

Documentação da API - Controle de Casa Inteligente

Grupo:

- Luana
- Sammya
- Teresa

Casas (Houses)

Base URL: /api/houses

1. Obter todas as casas cadastradas

- **Method:** GET
- **URL:** /api/houses
- **Parâmetros:** Nenhum
- **Corpo da Requisição (Body):** Não se aplica
- **Resposta (Response):**

```
[  
  {  
    "id_house": 1,  
    "name": "Casa da Família Silva",  
    "address": "Rua das Flores, 123"  
  },  
  {  
    "id_house": 2,  
    "name": "Casa da Família Souza",  
    "address": "Av. Central, 456"  
  }  
]
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 500 Internal Server Error → Erro do servidor

2. Obter uma casa por ID

- **Method:** GET
- **URL:** /api/houses/:id
- **Parâmetros (URL):** id (integer) → ID da casa que deseja buscar
- **Corpo da Requisição (Body):** Não se aplica
- **Resposta (Response):**

```
{  
  "id_house": 1,  
  "name": "Casa da Família Silva",  
  "address": "Rua das Flores, 123"  
}
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 404 Not Found → Casa não encontrada
 - 500 Internal Server Error → Erro do servidor

3. Criar uma nova casa

- **Method:** POST
- **URL:** /api/houses
- **Parâmetros:** Nenhum
- **Corpo da Requisição (Body):**

```
{  
  "name": "Casa da Família Pereira",  
  "address": "Rua Primavera, 789"  
}
```

- **Resposta (Response):**

```
{  
  "id_house": 3,  
  "name": "Casa da Família Pereira",
```

"address": "Rua Primavera, 789"

}

- **Status Codes:**
 - 201 Created → Casa criada com sucesso
 - 400 Bad Request → Campos obrigatórios ausentes ou inválidos
 - 500 Internal Server Error → Erro do servidor

4. Atualizar uma casa por ID

- **Method:** PUT
- **URL:** /api/houses/:id
- **Parâmetros (URL):**
 - id (integer) → ID da casa que deseja atualizar
- **Corpo da Requisição (Body):**

{

"name": "Casa Atualizada",

"address": "Rua Nova, 999"

}

- **Resposta (Response):**

{

"id_house": 1,

"name": "Casa Atualizada",

"address": "Rua Nova, 999"

}

- **Status Codes:**
 - 200 OK → Atualização realizada com sucesso
 - 404 Not Found → Casa não encontrada
 - 500 Internal Server Error → Erro do servidor

5. Remover uma casa por ID

- **Method:** DELETE
- **URL:** /api/houses/:id
- **Parâmetros (URL):** id (integer) → ID da casa que deseja remover
- **Corpo da Requisição (Body):** Não se aplica
- **Resposta (Response):**

```
{
  "message": "Casa removida com sucesso"
}
```

- **Status Codes:**
 - 200 OK → Casa removida
 - 404 Not Found → Casa não encontrada
 - 500 Internal Server Error → Erro do servidor

Cômodos (Rooms)

Base URL: /api/rooms

1. Obter todos os cômodos cadastrados

- **Method:** GET
- **URL:** /api/rooms
- **Parâmetros:** Nenhum
- **Corpo da Requisição (Body):** Não se aplica
- **Resposta (Response):**

```
[
  {
    "id_room": 1,
    "name": "Sala",
    "id_house": 1
  },
  {
    "id_room": 2,
```

```
"name": "Cozinha",  
"id_house": 1  
}  
]
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 500 Internal Server Error → Erro do servidor

2. Obter um cômodo por ID

- **Method:** GET
- **URL:** /api/rooms/:id
- **Parâmetros (URL):** id (integer) → ID do cômodo
- **Corpo da Requisição (Body):** Não se aplica
- **Resposta (Response):**

```
{  
"id_room": 1,  
"name": "Sala",  
"id_house": 1  
}
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 404 Not Found → Cômodo não encontrado
 - 500 Internal Server Error → Erro do servidor

