencial

[RNF 12] Autorização

Métrica: Controle de permissões por papel (administrador, usuário comum, convidado), restringindo acesso a funcionalidades específicas.

Casos de uso relacionados:

-

Prioridade: Essencial

[RNF 13] Criptografia de dados

Métrica: Dados sensíveis armazenados com AES-256 e transmitidos via HTTPS (TLS 1.3).

Casos de uso relacionados:

-

Prioridade: Essencial

### **5.2.5 Portabilidade**

[RNF 14] Multiplataforma

Descrição: O sistema deve ser totalmente funcional nos navegadores Google Chrome, Mozilla Firefox, Microsoft Edge e Safari em suas duas versões mais recentes. Além disso, deve possuir interface responsiva que se adapte automaticamente a telas de smartphones e tablets sem perda de funcionalidade.

Métrica: Testes de compatibilidade devem confirmar funcionamento em  $\geq 95\%$  das combinações de navegador + versão suportada.

Critério de Aceitação: Interface e funcionalidades críticas (login, CRUD de tarefas, notificações) devem ser executadas corretamente em desktop e dispositivos móveis.

Casos de uso relacionados:

-

Prioridade: Essencial

[RNF 15] Compatibilidade entre Sistemas Operacionais

Descrição: O cliente web deve ser acessível e funcional em Windows 10 ou superior, Ubuntu 20.04 ou superior e macOS 12 ou superior, sem necessidade de instalação adicional de software ou plug-ins de terceiros.



Métrica: Testes de execução devem validar a compatibilidade em  $\geq 95\%$  das combinações de navegador + sistema operacional suportados.

Critério de Aceitação: Usuários em qualquer um dos sistemas operacionais suportados devem acessar e utilizar todas as funcionalidades sem erros críticos.

Casos de uso relacionados:

-

Prioridade: Essencial

### **5.2.6 Compliance**

[RNF 16] LGPD Compliance

Métrica: Dados pessoais tratados em conformidade com LGPD.

Critério: Revisão legal concluída

Casos de uso relacionados:

-

Prioridade: Essencial

### **5.2.7 Internacionalização**

[RNF 17] Suporte a idiomas

Métrica: Sistema preparado (i18n-ready) para múltiplos idiomas.

Critério: Estrutura validada para tradução

Casos de uso relacionados:

-

Prioridade: Importante

### **5.2.8 Testabilidade**

#### [RNF 18] Cobertura de Testes Automatizados

Descrição: O sistema deve possuir suíte de testes automatizados (unitários, integração e ponta a ponta) cobrindo no mínimo 80% do código-fonte.

Métrica: Relatório de cobertura  $\geq 80\%$  gerado por ferramenta de CI/CD.

Critério de Aceitação: Build rejeitada caso cobertura  $< 80\%$ .

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RNF 19] Integração Contínua

Descrição: Todo commit deve acionar pipeline de integração contínua que execute testes automatizados e análise estática de código.

Métrica: 100% dos commits passam pelo pipeline.

Critério de Aceitação: Deploy permitido apenas se pipeline for concluído com sucesso.

Casos de uso relacionados:

-

Prioridade: Importante

### **5.2.9 Manutenibilidade**

#### [RNF 20] Tempo de Correção de Bugs Críticos

Descrição: Bugs classificados como críticos (impacto em funcionalidades principais ou segurança) devem ser corrigidos e disponibilizados em produção em no máximo 24

horas após detecção.

Métrica: SLA de correção  $\leq 24h$  para bugs críticos.

Critério de Aceitação: 100% dos bugs críticos corrigidos dentro do prazo.

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RNF 21] Documentação Técnica

Descrição: O sistema deve possuir documentação atualizada de arquitetura, APIs e fluxos de negócio, disponível para a equipe de desenvolvimento.

Métrica: Atualização realizada até 48h após cada alteração relevante.

Critério de Aceitação: Revisão semestral confirma consistência entre documentação e sistema.

Casos de uso relacionados:

-

Prioridade: Importante

### **5.2.10 Observabilidade**

#### [RNF 22] Logs Centralizados

Descrição: Todos os serviços devem enviar logs estruturados (JSON) para um sistema centralizado de observabilidade.

Métrica:  $\geq 99\%$  dos eventos críticos registrados em central de logs.

Critério de Aceitação: Validação em testes de carga e falhas simuladas.

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RNF 23] Alertas Automáticos

Descrição: O sistema deve gerar alertas automáticos em caso de falhas críticas (ex.: indisponibilidade, erro 5xx, falha de autenticação em massa).

Métrica: Tempo de notificação  $\leq 2$  minutos após ocorrência.

Critério de Aceitação: Equipe de suporte notificada em todos os incidentes simulados.

Casos de uso relacionados:

-

Prioridade: Essencial

#### [RNF 24] Dashboards de Métricas

Descrição: Deve haver dashboards em tempo real exibindo métricas de desempenho, uso e erros do sistema (ex.: tempo de resposta médio, taxa de erros, número de usuários ativos).

