

Rapport - PPE3

EuroForm

28/09/2022 - 14/12/2022



Sommaire

Contexte :	3
Objectifs:	3
Présentation de l'équipe	4
Outils pour la gestion du projet	4
Solution Techniques Proposées	5
Base de données	5
Idée de départ	5
Problèmes Rencontrés	6
Rendu Final	6
Site Internet	6
Idée de départ	6
Problèmes Rencontrés	6
Rendu Final	7
Fonctionnement Code Front-end	8
Fonctionnement Code Back-end	17
Serveur EuroForm	18
Idée de départ	18
Problèmes Rencontrés	18
Rendu Final	19
Option de serveur	21
Idée de départ	21
Problèmes Rencontrés	22
Rendue Final	22
CONCLUSION	23
Sources:	24

Contexte :

EUROFORM Groupe Gersois de Formation organise des formations. Ses activités s'organisent autour de plusieurs spécialités parmi lesquelles : la préparation aux BTS /Bac Pro, les formations en BTP ou bien l'enseignement de l'anglais. EuroForm est présent à Toulouse, Grenoble et Auch.

Le site d' Auch accueille les étudiants qui préparent un BTS ou un Bac Professionnel. Les spécialités proposées sont les suivantes :

BTS SIO et BTS MCO.

Objectifs:

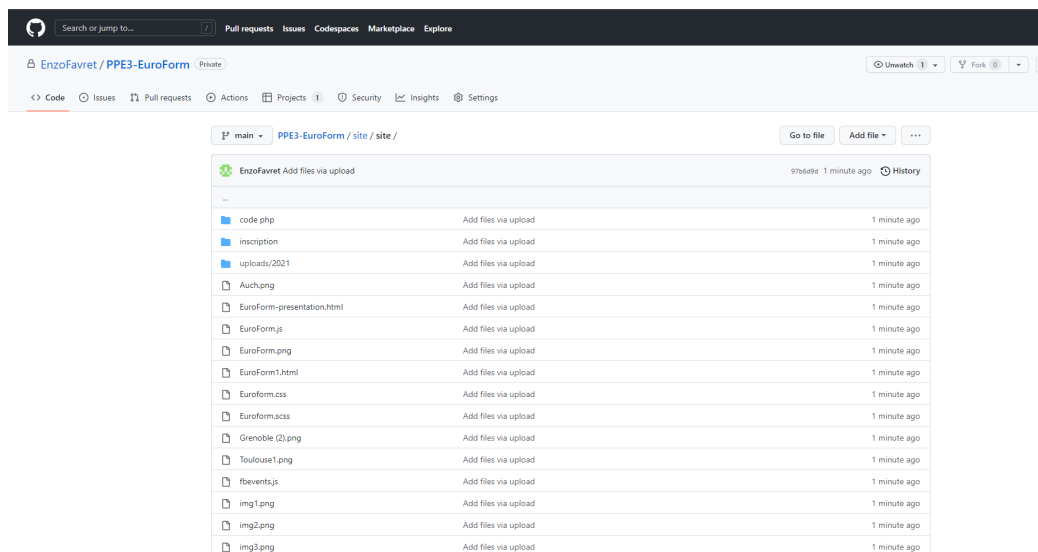
Permettre aux étudiants de pouvoir s'inscrire aux différentes formations via les différents lieux de formations. Il devra répondre aux besoins des utilisateurs (futurs élèves) et devra être un site intuitif et simple d'utilisation. Toutes les données devront être protégées, des données sensibles seront traitées.

Présentation de l'équipe

Notre équipe projet est composée de **Theo Mutel** , **Melvin LACIME** , **Noah FALLOT--VERDON** et **Enzo FAVRET** . Nous sommes étudiant en BTS SIO en deuxième année. Nos compétences sont variées et complémentaires pour la réalisation de ce projet, cela nous permet aussi d'apprendre des autres en travaillant ensemble.

Outils pour la gestion du projet

Tout au long de ce projet, nous avons utilisé le logiciel GanttProject afin de gérer le planning tout au long du projet. GanttProject est un logiciel libre de gestion de projet écrit en Java, ce qui permet de l'utiliser sur divers systèmes d'exploitation. Il permet d'éditer un diagramme de Gantt. Nous avons aussi utilisé Google Drive pour le partage de fichiers et la gestion des carnets de bord. GitHub était une solution imposée dans le cahier des charges, il n'a pas vraiment été utilisé lors de projet. Github, Inc est une entreprise de développement et services logiciels sise aux États-Unis. Github développe notamment la plateforme Github, l'éditeur de texte Atom ou encore la structure Electron.