3. Criar um novo cômodo

- **Method:** POST
- **URL:** /api/rooms
- **Parâmetros:** Nenhum
- **Corpo da Requisição (Body):**

```
{  
  "name": "Quarto",  
  "id_house": 1  
}
```

- **Resposta (Response):**

```
{  
  "id_room": 3,  
  "name": "Quarto",  
  "id_house": 1  
}
```

- **Status Codes:**

- 201 Created → Cômodo criado com sucesso
- 400 Bad Request → Campos obrigatórios ausentes ou casa não encontrada
- 500 Internal Server Error → Erro do servidor

4. Atualizar um cômodo por ID

- **Method:** PUT
- **URL:** /api/rooms/:id
- **Parâmetros (URL):** id (integer) → ID do cômodo
- **Corpo da Requisição (Body):**

```
{  
  "name": "Sala de Estar",  
  "id_house": 1  
}
```

- **Resposta (Response):**

```
{  
  "id_room": 1,  
  "name": "Sala de Estar",  
  "id_house": 1  
}
```

```
}
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 404 Not Found → Cômodo não encontrado
 - 500 Internal Server Error → Erro do servidor

5. Remover um cômodo por ID

- **Method:** DELETE
- **URL:** /api/rooms/:id
- **Parâmetros (URL):** id (integer) → ID do cômodo
- **Corpo da Requisição (Body):** Não se aplica
- **Resposta (Response):**

```
{
```

```
"message": "Cômodo removido com sucesso"
```

```
}
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 404 Not Found → Cômodo não encontrado
 - 500 Internal Server Error → Erro do servidor

6. Listar todos os cômodos de uma casa

- **Method:** GET
- **URL:** /api/rooms/house/:id_house
- **Parâmetros (URL):** id_house (integer) → ID da casa
- **Corpo da Requisição (Body):** Não se aplica
- **Resposta (Response):**

```
[
```

```
{
```

```
"id_room": 1,
```

```
"name": "Sala",  
"id_house": 1  
},  
{  
  "id_room": 2,  
  "name": "Cozinha",  
  "id_house": 1  
}  
]
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 400 Bad Request → Casa não encontrada
 - 500 Internal Server Error → Erro do servidor

Dispositivos (Devices)

Base URL: /api/devices

1. Obter todos os dispositivos

- **Method:** GET
- **URL:** /api/devices
- **Parâmetros:** Nenhum
- **Body:** Não se aplica
- **Response:**

```
[  
{  
  "id_device": 1,  
  "name": "Lâmpada",  
  "type": "LED",  
  "state": false,  
  "id_room": 1  
}
```



```
}
```

```
]
```

- **Status Codes:**

- 200 OK → Sucesso
- 500 Internal Server Error → Erro do servidor

2. Criar dispositivo

- **Method:** POST
- **URL:** /api/devices
- **Body:**

```
{
```

```
"name": "Ar Condicionado",
```

```
"type": "Split",
```

```
"state": false,
```

```
"id_room": 2
```

```
}
```

- **Response:**

```
{
```

```
"id_device": 2,
```

```
"name": "Ar Condicionado",
```

```
"type": "Split",
```

```
"state": false,
```

```
"id_room": 2
```

```
}
```

- **Status Codes:**

- 201 Created → Sucesso
- 400 Bad Request → Campos obrigatórios ausentes ou cômodo não encontrado
- 500 Internal Server Error → Erro do servidor

3. Obter dispositivo por ID

- **Method:** GET
- **URL:** /api/devices/:id_device
- **Parâmetros (URL):** id_device → ID do dispositivo
- **Body:** Não se aplica
- **Response:**

```
{  
  "id_device": 1,  
  "name": "Lâmpada",  
  "type": "LED",  
  "state": false,  
  "id_room": 1  
}
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 404 Not Found → Dispositivo não encontrado
 - 500 Internal Server Error → Erro do servidor

