

# JEGYZŐKÖNYV

Web technológia alapjai  
Magyar Sajt Áruház weboldal

Készítette: **Szabó Marcell**

Neptunkód: **FU7OMC**

Dátum: **2024. Május 5.**

**Miskolc, 2024**

# Tartalomjegyzék

Bevezetés.....	3
1. Mappa struktúra.....	3
1.1 Kezdőlap.....	4
1.2 Űrlap Elemek .....	5
1.3 Az űrlap validációja.....	7
1.4 Új div elem készítése .....	8
1.5 Videó megjelenítése: .....	9
2. Árucikkek, AJAX, jQuery, getJSON: .....	10
2.1 Kosár Működése.....	12
2.2 Áruház lap .....	12
2.3 getJSON.....	14
3. Szorgalmi: node.js inicializálása, package.json, server.js, package-lock.json .....	14

## Bevezetés

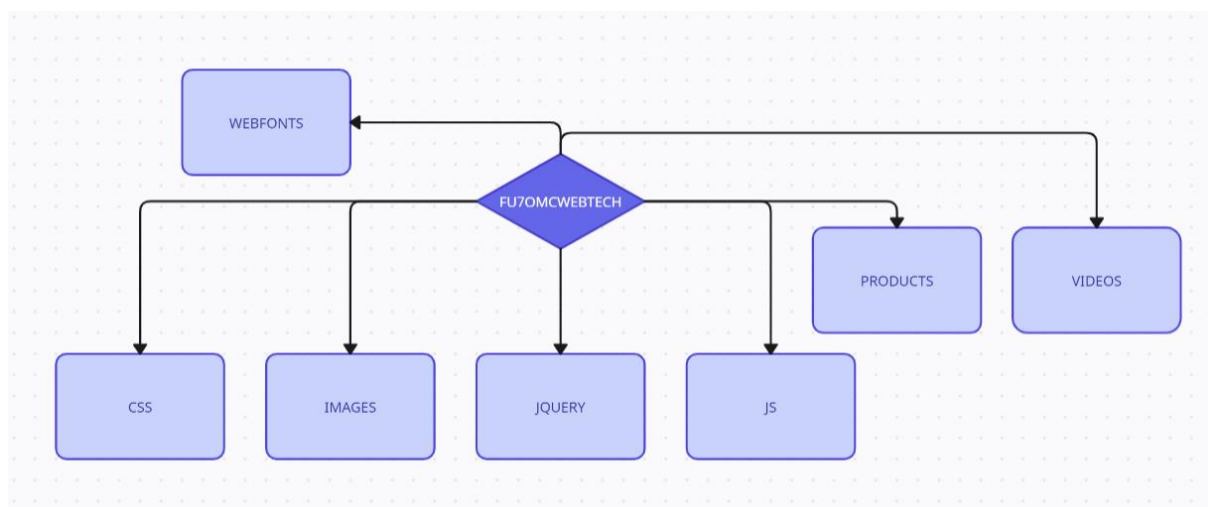
Amikor megkaptuk a feladatot, tudtam, hogy egy webshop szerű weboldalt szeretnék megvalósítani. A téma választásnál a medve sajtból fakadó szeretetem miatt és amiért más nem igazán jutott eszembe kitaláltam, hogy elkészítek egy “Magyar Sajt Áruház” név alatt futó weboldalt.

Mikor a tervezési szakaszba értem, felírtam magamnak, hogy mik azok a funkciók, részek, amiket viszont akarok majd látni és amik nélkül nem szeretném leadni a feladatomat. Ezek a funkciók a következők voltak:

1. Egy kezdőlap, ahol a legfontosabb információkat szeretném elhelyezni: navbar, animált bannerek, kiemelt áruk, egy rész ahol vásárló véleményeket tüntetek fel, egy kérdőív (form), és egy hely, ahol elhelyezem a videót.
2. Egy működő kosarat, amibe, ha belehelyezem az árukat, page váltásnál is a kosárban maradnak.
3. Egy bejelentkezés oldal, ami, habár nem funkcionál, formális jelentősége van mert a webshopok többségében van lehetőség regisztrációra/bejelentkezésre.
4. Egy kontakt és egy about oldal, amik ugyancsak nem rendelkeznek hatalmas funkciókkal, de minden weboldalnál szükséges szerintem.