Solution Techniques Proposées

Afin de répondre à leur demande nous avons réfléchi aux différentes solutions que l'on peut leur proposer. Nous avons donc divisé le projet en trois parties, une concernant la réalisation d'une base de données, une sur la réalisation d'une interface web pour l'inscription des élèves et un serveur pour supporter tout le système.

Base de données

Idée de départ

Nous avons eu pour idée de créer une base de données MongoDB pour répondre aux besoins de EuroForm. MongoDB est un système de gestion de base de données orienté documents, répartitionnable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données. Il est écrit en C++. Lors de notre réflexion, nous avons pensé à stocker les données sous cette forme:

```
{
  "nom": "Simone",
  "prenom": "Jean-Hubert",
  "adresse": [
    {
      "Commune": "paris",
      "rue": "charles de gaulles",
      "CP": "75 000",
      "Num_rue": "1"
    }
  ],
  "tel": "0659871465",
  "date_naissance": "02/06/1999",
  "formation": "MCO",
  "INE": "6453SDF45EERZ4",
  "formation_preced": "Bac STMG"
}
```

Problèmes Rencontrés

Lors de la mise en place de Mongodb sur notre serveur, nous avons rencontré un problème concernant la version de Debian qui n'avait pas de Mongodb compatible. La semaine suivante une version est sortie mais aucun tuto n'était disponible. Nous avons donc réussi à l'installer en se basant sur les installations classiques du SGDB. Après quelques problèmes avec la clé d'installation, nous avons réussi à l'installer correctement.

Rendu Final

Nous avons finalement bien réussi à installer et configurer Mongodb, un problème est apparu en suivant à cause de la connexion et l'envoi des données avec nodejs. Nous avons donc opté pour Mysql avec une base de données classique.

Base de données MYSQL

Site Internet

Idée de départ

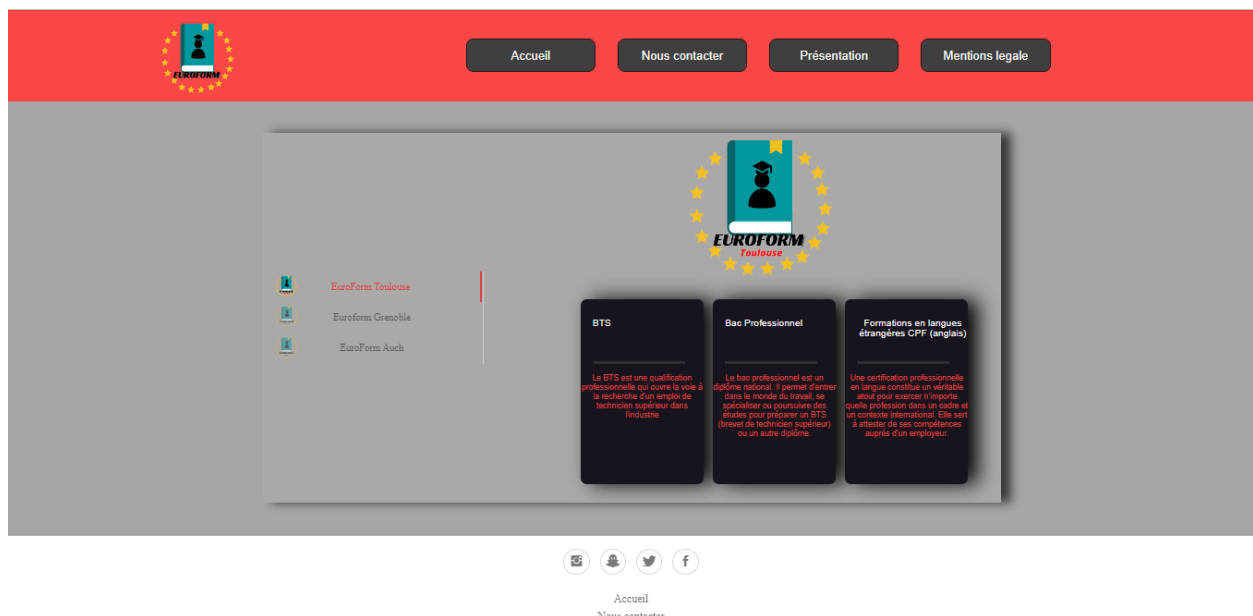
Notre Idée de départ était de faire un site plutôt facile d'usage et de compréhension tout en ayant un site plutôt esthétique, au niveau de la sélection des formation nous somme partie sur trois carte qui nous permet de sélectionner une formation est d'afficher un formulaire d'inscription relire à une base de donnée.