4. Obter dispositivos por cômodo

- **Method:** GET
- **URL:** /api/devices/room/:id_room
- **Parâmetros (URL):**
 - id_room → ID do cômodo
- **Body:** Não se aplica
- **Response:**

```
[  
  {  
    "id_device": 1,  
    "name": "Lâmpada",
```

```
"type": "LED",  
"state": false,  
"id_room": 1  
}  
]
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 400 Bad Request → Cômodo não encontrado
 - 500 Internal Server Error → Erro do servidor

5. Atualizar dispositivo

- **Method:** PUT
- **URL:** /api/devices/:id_device
- **Parâmetros (URL):**
 - id_device → ID do dispositivo
- **Body:**

```
{  
"name": "Ventilador",  
"type": "De Teto",  
"state": true,  
"id_room": 1  
}
```

- **Response:**

```
{  
"id_device": 1,  
"name": "Ventilador",  
"type": "De Teto",  
"state": true,  
"id_room": 1  
}
```

}

- **Status Codes:**
 - 200 OK → Sucesso
 - 400 Bad Request → Dados inválidos ou cômodo não encontrado
 - 404 Not Found → Dispositivo não encontrado
 - 500 Internal Server Error → Erro do servidor

6. Remover dispositivo

- **Method:** DELETE
- **URL:** /api/devices/:id_device
- **Parâmetros (URL):**
 - id_device → ID do dispositivo
- **Body:** Não se aplica
- **Response:**

{

"message": "Dispositivo removido com sucesso"

}

- **Status Codes:**
 - 200 OK → Sucesso
 - 400 Bad Request → Erro na requisição
 - 404 Not Found → Dispositivo não encontrado
 - 500 Internal Server Error → Erro do servidor

7. Ligar/Desligar dispositivo

- **Method:** PUT
- **URL:** /api/devices/:id_device/state
- **Parâmetros (URL):**
 - id_device → ID do dispositivo
- **Body:**

```
{  
  "state": true  
}
```

- **Response:**

```
{  
  "id_device": 1,  
  "name": "Lâmpada",  
  "type": "LED",  
  "state": true,  
  "id_room": 1  
}
```

- **Status Codes:**

- 200 OK → Sucesso
- 400 Bad Request → Campo state inválido ou dispositivo não encontrado
- 500 Internal Server Error → Erro do servidor

Cenas (Scenes)

Base URL: /api/scenes

1. Obter todas as cenas

- **Method:** GET
- **URL:** /api/scenes
- **Parâmetros:** Nenhum
- **Body:** Não se aplica
- **Response:**

```
[  
  {  
    "id_scene": 1,  
    "name": "Cena Noite",  
    "is_active": true
```

```
}
```

```
]
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 500 Internal Server Error → Erro do servidor

2. Criar cena

- **Method:** POST
- **URL:** /api/scenes
- **Body:**

```
{  
  "name": "Cena Manhã",  
  "is_active": true  
}
```

- **Response:**

```
{  
  "id_scene": 2,  
  "name": "Cena Manhã",  
  "is_active": true  
}
```

- **Status Codes:**
 - 201 Created → Sucesso
 - 400 Bad Request → Dados inválidos
 - 500 Internal Server Error → Erro do servidor

3. Obter cena por ID

- **Method:** GET
- **URL:** /api/scenes/:id
- **Parâmetros (URL):** id → ID da cena

- **Body:** Não se aplica

- **Response:**

```
{  
  "id_scene": 1,  
  "name": "Cena Noite",  
  "is_active": true  
}
```

- **Status Codes:**

- 200 OK → Sucesso
- 404 Not Found → Cena não encontrada
- 500 Internal Server Error → Erro do servidor

4. Atualizar cena

- **Method:** PUT
- **URL:** /api/scenes/:id
- **Parâmetros (URL):** id → ID da cena
- **Body:**

```
{  
  "name": "Cena Noite Atualizada",  
  "is_active": false  
}
```

- **Response:**

```
{  
  "id_scene": 1,  
  "name": "Cena Noite Atualizada",  
  "is_active": false  
}
```

- **Status Codes:**

- 200 OK → Sucesso

- 400 Bad Request → Dados inválidos
- 404 Not Found → Cena não encontrada
- 500 Internal Server Error → Erro do servidor

