



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**

Membre de

HONORIS UNITED UNIVERSITIES

RAPPORT DE MINI-PROJET

Thème :

**Création D'un Site de vente des
cours en ligne**

Réalisé par : HARAF Aymen

2022-2023

Table de matières :

Introduction	4
Chapitre 1 : Les différent étapes pour réaliser le projet.....	5
1) Les outils de développement de notre application	5
2) Les langages de programmation utilisé	6
Chapitre 2 : Présentation du site web	8
1) Sign UP.....	8
2) Page Home	13
3) Page courses.....	17
4) Page about us.....	19
Conclusion	20
Bibliographie	21

Table de figure :

Figure1 : Visual Studio Code	1
Figure 2 : Bootstrap	2
Figure 3 :HTML	3
Figure 4 : CSS	4
Figure 5 : JavaScript	5
Figure 6 : Sign Up	6
Figure 7 : Erreur des champs.....	7
Figure 8 : Validation de connexion	8
Figure 9 : Barre de Navigation	9
Figure 10 : Header	10
Figure 11 : cours aléatoires	11
Figure 12 : Pied courses	12
Figure 13 : Page about.....	13

Introduction :

« La réalisation réside dans la pratique » - Buddha.

Ainsi Mr. AMEKSA Mohammed nous a confié un Mini-projet afin d'améliorer nos connaissances en front end et également pratiquer tout ce qu'on vient de savoir concernant le JavaScript et le CSS et HTML, Pour s'adapter au monde professionnel est confronté tous les problèmes quand on peut se trouver devant. Le projet qui nous a été conçu et un site web spécialisé dans la vente Cours concernant HTML CSS et PHP

Ce rapport sera composé de deux chapitres :

Le Premier chapitre sera consacré aux logiciels et les langages de programmation utilisés afin de mettre en place le site web.

Le Deuxième chapitre va nous présenter les interfaces graphiques du site ainsi qu'une explication détaillée de chaque partie.

Le rapport prendra fin avec une petite conclusion et des impressions ainsi qu'une bibliographie présentant les différentes sources qui m'ont permis de mettre en place ce mini-projet.

Chapitre I :

Outils et Langages Utilisés :

1) Les outils de développement de notre application :



Figure 1 : Visual Studio Code

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS. Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré.



Figure 2 : Bootstrap

Bootstrap est une collection d'outils utiles à la création du design de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option.

2) Les langages de programmation utilisé :



Figure 3 : HTML

L'**HTML** Le HyperText Markup Language, généralement abrégé HTML ou, dans sa dernière version, HTML5, est le langage de balisage conçu pour représenter les pages web.

Ce langage permet d'écrire de l'hypertexte (d'où son nom), de structurer sémantiquement une page web, de mettre en forme du contenu, de créer des formulaires de saisie ou encore d'inclure des ressources multimédias dont des images, des vidéos, et des programmes informatiques. L'HTML offre également la possibilité de créer des documents interopérables avec des équipements très variés et conformément aux exigences de l'accessibilité du web.



Figure 4 : CSS

Le **CSS** Les feuilles de style en cascade¹, généralement appelées CSS de l'anglais Cascading Style Sheets, forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C). Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception de sites web et bien pris en charge par les navigateurs web dans les années 2000.