Problèmes Rencontrés

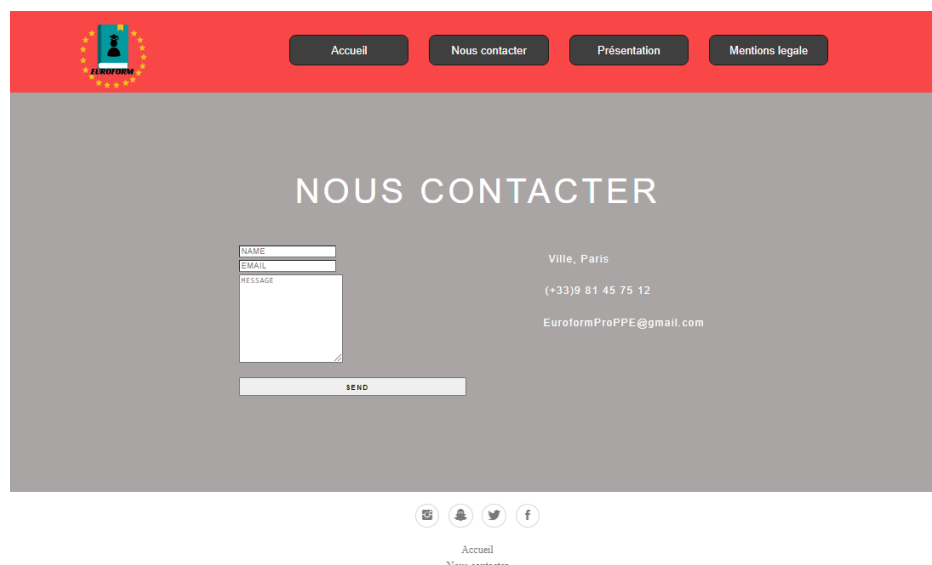
Lors du projet nous avons rencontrée différents problèmes que se soit au niveau de la gestion de l'envoi des informations du formulaire d'inscription à la base de données ou des problèmes de compatibilité entre différents ligne de code javascript

Rendu Final

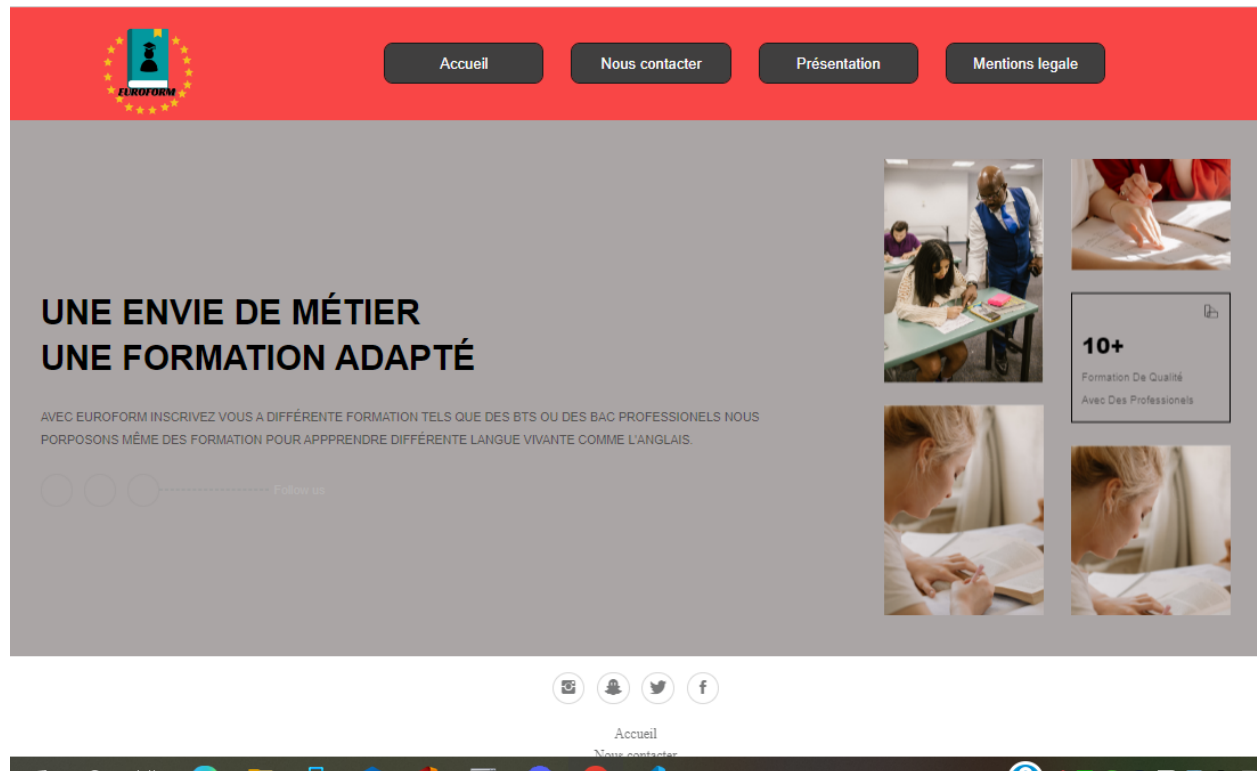
cette image est la page d'accueil du site EuroForm avec comme sité plus haut les 3 carte définissant les différente formation disponible dans les différent centre de formation de Euroform :



