



Rapport du projet tuteuré

SITE WEB AVEC JAVASCRIPT POUR LE COURS DE "RAISONNEMENT ET SCIENCE DE LA DÉCISION"

Yanrui GUO, Fei SHEN, Xiyue LIAO, Jiayu CHANG, Yiyang WANG

**Tuteurs :
Umberto Grandi**

Résumé :

Ce travail, qui entre dans le cadre du projet de fin de première année de master, consiste à concevoir et à développer un site web en utilisant le javascript pour améliorer son interactivité.

Ce document définit le projet et son contexte, décrit les étapes nécessaires à sa réalisation et explique le résultat.

Mots-clés

HTML, CSS, Javascript, JQuery, Carrousel, Glisser - déposer, Répertoire latéral, Canvas

Remerciements

Nous tenons particulièrement à remercier dans un premier temps, notre tuteur de projet Umberto Grandi Professeur assistant à l'IRIT, Université de Toulouse pour sa disponibilité et ses précieux conseils tant sur la partie conception que sur la partie rédaction du rapport qui nous ont permis de mener à bien ce projet.

Nous remercions également toute l'équipe pédagogique de Master 1 MIAGE parcours Ingénierie Métier, notamment pour l'ensemble des connaissances acquises qui nous ont permis de réaliser ce projet.

Table des matières

Résumé :	- 1 -
Mots-clés	- 1 -
Remerciements	- 2 -
I- Introduction	- 4 -
II- Conduite de projet	- 5 -
III- Conception du site web	- 6 -
1. Exigences du tuteur :	- 6 -
2. Environnement de développement	- 6 -
3. Contenu du site :	- 7 -
IV- Résultat réalisé	- 9 -
1. Carrousel des bannières	- 9 -
2. Répertoire latéral	- 11 -
3. Chapitre1-Recherche WEB :	- 13 -
4. Chapitre2-Réseaux sociaux :	- 15 -
5. Chapitre3-Logique et WEB sémantique I:	- 17 -
6. Chapitre4-Logique et WEB sémantique II :	- 19 -
7. Chapitre5-Intelligence collective apprentissage :	- 20 -
8. Chapitre6-Décisions collectives :	- 22 -
9. Chapitre7-Mécanismes d'enchères:	- 24 -
10. Chapitre8-Allocation des ressources:	- 26 -
11. Fonction supplémentaire	- 28 -
V- Problèmes rencontrés	- 29 -
VI- Conclusion	- 30 -
VII- Référence	- 31 -

I- Introduction

Le projet tuteuré à la fin de première année de master permet nous de déconforter les connaissances théoriques à l'exercice pratique et s'intégrer dans la vie professionnelle au futur. Nous avons choisi le projet du développement web frontal qui est le production HTML, CSS et JavaScript d'une page web qu'un utilisateur peut voir et avec lesquelles il peut interagir directement.

Le développement est basé sur le site web du cours de Raisonnement et Science de la Décision :

(<https://www.irit.fr/~Umberto.Grandi/teaching/raisonnement/>)

L'objectif de ce projet est de concevoir et d'implanter un page internet, avec animations Javascript et images qui expliquent les algorithmes présentés, pour améliorer l'interactivité du site web du cours. Dans ce projet, nous avons utilisé les compétences acquises au cours de notre Master 1 en ce qui concerne les langages HTML, CSS et Javascript, d'ailleurs, nous apprenons et utilisons aussi certaines de ses compétences de niveau supérieur. Les compétences en programmation sont nécessaires. De même, nous devez avoir compris le cours de Raisonnement et Science de la Décision.

En termes simples, le projet s'est déroulé en deux étapes : Premièrement, il s'agit améliorer le style du site à l'aide des CSS et Javascript. Deuxièmement, pour chacune de 8 sessions de cours, sélectionner l'algorithme ou le concept clé de la session, concevoir une animation qui permet de l'expliquer d'une façon visuelle et interactive, et la coder avec javascript.

II- Conduite de projet

Ce projet a commencé à 16/05/2018, qui a duré environ 7 semaines. Pendant cette période, le tuteur a organisé 4 réunions avec nous pour confirmer le degré d'achèvement du travail par étapes, discuter des arrangements de travail spécifiques pour chaque phase, faire des suggestions et des demandes pour notre conception et travail et répondre à nos questions. Notre plan et notre calendrier sont les suivants:

- **Semaine 1 (16/05 – 23/05)**

Réunion :16/05. Le tuteur met en avant les exigences de la page Web et organise les tâches pour chaque phase du projet.