Figure 5 : JavaScript

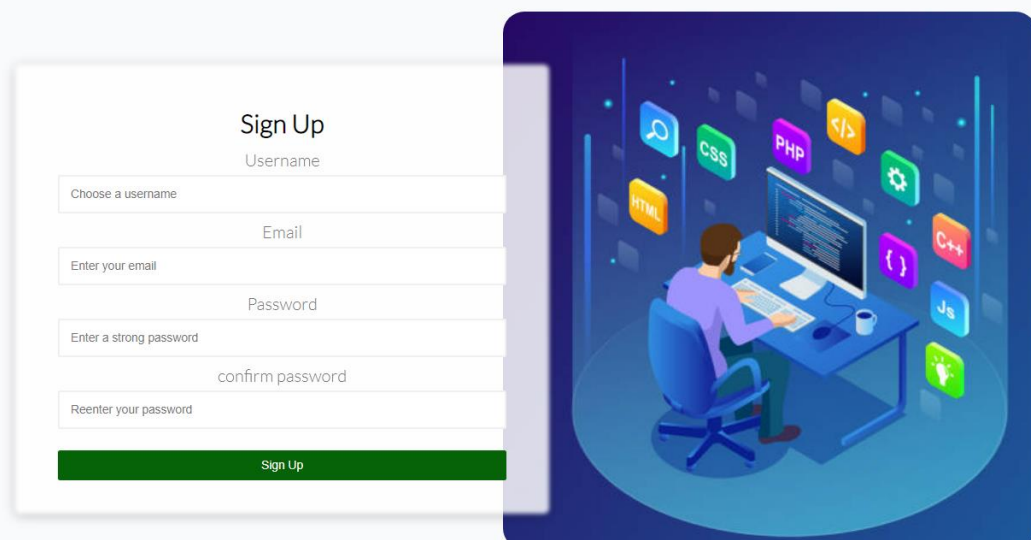
JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives et à ce titre est une partie essentielle des applications web. Avec les langages HTML et CSS, JavaScript est au cœur des langages utilisés par les développeurs web³. Une grande majorité des sites web l'utilisent⁴, et la majorité des navigateurs web disposent d'un moteur JavaScript⁵ pour l'interpréter.

JavaScript est aussi employé pour les serveurs Web⁶ avec l'utilisation (par exemple) de Node.js⁷ ou de Deno⁸.

Chapitre II :

Présentation du Site Web :

1) Sign UP :



The image is a composite of two parts. On the left is a 'Sign Up' form with a white background and a green 'Sign Up' button. The form fields are: 'Username' (with placeholder 'Choose a username'), 'Email' (with placeholder 'Enter your email'), 'Password' (with placeholder 'Enter a strong password'), and 'confirm password' (with placeholder 'Reenter your password'). On the right is a stylized illustration of a person with a beard sitting at a desk, working on a computer. The background is dark blue with various colorful icons representing programming languages and technologies: CSS, PHP, HTML, C++, Js, and a lightbulb icon.

Figure 6 : Sign Up

Cette page Sign Up est réalisé grâce à l'html et CSS

Sign Up

Username

az

Username must be at least 3 characters

Email

az

Email is not valid

Password

..

Weak password

confirm password

..

passwords do not match

Sign Up

Figure 7 : Erreur des champs

Comme on peut voir dans l'illustration ci-dessus on a mis des conditions grâce au JavaScript et si les valeurs entrées ne respectent pas les conditions données des messages d'erreur s'affiche

```
//check required fields
function checkRequired(inputArr){
  inputArr.forEach(function(input){
    if(input.value.trim() === ''){
      showError(input, `${getFieldName(input)} is required`);
    } else {
      showsuccess(input);
    }
  });
}
```

La condition `input.value.trim() === ""` vérifie si le champ est vide si oui il affiche un message d'erreur , si non il lance la fonction show success pour avoir un cadre vert qui indique que le champ est valide

```
function checkLength(input, min, max){
  if(input.value==""){
    showError(input, `${getFieldName(input)} is required`)
  }
  else if(input.value.length < min) {
    showError(input, `${getFieldName(input)} must be at least ${min} characters `);
  } else if(input.value.length > max){
    showError(input, `${getFieldName(input)} must be less than ${max} characters `);
  } else {
    showsuccess(input);
  }
}
```

Cette fonction vérifie la longueur du champ pour le username :