Sur le screen que l'on peut retrouver plus bas on peut apercevoir la page Nous Contacter qui comme son nom l'indique de contacter l'entreprise Euroform :



Sur la capture d'écran ci-dessous on peut apercevoir la page de présentation de l'entreprise qui comme son nom l'indique présente l'entreprise Euroform :

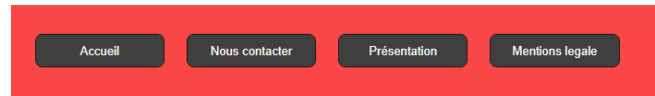


Fonctionnement Code Front-end

La partie front-end se compose de codes HTML , CSS et JavaScript ces différents langage permette de géré tout ce qui est visible par l'utilisateur lors qu'il se rends sur un site internet le but du front-end et de faire quelque chose qui est à la fois beau est simple d'utilisation pour que peut importe la personne qui se rends sur le site il puisse facilement comprendre son fonctionnement c'est ce que l'on a essayé de faire avec le site Euroform.

comme vous pouvez le voir plus bas la partie barre de navigation est vraiment facile a repéré ce sont juste quatre bouton

Le code de la barre de navigation :



```
<div class="banqueSIO--bouton1--v10" data-tilt data-tilt-max="10"
data-tilt-speed="1000" data-tilt-perspective="800" data-tilt-scale="1.1">

  <a href="EuroForm1.html">

    <p class="banqueSIO--bouton1--v10--p">Accueil</p>

  </a>

</div>
```

comme vous pouvez le voir plus bas la partie barre de navigation est vraiment facile a repéré ce sont juste quatre bouton

cette ligne de code nous montre comment fonctionne l'un des quatre bouton de la barre de navigation il sont tous crée de la même façon

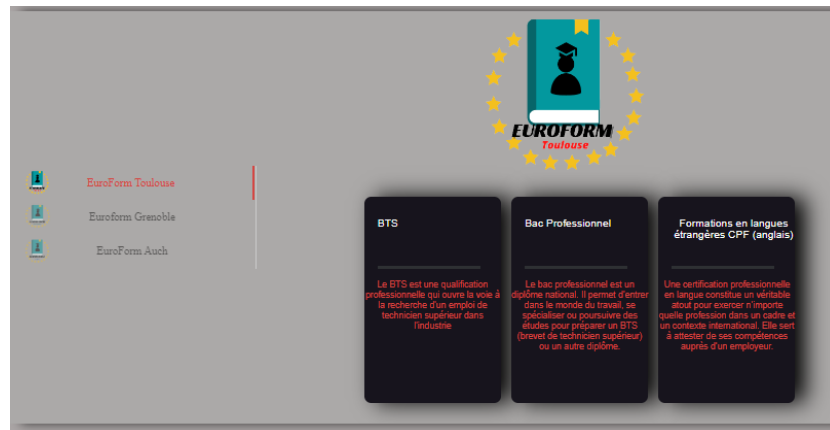
```
.banqueSIO--bouton1--v10 {
  width: 24.9vh;
  height: 4.9vh;
  border: 0.3vh solid #191919;
  background-color: #403f3f;
  border-radius: 12px;
  display: flex;
  align-items: center;
  align-content: center;
  justify-content: center;
  transition: all 0.2s ease;
  margin: 18px;
}
.banqueSIO--bouton1--v10--p{
  font-family: Poppins, sans-serif;
  font-size: 2vh;
```

```
color: white;
transition: all 0.2s ease;
}
.banqueSIO--bouton1--v10:hover{
background-color: #46f5e1;
box-shadow: 0px 0px 30px rgba(0,0,0,0.6);
transition: all 0.2s ease;
cursor: pointer;
}
.banqueSIO--bouton1--v10:hover .banqueSIO--bouton1--v10--p{
color: #191919;
transition: all 0.2s ease;
}
.banqueSIO--bouton1--v10:active{
transform: scale(0.9);
transition: all 0.2s ease;
}
```

ce code CSS définit la position du bouton, sa taille et nous permet de gérer l'animation de changement de couleur quand on passe dessus.

