

# **WEBTECHNOLÓGIÁK BEADANDÓ DOKUMENTÁLÁSA**

Szabó Martin

Programtervező informatikus BSC

Neptun-kód:WJFAOO

Miskolc

2023.02.14.

## Az alapkoncepcióm

Egy olyan oldalt szerettem volna elkészíteni a beadandómnak, amely eléggé modern hatású, teljesen reszponzív, valamint mobilbarát is egyaránt. A főoldalam, az index.html ezeket az általam kreált elvárásokat teljesíti, ezért is ez a beadandóm központi eleme.

## Az index.html oldalam bemutatása



Az első dologom az volt, hogy a `<head>` szekcióban hozzáadtam az alap dolgokat, mint például a dokumentum kódolását (`UTF-8`), a címet (`<title>`), hozzálinkeltem a CSS fájlt, ami majd hozzá fog tartozni és az `jQuery`-t is.

Ezután a `<body>` részben a navigációs sávot kezdtem elkészíteni `<div>`-ek bőszt használatával, hiszen a `class` attribútum használatával könnyen elkülöníthetők az elemek, valamint a `style.css`-ben is könnyedén hivatkozom a `class` ID-jára.

```
<!-- Navigation bar section -->
<nav class="navbar">
  <div class="navbar__container">
    <a href="#home" id="navbar__logo">Page of Colors</a>
    <div class="navbar__toggle" id="mobile-menu">
      <span class="bar"></span>
      <span class="bar"></span>
      <span class="bar"></span>
    </div>
    <ul class="navbar__menu">
      <li class="navbar__item">
        <a href="#home" class="navbar__links" id="home-page">Kezdőoldal</a>
      </li>
      <li class="navbar__item">
        <a href="#about" class="navbar__links" id="about-page">Rólam</a>
      </li>
      <li class="navbar__item">
        <a href="#studies" class="navbar__links" id="studies-page">Tanulmányok</a>
      </li>
      <li class="navbar__btn">
        <a href="#sign-up" class="button" id="signup">Elérhetőség</a>
      </li>
    </ul>
  </div>
</nav>
```

A *href* attribútumnál adtam meg, hogy ha rákattintunk az oldalam melyik részére ugorjon. Mint látni is lehet, az *index.html* 5 fő részből tevődik össze: navigációs sáv rész, a home szekció, rólam rész, tanulmányok rész, valamint az elérhetőség részből.

