

# User story

## Descrição

Como um usuário ,desejo poder controlar a iluminação, climatização e segurança da minha casa de maneira remota, eficaz e simples

---

## Critério de aceitação

### Iluminação

- Configurar as luzes de cada cômodo individualmente
- Criar modos de iluminação específicos. Exemplo: modo soneca, modo cinema, modo viagem etc

### Climatização

- Abrir e fechar janelas em cada cômodo;
- Controlar a temperatura e intensidade de ar condicionado em cada cômodo;

### Segurança residencial

- Controlar trancamento de portas em cada cômodo individualmente;
- Acesso ao circuito de câmeras de segurança;
- Ativação de alarmes e das autoridades competentes em caso de invasão ou acidentes;

# Modelagem de sistema

## Nome da classe

### Comodo

#### Responsabilidades

- **AdionarDispositivo da classe (terá um para cada dispositivo) int, string -> void:** Cria e armazena eu novo dispositivo
- **ListarDispositivos int -> void:** Mostra todos os dispositivos de um tipo disponíveis no cômodo
- **RemoverDispositivo int, string -> void:** Apaga um dispositivo, de um tipo, do cômodo.
- **ConfigurarTodos int -> void:** Configura todos os dispositivos de um tipo no cômodo.
- **SetNome string -> void:** Muda o nome do comodo;
- **Nome void -> string:** Retorna o nome do comodo;
- **Comodo void -> void:** Cria um comodo vazio e sem nome;

#### Colaboradores

- **map<string,Lampada> lampadas;**
- **map<string, Cortina> cortinas;**
- **map<string, ArCondicionado> ares\_condicionados;**
- **map<string, Tranca> trancas;**
- **map<string, Janela> janelas;**
- **class Lâmpada;**
- **class Cortina;**
- **class ArCondicionado;**
- **class Tranca;**
- **class Janela;**
- **string nome;**

## Nome da classe

### Casa

#### Responsabilidades

- **ListarComodos void -> void:** Lista todos os cômodos da casa;
- **ConfigurarModo string -> void:** Enquanto essa função estiver ativa, o usuário pode configurar todos os dispositivos da casa normalmente e todas as configurações feitas serão salvas em um modo;
- **AtivarModo string -> void:** Aplica as configurações de um modo salvo;
- **ListarModos void -> void:** Lista os modos da casa;
- **AdicionarComodo string -> void:** adiciona um cômodo para que o usuário possa edita-lo;
- **RemoverComodo string -> void:** Apaga um cômodo e todos os dispositivos nele;
- **SetNome string -> void:** Muda o nome da casa;
- **Nome void -> string:** Retorna o nome da casa;
- **Casa void -> void:** Cria uma casa vazia e sem nome;

#### Colaboradores

- **class Comodo;**
- **map<string, Comodo> comodoss;**
- **map<string, Casa> modos;**
- **string nome;**

## Nome da classe

### Lampada

#### Responsabilidades

- **SetIntensidade int -> void:** configura a intensidade da lâmpada. Precondição o valor da intensidade deve ser de 0 a 100
- **Intensidade void -> int:** retorna intensidade da lâmpada;
- **SetCor int -> void:** configura a cor da lâmpada;
- **Cor void -> string:** Retorna a cor atual da lâmpada;
- **ListarCores void -> void:** Lista as cores que a lâmpada pode mostrar
- **SetNome string -> void:** Muda o nome do dispositivo.
- **Nome void -> string:** Retorna o nome do dispositivo.
- **Lampada void -> void:** Cria uma lâmpada sem nome com intensidade 0;

#### Colaboradores

- **int intensidade;**
- **string cor;**
- **vector <string> cores;**
- **string nome;**

## Nome da classe

### Cortina

#### Responsabilidades

- **SetIntensidade int -> void :** configura a quantidade que a cortina esta fechada ou aberta. Precondição: o valor da intensidade deve ser de 0 a 100
- **Intensidade void -> int:** retorna o status de iluminação da cortina
- **SetNome string -> void:** Muda o nome do dispositivo.
- **Nome void -> string:** Retorna o nome do dispositivo.
- **Cortina void -> void:** Cria uma cortina sem nome;

#### Colaboradores

- **int intensidade;**
- **string nome;**

## Nome da classe

### ArCondicionado

#### Responsabilidades

- **SetLigado bool -> void:** liga ou desliga o ar-condicionado;
- **SetModo string -> void:** Muda o modo do air condicionado.Precondição: o modo inserido tem que ser cool,heat,dry, ou fan
- **SetTemperatura int -> void:** configura temperatura do ar-condicionado.Precondição:a temperatura deve ter valor entre 16 e 30
- **Modo void -> string:** Retorna qual modo do ar;
- **Temperatura void -> int:** retorna temperatura do ar condicionado;
- **Ligado void -> bool:** retorna true se o ar condicionado está ligado;
- **SetNome string -> void:** Muda o nome do dispositivo.
- **Nome void -> string:** Retorna o nome do dispositivo.
- **ArCondicionado void -> void:** Cria um ar-condicionado sem nome

#### Colaboradores

- **int temperatura;**
- **bool ligado;**
- **int intensidade;**
- **string nome;**

## Nome da classe

### Janela

#### Responsabilidades

- **SetIntensidade int -> void:** configura o nível de abertura da janela Precondições: a tranca deve ser 0, e a intensidade deve ter ser entre 0 e 100.
- **Intensidade void -> int:** retorna o status do quanto a janela está aberta.
- **SetTranca bool -> void:** tranca a janela Precondição: a janela deve ter intensidade 0;
- **Tranca void -> bool:** Retorna se a janela está aberta ou fechada
- **SetNome string -> void:** Muda o nome do dispositivo.
- **Nome void -> string:** Retorna o nome do dispositivo.
- **Janela void -> void:** Cria uma janela sem nome;

#### Colaboradores

- **int intensidade ;**
- **bool tranca;**
- **string nome;**

## Nome da classe

### Tranca

#### Responsabilidades

- **SetAtiva bool -> void:** tranca ou destranca a porta;
- **Ativa void -> bool:** Retorna se a tranca esta aberta ou fechada.
- **SetNome string -> void:** Muda o nome do dispositivo.
- **Nome void -> string:** Retorna o nome do dispositivo.
- **Tranca void -> void:** Cria uma tranca sem nome;

#### Colaboradores

- **bool ativa;**
- **string nome;**