au niveau du bandeau du header on a juste défini un div qui permet de réunir tout dans un seul est même bloc

nous allons à présent expliquer le fonctionnement du carrousel qui nous permet de sélectionner les différents centres de formation et la formation que l'on souhaite



```
<div id="flex-container" class="testimonials">
  <div id="left">
    <ul class="list">
      <li class="item">
        <input type="radio" id="radio_testimonial-1" name="basic_carousel"
checked="checked" class="lieux" onclick="clicklieux(this);"/>
        <label class="label_testimonial-1" id="lab1" for="radio_testimonial-1">EuroForm
Toulouse</label>
        <div class="content-test content_testimonial-1">
          <span class="picto"></span>
          <p class="testimonialFrom"><div class="container">
            <button id="toggl">
              <div class="card" id="card1">
                <h3 class="title">BTS</h3>
                <div class="bar">
                  <div class="emptybar"></div>
                  <div class="filledbar"></div>
                </div>
                <div class="circle">
                  <center> <FONT size="2pt"> Le BTS est une qualification
professionnelle qui ouvre la voie à la recherche d'un emploi de technicien supérieur dans
l'industrie </FONT></center>
                </div>
              </div>
            </button>
          </p>
        </div>
      </li>
    </ul>
  </div>
</div>
```

le carrousel fonction avec du HTML, CSS et Javascript pour pouvoir permettre au carrousel de fonctionner et d'être fluide on a utilisé un input de type radio qui fait en sorte que l'on ne peut selection que un seule des élément que l'on mets sur la liste pour notre carrousel ce sont les centre de formation en cliquant dessus on vas afficher tout ce que l'on a enregistrer dans cette input qui pour ce carrousel était les trois carte que l'on voit qui nous permet de sélectionner une formation précise mais nous reviendrons sur comment fonctionne les carte quand on vas parlée du CSS ,

```
#flex-container {
  display: -webkit-box;
  display: -ms-flexbox;
  display: flex;
  -ms-flex-wrap: wrap;
  flex-wrap: wrap;
  -webkit-box-pack: center;
  -ms-flex-pack: center;
  justify-content: center;
  -webkit-box-align: center;
  -ms-flex-align: center;
  align-items: center;
  width: 1200px;
  min-height: 400px;
  height: 110vh;
  margin: auto;
  background-color: #fff;
  -webkit-box-shadow: 0 1px 3px rgba(0, 0, 0, 0.12),
    0 1px 2px rgba(0, 0, 0, 0.24);
  box-shadow: 0 1px 3px rgba(0, 0, 0, 0.12), 0 1px 2px rgba(0, 0, 0, 0.24);
  overflow: hidden;
  -webkit-box-orient: vertical;
  -webkit-box-direction: normal;
  -ms-flex-direction: column;
  flex-direction: column;
  position: absolute;
}

#left{
  height: 50%;
  -webkit-box-flex: 0;
  -ms-flex: 0 0 auto;
  flex: 0 0 auto;
  display: -webkit-box;
  display: -ms-flexbox;
  display: flex;
  width: 100%;
}

.item input {
```

```
    display: none;
  }

  label {
    display: block;
    opacity: 0.5;
    height: 50px;
    text-align: center;
    line-height: 50px;
    position: relative;
  }

  label:hover {
    opacity: 0.75;
    cursor: pointer;
  }

  .content-test {
    position: absolute;
    right: 0;
    bottom: 0;
    opacity: 0;
    -webkit-transform: translateY(100%);
    -ms-transform: translateY(100%);
    transform: translateY(100%);
    height: 50%;
    width: 100%;
    -webkit-transition: 0.5s ease-out;
    -o-transition: 0.5s ease-out;
    transition: 0.5s ease-out;
    display: -webkit-box;
    display: -ms-flexbox;
    display: flex;
    -webkit-box-align: center;
    -ms-flex-align: center;
    align-items: center;
    -webkit-box-pack: center;
    -ms-flex-pack: center;
    justify-content: center;
    -webkit-box-orient: vertical;
    -webkit-box-direction: normal;
    -ms-flex-direction: column;
    flex-direction: column;
    pointer-events: none;
  }

  #left .list {
```

```

        border-right: 2px
    .content-test h1:first-letter {
        text-transform: uppercase;
    }

input:checked ~ label.label_testimonial-1 {
    c
    border-right: solid 4px #d64541;
}

label.label_testimonial-1:before {
    content: " ";
    display: block;
    position: absolute;
    width: 55px;
    height: 50px;
    margin-left: 14px;
    background-image: url(/uploads/Toulousel.png);
    background-position: center;
    background-size: 75% 75%;
    background-repeat: no-repeat;
    border-radius: 50%;
}