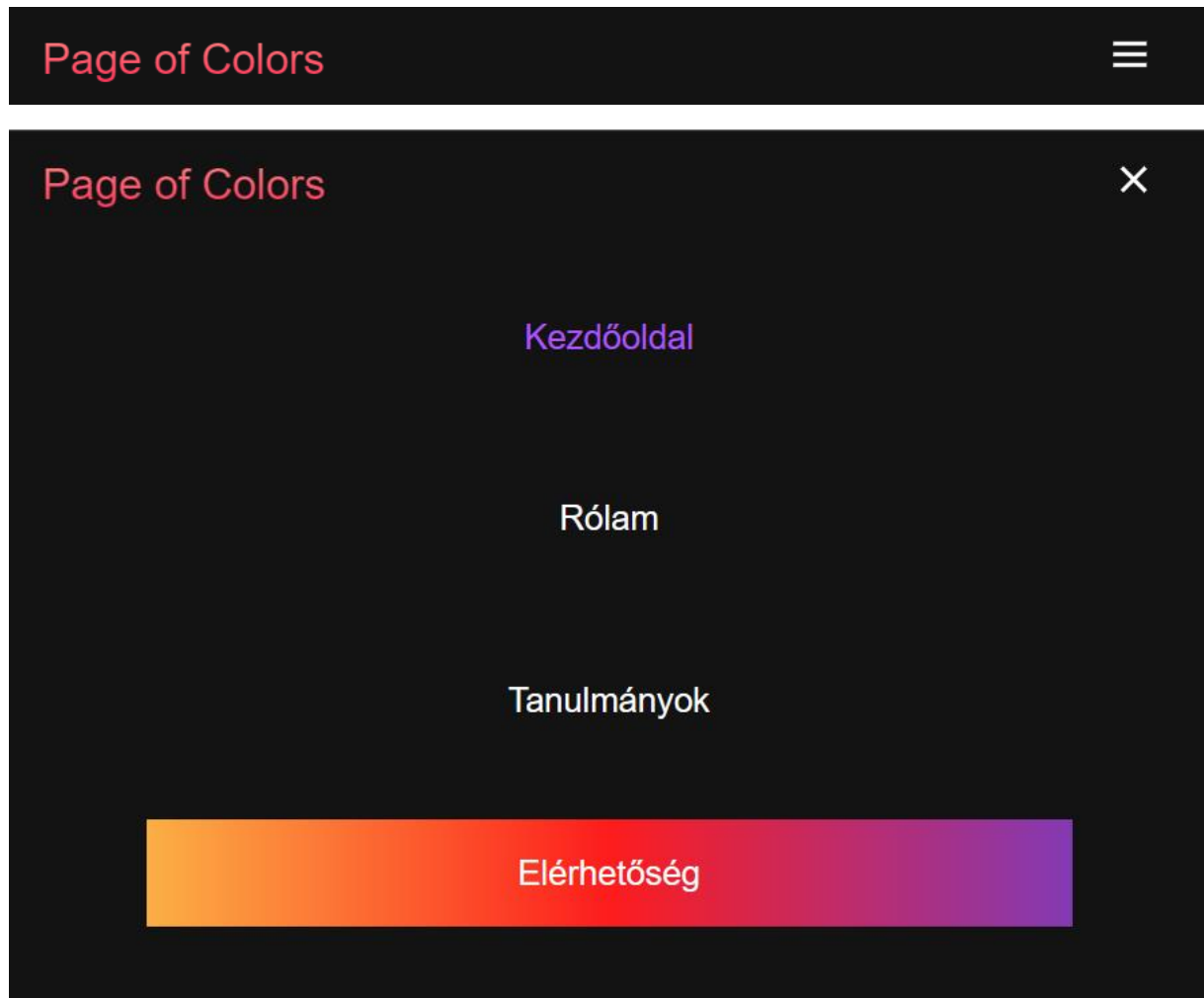
A *Navigation Bar Section*-höz tartozó CSS egy része az alábbi képen található meg. Azért csak egy része, mert ehhez a szekcióhoz tartozó CSS kód legalább 100 sor.

```
.navbar__container{
  display: flex;
  justify-content: space-between;
  height: 80px;
  z-index: 1;
  width: 100%;
  max-width: 1300px;
  margin: 0 auto;
  padding: 0 50px;
}

#navbar__logo{
  background-color: #ff8177;
  background-image: linear-gradient(to top,#ff0844 0%,#ffb199 100%);
  background-size: 100%;
  -webkit-background-clip: text;
  -moz-background-clip: padding;
  -webkit-text-fill-color: transparent;
  -moz-text-decoration-color: transparent;
  display: flex;
  align-items: center;
  cursor: pointer;
  text-decoration: none;
  font-size: 2rem;
}
```

Igyekeztem a stílus kialakítása során játszani egy kicsit a színekkel, mint az oldalam is mutatja, a színátmenetet a *linear-gradient()*-el értem el, itt tudtam megadni hogy merre legyen színátmenetes, milyen színeket mennyire alkalmazzon stb. A navigációs sáv kinézetét (hogy nem egymás alatt vannak az itemek, hanem egymás mellett) főleg a *display: flex;* módszernek köszönhetem. Ezekon kívül azt is beállítottam, hogy középre igazítva helyezkedjenek el az *itemek* (*align-items: center;*), mekkora legyen a méretük.

Ha a böngészőnket kisebbre állítjuk (jóval kisebbre), akkor a jobb felső sarokban megjelenik egy megnyitható menü, melyben a navigációs sávban található szavak lelik otthonukat. Az alábbi képeken ez meg is tekinthető.



