



Universidade do Estado de Santa Catarina
Centro de Ciências Tecnológicas – CCT
Departamento de Ciência da Computação

Dúvidas: Lucas Dallabeneta, Victor Eduardo Requia

Data: 28/06/2024

Manual de Instalação do Sistema de Gerenciamento de Salas

Pré-requisitos:

Antes de iniciar a instalação, verifique se seu sistema atende aos seguintes requisitos mínimos:

- **Python:** É necessário ter o Python instalado. Você pode baixar a versão mais recente em python.org.
- **Node.js e npm:** O Node.js já inclui o npm para gerenciar pacotes no cliente. Certifique-se de ter o Node.js instalado, disponível em nodejs.org.

Instalação e Início do Servidor:

Para configurar o servidor, siga estas etapas:

- **Instalação de Pacotes Python:** Abra o terminal e instale os pacotes necessários usando o pip:

```
pip install flask flask_cors pysnmp.hlapi
```

- **Iniciar o Servidor:** Após a instalação dos pacotes, execute o servidor com o seguinte comando:

```
python servidor.py
```

Instalação e Início do Cliente:

Para configurar o cliente web, proceda da seguinte maneira:

- **Instalação de Pacotes npm:** Navegue até o diretório do projeto no terminal e instale as dependências do cliente:

```
cd caminho/do/seu/projeto
npm install
```

- **Compilação do Cliente:** Após a instalação dos pacotes, compile o cliente com o seguinte comando:
`npm run build`
- **Hospedagem do Cliente:** Após a compilação, o cliente pode ser hospedado em qualquer servidor web de sua escolha.

É possível usar a biblioteca 'serve' para hospedar o cliente localmente. Para usá-la, siga os passos:

- Instale o pacote serve globalmente:

```
npm install -g serve
```

- Navegue até o diretório do seu projeto:

```
cd /caminho/para/seu/projeto
```

- Execute o serve:

```
serve -s build -l 3000
```

- Agora você pode acessar o servidor via seu navegador. Ele está disponível em:
<http://localhost:3000>

Uso do Sistema:

Algumas rotas não foram implementadas no frontend, então para o uso correto do sistema, é necessário criá-las usando alguma ferramenta que permita fazer requisições POST, como o [Bruno](#).

As rotas devem ser acessadas via o IP da aplicação com o acréscimo do nome da rota, como por exemplo: `http://10.90.90.92:5000/api/user`

Rotas Disponíveis para o Uso:

Rota	Método	Descrição	Exemplo de Corpo no Request
api/users	POST	cria um usuário	{ "username": "admin",

			<code>"password": "admin"</code> <code>}</code>
api/login	POST	realize o login na aplicação	<code>{</code> <code> "username": "admin",</code> <code> "password": "admin"</code> <code>}</code>
api/salas	GET	busca as salas	
api/sala	POST	cria uma sala	<code>{</code> <code> "username": "admin",</code> <code> "password": "admin"</code> <code>}</code>
api/computadores	GET	busca os computadores conectados	
api/agendamentos	GET	busca os agendamentos	
api/cancelar_agendamento	DELETE	cancela um agendamento	<code>{</code> <code> "id": "1"</code> <code>}</code>
api/administradores	GET	busca os administradores	
api/administradores	POST	cria um administrador	<code>{</code> <code> "username": "admin",</code> <code> "password": "admin"</code> <code>}</code>
api/desbloquear	POST	desbloqueia uma porta do switch	<code>{</code> <code> "porta": "22"</code> <code>}</code>
api/assign_sala	POST	associa um mac address à uma sala	<code>{</code> <code> "sala_id": "14",</code> <code> "mac_address": "AD-A9-80-AA-ED-50"</code> <code>}</code>
api/bloquear	POST	bloqueia uma porta do switch	<code>{</code> <code> "porta": "22",</code> <code> "startDate": "2024-05-01 13:22:00",</code> <code> "endDate": "2024-05-01 16:22:00",</code> <code>}</code>