

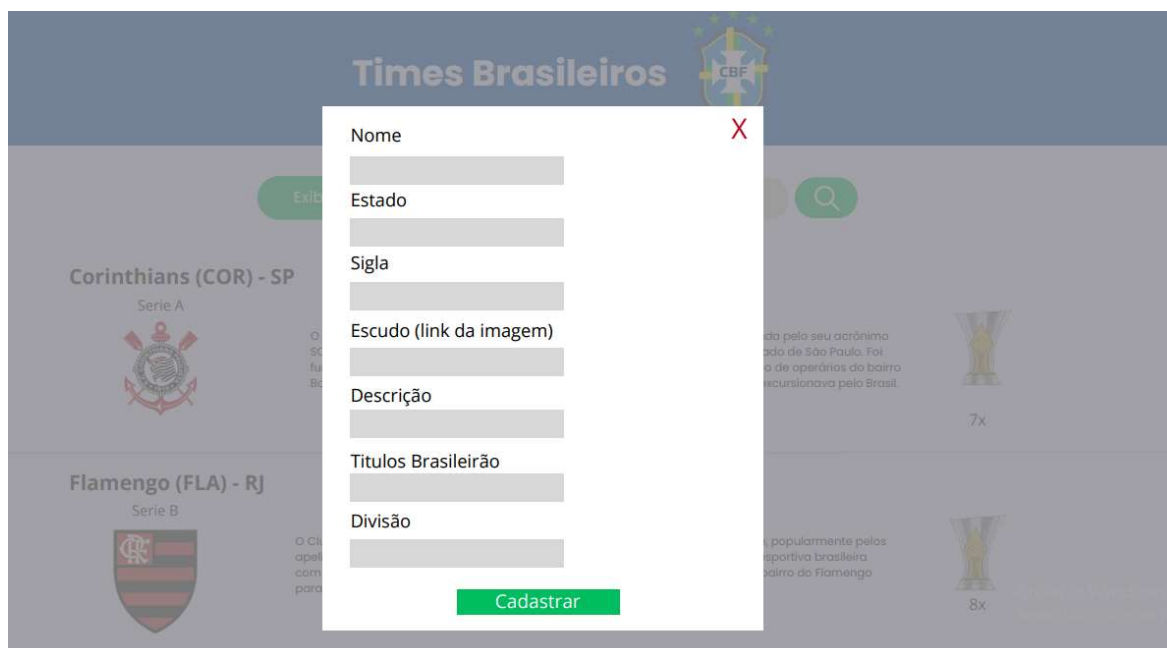
Victor Luis Carneiro Maia - 214072023 - Sprint 2

API - TIMES BRASILEIROS

Layout da Tela Inicial



Click botão cadastrar time



Código API - Programado em Node.Js:

Professor, estou estudando de forma autônoma o node.js, e acabei desenvolvendo minha api com o node.js JavaScript, pois tenho mais facilidade com a sintaxe diferente do php, agradeço a compreensão, obrigado!!

```
const express = require('express');
const bodyParser = require('body-parser');
const fs = require('fs').promises;
const path = require('path');
const cors = require('cors');

const app = express();
const port = 3000;

app.use(bodyParser.json());
app.use(cors({
  origin: '*'
}));

app.listen(port, () => {
  console.log(`Servidor rodando em http://localhost:${port}`);
});

const filePath = path.join(__dirname, 'times.json');

class Time {

  constructor(filePath){
    this.filePath = filePath;
  }

  createTime(nome, estado, sigla, escudo, descricao, xBrasileiro,
divisao){
    return {
      nome,
      estado,
      sigla,
      escudo,
      descricao,
      xBrasileiro,
      divisao
    }
  }
}
```

```

    async loadTimes() {
        try {
            const data = await fs.readFile(filePath, 'utf8');
            return JSON.parse(data);
        } catch (error) {
            return [];
        }
    }

    async saveTimes(times) {
        try {
            await fs.writeFile(filePath, JSON.stringify(times, null,
2), 'utf8');
        } catch (error) {
            console.error('Erro ao salvar os times:', error);
        }
    }

    async filterTimesByName(nome) {
        let times = await this.loadTimes();
        let filteredTimesName = times.filter(t => t.nome.toLowerCase()
=== nome.toLowerCase())

        return filteredTimesName
    }

    async filterTimesByState(estado){
        let times = await this.loadTimes();
        let filteredTimes = times.filter(t => t.estado.toLowerCase()
=== estado.toLowerCase())

        return filteredTimes
    }
}

const time = new Time(filePath);

// Adicionar time