Ahhoz, hogy telefonbaráttá tegyem az oldalam, az alábbi CSS kódot használtam. Itt ebben a részben ugyan úgy ki kellett alakítanom a gombok, itemek, containerek stílusát, mint eddig.

```
@media screen and (max-width: 960px)
```

A menü előhívásának van egy kisebb animációja is, amelyet JavaScript-el, valamint CSS kóddal oldottam meg. A CSS és a JavaScript kódokat az alábbi képek mutatják be.

```
#mobile-menu {
  position: absolute;
  top: 20%;
  right: 5%;
  transform: translate(5%,20%);
}

.navbar__toggle .bar {
  display: block;
  cursor: pointer;
}

#mobile-menu.is-active .bar:nth-child(2){
  opacity: 0;
}

#mobile-menu.is-active .bar:nth-child(1){
  transform: translateY(8px) rotate(45deg);
}

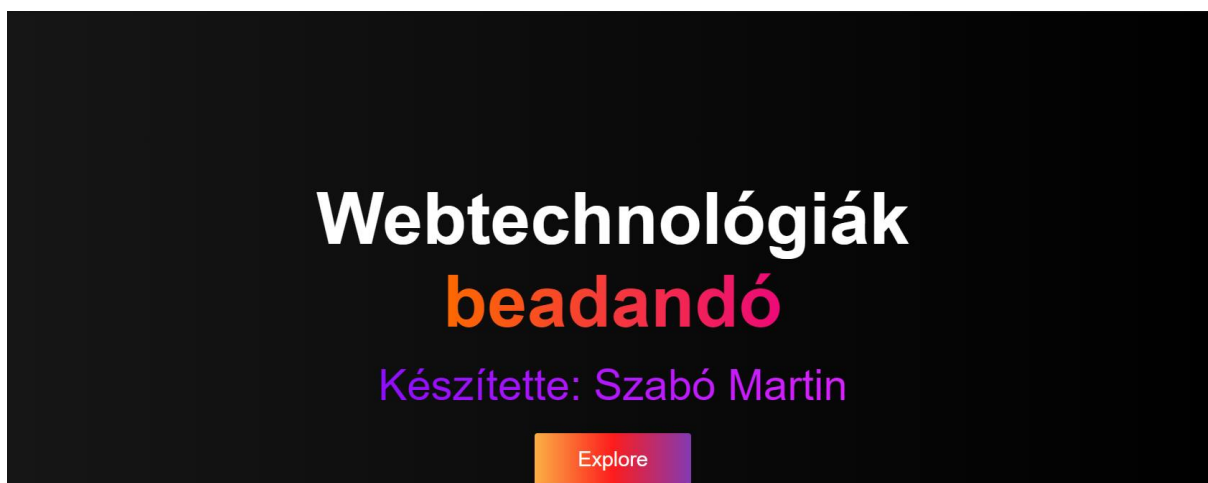
#mobile-menu.is-active .bar:nth-child(3){
  transform: translateY(-8px) rotate(-45deg);
}
```

```
const menu = document.querySelector(selectors: '#mobile-menu')
const menuLinks = document.querySelector(selectors: '.navbar__menu')

// Display Mobile Menu
const mobileMenu= () =>{
  menu.classList.toggle( tokens: 'is-active')
  menuLinks.classList.toggle( tokens: 'active')
}

menu.addEventListener( type: 'click',mobileMenu)
```

A következő szekció, amelyet be szeretnék mutatni, az a „home szekció”( a kódomban hero section). Ez az a rész, ami az oldalra látogató embert először fogadja.

