

Rapport d'activité

Sous le thème :

PREMIER PAS AU FRONTEND



Réaliser une interface utilisateur...



Réalisé par :
ACHRAF ABBASSI

Le 10/01/2020

Table des matières

Introduction	3
Chapitre 1 : PREMIERE APPROCHE THEORIQUE DU HTML ET DU CSS	4
Le langage HTML.....	4
Qu'est-ce que HTML ?	4
Définition :.....	4
Exemple de code HTML :	4
Lecture du code HTML :	5
Reconnaître une balise :	5
Balises ouvrantes et balises fermantes :	5
Structure de base :	6
Les liens :	7
Les images :	7
La balise :	7
Compléments sur HTML :	8
Marche à suivre.....	8
Le CSS.....	9
Introduction : qu'est-ce que CSS ?	9
La syntaxe :	9
Le sélecteur de classe :	9
Lier CSS à un document HTML :	10
Différence entre class et id	10
Balise div, marges et espacement	11
Chapitre 2 : Création CV en HTML & CSS	13
Mettre en page son CV grâce au CSS :	14

Introduction

Aujourd'hui, nous allons apprendre à créer un CV totalement personnalisé en HTML et en CSS et à le mettre en ligne. Cela pourrait être un bon moyen de vous différencier de vos concurrents et donc de vous faire embaucher plus facilement.

Chapitre 1 : PREMIERE APPROCHE THEORIQUE DU HTML ET DU CSS

Le langage HTML

Les pages qui suivent présentent le langage HTML. Il s'agit du langage qui permet d'afficher des informations sur une page web. En particulier, les points suivants sont discutés.

- les principes de base et la structure d'un document HTML
- comment mettre en forme du texte (gras, italique, etc...)
- comment afficher des images
- comment créer des liens vers d'autres documents.

Qu'est-ce que HTML ?

Définition :

HTML est un langage pour fonction de structurer et de donner du sens à du contenu. HTML est un acronyme pour Hyper Text Markup Language (langage de balisage d'hypertexte)

Exemple de code HTML :

Voici un exemple de code HTML :

```
<html>
  <head>
    <title>Le titre de la page</title>
  </head>
  <body>
    <h1>Mon premier titre</h1>
    <p>Mon premier <b>paragraphe</b></p>
  </body>
</html>
```

Figure 1 : Un exemple de code HTML

Lecture du code HTML :

Les navigateurs Internet (comme Internet explorer, Firefox ou Chrome par exemple) sont capables de lire les documents HTML et de les afficher, sous forme de pages web. Certains éléments, comme, etc.... ne sont pas affichés sur la page web. Ces éléments donnent des indications au navigateur sur la façon d'afficher le texte. Le code HTML ci-dessus, interprété par un navigateur, donne le résultat suivant :

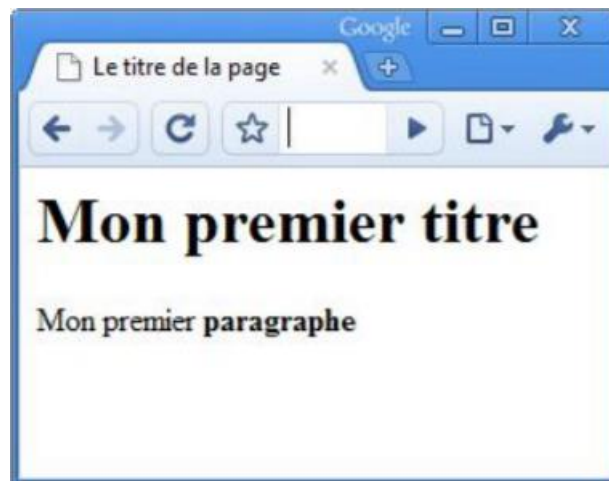


Figure 2 : Le code HTML de la figure 1 interprété par Google Chrome

Reconnaître une balise :

Document HTML = textes + balises

Pour composer une page web, on écrit donc du texte, et on y insère des caractères spéciaux qui permettront au navigateur (par exemple Firefox ou Internet Explorer), de mettre en forme ce texte. Ces caractères spéciaux se nomment des balises. Un document HTML est donc composé de