- `if(input.value.length < min)` vérifie si le username est identique au minimum demandé dans l'appel de la fonction.
- `(input.value.length > max)` de même pour le nombre de caractères maximum.

```
// check valid password
function checkValidPassword(input){

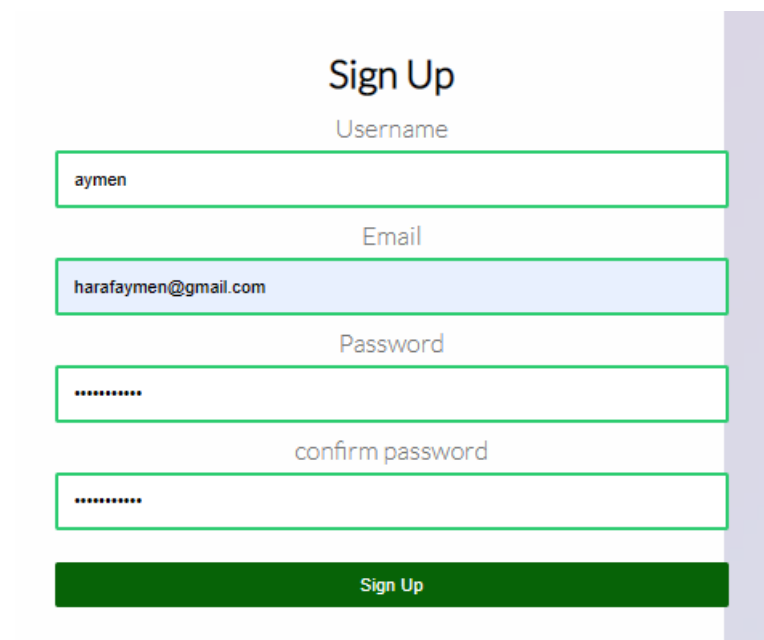
  var str = document.getElementById("password").value;
  if(input.value == ''){
    showError(input, `${getFieldName(input)} is required`)
  }
  else if (str.match( /[0-9]/g) &&
    str.match( /[A-Z]/g) &&
    str.match( /[a-z]/g) &&
    str.match( /^[^a-zA-Z\d]/g) &&
    str.length >= 8)
    showsuccess(input);
  else
    showError(input, 'Weak password ');
}
```

Pour vérifier si un mot de passe est fort, qui correspond à un modèle spécifié, on utilise **le RegExp**. Pour bien comprendre par exemple pour le symbole `[a-z]` signifie que le mot de passe doit contenir au moins un caractère minuscule.

```
//show input message error
function showError(input, message){
  const formControl = input.parentElement;
  formControl.className = 'form-control error';
  const small = formControl.querySelector('small');
  small.innerText = message;
}

//show input success message
function showsuccess(input){
  const formControl = input.parentElement;
  formControl.className= 'form-control success';
}
```

Pour la fonction **showError ()** et **showSuccess ()** deux fonctions l'une affiche un message d'erreur avec un border rouge et l'autre affiche un border vert, appelées lors de chaque vérification de champ.



The image shows a 'Sign Up' form with the following fields and values:

- Username:** aymen
- Email:** harafaymen@gmail.com
- Password:** [Redacted]
- confirm password:** [Redacted]

A green border highlights each input field, indicating successful validation. A green 'Sign Up' button is at the bottom. The form is set against a light gray background with a vertical purple bar on the right.

Figure 8 : Validation de Connexion

Si tous les champs respectent les conditions demandées une indication s'affichent que tt les champs sont valides

2) La Page Principale :



