

**REQUISITOS  
SISTEMA DE FUNCIONÁRIOS**

**Autor(a):** Nayara Emelly Pereira Mendonça

**Disciplina:** Engenharia de Requisitos e Análise  
de Requisitos

**Profº:** Gioliano Bertoni

Saquarema - RJ

2025

## 1. INTRODUÇÃO

**Nome do sistema:** Sistema de Gerenciamento de Funcionários

O sistema tem como objetivo permitir o cadastro, a listagem e a busca de funcionários, armazenando seus dados (nome, cargo e salário) em memória durante a execução. Trata-se de um sistema simples, executado via terminal, voltado para uso interno, como por exemplo pelo setor de Recursos Humanos (RH).

## 2. TÉCNICAS UTILIZADAS NO LEVANTAMENTO

### 2.1 Entrevista com Stakeholders

Consiste em conversar diretamente com os envolvidos no projeto (usuários, clientes e gestores) com o objetivo de identificar expectativas, necessidades e problemas existentes. Essa técnica fornece informações reais e confiáveis sobre o que o sistema precisa fazer.

### 2.2 Brainstorming

Trata-se de uma técnica colaborativa usada para reunir diversas ideias em grupo. Durante o brainstorming, não há julgamentos; o foco é gerar o máximo de sugestões possíveis para, posteriormente, selecionar e refinar as mais adequadas ao projeto.

### 2.3 Prototipagem

É a criação de um modelo simplificado do sistema, permitindo validar requisitos com o usuário antes da implementação final. A prototipagem ajuda a detectar falhas, ajustar funcionalidades e garantir que o sistema atenda aos objetivos desejados.

### 3. REQUISITOS FUNCIONAIS

#### [RF001] Adicionar Funcionário

O sistema deve permitir cadastrar um funcionário informando nome, cargo e salário.

#### [RF002] Listar Funcionários

O sistema deve exibir todos os funcionários cadastrados, mostrando nome, cargo e salário de forma organizada.

#### [RF003] Buscar Funcionário

O sistema deve permitir que o usuário busque um funcionário pelo nome e visualize seu cargo e salário.

#### [RF004] Encerrar Sistema

O sistema deve encerrar a execução quando o usuário selecionar a opção **0 - Sair**.

### 4. REQUISITOS NÃO-FUNCIONAIS

#### [RNF001] Linguagem

O sistema deve ser implementado na linguagem **Python 3**.

#### [RNF002] Interface

A interação deve ocorrer via **interface de linha de comando (CLI)**.

#### [RNF003] Armazenamento

O sistema deve permitir o cadastro de até **100 funcionários**, respeitando o limite utilizado na versão original em C.

#### [RNF004] Validação de Entrada

O sistema deve validar as informações digitadas, impedindo valores inválidos para campos numéricos como o salário.

#### [RNF005] Desempenho

Todas as operações (cadastro, listagem e busca) devem ser realizadas instantaneamente para até 100 registros.

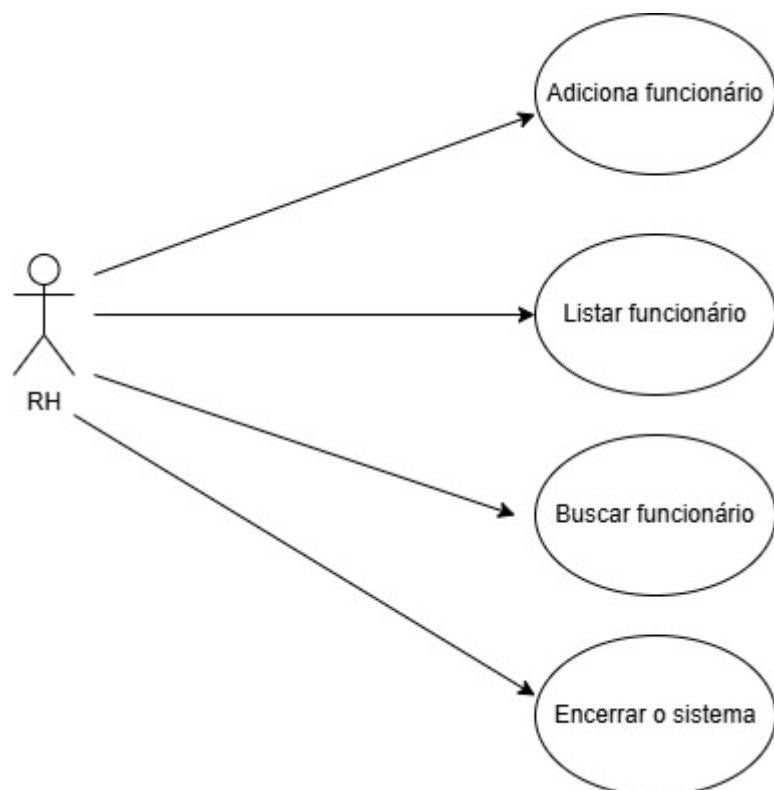
## 5. DIAGRAMA DE CASO DE USO

### DESCRIÇÃO:

Ator Principal: RH (Recursos Humanos) → É quem utiliza o sistema para cadastrar, consultar e gerenciar informações dos funcionários.

