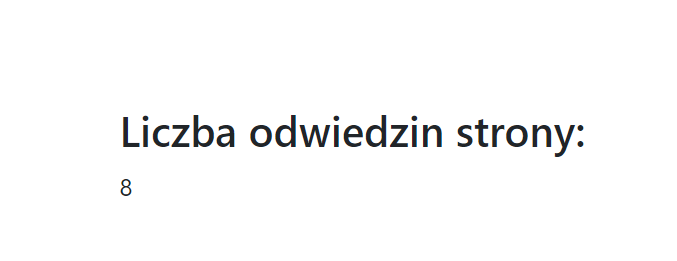
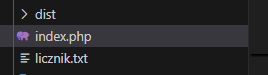
**Sprawozdanie**

|  |  |  |
| --- | --- | --- |
| **Projekt 2** | **Dynamiczna witryna www** | Wykonawca |
| **Technologia** | **Bootstrap5, jQuery 3.7.1, AJAX,(PHP)** | *Mulkityn, Arkadiusz, 173407* |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Cecha/użycie** | **Pkt** | **Kod (lub jego istotny fragment)** | | **Przeglądarka** | |
| Treść  (technologia web) | 1 |  | |  |
| Struktura witryny (katalogi pliki) | 1 |  | |  | |
| Witryna dynamiczna: szablon:  plik *index.html* | 1 |  | <footer class="bg-dark text-white text-center py-3 mt-5">  <p>&copy; 2025 TechForum</p>  </footer>  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>  <script src="./dist/js/script.js"></script>  </body>  </html> | |
| Witryna dynamiczna: podstrony: | 1 |  | |  | |
| Bootstrap5 | 0,5 | href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script> | |  | |
| Bootstrap5 – dostosowanie (CSS) | 0,5 | href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" | |  | |
| jQuery - animacja | 0,5 | function initJQueryEffects() {  $('.slide-animation').click(function() {  $(this).slideUp(500).slideDown(500);  });  $('.fade-animation').hover(  function() { $(this).fadeOut(300).fadeIn(300); },  function() { $(this).fadeIn(300); }  );  } | |  | |
| jQuery - efekt | 0,5 | function() { $(this).fadeOut(300).fadeIn(300); },  function() { $(this).fadeIn(300); } | |  | |
| AJAX – dynamiczne ładowanie plików html | 1 | </code></pre>  </div>  </div>  </div>  <div class="col-md-4">  <div class="card">  <div class="card-body">  <h5 class="card-title">XMLHttpRequest</h5>  <p class="card-text">Traditional AJAX method</p>  <pre class="bg-light p-3"><code>  const xhr = new XMLHttpRequest();  xhr.open("GET", url);  xhr.onload = function() {  if (xhr.status === 200) {  const data =  JSON.parse(xhr.response);  handleData(data);  }  };  xhr.send();</code></pre>  </div>  </div>  </div>  <div class="col-md-4">  <div class="card">  <div class="card-body">  <h5 class="card-title">jQuery AJAX</h5>  <p class="card-text">jQuery's AJAX implementation</p>  <pre class="bg-light p-3"><code>  $.ajax({  url: url,  method: "GET",  dataType: "json",  success: function(data) {  handleData(data);  },  error: function(xhr, status, error) {  handleError(error);  }  });</code> | |  | |
| AJAX – dynamiczne ładowanie plików txt | 1 | ocument.getElementById('loadTxtBtn').addEventListener('click', function() {      fetch('./data/dane.txt')          .then(response => response.text())          .then(content => {              document.getElementById('txtContent').innerHTML =                  `<div class="alert alert-info">${content}</div>`;          })          .catch(error => console.error('Error:', error)); | |  | |
| AJAX – dynamiczne ładowanie plików json | 1 | async function fetchFrameworks() {  try {  const response = await fetch("./data/frameworks.json");  if (!response.ok) {  throw new Error(`HTTP error! status: ${response.status}`);  }  const data = await response.json();  const frameworksList = document.getElementById("frameworksListFetch");  frameworksList.innerHTML = '<h5>Fetch API Results:</h5>';    data.forEach((item) => {  const listItem = document.createElement("li");  listItem.textContent = item.name;  listItem.className = 'list-group-item';  frameworksList.appendChild(listItem);  });  } catch (error) {  console.error("Error fetching data:", error);  }  }  function