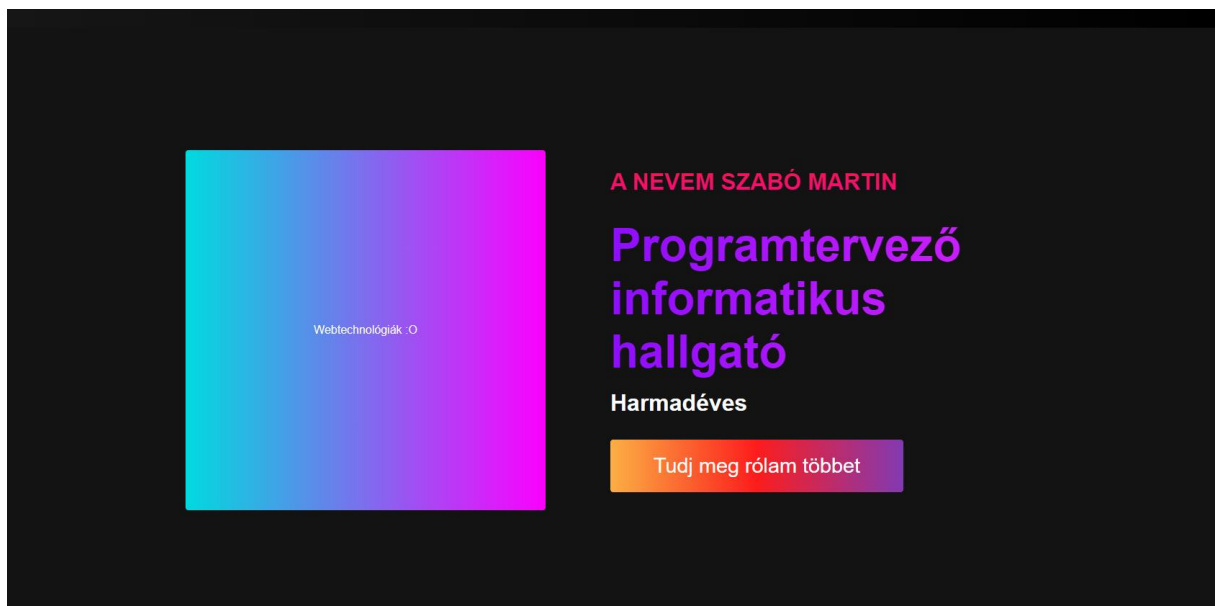


Ezt a rész szintén <div>-ekkel és class-okkal oldottam meg, amelyet az alábbi kód reprezentál.

```
<!-- Hero Section-->
<div class="hero" id="home">
  <div class="hero__container">
    <h1 class="hero__heading">Webtechnológiák<span> beadandó</span></h1>
    <p class="hero__description">Készítette: Szabó Martin</p>
    <button class="main__btn"><a href="#">Explore</a></button>
  </div>
</div>
```

Ha az *Explore* gombra ráviszem a kurzort, akkor a színe megváltozik balról jobbra haladó átmenetben. Ezt a CSS-ben oldottam meg a *linear-gradient()* valamint a *-webkit-linear-gradient()* metódusokkal. Természetesen a @media screen -t itt is használtam a mobilbarát megjelenítés érdekében.

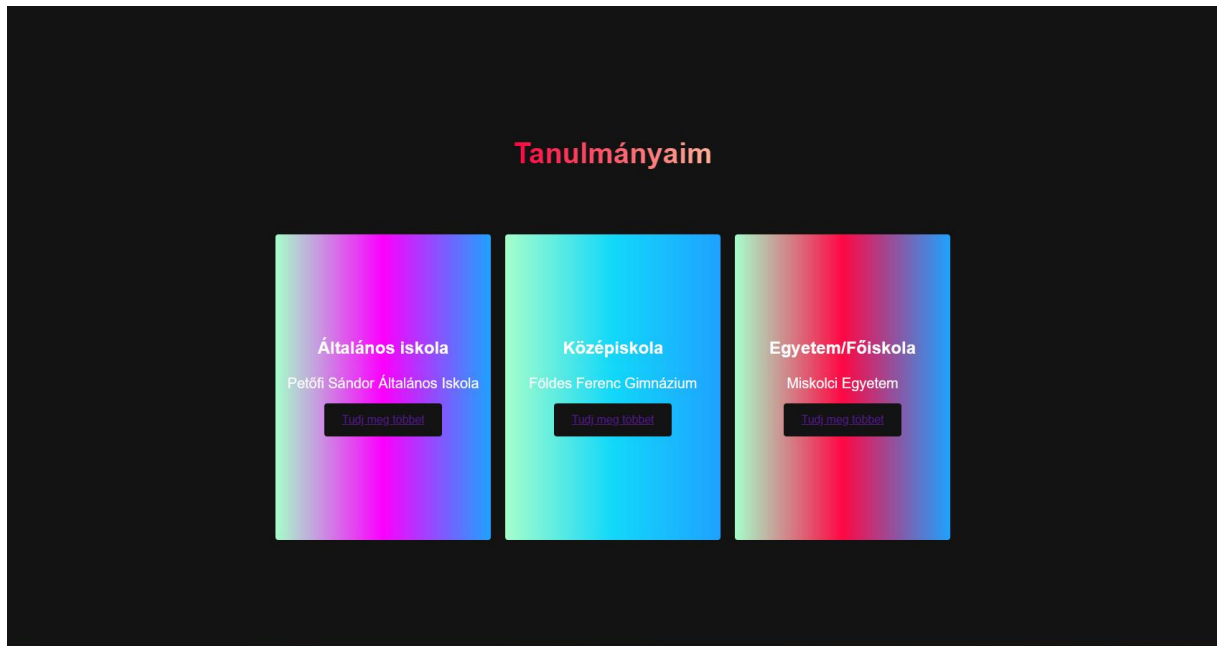
A következő szekció a „rólam” szekció. Itt is szinte ugyan úgy jártam el, mint a hero sectionnél.