Home Courses About

Figure 9 : Barre de Navigation

Cette barre de navigation a été conçue avec html et css en ajoutant des effets Hover pour un peu plus de beauté pour notre site

```
<div class="navigationbar">
  <ul>
    <li><a href="../index/index.html"> Home</a></li>
    <li><a href="../3rd page/courses.html"> Courses</a></li>
    <li><a href="../about/indexx.html"> About</a></li>
  </ul>
</div>
```

```
.navigationbar {
  background-color: ■ rgb(238, 238, 238);
  border: 10px;
  border-color: □ rgb(0, 0, 0);
  margin: 0;
  padding: 0;
}

ul {
  list-style-type: none;
  font-family: 'Lato', sans-serif;
  margin: 0px 100px;
  padding: 0;
  display: flex;
  overflow: hidden;
}

ul li {
  float: left;
  margin: 0px 5px;
  transition: 0.5s;
}

ul li a {
  display: block;
  position: relative;
  text-decoration: none;
  padding: 10px;
  font-size: 18px;
  text-align: center;
}
```

```
ul li a:hover {
  background-color: ■ rgb(214, 209, 209);
  transform: scale(1.2);
  opacity: 1;
  filter: blur(0);
  text-decoration: none;
  color: □ black;
}

ul li a:before{
  content: '';
  position: absolute;
  top: 0;
  left: 0;
  width: 100%;
  height: 100%;
  background: ■ orange;
  transition: 0.5s;
  transform-origin: right;
  transform: scaleX(0);
  z-index: -1;
}

ul li a:hover:before {
  transition: transform 0.5s;
  transform-origin: left;
  transform: scaleX(1);
}
```