5. Deletar cena

- **Method:** DELETE
- **URL:** /api/scenes/:id
- **Parâmetros (URL):** id → ID da cena
- **Body:** Não se aplica
- **Response:**

```
{
  "message": "Cena removida com sucesso"
}
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 404 Not Found → Cena não encontrada
 - 500 Internal Server Error → Erro do servidor

6. Ativar/Desativar cena

- **Method:** PATCH
- **URL:** /api/scenes/:id/toggle
- **Parâmetros (URL):** id → ID da cena
- **Body:**

```
{
  "is_active": true
}
```

- **Response:**

```
{
  "id_scene": 1,
```



```
"name": "Cena Noite",  
"is_active": true  
}
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 400 Bad Request → Dados inválidos
 - 500 Internal Server Error → Erro do servidor

7. Acionar cena

- **Method:** POST
- **URL:** /api/scenes/:id/execute
- **Parâmetros (URL):** id → ID da cena
- **Body:** Não se aplica
- **Response:**

```
{  
  "message": "Cena 'Cena Noite' executada com sucesso!"  
}
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 400 Bad Request → Cena desativada ou dados inválidos
 - 500 Internal Server Error → Erro do servidor

Ação da Cena (SceneDevice)

Base URL: /api/scene-devices

1. Obter todos os dispositivos de todas as cenas (ações)

- **Method:** GET
- **URL:** /api/scene-devices
- **Parâmetros:** Nenhum
- **Body:** Não se aplica

- **Response:**

```
[  
  {  
    "id_scene_device": 1,  
    "id_scene": 1,  
    "id_device": 2,  
    "order": 1,  
    "interval": 5,  
    "action": "turn_on"  
  }  
]
```

- **Status Codes:**

- 200 OK → Sucesso
- 500 Internal Server Error → Erro do servidor

2. Obter dispositivo da cena (ação) por ID

- **Method:** GET

- **URL:** /api/scene-devices/:id

- **Parâmetros (URL):** id → ID do dispositivo da cena

- **Response:**

```
{  
  "id_scene_device": 1,  
  "id_scene": 1,  
  "id_device": 2,  
  "order": 1,  
  "interval": 5,  
  "action": "turn_on"  
}
```

- **Status Codes:**

- 200 OK → Sucesso
- 404 Not Found → Ação da cena não encontrada
- 500 Internal Server Error → Erro do servidor

3. Adicionar dispositivo a uma cena (criar uma ação)

- **Method:** POST
- **URL:** /api/scene-devices
- **Body:**

```
{
  "id_scene": 1,
  "id_device": 2,
  "action": "turn_on",
  "order": 1,
  "interval": 5
}
```

- **Response:**

```
{
  "id_scene_device": 1,
  "id_scene": 1,
  "id_device": 2,
  "order": 1,
  "interval": 5,
  "action": "turn_on"
}
```

- **Status Codes:**
 - 201 Created → Sucesso
 - 400 Bad Request → Dados inválidos ou cena/dispositivo não encontrado
 - 500 Internal Server Error → Erro do servidor

4. Atualizar dispositivo da cena (ação)

- **Method:** PUT
- **URL:** /api/scene-devices/:id
- **Parâmetros (URL):** id → ID do dispositivo da cena
- **Body:**

```
{  
  "action": "turn_off",  
  "order": 2,  
  "interval": 3  
}
```

- **Response:**

```
{  
  "id_scene_device": 1,  
  "id_scene": 1,  
  "id_device": 2,  
  "order": 2,  
  "interval": 3,  
  "action": "turn_off"  
}
```

- **Status Codes:**
 - 200 OK → Sucesso
 - 400 Bad Request → Dados inválidos
 - 404 Not Found → Ação da cena não encontrada
 - 500 Internal Server Error → Erro do servidor

5. Remover dispositivo da cena (remover ação)

- **Method:** DELETE
- **URL:** /api/scene-devices/:id
- **Parâmetros (URL):** id → ID do dispositivo da cena

- **Response:**

```
{  
  "message": "Ação da cena removida com sucesso"  
}
```

- **Status Codes:**

- 200 OK → Sucesso
- 404 Not Found → Ação da cena não encontrada
- 500 Internal Server Error → Erro do servidor