A feladat megoldása során több nyílt forrásból elérhető stíluslapot és javascriptet használtam pl: bootstrap.css vagy minicart.js

## 1. Mappa struktúra



## 1.1 Kezdőlap

A weboldal célja a saját áruház, annak termékeinek, szolgáltatásainak bemutatása. A weboldal szerkezete tartalmazza a következő HTML elemeket: *section*, *nav*, *header*, *footer*.

<nav> és <header> - tagok használata a navigációs sávhoz.

```
<nav class="navbar navbar-expand-lg navbar-light bg-light top-header mb-2">
  <button class="navbar-toggler mx-auto" type="button" data-toggle="collapse"
    data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
    aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav nav-mega mx-auto">
      <li class="nav-item active">
        <a class="nav-link ml-lg-0" href="index.html">Kezdőlap</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="about.html">Rólunk</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="shop.html">Áruház</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="contact.html">Kapcsolat</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href=" ../.. /forras.html">Források</a>
      </li>
    </ul>
  </div>
</nav>
</header>
```

<section> - tag használata szövegek megjelenítésére az árucikkekhez:

```
<section class="banner-bottom-wthreelayouts py-lg-5 py-3">
  <div class="container-fluid">
    <div class="inner-sec-shop px-lg-4 px-3">
      <h3 class="tittle-w3layouts my-lg-4 my-4">Kiemelt ajánlataink </h3>
      <div class="row">
```

<footer>-tag használata a lábléc elkészítéséhez

```

<!--footer -->
<footer class="py-lg-5 py-3">
  <div class="container-fluid px-lg-5 px-3">
    <div class="row footer-top-w3layouts">
      <div class="col-lg-3 footer-grid-w3ls">
        <div class="footer-title">
          <h3>Rólunk</h3>
        </div>
        <div class="footer-text">
          <p>Kalandozz az ízek világában a mi sajtkollekcióinkkal, amely minden ízlésnek és vágyaknak megfelel! Fedezd fel a világ különböző tájairól származó sajtókészítő mesterek remekeit egyetlen helyen, ahol a hagyomány és a kreativitás találkozik.</p>
          <ul class="footer-social text-left mt-lg-4 mt-3">

```

## 1.2 Űrlap Elemek

Az űrlapot egy <div> elembe helyeztem, amelyet azonosítóval látok el id attribútummal. Erre az azonosítóra később hivatkozom majd a CSS és JavaScript kódban. Az űrlapon számos HTML elemet használtam a feladatnak megfelelően. A <form> elemek között megtalálhatók a feladatban megadott elemek, mint a szöveges beviteli mező egy vagy több soros, adatlista, jelölőnégyzet, rádiógomb, színválasztó, dátumválasztás, valamint gombok.

```

<h4>Töltsd ki az űrlapot!</h4>
<form action="">
  <label for="name">Név:</label>
  <input type="text" id="name" name="name"><br>
  <label for="name" class="error" id="error-name"></label><br>
  <label for="email">E-mail:</label>
  <input type="text" id="email" name="email"><br>
  <label for="email" class="error" id="error-email"></label><br>

  <label for="dateOB">Szül. idő:</label>
  <input type="date" id="dateOB" name="dateOB"
  min="1900-01-01" max="2025-01-01"><br><br>

  <input type="radio" id="gender" name="gender" value="Férfi">
  <label for="gender">Férfi</label>

  <input type="radio" id="female" name="gender" value="Nő">
  <label for="female">Nő</label><br><br>