- textes
 - balises
- Sur la figure 1, les balises sont :

```
<html>, <head>, <title>, </title>, </head>,  
<body>, <h1>, </h1>, <p>, <b>, </b>, </p>,  
</body>, </html>
```

Balises ouvrantes et balises fermantes :

Les balises vont par paire : une balise ouvrante et une balise fermante. Les balises fermantes s'écrivent avec un slash (/). Voici les balises ouvrantes et les balises fermantes de l'exemple ci-dessus :

Balises ouvrantes: <html>, <head>, <title>, <body>, <h1>, <p>,

Balises fermantes: </title>, </head>, </h1>, </p>, , </body>, </html>

Le texte écrit entre la balise ouvrante et la balise fermante est le contenu de la balise. Dans l'exemple suivant

Mon premier paragraphe

Le terme paragraphe est le contenu de la balise

Voici le contenu de certaines balises :

- Contenu de la balise <head> : <title> Le titre de la page </title>
- Contenu de la balise <h1> : Mon premier titre </h1>
- Contenu de la balise <body> : <h1> Mon premier titre </h1><p>Mon premier paragraphe

Structure de base :

La structure de base de tout document HTML est la suivante :

```
<html>
  <head>
  </head>
  <body>
  </body>
</html>
```

L'entier d'un document HTML est contenu entre les balises <html> et </html>. Entre les balises <head> et </head> se trouvent des informations supplémentaires, qui n'apparaissent pas directement sur la page, comme par exemple les styles utilisés sur la page, l'encodage des caractères, le nom donné à la page, ou encore le code des fonctions qui seront utilisées sur la page. Les informations qui apparaîtront directement sur la page sont écrites entre les balises <body> et </body>.

<title>

La balise <title>, inscrite dans la partie head correspond en fait au texte qui sera affiché comme titre de l'onglet (voir figure 2). Le contenu de cette balise ne correspond pas à un titre qui apparaîtrait directement sur la page.

<h1>

La balise <h1> signifie heading 1. Cette balise permet de faire des titres, à l'intérieur de la page. Les caractères entre <h1> et </h1> sont donc affichés avec une taille de police plus grande que les autres caractères de la page. Il est également possible de définir des sous-titres, des titres de section, etc.... en utilisant les balises suivantes : <h1>, <h2>, <h3>, <h4>, <h5>, <h6>. <h1> sera utilisée pour les titres principaux, <h2> pour les sous-titres, <h3> pour les titres de section, etc...

<p>

La balise permet de définir les paragraphes

La balise (abréviation pour bold) permet de mettre du texte en gras. Pour mettre du texte en italique, on utilise <i> (pour italic) ou la balise (*pour emphasis*).

Les liens :

La balise, avec son attribut href permet de créer un hyperlien vers une autre page. Voici un exemple :

```
<a href="http://www.google.ch">GOOGLE</a>
```

Quelques explications sur cet exemple :

- Dans cet exemple, le terme GOOGLE apparaît sur la page. Lorsque l'utilisateur clique sur le lien, la page <http://www.google.ch> apparaîtra.
- Un lien, nommé GOOGLE, et pointant vers <http://www.google.ch> est créé.
- href est l'attribut de la balise a
- la valeur de l'attribut href donne l'adresse vers laquelle l'utilisateur sera dirigé – Le texte qui est placé entre la balise <a...> et apparaît sur la page.

Les images :

La balise permet d'insérer une image dans un document HTML. Cette balise possède un attribut nommé src qui indique où prendre l'image à afficher. Par exemple, l'instruction suivante :

```
<img src='mon_img.jpg' />
```

Affiche l'image nommée mon_img.jpg sur la page.

La balise :

La balise span n'a aucun effet ! Par exemple, écrire Un peu de texte

Ou

Un peu de texte

Produira exactement le même résultat. Cependant, une telle balise est utile pour délimiter une portion du texte, à savoir le texte présent entre la balise ouvrante et la balise fermante .

Compléments sur HTML :

Lors de la création d'un page Internet en HTML, quelques règles s'imposent quand à la présentation du code HTML et à la marche à suivre pour créer cette page.