```

nous avons ici une partie du CSS qui gère le carrousel cette partie gère une partie des animation du carrousel qui se passe au niveau des input type radio est de ce qui contient il nous permet de définir la taille et l'emplacement des input de la couleur que l'on veut leur définir et le plus important l'emplacement ou on veut afficher le contenu qui est enregistré dans l'input sélectionné bien sûr une autre partie est gérée en Javascript

```

var testimonialItems = document.querySelectorAll(".item label");
var timer;
function cycleTestimonials(index) {
    timer = setTimeout(function() {
        var evt;
        if (document.createEvent){
            evt = document.createEvent('MouseEvent');
            evt.initMouseEvent('click', true, true, window, 0, 0, 0, 0, 0, false, false, false,
false, 0, null);
        } else {

```

```
    evt = new MouseEvent("click", {
        view: window,
        bubbles: true,
        cancelable: true,
        clientX: 20
    });
}
index++;
if (index >= testimonialItems.length) {
    index = 0;
}
cycleTestimonials(index);
document.querySelector(".testimonials").addEventListener("click", function() {
    clearTimeout(timer);
});
}, 2000);
}

cycleTestimonials(0);
```

le Javascript liée au Carrousel permet de faire défiler les différentes radio avec une animation très fluide et lente quand on clique sur un .testimonial, celui qui était sélectionné sera retiré en se glissant vers le bas alors que le celui que l'on vient de sélectionner s'affiche vers le haut.

nous pouvons à présent revenir sur les cartes qu'on vous parlais plus tôt comme expliquée plus haut c'est la carte nous permet de sélectionner une des formations du centre de formation mais avec du CSS on a pu faire en sorte qu'elle s'anime un peu pour rendre ça plus jolie et fluide le code CSS en dessous nous définit la position des cartes leur taille et surtout les animations telles que les transitions ou autre tout cela est géré dans le CSS

```
.card {
    display: flex;
    height: 300px;
    width: 200px;
    background-color: #17141d;
    border-radius: 10px;
```

```

    box-shadow: 1rem 0 2rem #000;
    /* margin-left: -50px; */
    transition: 0.4s ease-out;
    position: relative;
    left: 0px;
    display: none;
}

.card:not(:first-child) {
    margin-left: 10px;
}

.card:hover {
    transform: translateY(-20px);
    transition: 0.2s ease-out;
}

.card1:hover ~ .card {
    position: relative;
    left: 50px;
    transition: 0.4s ease-out;
}

```

quand on clique sur l'une des cartes un formulaire apparaît nous demandant d'entrer nos informations personnelles et de sélectionner notre formation pour éviter toute erreur.

```

let togg1 = document.getElementById("togg1");
let togg2 = document.getElementById("togg2");
let togg3 = document.getElementById("togg3");
let togg4 = document.getElementById("togg4");
let togg5 = document.getElementById("togg5");
let togg6 = document.getElementById("togg6");
let togg7 = document.getElementById("togg7");
let togg8 = document.getElementById("togg8");
let togg9 = document.getElementById("togg9");
let left = document.getElementById("left");
let retour = document.getElementById("retour");
let formulaire = document.getElementById("formulaire");
let lab1 = document.getElementById("lab1");
let lab2 = document.getElementById("lab2");
let lab3 = document.getElementById("lab3");
let card1 = document.getElementById("card1");
let card2 = document.getElementById("card2");
let card3 = document.getElementById("card3");
let card4 = document.getElementById("card4");
let card5 = document.getElementById("card5");
let card6 = document.getElementById("card6");
let card7 = document.getElementById("card7");
let card8 = document.getElementById("card8");
let card9 = document.getElementById("card9");

togg1.addEventListener("click", () => {
    if(getComputedStyle(left).display != "none"){
        left.style.display = "none";
        formulaire.style.display = "block";
    } else {
        left.style.display = "block";
    }
})

togg2.addEventListener("click", () => {
    if(getComputedStyle(left).display != "none"){
        left.style.display = "none";
        formulaire.style.display = "block";
    } else {
        left.style.display = "block";
        formulaire.style.display = "none";
    }
})

```

Le but de ce code Javascript est d'afficher le Formulaire d'inscription lorsque l'on clique sur l'une des trois cartes de formation.

Fonctionnement Code Back-end

Pour le Back-end, il se compose de code php. Celui-ci permet de réaliser l'enregistrement des étudiants sur la plateforme d'EuroForm et d'afficher leurs données après inscription.

A cause d'un problème de temps, il ne se compose que de deux fichiers.