```

```

<label for="Sajtok">Kedvenc sajtod:</label>
<datalist id="sajt">
<option value="Mizo Sajt">
<option value="Karaván Sajt">
<option value="Medve Sajt">
<option value="Pannonia Sajt">
</datalist>
<input list="sajt" id="favSajt" name="favSajt"><br><br>

<label for="color">Kedvenc színed:</label>
<input type="color" id="selectColor" name="selectColor"><br><br>

<label for="satisfaction">Mennyire voltál megelégedve webshopunkkal? (1-10)</label>
<input type="range" id="satisfaction" name="satisfaction" min="1" max="10" value="5">
<div id="satisfactionValue">5</div>

<label for="agree">
<input type="checkbox" id="agree" name="agree">
Hozzájárulok adataim marketing célú felhasználásához.
</label>
<br><br>
<input id="form-btn" type="reset" value="Reset">
<input id="form-btn" type="submit" value="Küldés" onclick="showAlert()">
</form>

```

Az űrlap alapvetően rejtve van az oldalon (display: none;). Egy gomb létrehozásával az oldalon, az id attribútumra hivatkozva JQuery segítségével jelenítem meg az űrlapot. A gombra kattintva az űrlap fokozatosan megjelenik (fade effect). Emellett a gombon is alkalmaztam egy JQuery animációt, amely a hover funkciót használja, így amikor az egér fölé kerül, megváltozik a gomb átláthatósága (opacity).

```

$(document).ready(function () {

    $('.content').mouseenter(function () {
        $(this).find('.content-text').show();
    }).mouseleave(function () {
        $(this).find('.content-text').hide();
    });

    $('#formBtn').hover(function () {
        $(this).stop().animate({ opacity: 0.7 }, 'fast');
    }, function () {

```

```
$(this).stop().animate({ opacity: 1 }, 'fast');
});
```

### 1.3 Az űrlap validációja

Az űrlap beküldésének figyelőjét az űrlap azonosítójával, azaz #popup-formmal hivatkozva JQuery segítségével lehet meghívni. Amikor az űrlap beküldésre kerül, a kód először meghívja az e.preventDefault() függvényt, hogy megakadályozza az alapértelmezett űrlap beküldési működést, azaz az oldal újratöltését. Ezután a kód lekéri az input mezők értékeit, majd elvégzi a kötelező mezők, azaz a név és az e-mail mező validációját. Ha a név vagy az e-mail mező üres, akkor a kód piros keretet ad a mezőkhöz, és megjeleníti a megfelelő hibaüzenetet. Ha az e-mail formátuma nem megfelelő, akkor szintén piros keretet ad a mezőhöz, és megjeleníti a hibaüzenetet.

```
$('#popup-form').submit(function (e) {
    e.preventDefault();
    $('#error').empty();
    let name = $('#name').val();
    let email = $('#email').val();

    if (name === '') {
        $('#name').css('border', '1px solid red');
        $('#error-name').text('Írd be a neved!');
    } else {
        $('#name').css('border', '1px solid #ccc');
    }

    if (email === '') {
        $('#email').css('border', '1px solid red');
        $('#error-email').text('Írd be az Email címed!');
    } else if (!isValidEmail(email)) {
        $('#email').css('border', '1px solid red');
        $('#error').text('Nem érvényes Email cím!');
    } else {
        $('#email').css('border', '1px solid #ccc');
    }

});
});
```

