

# **Documento de requisitos do Cod.ed**

**Versão 1.0**

---

**SUMÁRIO**

---

<b>1. Introdução</b>	<b>3</b>
<b>1.1. Finalidade</b>	<b>3</b>
<b>1.2. Escopo</b>	<b>3</b>
<b>1.3. Definições acrônimos e abreviações</b>	<b>3</b>
<b>1.4. Visão Geral</b>	
<b>2. Requisitos do Sistema</b>	<b>3</b>
<b>2.1. Requisitos Funcionais</b>	<b>3</b>
<b>2.2. Requisitos Não Funcionais</b>	
<b>2.3. Regras de Negócio</b>	
<b>3. Critérios de Aceitação do Produto</b>	

## **1. Introdução**

### **1.1. Finalidade**

Este documento especifica oficialmente os requisitos funcionais, requisitos não funcionais e estabelece o escopo do Sistema de aprendizado online Cod.ed bem como restrições e informações do domínio. Serve como base para a equipe de desenvolvimento, na produção de outros artefatos do sistema e como forma de demonstrar suas funcionalidades para o cliente e usuários finais, ao longo de todo o ciclo de vida do produto.

### **1.2. Escopo**

O produto desenvolvido a partir deste documento de requisitos e critérios de aceitação é o Sistema de aprendizado online Cod.ed. Seu objetivo é facilitar a busca por informações relacionadas a problemas e soluções no rumo da programação afim de facilitar o aprendizado e facilitar o caminho da informação e conhecimento.

### **1.3. Definições acrônimos e abreviações**

Esta seção fornece as definições de todos os termos, acrônimos e abreviações requeridas para interpretar este documento de requisitos.

As abreviações usadas no documento são:

RF00X – Requisito Funcional;

RNF00X – Requisito Não Funcional;

RN00X – Regra de Negócio (requisitos de domínio).

#### 1.4. Visão Geral

Além desta seção introdutória, as seções seguintes estão organizadas como descrito abaixo.

1. **Seção 2 – Requisitos funcionais:** apresenta os requisitos funcionais do sistema juntamente com a descrição de cada requisito;
2. **Seção 3 – Requisitos não funcionais:** apresenta os requisitos não funcionais do sistema juntamente com a descrição de cada requisito;
3. **Seção 4 – Regras de Negócio:** apresenta as regras de negócio do sistema juntamente com a descrição de cada uma delas.

## 2. Requisitos do Sistema

### 2.1. Requisitos Funcionais

*RF001: O sistema deve fornecer recomendações de cursos relevantes com base nas pesquisas recentes do usuário, levando em consideração os tópicos de interesse identificados.*

*RF002: O sistema deve permitir que o usuário realize pesquisas sucessivas no mesmo contexto, mantendo a interatividade entre as solicitações e fornecendo resultados relevantes para cada pesquisa.*

*RF003: O sistema deve ser capaz de buscar, explicar e detalhar o funcionamento e a lógica de códigos fornecidos pelo usuário, permitindo a compreensão de códigos de autoria externa.*

*RF004: O sistema deve sugerir melhorias para o código fornecido pelo usuário.*

*RF005: O sistema deve fornecer informações específicas sobre a correção de bugs, permitindo que o usuário resolva problemas de forma eficiente e economize tempo.*

*RF006: O sistema deve apresentar recomendações de conteúdo educacional em uma seção separada, facilitando o acesso e evitando confusões e distrações desnecessárias.*

*RF007: O sistema deve permitir que o usuário consulte erros em seus códigos e tire dúvidas de forma interativa.*

*RF008: O sistema deve oferecer acesso a fóruns e comunidades de programação, permitindo ao usuário interagir, conversar e obter respostas para problemas específicos.*

*RF009: O sistema não deve salvar os códigos compartilhados pelos programadores, a fim de evitar vazamento de dados sensíveis.*

*RF010: O sistema deve fornecer explicações teóricas detalhadas sobre bugs, permitindo que os programadores compreendam a origem e as soluções para esses problemas.*