Marche à suivre

- La structure de la page doit être définie AVANT de commencer à écrire du code. Cela ne veut pas dire que l'on ne peut pas changer cette structure par la suite pour y apporter des améliorations.
- Avant de mettre du texte et des images sur la page, commencer par créer un squelette, c'est-à-dire par écrire les tags HTML sans texte ainsi que les informations de taille pour que cela ressemble le plus possible à la structure de la page finale. Par exemple :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
  <head>

    <title></title>

    <meta http-equiv="Content-Language" content="fr" />
    <meta http-equiv="content-type" content="text/html; charset=UTF-8" />

  </head>
  <body>

    <table width = "800px">
      <tr>
        <td width = "50%">
        </td>
        <td>
        </td>
      </tr>
    </table>

    ...
  </body>
</html>
```

- le code doit être présenté de manière lisible. Une bonne habitude à prendre est d'indenter, c'est-à-dire décaler, le code comme ci-dessus. Ainsi les balises ouvrantes et fermantes se trouvent alignées dans une même colonne ce qui rend le code plus lisible et donc permet de trouver plus facilement les erreurs. Cela devient particulièrement important lorsque vous avez des milliers de lignes de code et que plusieurs personnes travaillent sur un même projet !
- une fois la structure de la page définie, introduire le texte et les images.
- le style vient en dernier. Les couleurs, les types de polices de caractère, leur taille, etc se met tout à la fin. Nous verrons d'ailleurs qu'avec CSS, le style est en général stocké dans un fichier séparé qu'on ne remplit qu'une fois le fichier html terminé.

Le CSS

Introduction : qu'est-ce que CSS ?

Cette section présente le langage CSS. CSS vous permet de définir l'apparence des textes (comme la police, la couleur, la taille, etc...), ainsi que l'agencement de la page (comme les marges, l'arrière-plan, etc...). CSS définit donc la présentation du document. CSS est l'abréviation de Cascading Style Sheets. Un style définit la façon dont un élément HTML (par exemple `<h1>`) sera affiché.

Ces styles peuvent être définis dans une feuille de style externe (un fichier .css). Une feuille de style peut être utilisée pour définir la présentation de plusieurs documents HTML, ce qui permet de gagner beaucoup de temps. HTML a été conçu pour définir la structure d'un document pas sa présentation.

Par conséquent tout ce qui est lié à la présentation d'un document devrait être défini à l'aide de CSS.

Typiquement, il faut préférer CSS à l'utilisation de balises HTML permettant de définir la présentation d'un document (comme par exemple ``)

La syntaxe :

La syntaxe de base de CSS est composée de 3 parties :

- un sélecteur
- une propriété
- une valeur

Ces trois parties sont écrites de la façon suivante :

Sélecteur {propriété:valeur}

Un sélecteur correspond à une balise HTML (`<p>`, `<h1>`, etc. ...) et la propriété est un attribut dont on veut changer la valeur.

Exemple : `h1 {font-size:100px}`

Dans cet exemple, tous les titres principaux (`<h1>`) du document auront une taille de 100 pixels. Si la valeur d'un attribut contient un espace, alors la valeur de l'attribut s'écrit en guillemets : `h2 {font-family:"sans serif"}` Il est possible de définir plusieurs attributs pour un même sélecteur. Dans ce cas, chaque propriété sera séparée par un point-virgule : `p {font-family:"sans serif"; font-size: 90%; color: blue}` En écrivant une propriété par ligne, la lisibilité est meilleure : `p { font-family:"sans serif"; font-size: 90%; color: blue; }`

Le sélecteur de classe :

Avec le sélecteur de classe il est possible d'appliquer des styles différents pour des éléments HTML d'un même type. Par exemple, si nous souhaitons que certains paragraphes (`<p>`) soient affichés en

rouge (disons, pour mettre en évidence des paragraphes importants), et que d'autres paragraphes aient un alignement centré, nous pouvons définir deux classes pour chacun de types de paragraphe :

```
p.introduction{text-align:center}  
p.important{color:red}
```

Dans le document HTML, l'attribut class permet de choisir la classe qui sera utilisée :