Ugyan úgy <div>-eket használtam eme szekció létrehozásához. A gomb, amelyen a „Tudj meg rólam többet” szöveg szerepel az aboutme.html oldalra mutat, melyet a href=„aboutme.html” attribútummal oldottam meg.

```
<!-- About Section-->
<div class="main" id="about">
  <div class="main__container">
    <div class="main__img--container">
      <div class="main__img--card">Webtechnológiák :O</div>
    </div>
    <div class="main__content">
      <h1>A nevem Szabó Martin</h1>
      <h2>Programtervező informatikus hallgató</h2>
      <p>Harmadéves</p>
      <button class="main__btn"><a href="aboutme.html">Tudj meg rólam többet</a></button>
    </div>
  </div>
</div>
```

A következő szekció a tanulmányaim szekció, melyben szintén szerep `<div>`, `<a>`, `<button>`, `<h2>`, `<p>`.



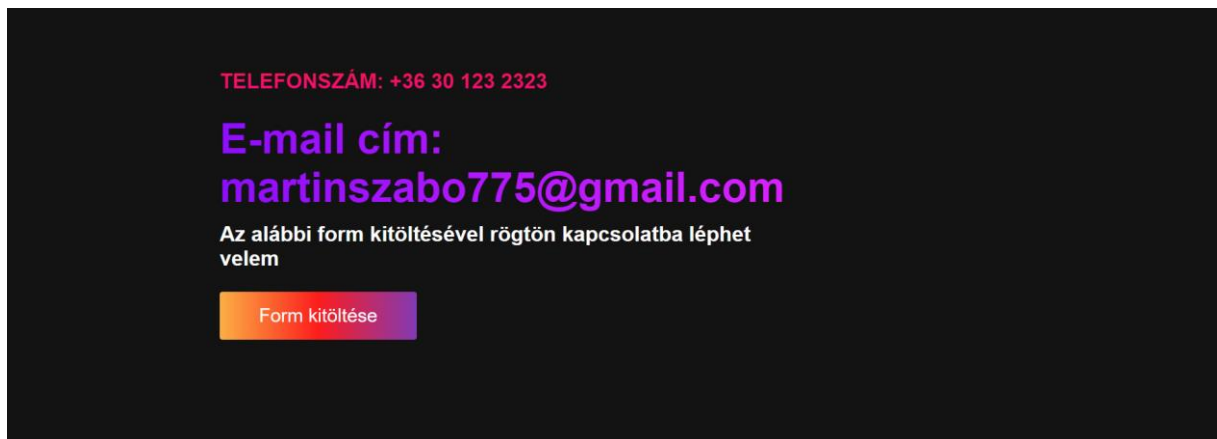
Ebben a részben azokat az iskolákat sorolom fel, ahová jártam, valamint ugye a Miskolci egyetemet, ahová most járok. a Tudj meg többet gomb elnavigál a studies.html oldalra, ahol a látogató többet megtudhat az alábbi iskolákról (főként a tapasztalataimról az iskolával kapcsolatban). A szekció egy kódrészlete az alábbi képen látható.

```
<!-- Studies Section -->
<div class="studies" id="studies">
  <h1>Tanulmányaim</h1>
  <div class="studies__wrapper">
    <div class="studies__card">
      <h2>Általános iskola</h2>
      <p>Petőfi Sándor Általános Iskola</p>
      <div class="studies__btn"><button><a href="studies.html">Tudj meg többet</a></button></div>
    </div>
```