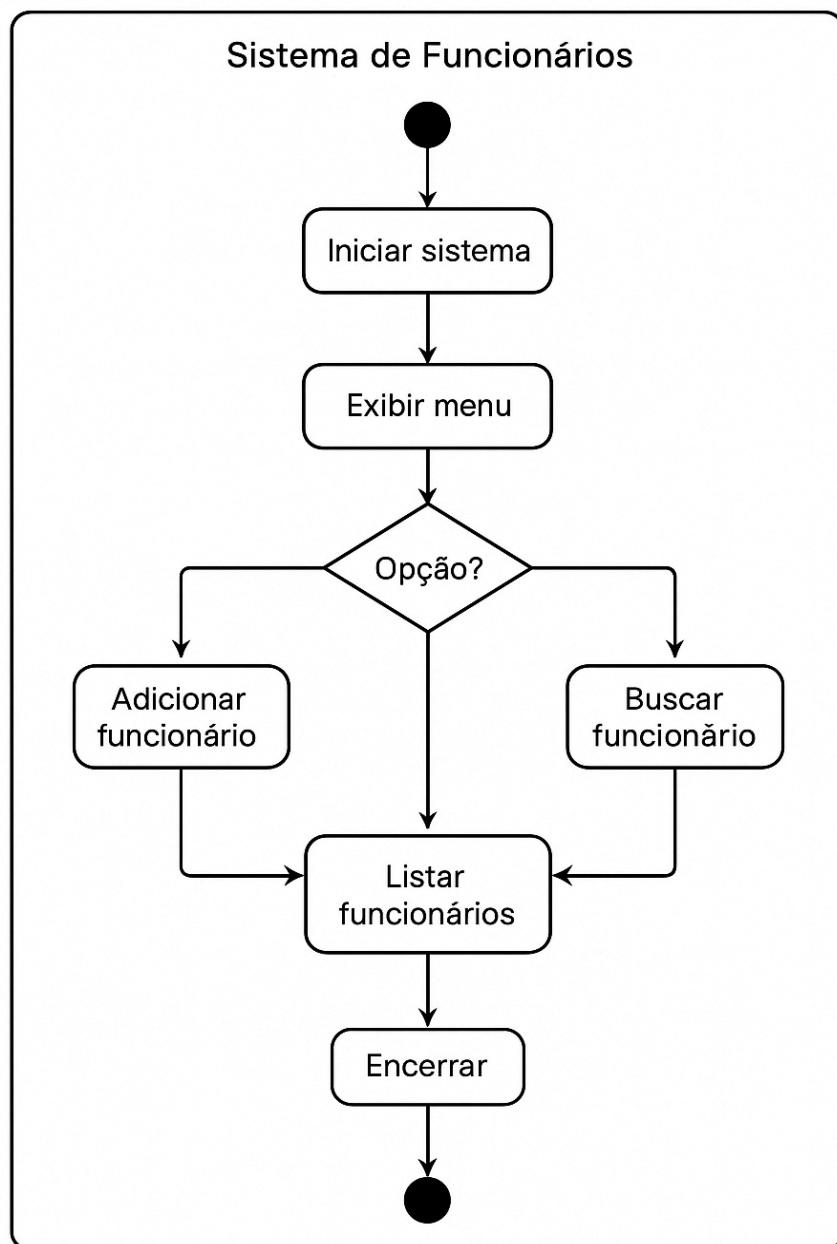
Caso de Uso Geral:

- Cadastrar Funcionário(ADICIONA)
- Listar Funcionários
- Buscar Funcionário
- Sair do Sistema(ENCERRA)



## 6. DIAGRAMA DE ATIVIDADEDE

Diagrama de Atividades



**REQUISITOS II**  
**SISTEMA DE ALUNOS**

**Autor(a):** Nayara Emelly Pereira Mendonça

**Disciplina:** Engenharia de Requisitos e Análise  
de Requisitos

**Profº:** Gioliano Bertoni

**Saquarema - RJ**

**2025**

## 7. INTRODUÇÃO

**Nome do sistema:** Sistema de Gerenciamento de Alunos

O sistema permite cadastrar alunos, listar todos os alunos cadastrados e buscar a nota de um aluno específico. Os dados incluem o nome e a nota final do aluno. Todas as informações são armazenadas em memória durante a execução do programa, que funciona por meio de uma interface de linha de comando (CLI).