```
<p class="introduction">Un paragraphe d'introduction </p>  
<p class="important">Un paragraphe important</p>
```

Pour définir une classe qui s'applique à tous les éléments HTML, et pas uniquement à un seul élément HTML (comme <p> dans l'exemple ci-dessus), on utilise la syntaxe suivante : l'élément HTML n'est pas écrit ; on commence directement par un point, suivi du nom de la classe.

Exemple :

```
.important{color:red}
```

Ainsi, la classe important pourra être utilisée pour tous les éléments HTML :

```
<h1 class="important">Un titre important</h1>  
<p class="important">Un paragraphe important <.p>
```

Lier CSS à un document HTML :

Pour intégrer du CSS dans un document HTML, il existe trois possibilités :

- une feuille de style externe
- une feuille de style interne
- un style défini directement dans la balise (inline style)

Les feuilles de style externes Les feuilles de style externes sont idéales lorsque les mêmes styles sont appliqués sur plusieurs documents HTML. La balise <link>, à écrire dans la partie <head> du document HTML permet de lier une feuille de style à un document HTML :

```
<head>  
< link rel="stylesheet" type="text/css" href="mon_style.css" />  
</head>
```

Généralement, les fichiers contenant les styles ont l'extension .css

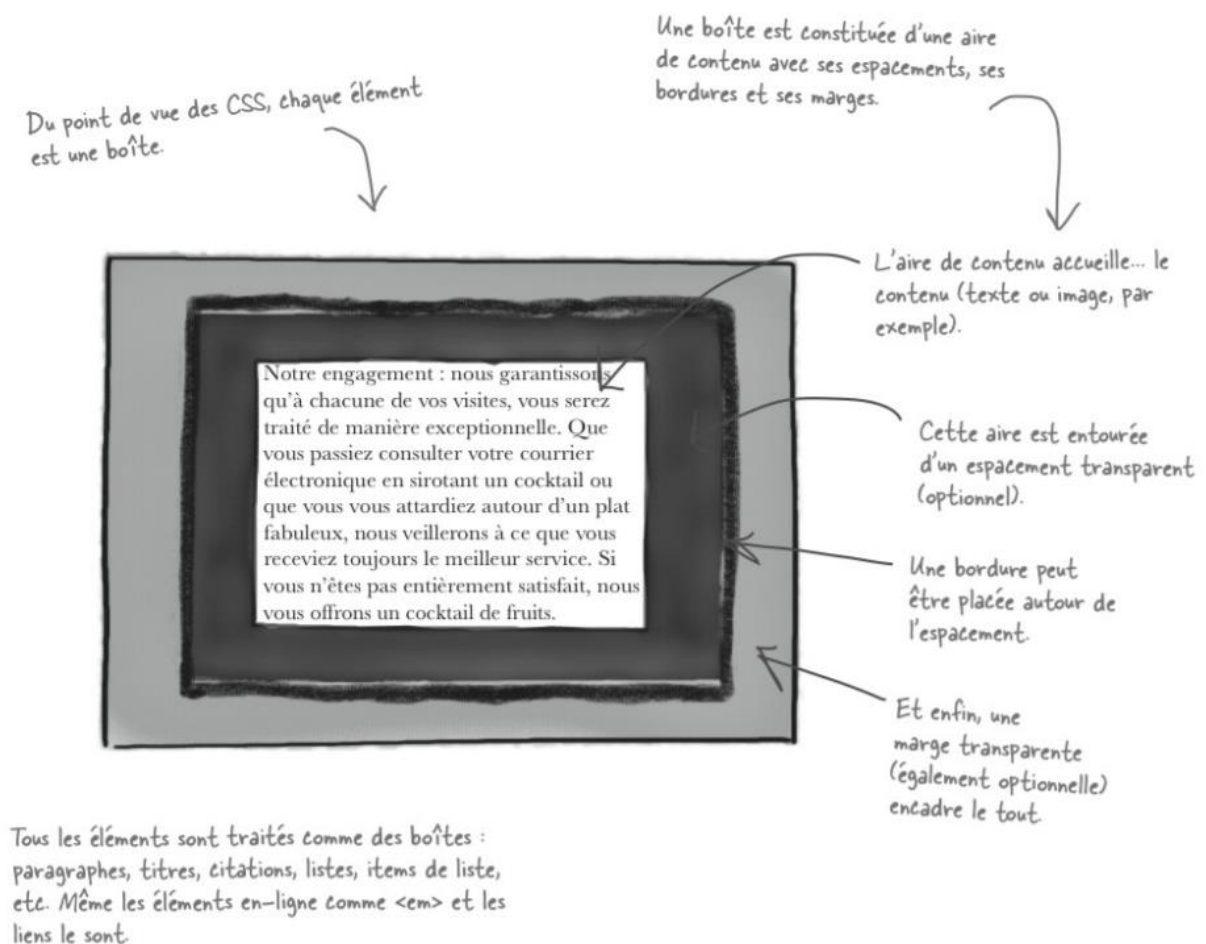
Différence entre class et id

Servez-vous de l'attribut class pour styler un groupe d'éléments, c'est à dire pour appliquer un même style à plusieurs éléments d'une page. L'attribut id permet de donner un nom unique à un élément. Il

sert aussi à donner un style spécifique à un élément. Il ne doit y avoir qu'un seul élément par page portant le même id.

Balise div, marges et espacement

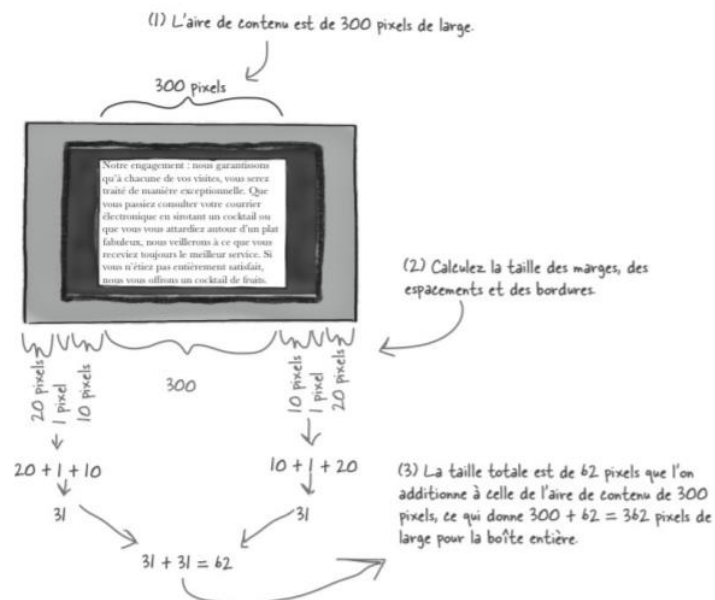