## 1.4 Új div elem készítése

Az előző bemutatott űrlap, amelyet a felhasználó kitölt, lehetővé teszi, hogy az adatokat ellenőrizze. Ha az adatok megfelelőek, az oldal azonnal megjeleníti azokat, amikor a felhasználó az elküldés gombra kattint. Az eseménykezelő függvény, amely az űrlapot és az adatok megjelenítésére szolgáló tartományt beállítja, automatikusan lefut az oldal betöltésekor ('DOMContentLoaded' esemény). Ezt követően az eseménykezelő függvény hozzáad egy figyelőt az űrlap elküldéséhez (form.addEventListener('submit', ...)).

```
document.addEventListener('DOMContentLoaded', function () {
  let form = document.getElementById('popup-form');
  let displayData = document.getElementById("displayData");

  form.addEventListener('submit', function (event) {
    event.preventDefault();

    let name = document.getElementById('name').value;
    let email = document.getElementById('email').value;
    let bornDate = document.getElementById('dateOB').value;
    let gender = document.querySelector('input[name="gender"]:checked').value;

    const dataList = document.querySelector('#sajt');
    const options = dataList.querySelectorAll('option');
    let favouriteSajtok = [];
    options.forEach(option => {
      favouriteSajtok.push(option.value);
    });

    const messageInput = document.querySelector('#message');
    const message = messageInput.value;

    let datasDiv = document.createElement('div');
    datasDiv.innerHTML = '<h3>Válaszaid</h3>' +
      '<p><strong>Név:</strong> ' + name + '</p>' +
      '<p><strong>E-mail:</strong> ' + email + '</p>' +
      '<p><strong>Szül. idő:</strong> ' + bornDate + '</p>' +
      '<p><strong>Nem:</strong> ' + gender + '</p>' +
      '<p><strong>Kedvenc sajt:</strong> ' + favouriteSajtok[0] + '</p>'
    +
      '<p><strong>Üzenet:</strong> ' + message + '</p>';
    displayData.appendChild(datasDiv);

    form.style.display = 'none';
  });
});
```



## 1.5 Videó megjelenítése:

Videó implementáció:

```
<!-- video -->
<div class="testimonials2 py-lg-4 py-3 mt-4">
  <div class="testimonials-inner py-lg-4 py-3">
    <div id="carouselExampleControls" class="carousel slide" data-
ride="carousel">
      <div class="carousel-inner" role="listbox">
        <video id="videoPlayer" controls loop autoplay>
<source src="videos/mizoreklam.mp4" type="video/mp4">
Your browser does not support the video tag.
</video>
<br>
<div class="videoButtons">
<button class="videoButtons2" onclick="togglePlayPause()">Indítás /
Leállítás</button>
<button class="videoButtons2" onclick="decreaseVolume()">Halk</button>
<button class="videoButtons2" onclick="increaseVolume()">Hangos</button>
<button class="videoButtons2" onclick="mute()">Néma</button>
      </div>
    </div>
  </div>
</div>
</div>
```

Videóhoz tartozó gombok:

```
const video = document.getElementById('videoPlayer');

function togglePlayPause() {
  if (video.paused) {
    video.play();
  } else {
    video.pause();
  }
}

function mute() {
  video.muted = !video.muted;
}

function decreaseVolume() {
  if (video.volume > 0) {
    video.volume -= 0.5;
  }
}
```

```
function increaseVolume() {
    if (video.volume < 1) {
        video.volume += 0.5;
    }
}

const videoPlayer = document.getElementById('myVideo');
video.loop = true;
```

## 2. Árucikkek, AJAX, jQuery, getJSON:

4 oldal tartozik ide, ezeknek a struktúrája szinte ugyanaz csak adatokban térnek el.

Minden cikkhez tartozik egy “Előnézet” oldal amire kattintva több információt tudhatunk meg az árukról. Az itt megjelenített információkat JSON fájlban tároltam és a jQuery AJAX módszerével jelenítettem meg egy táblázatban:

Mizo.json:

```
{
  "termek_neve": "Mizo Trappista sajt",
  "marka": "Mizo",
  "sajttipus": "Trappista",
  "tomeg": 1000,
  "aras": 3000,
  "osszetevek": [
    "tej",
    "só",
    "kalcium-klorid",
    "tejzára"
  ],
  "szallitas": {
    "szallito": "SajtExpressz",
    "szallitasi_ido": "1-2 munkanap"
  }
}
```

