

ANO
2023



UNINTER

**CADERNO DE RESPOSTAS DA
ATIVIDADE PRÁTICA DE:**

**ANÁLISE E MODELAGEM DE
SISTEMAS**

ALUNO: THIAGO OLIVEIRA DA ROCHA

**Caderno de Resposta Elaborado por:
Prof. MSc. Guilherme Ditzel Patriota**

Prática 01 – COLETA DE REQUISITOS, CRIAÇÃO DE DIAGRAMA DE CASO DE USO E CRIAÇÃO DE DIAGRAMA DE CLASSES.

Questão 01 – Criação de diagrama de caso de uso

ENUNCIADO: Veja o Roteiro da Atividade Prática para mais detalhes.

I. Apresentação dos requisitos funcionais e não funcionais (mínimo 3 de cada):

Requisitos funcionais:

1. **Controle de Portas:** O sistema deve ser capaz de receber comandos de voz para abrir e fechar portas de forma segura e confiável. Controle de
2. **Luzes:** O sistema deve permitir que o usuário acione e desligue as luzes através de comandos de voz.
3. **Controle de Equipamentos:** O sistema deve ser capaz de controlar outros dispositivos eletrônicos, como aparelhos de ar condicionado, televisões, entre outros, através de comandos de voz.

Requisitos Não Funcionais:

1. **Segurança:** O sistema deve implementar medidas de segurança para garantir que apenas comandos autorizados sejam executados, evitando acesso não autorizado às portas e equipamentos.
2. **Disponibilidade:** O sistema deve estar sempre disponível para receber comandos de voz, garantindo uma resposta rápida e eficiente.
3. **Integração com Dispositivos:** O sistema deve ser capaz de se integrar com os dispositivos existentes, como fechaduras eletrônicas, lâmpadas inteligentes e outros equipamentos, para realizar as ações solicitadas.

II. Apresentação do Diagrama de Caso de Uso (não esquecer do identificador pessoal):

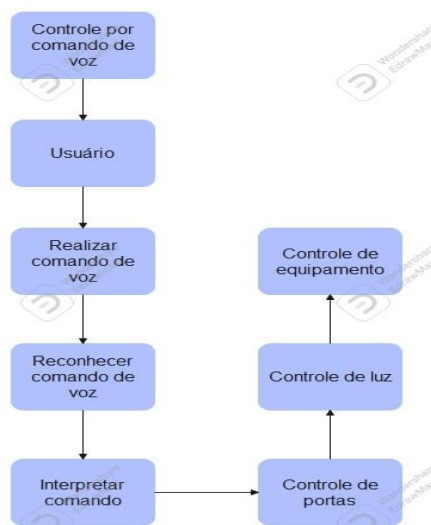


Figura 1: Neste exemplo atualizado, temos os casos de uso "Controle de Portas", "Controle de Luzes" e "Controle de Equipamentos" como funcionalidades específicas do sistema.

III. Responda à pergunta: Dos requisitos que você coletou, como é realizada a identificação de qual requisito é funcional e qual é requisito não funcional?

Resposta: Os requisitos funcionais referem-se às funcionalidades específicas que o sistema deve possuir, descrevendo as ações que ele deve ser capaz de realizar. Eles são geralmente verbos que descrevem o que o sistema deve fazer, como "reconhecer comandos de voz", "controlar portas" e "acionar luzes".

Por outro lado, os requisitos não funcionais dizem respeito às características gerais que o sistema deve ter, como qualidade, desempenho, segurança e integração. Eles definem as restrições e condições sob as quais o sistema deve operar, como "alta



precisão no reconhecimento de voz", "medidas de segurança implementadas" e "integração com dispositivos existentes".

Em resumo, os requisitos funcionais descrevem o que o sistema deve fazer, enquanto os requisitos não funcionais especificam as qualidades e características que o sistema deve possuir.

Prática 01 – COLETA DE REQUISITOS, CRIAÇÃO DE DIAGRAMA DE CASO DE USO E CRIAÇÃO DE DIAGRAMA DE CLASSES.

Questão 02 – Criação de diagrama de Classes.

ENUNCIADO: Veja o Roteiro da Atividade Prática para mais detalhes.

IV. Apresentação dos requisitos funcionais e não funcionais (mínimo 3 de cada diferentes da questão 1):

Requisitos funcionais:

1. **Controle de Portas:** O sistema deve ser capaz de receber comandos de voz para abrir e fechar portas de forma segura e confiável. Controle de
2. **Luzes:** O sistema deve permitir que o usuário acione e desligue as luzes através de comandos de voz.
3. **Controle de Equipamentos:** O sistema deve ser capaz de controlar outros dispositivos eletrônicos, como aparelhos de ar condicionado, televisões, entre outros, através de comandos de voz.

Requisitos Não Funcionais:

1. **Segurança:** O sistema deve implementar medidas de segurança para garantir que apenas comandos autorizados sejam executados, evitando acesso não autorizado às portas e equipamentos.
2. **Disponibilidade:** O sistema deve estar sempre disponível para receber comandos de voz, garantindo uma resposta rápida e eficiente.
3. **Integração com Dispositivos:** O sistema deve ser capaz de se integrar com os dispositivos existentes, como fechaduras eletrônicas, lâmpadas inteligentes e outros equipamentos, para realizar as ações solicitadas.

V. Apresentação do Diagrama de Classe (não esquecer do identificador pessoal):

Classe: VozController

Atributos:

vozReconhecimento:

ReconhecimentoDeVoz

Métodos:

abrirPorta(): void

fecharPorta(): void

acionarLuz(): void

desligarLuz(): void

controlarEquipamento(equipamento: Equipamento): void

Classe: ReconhecimentoDeVoz

Métodos:

reconhecerComando(): Comando

Classe: Comando

Atributos:

acao: String

Métodos:

getAcao(): String

Classe: Equipamento

Atributos:

nome: String

Métodos:

ligar(): void

desligar(): void

Figura 2: Essa é uma representação básica do diagrama de classes para o sistema de controle por comando de voz. A classe "VozController" é responsável por receber os comandos de voz, utilizar a classe "ReconhecimentoDeVoz" para reconhecer o comando, e então executar a ação apropriada com base no comando recebido. A classe "Equipamento" representa os diferentes dispositivos eletrônicos que podem ser controlados, e a classe "Comando" representa o comando de voz reconhecido, que possui uma ação associada.

I. Responda à pergunta: Como fazemos para converter um requisito ou um grupo de requisitos em uma classe para o diagrama de classes?

Resposta: (A conversão de requisitos em classes para um diagrama de classes ocorre por meio de uma análise das necessidades do sistema.

O diagrama de classes é amplamente utilizado por engenheiros de software e oferece um mapa conceitual do que é necessário para modelar um sistema. Ele representa as classes, os objetos e as relações entre eles.

Durante a análise dos requisitos do sistema, são identificadas as "entidades do mundo real", como clientes, empresas, entre outros, e as operações conceituais,



como cadastro, venda, entre outras, que são consideradas necessárias para o sistema. Com base nessa análise, essas entidades e operações são integradas ao diagrama de classes.

Em resumo, a conversão de requisitos em classes para o diagrama de classes envolve identificar as entidades e as operações relevantes, representá-las como classes e métodos correspondentes, estabelecer relacionamentos entre as classes e definir atributos, quando necessário.

Essa abordagem permite mapear os requisitos do sistema em uma representação visual, que facilita a compreensão das estruturas e interações necessárias para a implementação do software.