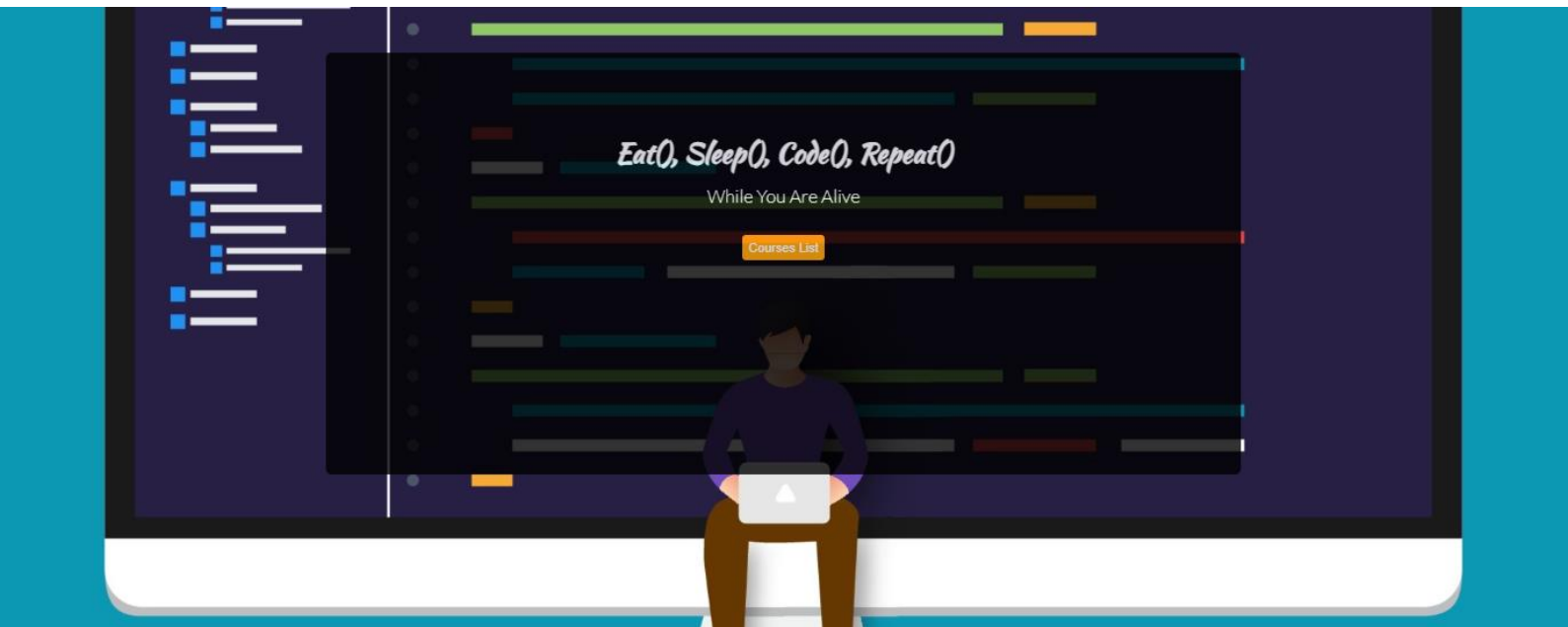


Figure 10 : Header

Cette partie presente une section qui contient le tout du header ensuite un container qui contient le text et le bouton dedans, l'image est appelée via background-image sur CSS.

```
<section id="home">
  <div class="text-container">
    <h1>Eat(), Sleep(), Code(), Repeat()</h1>
    <p id="p2">While You Are Alive</p>
    <button><a href="#courses"></a> Courses List</button>
  </div>
</section>
```

```
#home {
  background-image: url("../image/coding.jpg");
  position: relative;
  width: 100%;
  height: 70vh;
  background-size: cover;
  background-position: center;
  display: flex;
  flex-direction: column;
  align-items: center;
}
```

```
#home .text-container {
  background-color: black;
  justify-content: center;
  display: block;
  color: white;
  margin: 40px;
  padding: 70px;
  border-radius: 5px;
  width: 50em;
  height: 23em;
  position: static;
  filter: opacity(80%);
  right: 550px;
  left: 300px;
}
#home .text-container h1 {
  font-family: 'Kaushan script', sans-serif;
  text-align: center;
}
#home .text-container p {
  text-align: center;
  font-family: 'Lato', sans-serif;
  padding: 10px;
  margin-bottom: 15px;
}
}
```

```
#home .text-container button {
  cursor: pointer;
  background-color: #ff9900;
  border: 2px solid black;
  border-radius: 3px;
  color: white;
  display: block;
  font-size: 11px;
  padding: 5px;
  margin-top: 5px;
  margin: auto;
}
}
```

/ Courses



CSS Selectors
69.9\$



Variables And Sata Type Of Javascript
19.9\$



Basics of HTML
5.9\$

[all courses](#)

Figure 11 : cours aléatoires

Cette Partie présente les différents cours disponibles dans le site et ils changent à chaque fois que la page est rafraîchie. Aussi un bouton permettant d'accéder à la page des cours qui a aussi été mis en place comme le Bouton précédent.

```
var cours = [  
  ["../image/html attributes.jpg",  
   "HTML attributes and values",  
   "19.9$"],  
  ["../image/crud php mySQL bootstrap.jpg",  
   "Manipulating Crud Using Php",  
   "45.9$"],  
  ["../image/css selectors.jpg",  
   "CSS Selectors",  
   "69.9$"],  
  ["../image/Basics of HTML.png",  
   "Basics of HTML",  
   "5.9$"],  
  ["../image/js variables.jpg",  
   "Variables And Sata Type Of Javascript",  
   "19.9$"],  
  ["../image/What is javascript.png",  
   "What Is Javascript ?",  
   "9.9$"],  
];
```

D'abord un tableau d'éléments dont on va a chaque rrefresh de la page web, prendre un index du tableau pour afficher l'element dedans avec les fonction qui suivent :


```

let count1 = Math.floor(Math.random()*cours.length),
count2 = Math.floor(Math.random()*cours.length),
count3 = Math.floor(Math.random()*cours.length);
while(count1 == count2 ){
    count2 = Math.floor(Math.random()*cours.length);
}
while(count2 == count3 ){
    count3 = Math.floor(Math.random()*cours.length);
}
while(count1 == count3 ){
    count2 = Math.floor(Math.random()*cours.length);
}
const tab = [cours[count1],cours[count2],cours[count3]]