*RF011: O sistema deve oferecer acesso a conteúdos base relacionados ao assunto pesquisado pelo programador, proporcionando uma variedade de opções para melhor compreensão de seu questionamento.*

*RF012: O sistema deve permitir a busca de soluções por nível de relevância, para que os programadores possam resolver problemas com maior precisão e otimização de tempo.*

*RF013: O sistema deve oferecer a possibilidade de filtrar soluções por linguagens de programação, permitindo que os programadores encontrem soluções específicas para suas linguagens de preferência, economizando tempo na resolução de problemas.*

*RF014: O sistema deve permitir que os programadores busquem resoluções para problemas relacionados ao trabalho, visando otimizar o desempenho e a eficiência.*

*RF015: O sistema deve fornecer recomendações de vídeo aulas explicativas e intuitivas para os estudantes de programação, oferecendo um ambiente seguro para aprendizado.*

*RF016: O sistema deve apresentar informações relevantes sobre bugs, explicando sua origem e impactos, de forma a auxiliar na solução efetiva dos problemas, evitando soluções paliativas.*

*RF017: O sistema deve ser capaz de recomendar conteúdo acessível com base no nível técnico do usuário, permitindo uma melhor compreensão dos conteúdos de acordo com o seu nível de habilidade.*

*RF018: O sistema deve oferecer recomendações de conteúdo em PDF, como livros, facilitando o acesso e o download dos materiais recomendados de forma fácil e rápida.*

*RF019: O sistema deve oferecer um mecanismo de feedback, permitindo que os usuários avaliem a qualidade das respostas e soluções fornecidas, contribuindo para a melhoria contínua do sistema.*

*RF020: O sistema deve ter a capacidade de se adaptar às preferências dos usuários, aprendendo com as interações e fornecendo resultados cada vez mais personalizados e relevantes ao longo do tempo.*

*RF021: O sistema deve oferecer suporte a diferentes linguagens de programação, permitindo que os programadores selecionem a linguagem de sua preferência ao buscar por soluções e respostas.*

*RF022: O sistema deve ter uma opção para alternar para o modo seguro.*

*RF023: O sistema deve possuir um modo seguro onde todas as pesquisas e códigos compartilhados ficam seguros da internet e do sistema em si.*

## **2.2. Requisitos Não Funcionais**

*RNF001: O sistema deve atender a requisitos de desempenho, sendo capaz de lidar com um alto volume de solicitações simultâneas de usuários, fornecendo uma resposta rápida e eficiente.*

*RNF002: O sistema deve garantir alta disponibilidade, estando disponível e acessível aos usuários na maior parte do tempo. Medidas como redundância em servidores, balanceamento de carga, monitoramento constante e planos de contingência podem ser necessárias para alcançar esse objetivo.*

*RNF003: O sistema deve garantir a segurança dos usuários e dos dados, implementando medidas como criptografia de dados, autenticação de usuários, proteção contra ataques de injeção de código e adoção de políticas de privacidade.*

*RNF004: O sistema deve ser projetado para ser escalável, capaz de lidar com o aumento no número de usuários, recursos e dados sem comprometer o*

*desempenho. Estratégias de escalabilidade horizontal, como o uso de serviços em nuvem e balanceadores de carga, podem ser consideradas.*

*RNF005: O sistema deve ser intuitivo e fácil de usar, fornecendo uma experiência amigável e eficiente para os usuários. Uma interface de usuário bem projetada, recursos de pesquisa e filtragem eficientes podem contribuir para a usabilidade do sistema.*

*RNF006: O sistema deve ser desenvolvido de forma modular e bem documentada, facilitando a manutenção, atualização e expansão futura. O uso de código limpo, boas práticas de desenvolvimento e ferramentas de controle de versão podem auxiliar na manutenção do sistema.*

*RNF007: O sistema deve ser confiável e consistente em suas respostas, garantindo que os resultados gerados sejam consistentes e corretos.*