Tâche: Conception de la maquette CSS, Identification des algorithmes pour les figures et les animations Javascript, Identification des outils Javascript à utiliser.

- **Semaine 2&3 (23/05 – 05/06)**

Réunion :23/05. Le tuteur a adopté notre conception de la page Web. Il a validé le choix d'algorithme pour chaque cours et il a donné des suggestions et des directions de l'amélioration pour l'animation Javascript de chaque cours.

Tâche: Implantation du CSS et html (sans figures et animations) pour validation, conception des animations et figures sur papier, premier test d'une figure et une animation avec l'outil choisi

- **Semaine 4&5 (05/06 – 20/06)**

Réunion :05/06. Le tuteur a validé la conception globale de notre site Web (CSS) et a montré la partie JavaScript actuellement écrite.

Tâche: Implantation des figures et des animations. Première validation du site final.

- **Semaine 6&7 (20/06 - 30/06)**

Réunion :20/06. Nous avons testé le site dans différents navigateurs et corrigé les erreurs et assuré l'acceptabilité.

Tâche: Finalisation du site, mise en ligne, et écriture du rapport

III- Conception du site web

1. Exigences du tuteur :

Le site internet doit être composé d'une seule page. Le contenu (texte, sections, etc.) doit être le même que la version existante. On doit pouvoir télécharger les transparences, et les liens doivent bien fonctionner. L'image initiale peut être changée. Le style du site doit être sobre et sérieux, il s'agit d'un site qui s'adresse aux collègues professeurs qui veulent enseigner la même matière. Chaque cours contient plusieurs algorithmes, et le tuteur voudrais qu'un soit sélectionné comme le plus adapté à être mis en image ou en animation. N'importe quel outil javascript peut être utilisé.

2. Environnement de développement

Afin de mettre en route notre site, on va s'intéresser tout d'abord à la définition de l'environnement du travail.

- **Système d'exploitation:** Windows 10
- **Langue d'écriture:**

a) HTML

Le HTML (« HyperText Mark-Up Language ») est un langage dit de « balisage » ou de « structuration » permettant la conception de pages web avec des balises de formatage. Les balises permettent d'indiquer la façon dont doivent être présentés le document et les liens qu'il établit avec d'autres documents. C'est le langage basic de création un site web.

b) CSS

Le CSS (« Cascading Style Sheets » : feuilles de style en cascade) est un langage informatique complétant le HTML. Alors que le HTML structure la page Web, le CSS va la mettre en forme en y apportant du style. CSS nous aide de faire le site web plus joli.

c) JAVASCRIPT

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives. Ça nous permet de créer les animations et dynamiser le page web.

d) JQuery

jQuery est une bibliothèque JavaScript libre et multiplateforme créée pour faciliter l'écriture de scripts côté client dans le code HTML des pages web. Nous l'utilisons pour favoriser l'interactivité du site web.

- **Outils de développement:**

1. Notepad ++

Notepad++ est un éditeur de texte libre générique, fonctionnant sous Windows, qui intègre la coloration syntaxique de code source pour les langages et fichiers de programmation ainsi que pour tout autre langage informatique, car ce logiciel propose la possibilité de créer ses propres colorations syntaxiques pour un langage quelconque.

2. JSFiddle (<https://jsfiddle.net/>)

JSFiddle est une communauté en ligne pour tester et présenter des extraits de code HTML, CSS et JavaScript créés par les utilisateurs et collaboratifs. Cela permet des appels AJAX simulés. Le site est hébergé par DigitalOcean.