## 8. TÉCNICAS UTILIZADAS NO LEVANTAMENTO

### 8.1 Entrevista com Stakeholders

Consiste em conversar diretamente com usuários, clientes ou gestores para identificar suas necessidades, expectativas e objetivos. Essa técnica fornece informações reais sobre o que o sistema precisa oferecer.

### 8.2 Brainstorming

Técnica colaborativa utilizada para gerar ideias e possíveis soluções. Os participantes discutem livremente funcionalidades e melhorias que podem ser implementadas no sistema.

### 8.3 Prototipagem

Criação de uma versão simples do sistema, permitindo validar requisitos, testar fluxos de operação e avaliar usabilidade antes da implementação final.

## 9. REQUISITOS FUNCIONAIS

### [RF001] Cadastrar Aluno

O sistema deve permitir que o usuário cadastre um aluno informando seu nome e sua nota.

### [RF002] Listar Alunos

O sistema deve exibir todos os alunos cadastrados, incluindo seus nomes e notas.

### [RF003] Buscar Nota de um Aluno

O sistema deve permitir buscar um aluno pelo nome e exibir sua nota.

### [RF004] Exibir menu principal

O sistema deve apresentar um menu de opções interativas para o usuário selecionar as funcionalidades.

### [RF005] Encerrar programa

O sistema deve finalizar a execução ao usuário selecionar a opção correspondente.

## 10. REQUISITOS NÃO-FUNCIONAIS

### [RNF001] Interface textual

O sistema deve ser operado por meio de linha de comando (CLI).

### [RNF002] Validação de dados

O sistema deve validar entradas do usuário, impedindo notas inválidas e nomes em branco.

### [RNF003] Armazenamento em memória

Os dados devem ser mantidos em estruturas de memória (listas, dicionários etc.) enquanto o programa estiver em execução.

### [RNF004] Usabilidade simples

A interface deve ser fácil de usar e intuitiva para qualquer usuário.

**[RNF005] Linguagem**

O sistema deve ser implementado em Python 3.

**[RNF006] Desempenho**

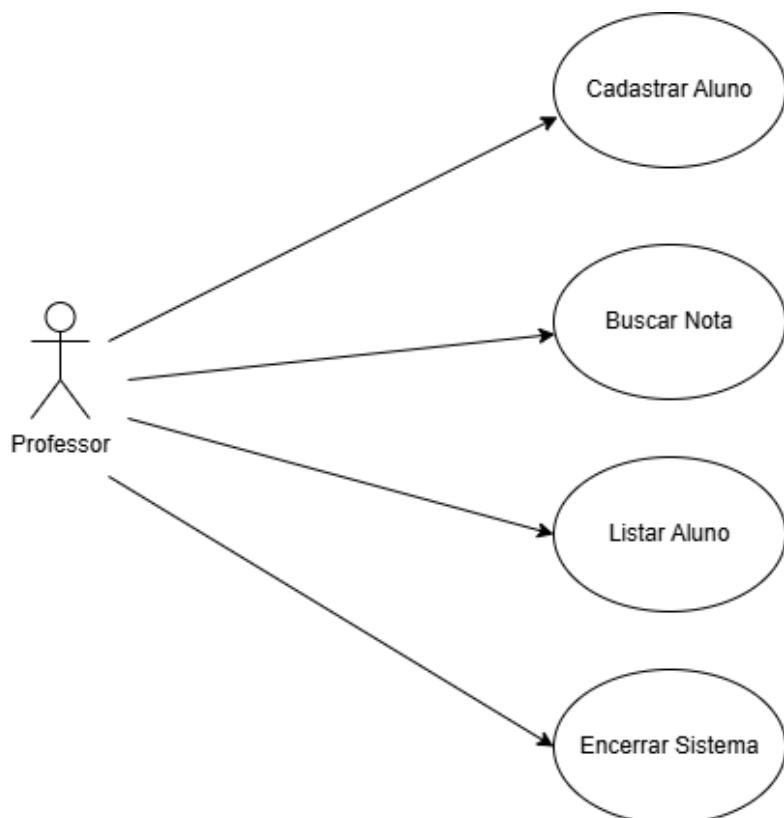
As operações devem ocorrer de forma instantânea, mesmo com grande quantidade de alunos cadastrados.

## 11. DIAGRAMA DE CASO DE USO

**Autor Principal: Usuário do sistema (ex.: professor, secretário escolar).**

**Casos de Uso:**

- Cadastrar aluno
- Buscar nota
- Listar alunos
- Encerrar sistema



## 12. DIAGRAMA DE ATIVIDADE