Mizo.js:

```
$(document).ready(function(){
    $.ajax({
        url: 'mizo.json',
        dataType: 'json',
        success: function(data) {
            $('#termek_neve').text(data.termek_neve);
        }
    });
});
```

```

    $('#marka').text(data.marka);
    $('#sajttipus').text(data.sajttipus);
    $('#tomeg').text(data.tomeg);
    $('#aras').text(data.aras);

    var osszetevek = data.osszetevek;
    $.each(osszetevek, function(index, value) {
        $('#osszetevek').append('<li>' + value + '</li>');
    });

    $('#szallito').text(data.szallitas.szallito);
    $('#szallitasi_ido').text(data.szallitas.szallitasi_ido);
    }
    });
});

```

Mizo.html:

```

<table id="sajt_table" border="1">
    <tr>
    <td>Termék neve:</td>
    <td><span id="termek_neve"></span></td>
    </tr>
    <tr>
    <td>Márka:</td>
    <td><span id="marka"></span></td>
    </tr>
    <tr>
    <td>Sajttípus:</td>
    <td><span id="sajttipus"></span></td>
    </tr>
    <tr>
    <td>Tömeg:</td>
    <td><span id="tomeg"></span> g</td>
    </tr>
    <tr>
    <td>Ár:</td>
    <td><span id="aras"></span> Ft</td>
    </tr>
    <tr>
    <td>Összetevők:</td>
    <td><span id="osszetevek"></span></td>
    </tr>
    <tr>
    <td>Szállítás:</td>
    <td>

```

```

<p class="table2" style="font-size: 16px">Szállító: <span class="table"
id="szallito"></span></p>
<p class="table2" style="font-size: 16px">Szállítási idő: <span class="table"
id="szallitasi_ido"></span></p>
</td>
</tr>
</table>
<script src="mizo.js"></script>

```

Az összes többi cikknél hasonlóan jártam el.

## 2.1 Kosár Működése

Egy nyílt forrásból letöltött “minicart.js” segítségével csináltam meg.

A navbar-ban található a gomb, ami a kosár tartalmát megjeleníti.

Az árucikkek a következőképpen vannak meghatározva:

```

<form action="#" method="post">
<input type="hidden" name="cmd" value="_cart">
<input type="hidden" name="add" value="1">
<input type="hidden" name="googles_item" value="Mizo Trapista Sajt">
<input type="hidden" name="amount" value="3000">
<button type="submit" class="googles-cart pgoogles-cart">
    Kosárhoz adás
</button>
</form>

```

Amikor azt szeretnénk, hogy valami a kosárba kerüljön azt a script ezek alapján az adatok alapján azonosítja. Például, ha csak egy betű vagy egy szám eltérne azt már egy másik árucikknek azonosítaná. Ez a Script végzi a végösszeg kiszámítását, a valuta kiválasztását, illetve ez kezeli a gombot, ami később a checkout oldalra visz.

## 2.2 Áruház lap

Ezen az oldalon igazából az árucikkek vannak megjelenítve a kezdőlaphoz hasonlóan csak átméretezve. Ami talán kiemelendő vagy kicsit új azaz oldalsáv ami ezen az oldalon található ami tartalmazza a kiemelt ajánlatokat, egy keresősávot (nem működik) és egy ár beállító slidert ami a húzogatáson kívül ugyancsak nem működik:

```

<div class="search-hotel">
<h3 class="agileits-sear-head">Keresés</h3>
<form action="#" method="post">

```

```

<input      class="form-control"      type="search"      name="search"
placeholder="Keresés..."
      required="">
  <button class="btn1">
    <i class="fas fa-search"></i>
  </button>
  <div class="clearfix"> </div>
</form>
</div>
<!-- price range -->
<div class="range">
<h3 class="agileits-sear-head">Ár</h3>
<ul class="dropdown-menu6">
  <li>
    <div id="slider-range"></div>
    <input type="text" id="amount"
      style="border: 0; color: #ffffff; font-weight: normal; width:
300px;" />
  </li>
</ul>
</div>
<!-- //price range -->