*RNF008: O sistema deve ser capaz de se integrar com recursos externos, como vídeos, cursos, links e sites. Isso pode envolver o desenvolvimento de APIs, integração com plataformas de e-learning ou serviços de terceiros, e garantir a compatibilidade de formatos de dados e comunicação.*

*RNF009: O sistema deve permitir personalização por parte dos usuários, possibilitando que configurem suas preferências, como seleção de categorias de respostas, linguagens específicas ou relevância.*

*RNF010: O sistema deve ser facilmente testável, permitindo a realização de testes para garantir a precisão e relevância das respostas geradas. O desenvolvimento de testes automatizados, pode ser considerado para garantir a qualidade do sistema.*

## **2.3 Regras de Negócio**

*RN001: As buscas devem ser apenas de conteúdos relacionados a programação e desenvolvimento;*



*RN002: Códigos e buscas no modo seguro não podem em hipótese alguma serem expostas ou vazadas;*

*RN003: Todo conteúdo recomendado ou usado na plataforma deve ser de fontes seguras e atualizadas;*

*RN004: O sistema deve estar em conformidade com regulamentações de proteção de dados e privacidade, garantindo que as informações dos usuários sejam tratadas de forma segura e de acordo com as leis vigentes.*

### **3. Critérios de aceitação do produto**

*CA001: O sistema deve ser capaz de fornecer recomendações de cursos relevantes com base nas pesquisas recentes do usuário, considerando os tópicos de interesse identificados.*

*- Critério de Aceitação: Os cursos recomendados devem estar relacionados aos interesses do usuário, demonstrando relevância e correspondência com as pesquisas realizadas.*

*CA002: O sistema deve permitir que o usuário realize pesquisas sucessivas no mesmo contexto, mantendo a interatividade entre as solicitações e fornecendo resultados relevantes para cada pesquisa.*

*- Critério de Aceitação: As pesquisas sucessivas devem retornar resultados atualizados e relacionados ao contexto das pesquisas anteriores, garantindo a continuidade e coerência das informações fornecidas.*

*CA003: O sistema deve buscar, explicar e detalhar o funcionamento e a lógica de códigos fornecidos pelo usuário, permitindo a compreensão de códigos de autoria externa.*

*- Critério de Aceitação: As explicações e detalhamentos dos códigos fornecidos devem ser claros, precisos e abordar os aspectos relevantes para a compreensão do funcionamento e da lógica dos códigos.*

*CA004: O sistema deve sugerir melhorias para o código fornecido pelo usuário, visando aprimorar os projetos pessoais de programação.*

*- Critério de Aceitação: As sugestões de melhorias devem ser relevantes, específicas e ajudar os usuários a aprimorar seus códigos, considerando boas práticas e eficiência.*

*CA005: O sistema deve fornecer informações específicas sobre a correção de bugs, permitindo que o usuário resolva problemas de forma eficiente e economize tempo.*

*- Critério de Aceitação: As informações fornecidas sobre a correção de bugs devem ser precisas, completas e auxiliar os usuários na identificação e solução dos problemas de maneira efetiva.*

*CA006: O sistema deve apresentar recomendações de conteúdo educacional em uma seção separada, facilitando o acesso e evitando confusões e distrações desnecessárias.*

*- Critério de Aceitação: As recomendações de conteúdo educacional devem ser organizadas, de fácil acesso e fornecer opções relevantes e diversificadas para os usuários.*

*CA007: O sistema deve permitir que o usuário consulte erros em seus códigos e tire dúvidas de forma interativa, sem envolver outras pessoas.*

*- Critério de Aceitação: O sistema deve fornecer uma interface interativa e intuitiva para a consulta de erros e dúvidas, permitindo ao usuário obter respostas e esclarecimentos de forma autônoma.*

*CA008: O sistema deve oferecer acesso a fóruns e comunidades de programação, permitindo ao usuário interagir, conversar e obter respostas para problemas específicos.*

*- Critério de Aceitação: O sistema deve fornecer uma plataforma de fórum que permita a interação entre os usuários, facilitando a troca de conhecimentos, discussões e obtenção de respostas relevantes.*