Métrica: Disponibilidade de dashboards  $\geq 99\%$ .

Critério de Aceitação: Métricas atualizadas em tempo real ( $\leq 30s$  de defasagem).

Casos de uso relacionados:

-

Prioridade: Importante

## 6. Especificação de Requisitos

<Nesta seção, apresente **APENAS 3** casos de uso mais importantes/relevantes do sistema. Em sistemas grandes é comum haver muitos casos de uso e, para facilitar a visualização deste documento, você pode agrupá-los em subseções de casos de uso correlacionados. Os nomes das subseções devem ser únicos e pequenos (3 palavras no máximo) e podem ser formados por palavras, números e/ou abreviações.

Cada um dos casos de uso deve ser descrito em um bloco específico, seguindo o modelo descrito abaixo. O identificador do bloco deve conter o número do caso de uso (por exemplo, [UC001]) e o seu nome. Se os casos de uso forem agrupados em subseções específicas, a numeração deles deve ser reiniciada a cada subseção (dentro de uma mesma subseção, todo caso de uso deve ter um número de identificação único).

Quando a primeira versão deste documento for disponibilizada para a equipe de desenvolvimento, os nomes das subseções e os números dos casos de uso não devem ser modificados ou reaproveitados, para não invalidar referências externas feitas a eles.>

### <Nome de subseção para agrupar casos de uso correlacionados>

<Utilize este espaço para descrever características comuns dos casos de uso desta seção, explicitando o motivo do seu agrupamento em uma seção única.

Se todos os casos de uso desta seção estiverem relacionados com o mesmo ator você pode informar isso aqui, especificando qual é o ator em questão, e eliminar o campo “Ator:” das descrições dos casos de uso feitas nos blocos a seguir.>

[UC001] <Nome do caso de uso>

<Opcional – forneça uma pequena explicação do propósito do caso de uso (útil quando o nome do caso de uso não deixa suficientemente claro qual é o seu objetivo) e o(s) seu(s) respectivo(s) ator(es). Em seguida, substitua um dos símbolos abaixo por p, para indicar a prioridade do caso de uso.>

**Ator:** <informe o(s) ator(es) do caso de uso >

<b>Prioridade:</b>		Essencial		Importante		Desejável
--------------------	--	-----------	--	------------	--	-----------

**<Opcional> Interface(s) associada(s):** <inclua aqui o(s) identificador(es) da(s) respectiva(s) interface(s) do caso de uso (descrita(s) na Seção 5).>

**Entradas e pré condições:** <Liste aqui todas as entradas e/ou pré condições do caso de uso. Pré condição de um caso de uso é o estado em que o sistema deve estar para realizar o caso de uso.>

**Saídas e pós condições:** <Liste aqui todas as saídas e/ou pós condições do caso de uso. Pós condição de um caso de uso é a lista de possíveis estados em que o sistema pode estar imediatamente após o término da realização do caso de uso.>

#### **Fluxo de eventos principal**

<Descreva aqui o fluxo de eventos principal que ocorre durante a execução do caso de uso.>

#### **<Opcional> Fluxos secundários (alternativos e de exceção)**

<Fluxo secundário XXX>

<Use este espaço para descrever o fluxo secundário XXX do caso de uso.>

<Fluxo secundário YYY>

<Prossiga na descrição dos fluxos secundários do caso de uso, descrevendo cada um deles separadamente.>

[UC002] <Nome de outro caso de uso>

<Utilize os mesmos campos mostrados no bloco anterior para descrever este e os demais requisitos funcionais (casos de uso) desta subseção.>

#### **<Nome de outra subseção para agrupar outros casos de uso correlacionados>**

<Prossiga de maneira similar à subseção anterior para descrever quaisquer outras subseções que forem usadas para agrupar requisitos funcionais.>

## **6.1. Diagrama Casos de Uso**

<Inserir aqui o diagrama de casos de uso que contemple todos os requisitos funcionais>

## **7. Análise de casos de uso (diagrama de classes de análise)**

<Esta seção deve conter um diagrama de classes com a análise de todos casos de uso levantados anteriormente. Não é necessário incluir atributos e métodos nas classes. Porém, os relacionamentos entre as classes (associações, herança e dependências) devem ser já ser definidos>.

## **8. Descrição da interface com o usuário**

<Esta seção deve conter desenhos ou rascunhos (mockups) das telas do sistema que forem necessários ou convenientes para esclarecer algum dos requisitos do sistema. Para sistemas que possuem protótipos ou versões já desenvolvidas é possível capturar as telas e apresentar figuras das mesmas.



## **9. Diagramas de Arquitetura**

<Esta seção deve conter pelo menos um diagrama inicial de arquitetura lógica e física do sistema. Estudar livro referência da disciplina. A arquitetura deve responder como serão contemplados cada requisito não-funcional>