Un div est comme un conteneur dans lequel on met des éléments pour les garder ensemble. Cela permet, par exemple, d'appliquer un même style à un groupe d'éléments ou encore de placer ces différents éléments ensemble sur une page.



Pour ajouter de l'espacement, on utilise la propriété `padding`. Pour ajouter une marge, on utilise la propriété `margin`. Pour avoir une idée de la largeur de toute la boîte, il faut additionner la largeur de l'aire de contenu à celles des marges de gauche et de droite plus celle de la bordure qu'il faudra compter deux fois puisqu'il y en a une à gauche et une autre à droite.



Voici un exemple :





Chapitre 2 : Création CV en HTML & CSS

Nous allons commencer par rédiger notre CV en HTML en gardant à l'esprit que l'on va devoir le styliser avec le CSS par la suite.

N'oubliez pas que, normalement, vous devez toujours créer une maquette avant de vous lancer et savoir où vous allez et notamment comment et à quels éléments vous allez appliquer des styles en CSS.

Si vous ne faites pas ce travail préalable, vous risquez de perdre énormément de temps à faire des allers retours entre vos feuilles HTML et CSS et à ajuster sans cesse votre code HTML.

Ensuite, chaque section (formation, expériences professionnelles, etc.) aura son div. Comme les sections ont une mise en page équivalente, on pourra peut-être se contenter de n'appliquer certains styles qu'une seule fois en attribuant là même class à différents div.

Enfin, je vais grouper tout mon CV dans un gros div pour lui donner une taille fixe et pour pouvoir éventuellement ajouter une couleur ou une image de fond à ma page web.

Cela nous donne donc un premier squelette.

Ensuite, si le travail de réflexion préliminaire a été bien fait, nous n'avons plus qu'à remplir les différentes parties.

Commençons avec le premier gros div. J'ai pris soin au préalable d'enregistrer ma photo dans le même dossier que ma page HTML sous le nom « 1.JPG ».