```

Comme ça on prend un index et on test si 2 index se ressemblent, si oui on retient le premier et lance la recherche aléatoire d'un index pour le deuxième count (pour ne pas avoir 2 éléments qui se ressemblent) avec les fonction math.random et math.floor pour avoir les index 0,1,2,3 Ensuite on lance une fonction pour créer la div du cours avec un image-container pour Contrôler l'image, h3 pour le Titre et p pour le prix, après la création de chaque élément et donner à chacun l'élément qui lui appartient d'après le tableau, on l'affiche.

```

function test(courseAl){
    var container=document.getElementById("courses");

    var course=document.createElement('div');
    var image=document.createElement('img');
    course.classList="course";
    var image_container=document.createElement('div');
    image_container.classList="image_container";
    image_container.appendChild(image);
    var phrase=document.createElement('h3');
    var taman=document.createElement('p');

    phrase.innerHTML=courseAl[1];
    taman.innerHTML=courseAl[2];
    image.src=courseAl[0];

    course.appendChild(image_container);
    course.appendChild(phrase);
    course.appendChild(taman);

    container.appendChild(course);
}

tab.forEach(element => {
    test(element)
});

```

3) La Page Cours :

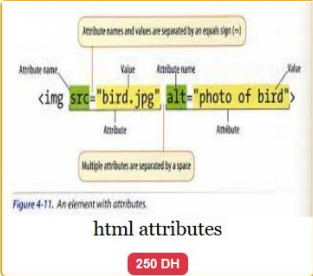
[Home](#) [Courses](#) [About](#)

Category

- all
- html
- css
- js
- php


Price

value : 100 DH



html attributes


250 DH



HTML Basic

basics of HTML


120 DH



HTML tags

HTML elements and tags

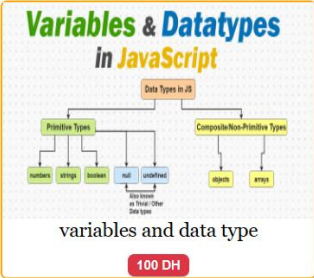
199 DH



ADVANCED CSS SELECTORS

CSS selectors


299 DH



Variables & Datatypes in JavaScript

variables and data type


100 DH




JavaScript Operators

JavaScript operators


350 DH



JavaScript



JavaScript DOM



JavaScript DOM

Figure 12: Page courses.

Dans cette page on affiche tous les cours disponibles, avec une barre de recherche et des filtres et une range barre pour le filtre de prix

```

let divis = document.createElement('div')

let Cname = document.createElement('h2');
Cname.setAttribute('class', 'course-title');
let Ctit = document.createTextNode(titre);
Cname.appendChild(Ctit);

let span = document.createElement('span');
span.setAttribute('class', 'badge badge-pill badge-danger bg-danger text-white');
span.appendChild(document.createTextNode(prix + ' DH'));

div.appendChild(img);
divis.appendChild(Cname);
divis.appendChild(span);
div.appendChild(divis)
cont.append(div)
}

```

Là on a créé les divisions qui vont contenir l'image, le titre et le prix

```

let Categories = (cat)=>{
  let li = document.createElement('li');
  li.appendChild(document.createTextNode(cat));
  li.setAttribute('class', 'list-group-item');
  li.id=cat;
  listing.append(li);
}

let categories = ['all', ...new Set(courses.map((v) => v.category.toLowerCase()))]

categories.forEach((v) => {
  Categories(v);
})

```

Cette fonction est pour créer les éléments du filtre en lui donnant une classe List-group-item et on affiche les éléments grâce à forEach .

La Fonction filtrage avec prix :

```
let Valeur={()=>{
  let value =document.getElementById('range').value
  document.getElementById('valeur').innerHTML = "Value : "+ value +"DH"
  document.getElementById('range').addEventListener('click',()=>{
    document.getElementById('cards').innerHTML = " "
    for(let i =0;i<courses.length;i++){
      if(courses[i].price <= value){
        Cours(courses[i].image,courses[i].title,courses[i].price)
      }
    }
  })
}
```

Dans cette fonction on récupère les éléments html du range input et on la mise à jour avec les nouvelles valeurs selon le niveau de la barre après on parcourt tous les éléments du tableau en cherchant les prix inférieurs ou égaux le prix afficher par la range barre grâce à la boucle for.

