

Projeto: Interface para Controle de Salas e Laboratórios

Integrantes: Augusto Pereira Aguiar, João Gabriel Klug e Pedro Henrique Filla Decolin

Orientador: Fábio Albini

Co-orientador: Ricardo Arruda

Requisitos/instalações necessárias:

Node.js: <https://nodejs.org/pt-br/download/>

Linguagens de programação:

- HTML: foi utilizado para a estruturação das páginas do site;
- CSS: foi utilizado para a estilização das páginas do site;
- JavaScript: foi utilizado para implementar funcionalidades nas páginas do site;
- Nodejs: foi utilizado para a construção do backend e comunicação com o banco de dados;

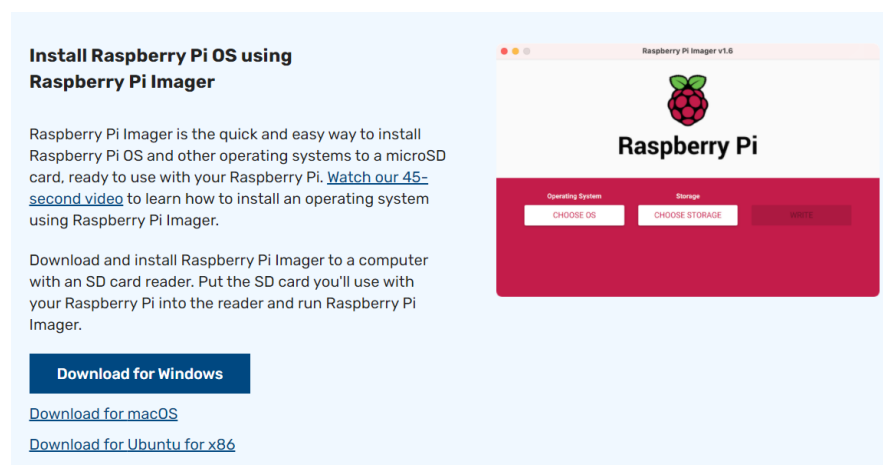
IDE:

A IDE utilizada para a programação do site, tanto frontend quanto backend, foi o Visual Studio Code, da Microsoft. (<https://code.visualstudio.com/download>.)

Orientações para instalação do servidor:

Orientações para instalação do sistema do servidor:

1. Em um computador, baixe a ferramenta do Raspberry Pi Imager (<https://www.raspberrypi.com/software/>)



2. Instale a ferramenta com as opções padrão

3. Insira um cartão microSD no computador
4. Execute o Raspberry Pi Imager. Em “Chose OS”, vá em “Raspberry Pi OS (Other)”. Selecione a opção “Raspberry OS Lite (64-bit)”. Em “Chose Storage”, selecione a unidade do cartão microSD. Depois, clique em “Write” e aguarde o processo;
5. Depois que terminar, retire o cartão microSD e o insira novamente no computador. Uma partição “boot” irá aparecer. Vá para ela e abra o arquivo “cmdline.txt”. Ao final do arquivo adicione a linha
“ip=**192.168.18.12::192.168.18.1:255.255.255.0:servidor:eth0:off**”
 - a. IP desejado do servidor
 - b. Gateway padrão para o servidor
 - c. Máscara de rede
 - d. Hostname
 - e. Interface de rede utilizada (por padrão, a placa de ethernet do raspberry pi é “eth0”)
 - f. Desliga opções para atribuição de IP automático (deixar como está, “off”)
6. Abra o powershell. Vá para a partição 'boot' digitando 'E:' ('E' é a unidade que o windows colocou para a partição). Crie um arquivo chamado 'ssh' com o comando: 'new-item ssh'. Não coloque nenhuma extensão de arquivo. Isso vai habilitar o SSH na próxima vez que o raspi bootar;
7. Os passos 5 e 6 definem um IP fixo para o servidor e habilitam o SSH para o acesso remoto, caso precise;
8. Insira o cartão microSD no raspberry e ligue-o. É importante que o cabo HDMI esteja conectado no monitor e que o cabo ethernet esteja conectado no dispositivo gateway (com o mesmo IP definido no passo 5). Aguarde até ligar e depois faça as configurações iniciais do sistema, referentes à localização e idioma. Caso uma janela pedindo atualização apareça, é recomendado que antes de tudo se atualize o sistema. Caso apareça uma janela para mudar a senha padrão, recomenda-se fortemente que a mude para uma senha forte;

Orientações para criação do banco de dados:

1. Abra o terminal do Raspberry Pi OS. Digite o comando “sudo apt update” e, depois, “sudo apt upgrade”. Isso atualizará a lista de repositórios do sistema. Depois, digite o comando “sudo apt install postgresql libpq-dev postgresql-client postgresql-client-common -y”. Isso instalará o Postgresql e suas dependências. Caso alguma outra dependência seja requisitada, instale-a com a fórmula “sudo apt install <nome do pacote>”. Após isso, entre no usuário do Postgresql com o comando “sudo su postgres”. Crie um usuário para o Postgres com o comando “createuser pi -P --interactive”. **Guarde o nome de usuário.** Feche o terminal e abra-o novamente. Digite “sudo su postgres” e depois “psql”, para entrar no utilitário do Postgresql. Após isso, copie o conteúdo do arquivo “ifprfechadura.sql”, na raiz do projeto, e cole-o no terminal. Este é o script de criação do banco e das tabelas.

Configurações antes de executar o sistema:

1. Nos arquivos “/Node/acesso/funcoes/liberarAcesso.js”, “/Node/acesso/funcoes/retornarEstadoFechadura.js” e nos 5 arquivos de “/Node/tables” haverá um trecho de código assim:

```
//CONFIGURA O BD
const Client = require('pg').Client;
const cliente = new Client({
  user: "pi",
  password: "SenhaPostgres5432",
  host: "localhost",
  port: 5432,
  database: "db_tcc"
})
```

Preste atenção: em ‘user’ coloque o nome do usuário criado anteriormente e em ‘password’ a sua respectiva senha. ‘host’, ‘port’ e ‘database’ podem permanecer como estão.

Executando o sistema:

1. No terminal, instale o Nodejs com o comando: ‘sudo apt install nodejs’

2. Baixe/copie a pasta do projeto (InterfaceLaboratorios) para a pasta 'Documents' do usuário 'pi' do Raspberry Pi OS. Entre nela e na aba 'Ferramentas' clique em 'Abrir a pasta atual no terminal'. Rode o comando 'node app.js'. O servidor está no ar. Para acessá-lo do proprio servidor, digite no navegador 'localhost:3000'. Para acessar de outro computador na rede digite '192.168.18.12:3000'. Note que deve ser colocado o IP referente ao servidor.