Le premier fichier consiste à la connexion à la base de données mysql:

```
<?php

// Informations d'identification

define('DB_SERVER', 'localhost');

define('DB_USERNAME', 'root');

define('DB_PASSWORD', '');

define('DB_NAME', /*Nom de la base de données*/);

$connect = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD, DB_NAME);

if($connect === false){

    die("ERREUR : Impossible de se connecter. " . mysqli_connect_error());

}

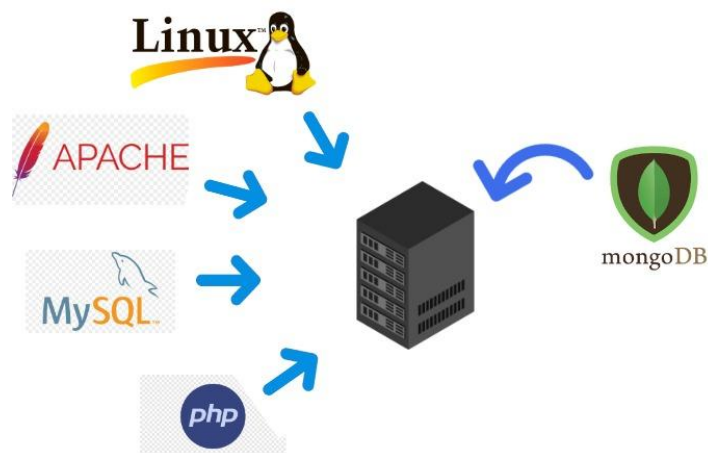
?>
```

Le second fichier est la page d'inscription du site, le code est directement intégré dans le code html pour faciliter le lien des fonctions. Le [Fichier](#) sert au fonctionnement du formulaire d'inscription du site.

Serveur EuroForm

Idée de départ

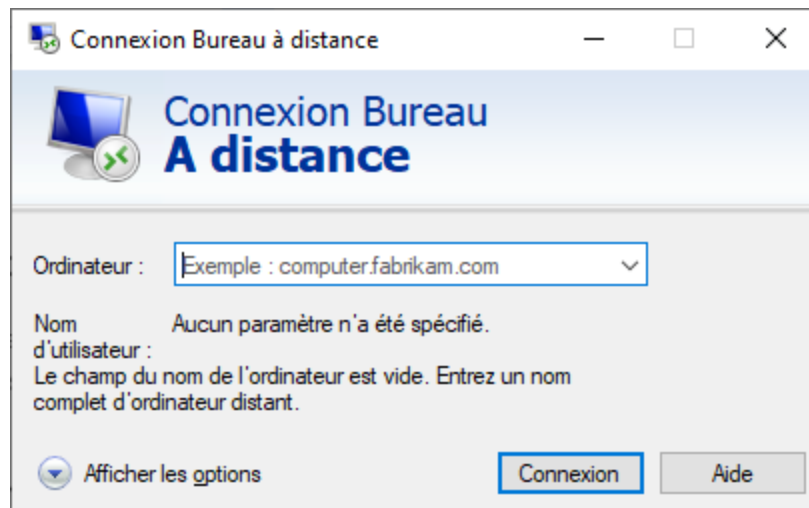
Pour pouvoir accéder au site que nous avons créé, nous avons décidé de réaliser un espace virtuel où nous pourrions stocker et héberger le site internet de EuroForm via un serveur LAMP



Cette idée à évolué avec le temps et un serveur samba à été installé en plus sur la machine afin de permettre une certaine autonomie dans le groupe.

Problèmes Rencontrés

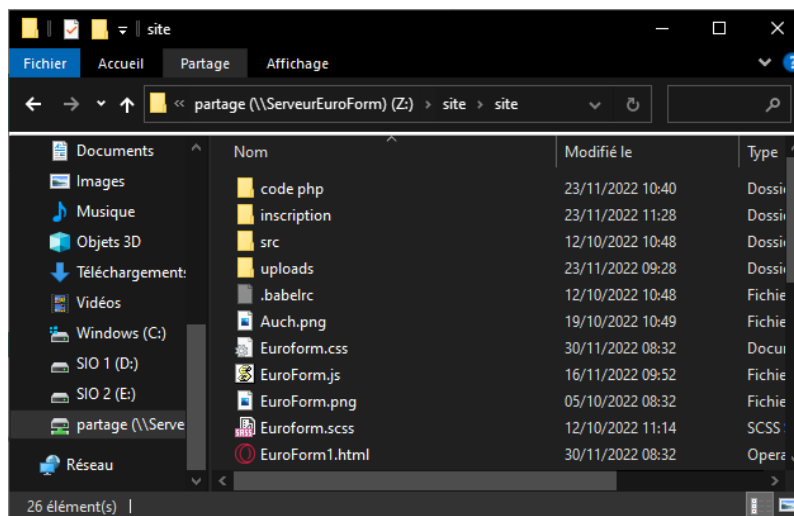
Au départ, nous avons remarqué que Windows mettait à disposition une interface de “Connexion Bureau à distance “. Mais nous nous sommes vite aperçu que cet outil n’était pas adapté à nos besoins en raison de nombreux crashes. De plus, les versions de MongoDB présentes dans les “repository “ linux étaient obsolètes et son installation demandait l'ajout de liens et de clefs de repository ce qui à pris pas mal de temps en raison des liens périmés.