4) ABOUT

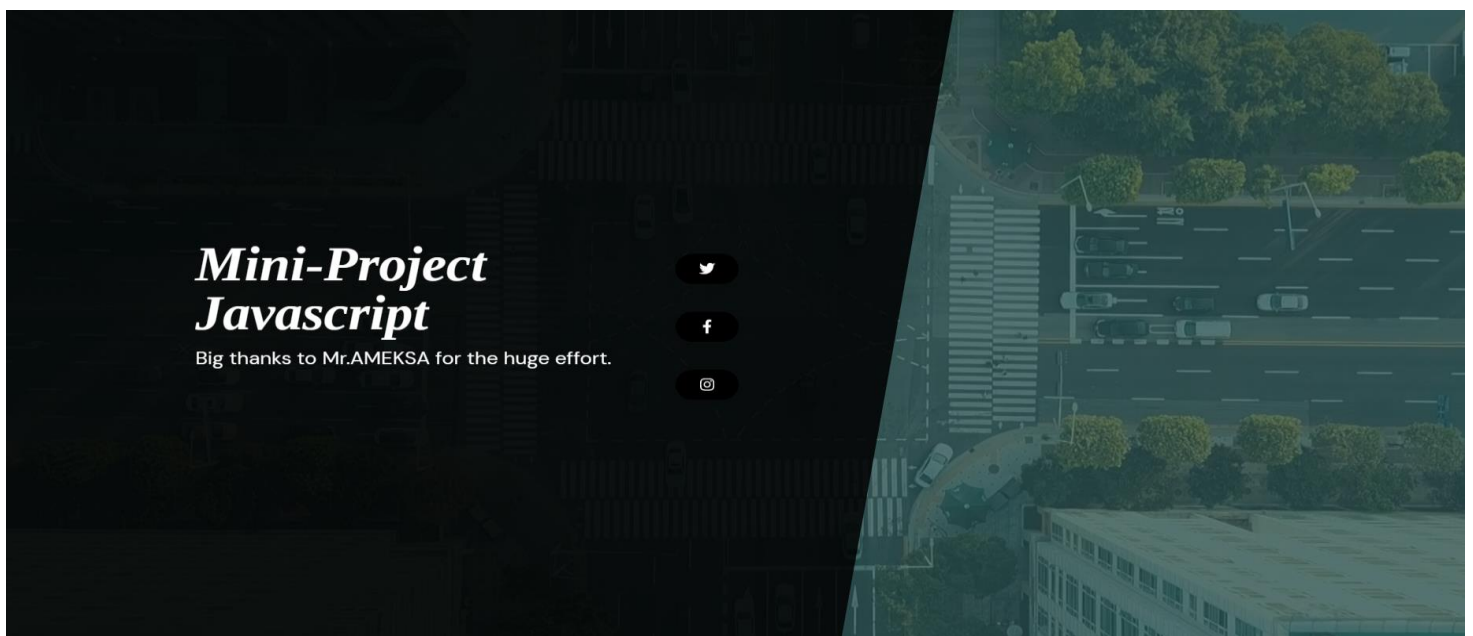


Figure 13: Page About.

Cette page est conçue en utilisant Bootstrap et contient un arrière-plan dynamique avec un mp4, un petit message de remerciement et mes réseaux sociaux

Conclusion

Je pense qu'on peut généraliser que ce projet nous a aidé énormément à bien pratiquer tout ce qu'on a parcouru durant ce dernier semestre autant sur le plan HTML CSS que JS, grâce à vous Monsieur et ce projet j'ai plus savoir plus sur le DOM et surtout et l'utilisation du bootstrap dont je ne savais rien et donc je vous remercie encore une fois.

BIBLIOGRAPHIE :

<https://www.youtube.com/@freecodecamp>

<https://getbootstrap.com/docs/5.3/components/badge/>

Google image pour tous les images

<https://getbootstrap.com/docs/5.3/components/card/>

<https://getbootstrap.com/docs/5.3/components/buttons/>

<https://www.w3schools.com/js/default.asp>

<https://mdbootstrap.com/docs/standard/extended/login/>

JavaScript, HTML, CSS cheat sheets .