```
<header class="presentation">
  <div></div>
  <div><h1 class="myname">Abbassi Achraf</h1></div>
  <div><h3 class="option" style="color: blue;">Technicien Supérieur en Electromécanique et Systèmes Automatisés</h3></div>
  <div class="plusinfo">
    <div class="info"><a href="https://www.linkedin.com/in/achraf-abbassi-21453b18a/" target="_blank">Linkedin.com</a>
    <div class="info"><a href="mailto:hraf.abbassi@gmail.com">hraf.abbassi@gmail.com</a></div>
    <div class="info"><a href="tel:+212 621969508">+212 621969508</a></div>
  </div>
  <div class="plusinfo"><p>Je m'appelle Achraf Abbassi j'ai 22 ans et je suis un technicien supérieur en Electromécanique et Systèmes Automatisés marocain.
  Je suis passionné par les nouvelles technologies surtout les véhicules embarqués, d'une formation approfondie en électronique et traitement de signal
  Grâce à mes différentes expériences, j'ai pu développer ces dernières années des compétences en matière de:
  Système temps réel & embarqués. <br> J'ai également développé des soft skills telles que:
  La capacité d'apprentissage, la capacité d'adaptation.</p></div>
</header>
```

Rappelez-vous que pour le moment on ne s'occupe que du contenu ; le design ne viendra qu'à la fin. Poursuivons donc en remplissant nos différentes sections.

Ensuite, nous n'avons plus qu'à remplir le contenu de nos divs et on pourra passer à la mise en page en CSS !

Mettre en page son CV grâce au CSS :

Maintenant que nous avons tout notre contenu, nous pouvons passer à la mise en forme de notre CV.

Pour cela, nous allons créer un nouveau fichier « style.css » et allons lier notre fichier HTML et notre fichier CSS comme on sait le faire, avec l'élément link :

```
1  <!DOCTYPE html>
2  <html lang="fr">
3  <head>
4    <meta charset="UTF-8">
5    <title>Mon CV</title>
6    <link rel="stylesheet" href="style.css">
```

On va commencer par écrire les styles généraux de notre CV. Pour cela, on va cibler la class « page ».

Nous allons commencer par donner une largeur à notre CV (nous ne voulons pas que les éléments flottants se baladent de gauche à droite selon que les visiteurs agrandissent ou rétrécissent leur fenêtre) avec la propriété width.

Ensuite, nous allons centrer notre contenu. Pour centrer le contenu à l'intérieur d'un width qui possède une largeur, c'est très simple, il suffit de donner la valeur auto à la propriété margin. Enfin, nous allons spécifier les polices souhaitées pour notre CV (je vous conseille de ne pas utiliser de polices trop originales ou qui pourraient être peu lisibles pour un CV) ainsi que la taille par défaut de nos textes.

```
22 }
23 .page{
24     max-width: 960px;
25     height: auto;
26     margin: 40px auto;
27     background: white;
28     padding: 20px;
29     box-shadow: -1px 1px 12px 12px gray;
30 }
```

Pour l'heure, nous allons continuer à mettre en forme nos sections et notamment le texte à l'intérieur de nos sections.

Nous allons déjà justifier le texte afin que notre CV paraisse plus professionnel, puis changer la taille et la couleur des titres et les mettre en majuscule. Nous allons également changer le poids de nos sous-titres et de nos dates.