```

```

app.post('/times', async (req, res) => {
  const { nome, estado, sigla, escudo, descricao, xBrasileiro, divisao} =
    req.body;

  if (!nome || !estado || !sigla || !escudo || !descricao || !xBrasileiro
) {
    return res.status(400).send("Por favor, forneça o nome e o estado
do time.");
  }

  let times = await time.loadTimes();

  if (times.some(t => t.sigla === sigla || t.nome === nome)){
    return res.status(400).send("Time já existe");
  }

  const novoTime = time.createTime(nome, estado, sigla, escudo,
descricao, xBrasileiro, divisao)

  times.push(novoTime);
  await time.saveTimes(times);

  res.status(201).send(novoTime);
});

// Remover time

app.delete('/times/:nome', async (req, res) => {
  const { nome } = req.params

  let times = await time.loadTimes()

  const index = times.findIndex(t => t.nome.toLowerCase() ===
nome.toLowerCase());

  if(index === -1){

```

```

        return res.status(404).send("Time não encontrado")
    }

    times.splice(index, 1);

    await time.saveTimes(times);

    res.status(200).send("Time deletado com sucesso")
  })

app.get('/times/todos', async (req, res) => {

  try {
    let times = await time.loadTimes();
    res.status(200).send(times);
  } catch (error) {
    res.status(500).send("Erro ao carregar os times.");
  }

})

app.get('/times/:nome', async (req, res, next) => {
  const { nome } = req.params;

  try {
    let times = await time.filterTimesByName(nome);

    if (times.length === 0) {
      throw new Error(`Time ${nome} não cadastrado na nossa base de
dados!`);
    }

    res.status(200).send(times);
  }

```

```

    } catch (error) {
      next(error);
    }
  })

app.get('/times/:estado', async (req, res, next) => {
  const { estado } = req.params;

  try {
    let times = await time.filterTimesByState(estado);

    if (times.length === 0) {
      throw new Error('Estado não possui times cadastrados');
    }

    res.status(200).send(times);
  } catch (error) {
    next(error);
  }
}

```

Código API Consumida pelo JavaScript

```

document.addEventListener('DOMContentLoaded', () => {
  const timesContainer =
document.getElementById('times-container');

  // Função para buscar times
  async function fetchTimes() {
    try {
      const response = await
fetch('http://localhost:3000/times/todos');
      if (!response.ok) {
        throw new Error('Erro ao buscar os times');
      }
    }
  }
}

```

```

        const times = await response.json();
        displayTimes(times);
    } catch (error) {
        console.error('Erro:', error);
    }
}

// Função para exibir os times
function displayTimes(times) {
    timesContainer.innerHTML = '';
    times.forEach(time => {
        const timeElement = document.createElement('div');
        timeElement.classList.add("timesBox")
        timeElement.innerHTML = `
            <div class="timeInfos">
                <h2>${time.nome} (${time.sigla}) -
${time.estado}</p></h2><br>
                Serie ${time.divisao}

                
            </div>

            <div class="history">

                ${time.descricao}
                <div class="titlesBox">
                    ${time.xBrasileiro}x</p>
                <div>

                <div>
                `;
        timesContainer.appendChild(timeElement);
    });
}

```