```

```

<div class="deal-leftmk left-side">
<h3 class="agileits-sear-head">Kiemelt Ajánlatok</h3>
<div class="special-sec1">
  <div class="img-deals">
    
  </div>
  <div class="img-deal1">
    <h3>Mizo Trapista Sajt</h3>
    <a href="../products/mizo.html">3000 Ft</a>
  </div>
  <div class="clearfix"></div>
</div>
<div class="special-sec1">
  <div class="col-xs-4 img-deals">
    
  </div>
  <div class="col-xs-8 img-deal1">
    <h3>KARAVÁN Füstölt Sajt</h3>
    <a href="../products/karavan.html">4000 Ft</a>
  </div>
  <div class="clearfix"></div>
</div>

```

## 2.3 getJSON

Elvégeztem az adat meghívásokat a getJson metódussal is amelynek az alábbi a script kódja.

```
$(document).ready(function(){
    $.getJSON('camem.json', function(data) {
        $('#termek_neve').text(data.termek_neve);
        $('#marka').text(data.marka);
        $('#sajttipus').text(data.sajttipus);
        $('#tomeg').text(data.tomeg);
        $('#aras').text(data.aras);

        var osszetevek = data.osszetevek;
        $.each(osszetevek, function(index, value) {
            $('#osszetevek').append('<li>' + value + '</li>');
        });

        $('#szallito').text(data.szallitas.szallito);
        $('#szallitasi_ido').text(data.szallitas.szallitasi_ido);
    })
    .fail(function() {
        console.error('Hiba történt a JSON betöltése közben.');
```

## 3. Szorgalmi: node.js inicializálása, package.json, server.js, package-lock.json

A terminálban kiadott 'npm init' parancs segítségével egy új Node.js projektet inicializáltam. Ezt követően a Node Package Manager több kérdést tett fel a konfigurációval kapcsolatban, például a projekt neve, verziószáma, leírása, fő fájl neve és egyéb metainformációk. Ennek eredményeként létrejött a package.json fájl, amely tartalmazza a konfigurációt és a függőségeket.

```

git repository: https://github.com/marcellszabo921/FU7OMCWebTech.git
keywords: sajt
author: Szabo Marcell
license: (ISC)
About to write to C:\Users\marce\Desktop\WEBTECH\FU7OMCWebTech\WebTechFU7OMC\package.json:

{
  "name": "webtechfu7omc",
  "version": "1.0.0",
  "description": "magyar sajt aruhaz weboldala",
  "main": "server.js",
  "scripts": {
    "test": "ng test",
    "start": "node server.js"
  },
  "repository": {
    "type": "git",
    "url": "git+https://github.com/marcellszabo921/FU7OMCWebTech.git"
  },
  "keywords": [
    "sajt"
  ],
  "author": "Szabo Marcell",
  "license": "ISC",
  "bugs": {
    "url": "https://github.com/marcellszabo921/FU7OMCWebTech/issues"
  },
  "homepage": "https://github.com/marcellszabo921/FU7OMCWebTech#readme"
}

Is this OK? (yes) █

```

```

{
  "name": "webtechfu7omc",
  "version": "1.0.0",
  "description": "magyar sajt aruhaz weboldala",
  "main": "server.js",
  "scripts": {
    "test": "ng test",
    "start": "node server.js"
  },
  "repository": {
    "type": "git",
    "url": "git+https://github.com/marcellszabo921/FU7OMCWebTech.git"
  },
  "keywords": [
    "sajt"
  ],
  "author": "Szabo Marcell",
  "license": "ISC",
  "bugs": {
    "url": "https://github.com/marcellszabo921/FU7OMCWebTech/issues"
  },
  "homepage": "https://github.com/marcellszabo921/FU7OMCWebTech#readme",
  "dependencies": {
    "figlet": "^1.7.0"
  }
}