```
39 .option{
40     margin-left: 210px;
41 }
42 }
43 .info{
44     display: inline;
45     margin-right: 60px;
46 }
47 .plusinfo{
48     margin-left: 210px;
49     margin-top: 20px;
50 }
51 .presentation{
52     border-bottom: 1px solid darkgrey;
53     padding: 24px 0;
54 }
55 .ste{
56     display: inline;
57 }
58 }
59
60 .date{
61     float: right;
62 }
63 .experience{
64     margin-top: 20px;
65 }
66 a{
67     text-decoration: none;
68     color: black;
69 }
70 a:hover{
71     color: blue;
72 }
73 .plusin{
74     margin-left: 210px;
75     margin-top: 20px;
76     border-bottom: 1px solid darkgrey;
```




Abbassi Achraf

Technicien Supérieur en Electromécanique et Systèmes Automatisés

🌐 LinkedIn.com

✉️ hrab.abbassi@gmail.com

☎️ +212 621969508

Je m'appelle Achraf Abbassi j'ai 22 ans et je suis un technicien supérieur en Electromécanique et Systèmes Automatisés marocain. Je suis passionné par les nouvelles technologies surtout les véhicules embarqués, d'une formation approfondie en électronique et traitement de signal et comportement des Systèmes. Enthousiaste et patient et à la recherche de concrétiser mes connaissances théoriques en Système Embarquée. Grâce à mes différentes expériences, j'ai pu développer ces dernières années des compétences en matière de: Système temps réel & embarqués. J'ai également développé des soft skills telles que: La capacité d'apprentissage, la capacité d'adaptation.

Experience

OCP

Avr 2019 – Juin 2019

Technicien d'automatisation

- Etude et automatisation des trémies
- Programmation des entrées et des sorties analogiques
- Utilisation et programmation des blocs de régulation de RsLogix
- Programmation de la régulation sur un process reel
- Supervision du process et de la boucle de régulation avec RsView32
- Réaliser l'interface de supervision de la boucle de régulation
- Réaliser la commande du process depuis l'interface de supervision
- Afficher les courbes des informations analogiques
- Choix de variateur de vitesse
- Paramétrage, câblage et mise en service d'un variateur de vitesse

OCP

Mai 2018 – Juin 2018

Technicien d'automatisation

- Simatic step7 : diagnostic, configuration et programmation des API siemens
- WIN CC explorer : développement des interfaces de supervision des process industriels
- Câblage électrique des cartes E/S DES API S7 300

OCP

Avr 2016 – Mai 2016

Technicien de maintenance électrique

- Etude des MCC
- Etude des MAS
- Bobinage des moteurs électrique

Education

Université Ibn Zohr EST de GUELMIM Licence Professionnelle	Sept 2018 – Juin 2019
Option: Instrumentation et systèmes	
Institu Spécialisé De Technologie Appliquée YOUSOUFIA DIPLÔME de Technicien	Sept 2017 – Juin 2019
Option: Electricité d'Entretien Industriel	
Lycée Technique ERRAZI d'El Jadida Brevet de Technicien Supérieur	Sept 2015 – Juin 2018
Option: Electromécanique et Systèmes Automatisés	
Lycée Jaber Ben Hayane YOUSOUFIA Baccalauréat Sciences et Technologies	Juin 2015
Option: Sciences et Technologies Électriques	

Compétences

Gestion :

- – Automates programmable
- – Commandes électriques et électroniques des machines électriques
- – Régulation PID

Électronique :

- – Numérique et analogique.
- – Réalisation des cartes électroniques.
- – Langages de programmation: C, C++
- – Plateformes embarquées: Arduino, Pic (16F..).

Informatiques :

- – Microsoft Office.
- – RS Logix, Step7, LabVIEW, Matlab, ISIS Proteus.
- – Connaissance générale des différents systèmes d'exploitation usuels (Windows et Linux)
- – Détecter et diagnostiquer les problèmes dans les systems d'exploitation

Centre d'interet



Langues

Arabic :

Langue maternelle

Français :

Moyen

Anglais :

Moyen



Ça y est, notre page de CV est prête ! Il ne nous reste plus qu'à la mettre en ligne !

Pour cela, il va vous falloir choisir un nom de domaine et un hébergeur avec une formule d'hébergement.

Pour le nom de domaine, on va faire simple, je vous conseille vivement de choisir votre propre nom, tout simplement. Pour l'extension, prenez un « .com » si disponible .

Ensuite, pour l'hébergement, si vous ne comptez héberger que votre CV les offres d'entrée de gamme feront parfaitement l'affaire.