A stílust, hogy a három iskola egymás mellett van egy színes téglalapban a CSS-ben oldottam meg gridek alkalmazásával, amelyet a studies\_\_wrapper class id-hoz rendeltem.

```
.studies__wrapper{
  display: grid;
  grid-template-columns: 1fr 1fr 1fr;
  grid-template-rows: 1fr;
}
```

Az elérhetőség szekciót is <div>-ekkel oldottam meg, mint az előző hármat. A stílusa ugyan az, mint a „hero section”-nek.



A „Form kitöltése” gomb elnavigál a contactme.html oldalra, ahol egy formot tudunk kitölteni, amellyel kapcsolatba léphet a látogató velem.

## A három mellékoldal bemutatása

Az aboutme.html és a studies.html oldal nagyon egyszerű HTML oldalak, melyeken a látogató több információt tudhat meg rólam, valamint az iskoláimról. Egy egyszerű stílust állítottam be hozzájuk, az aboutme.html oldalon található magamról egy kép, amelyet `` kóddal illesztettem be. Természetesen mindegyik HTML oldalon található egy link, amely a Kezdőoldalra visz.

A contactme.html oldalon egy formot tölthetünk ki, amellyel kapcsolatba léphet velem az illető.

A contact form on a blue background. It includes fields for Name, Email, a message, gender selection, a rating, and a date, followed by a Submit button.



Írtam egy JS scriptet is annak érdekében, ha az illető nem tölti ki a „Név”, „Email”, „Ok, amiért keres” mezőket, akkor egy figyelmeztetést kapjanak arról, hogy ezek kitöltése kötelező. A JavaScript kód megtalálható alább. Főként function-ökkel oldottam meg, amelyekben meghívok egy set metódust. Az eventListener annyit csinál, hogy megnézi, hogy érvényes emailt, nevet adott-e meg az illető, és ha igen akkor hajtódik végre a submit gomb.

```
const nameInput = document.getElementById("name");
const emailInput = document.getElementById("email");
const form = document.querySelector("form");

function validateName() {
  const nameValue = nameInput.value.trim();
  if (nameValue === "") {
    setError(nameInput, "A neve szükséges!");
    return false;
  } else {
    setSuccess(nameInput);
    return true;
  }
}

function validateEmail() {
  const emailValue = emailInput.value.trim();
  if (emailValue === "") {
    setError(emailInput, "Az email címe szükséges!");
    return false;
  } else if (!isValidEmail(emailValue)) {
    setError(emailInput, "Kérem adjon meg működő email címet.");
    return false;
  } else {
    setSuccess(emailInput);
    return true;
  }
}

function setError(input, message) {
  const parent = input.parentElement;
  const error = parent.querySelector(".error");
  input.classList.add("error");
  error.innerText = message;
}

function setSuccess(input) {
  const parent = input.parentElement;
  const error = parent.querySelector(".error");
  input.classList.remove("error");
  error.innerText = "";
}

function isValidEmail(email) {
  const emailRegex = /^[^\w-]+\.[\w-]+@([\w-]+\.)+[\w-]{2,4}$/;
  return emailRegex.test(email);
}

form.addEventListener("submit", function (event) {
  event.preventDefault();
  let isNameValid = validateName();
  let isEmailValid = validateEmail();
  if (isNameValid && isEmailValid) {
    form.submit();
  }
});
```

```
}  
}) ;
```