Nous avons donc opté pour une autre solution qui était Samba.

Rendu Final

Nous avons fini par réaliser un espace partagé avec notre site Euroform qui est accessible par n'importe quel ordinateur possédant les accès au serveur. Cet espace permet également un accès au site via un navigateur avec l'adresse du site.



Le serveur apache est en état de marche et le site est disponible à cette adresse:

<http://192.168.1.116/html/site/site/EuroForm1.html>

Toute la procédure d'installation de ce serveur est disponible ici :

https://docs.google.com/document/d/12oXhGeWXqFZwCtYMdZqGBvVCwPBP4WUxe5pWIm_-pUI/edit?usp=sharing

Option de serveur



Idée de départ

Pour pouvoir accéder à notre base de données, précédemment choisi pour être de type document (MongoDB). Le plus adapté était de trouver une alternative à PHP qui est actuellement obsolète pour beaucoup de projets et notamment pour la communication de données vers des bases de données documents.

Pour palier à ce problème la solution était de configurer un serveur node.js, ce serveur fonctionne sous le langage JavaScript. Il est très complet et permet de pallier beaucoup de problèmes. Notamment pour les applications Web, cependant il est plutôt compliqué à utiliser.

Problèmes Rencontrés

Pour lancer un serveur sur node.js ce n'est pas très compliqué, deux lignes suffisent à lancer un serveur en local. Cependant cela n'affecte qu'une seule page unique, pour ajouter de la complexité il est nécessaire d'utiliser un système de "routes" sur une machine. Ces routes représentent les fichiers et différents dossiers dans lesquels sous tel ou tel condition l'utilisateur doit être redirigé.

L'un des problèmes est que pour simplifier l'utilisation de ces routes il y existe des frameworks. Notamment "Express.js" ce framework est beaucoup utilisé par de nombreux développeurs. Mais la documentation n'est pas toujours très claire et même en ayant des instructions simples il est facile de se tromper. Seulement à ce moment-là du développement, il était impossible de le déterminer. Beaucoup de temps pour la compréhension de node.js natif ainsi que pour l'utilisation de express.js ont été perdus.

Vient s'ajouter à cela la gestion des données envoyées du côté client, la gestion du framework JQuery est plutôt simple de ce côté là. Mais les choses se compliquent lorsqu'il en vient à les utiliser du côté du serveur. Un autre framework vient s'ajouter à ceux-ci, celui de MongoDB. Pour l'utilisation des données il est alors nécessaire de les transmettre du serveur web au serveur base de données.

Pour transmettre ses données il y a une procédure plutôt facile à comprendre sur le principe mais un peu chronovore à mettre en place. Il s'agit de créer des modèles utilisables et exploitables sur le serveur web. On les utilise lors de l'envoi de données vers le serveur, mais l'utilisation de ces modèles requiert une compréhension du langage JavaScript plutôt avancé.

Pour faire simple le travail était un peu trop ambitieux pour le temps données ainsi que l'effectif attribué.

Alors pour pallier ce problème il a été choisi de tout changer, la base de données est-elle passer en relationnel (MySQL). Les données envoyées par les utilisateurs sont alors redirigées par PHP.

Rendue Final

Comme mentionné au-dessus, l'option la plus simple à était choisie. La plupart des personnes étaient familières avec l'utilisation de PHP ainsi que la gestion de bases de données relationnelles. Il à alors été possible de lier le site avec la base de données en très peu de temps. Finalement l'envoi des données est effectif.

CONCLUSION

Dans l'ensemble le projet est une réussite, l'application n'est pas totalement opérationnelle mais les fonctions de bases pour son utilisation sont présentes.

La gestion de ce projet via le partage de tâche avec gantz et l'utilisation de la plateforme Github était un élément nouveau plutôt intéressant à mettre en place. Surtout pour l'utilisation de Github par VScode.

En addition de cela beaucoup d'imprévue se sont ajoutés sur au projet, comme l'absence répété du chef de projet ou de certains de ses membres. Ou bien encore par la complexité des éléments choisis initialement.

Nous avons pu apprendre de nouvelles méthodes de travail et de nouvelles compétences via l'utilisation de nouveaux outils et de nouvelles technologies.

Nous pouvons donc affirmer que ce projet est aller plus loin que ce qu'il était initialement prévu, même si nous ne sommes pas satisfait pleinement du produit final.

Sources:

www.wikipedia.fr

www.stackoverflow.com

www.sql.sh