**APP DASHBOARD – AJAX**

**Ajax com jQuery – Conceitos básicos / iniciando o App Dashboard**

**Ajax** – Asynchrounous JS and XML.

**XMLHttpRequest** – objeto que pode ser instanciado de forma nativa através da linguagem JS.

Dshboard

Aplicação front-end;

Irá consumir dados do back-end através de requisições assíncronas;

PHP;

Banco de dados MySQL.

**Requisições para o back-end com load(), get() e post()**

Funcionam como uma interface para a manipulação do objeto XMLHttpRequest;

Objeto utilizado para realizar e controlar requisições HTTP, é como uma espécie de facilitador desse processo;

Através desses métodos podemos fazer essas requisições de forma enxuta, prática.

<li><a href="#" id="documentacao">

<li><a href="#" id="suporte">

Adicionando id aos links

$('#documentacao').on('click', ()=>{

        console.log('link documentacao clicado')

    })

    $('#suporte').on('click', ()=>{

        console.log('link suporte clicado')

    })

Testando links

**Método load()**

$("#documentacao").on("click", () => {

    $('#pagina').load('documentacao.html')

  });

  $("#suporte").on("click", () => {

    $('#pagina').load('suporte.html')

  });

Assim que elemento com id documentação for clicado, página recebe conteúdo do arquivo ‘documentação.html’;

Ocorre uma sobreposição do conteúdo;

Através de poucas linhas de código, foi possível fazer um Request.

    $.get("documentacao.html", (data) => {//podereia ser qualquer parâmetro (está sendo recuperado o conteúdo em si da resposta) - data, valor..

através da instancia do jQuery ($) será executado o método get (.get()) que espera a url .get('') e uma ação .get('', data => {}) - data pois estamos recuperando o conteúdo em si da resposta.

  $.post("suporte.html", (data) => {

      $("#pagina").html(data);

    });

Post é feito de forma análoga ao get