```

A npm install parancs futtatásakor létrejön a package-lock.json fájl. Ez a fájl részletesen dokumentálja, hogy melyik verziója települt le a függőségeknek, valamint ezek függőségei és azok verziói.

```
{
  "name": "webtechfu7omc",
  "version": "1.0.0",
  "lockfileVersion": 3,
  "requires": true,
  "packages": {
    "": {
      "name": "webtechfu7omc",
      "version": "1.0.0",
      "license": "ISC",
      "dependencies": {
        "figlet": "^1.7.0"
      }
    },
    "node_modules/figlet": {
      "version": "1.7.0",
      "resolved": "https://registry.npmjs.org/figlet/-/figlet-1.7.0.tgz",
      "integrity": "sha512-gO8l3wvqo0V7wEFLXPbkX83b7MVjRrk1oRLfYlZXol8nEpb/ON9pcKLI4qpBv5YtOTfrINTqb7b40iYY2FTWFg==",
      "bin": {
        "figlet": "bin/index.js"
      },
      "engines": {
        "node": ">= 0.4.0"
      }
    }
  }
}
```

Készítettem egy server.js fájlt, ami egy egyszerű Node.js szerveralkalmazást definiál. A szerver működtetéséhez és a statikus fájlok szolgáltatásához beépített Node.js modulokat használ, mint például az http, fs és path. Ezek a modulok részei a Node.js alaptelepítésének, így külön telepíteni őket az npm-en keresztül nem szükséges.

```
const http = require('http');
const fs = require('fs');
const path = require('path');
const figlet = require('figlet');

figlet('Magyar Sajt Aruhaz', function(err, data) {
  if (err) {
```



```

    console.log('Something went wrong...');
    console.dir(err);
    return;
}
console.log(data);

const server = http.createServer((req, res) => {

    let filePath = '.' + req.url;
    if (filePath === './') {
        filePath = './index.html';
    }

    const extname = path.extname(filePath);
    let contentType = 'text/html';

    switch (extname) {
        case '.js':
            contentType = 'text/javascript';
            break;
        case '.css':
            contentType = 'text/css';
            break;
        case '.json':
            contentType = 'application/json';
            break;
        case '.png':
            contentType = 'image/png';
            break;
        case '.jpg':
            contentType = 'image/jpeg';
            break;
    }

    fs.readFile(filePath, (err, content) => {
        if (err) {
            if (err.code === 'ENOENT') {
                res.writeHead(404);
                res.end('File not found');
            } else {
                res.writeHead(500);
                res.end('Server error');
            }
        } else {
            res.writeHead(200, { 'Content-Type': contentType });
            res.end(content, 'utf-8');
        }
    })
})

```

```
    });  
  });  
  
  const PORT = process.env.PORT || 3000;  
  server.listen(PORT, () => console.log(`Server running on port ${PORT}`));  
});
```

A node\_modules mappa létrehozásához be kell húznom egy függőséget. Egy szimpla modult választottam, aminek a neve figlet. Az npm install figlet paranccsal lehúztam ezt a modult a node.js projektembe.

Ha futtatom a node server.js paranccsal akkor kifogja rajzolni a konzolra a figlet függvény által visszaadott értéket.

```
found 0 vulnerabilities  
PS C:\Users\marce\Desktop\WEBTECH\FU70MCWebTech\WebTechFU70MC> npm install figlet
```

```
added 1 package, and audited 2 packages in 2s
```

```
found 0 vulnerabilities  
PS C:\Users\marce\Desktop\WEBTECH\FU70MCWebTech\WebTechFU70MC> node server.js
```

```
  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  
 /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_  
/  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  /  
_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/_/__
```

```
Server running on port 3000
```

```
█
```

És ezek után már elérhető a weboldal a localhost:3000 címen.