fetchFrameworksXHR() {  const xhr = new XMLHttpRequest();  xhr.open("GET", "./data/frameworks.json", true);  xhr.onload = function() {  if (xhr.status >= 200 && xhr.status < 300) {  try {  const data = JSON.parse(xhr.responseText);  const frameworksList = document.getElementById("frameworksListXHR");  frameworksList.innerHTML = '<h5>XMLHttpRequest Results:</h5>';    data.forEach((item) => {  const listItem = document.createElement("li");  listItem.textContent = item.name;  listItem.className = 'list-group-item';  frameworksList.appendChild(listItem);  });  } catch (error) {  console.error("JSON parsing error:", error);  }  }  };  xhr.onerror = function() {  console.error("Request error");  };  xhr.send();  }  function fetchFrameworksJQ() {  $.ajax({  url: "./data/frameworks.json",  method: "GET",  dataType: "json",  success: function(data) {  const frameworksList = $("#frameworksListJQ");  frameworksList.html('<h5>jQuery AJAX Results:</h5>');    data.forEach(function(item) {  const listItem = $("<li></li>")  .text(item.name)  .addClass('list-group-item');  frameworksList.append(listItem);  });  },  error: function(xhr, status, error) {  console.error("AJAX error:", error);  }  });  } | |  | |
| AJAX – użycie 3 metod   1. obiekt XMLHttpRequest 2. funkcja Fetch API 3. metoda jQuery Ajax | 1 | async function fetchFrameworks() {  try {  const response = await fetch("./data/frameworks.json");  if (!response.ok) {  throw new Error(`HTTP error! status: ${response.status}`);  }  const data = await response.json();  const frameworksList = document.getElementById("frameworksListFetch");  frameworksList.innerHTML = '<h5>Fetch API Results:</h5>';    data.forEach((item) => {  const listItem = document.createElement("li");  listItem.textContent = item.name;  listItem.className = 'list-group-item';  frameworksList.appendChild(listItem);  });  } catch (error) {  console.error("Error fetching data:", error);  }  }  function fetchFrameworksXHR() {  const xhr = new XMLHttpRequest();  xhr.open("GET", "./data/frameworks.json", true);  xhr.onload = function() {  if (xhr.status >= 200 && xhr.status < 300) {  try {  const data = JSON.parse(xhr.responseText);  const frameworksList = document.getElementById("frameworksListXHR");  frameworksList.innerHTML = '<h5>XMLHttpRequest Results:</h5>';    data.forEach((item) => {  const listItem = document.createElement("li");  listItem.textContent = item.name;  listItem.className = 'list-group-item';  frameworksList.appendChild(listItem);  });  } catch (error) {  console.error("JSON parsing error:", error);  }  }  };  xhr.onerror = function() {  console.error("Request error");  };  xhr.send();  }  function fetchFrameworksJQ() {  $.ajax({  url: "./data/frameworks.json",  method: "GET",  dataType: "json",  success: function(data) {  const frameworksList = $("#frameworksListJQ");  frameworksList.html('<h5>jQuery AJAX Results:</h5>');    data.forEach(function(item) {  const listItem = $("<li></li>")  .text(item.name)  .addClass('list-group-item');  frameworksList.append(listItem);  });  },  error: function(xhr, status, error) {  console.error("AJAX error:", error);  }  });  } | |  | |

*Dodatkowo*: PHP

<div class="container mt-5">

<h3>Liczba odwiedzin strony:</h3>

<p>

<?php

$filename = 'licznik.txt';

if (!file\_exists($filename)) {

file\_put\_contents($filename, 0);

}

$licznik = (int)file\_get\_contents($filename);

$licznik++;

file\_put\_contents($filename, $licznik);

echo $licznik;

?>

</p>

</div>