3. Contenu du site :

Selon les exigences du tuteur, nous avons conçu le contenu de la page Web. Le site comporte les éléments suivants :

- ✧ La tête du page: c'est la première partie que les utilisateurs peuvent voir. Il comporte le titre du cours et l'affichage de défilement des images concernées.
- ✧ Une présentation du "Description du cours" : cette partie donne la petite présentation du cours <Raisonnement et Science de la Décision>.
- ✧ Support du cours: Dans cette partie, nous présentons brièvement chaque cours en utilisant l'animation. Et chaque cours est présenté par un algorithme sélectionné de ce cours. Ils sont les suivants:
 1. Recherche WEB : l'algorithmes de recherche fondés sur le réseau des liens (PageRank 1.0)
 2. Réseaux sociaux : algorithme de Dijkstra
 3. Logique et WEB sémantique I: l'langage de la logique propositionnelle
 4. Logique et WEB sémantique II : Linked Data, DBPédia
 5. Intelligence collective apprentissage :collaborative filtering (item-based)

6. Décisions collectives : règles de vote (Borda)
 7. Mécanismes d'enchères: enchères double
 8. Allocation des ressources: programmation par contraintes (problème de sac à dos)
- ✧ Bibliographie: il contient les références du cours. Nous utilisons le même contenu que la page web originale.

IV- Résultat réalisé

1. Carrousel des bannières

Cinq images sont mise en place afin de représenter les différents sujets des cours, alternant de l'un à l'autre automatiquement toutes les deux secondes.

Dans les codes html, les bannières sont déposées dans une ligne. La fonction va capturer les image sous la classe « top_img » et calculer le longueur d'ensemble des images. En fait, l'animation effectue sur l'ensemble de la classe « .top_img », mais pas sur chaque image. L'idée est que tous les contenus hors le champs de cette classe n'affiche pas, et l'ensemble de la classe déplace à gauche la largeur d'un image chaque fois.

```
1. //Banner
2. $(document).ready(function() {
3.     $(function(){
4.         //restituer la largeur de chaque photo et le nombre des photos
5.         var distance=$( ".top .top_img img" ).width();
6.         var list=$( '.top_img img' ).length;
7.         //définir la largeur de la section où se trouvent les photots
8.         $( '.top_img' ).css({
9.             width:list*distance,
10.        });
11.        //ouvrir un compteur
12.        //Dérouler automatiquement toutes les 2 secondes
13.        var timer='';
14.        var num=0;
15.        timer=setInterval(anime,2000);
16.        //faire déplacer les photos à gauche
17.        function anime(){
18.            num++;
19.            if(num>parseFloat(list)-1){
20.                num=0;
21.                $( '.top_img' ).animate({left:num*-distance},"slow");
22.            }else{
23.                $( '.top_img' ).animate({left:num*-distance},"slow");
24.            }
25.        }
26.        //arrêter l'animation quant les photos sont invisibles
27.        document.addEventListener("visibilitychange", function(){
28.            if (document.hidden || document.webkitHidden) {
29.                clearInterval(timer);
30.            }else {
```

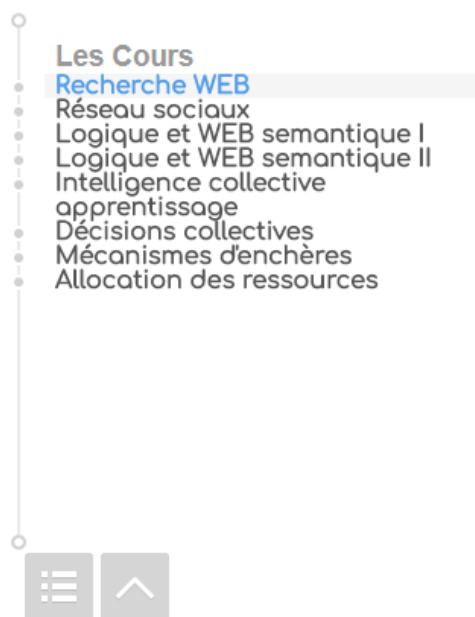
```
31.         timer = setInterval(anime,2000);
32.     }
33.     }, false);
34. });
35. });
```

2. Répertoire latéral

Pour que l'utilisateur peut consulter les différents sujets et revenir au début de la page pendant lisant les cours, on a construit un répertoire latéral en bas et à droite de la page. L'utilisateur peut aussi le rendre invisible.

La fonction cherche tous les titres (h1~h6) dans le html qui appartient au DOM '#touslescours', et après traverse tous les header. Les titres sont déposés dans un DOM dont l'identifiant est catalog_item.

On utilise la fonction scrollspy de bootstrap afin de mettre en évident le titre concernant le contenu actuel.



```
1. headers = s.find(':header'); //chercher tous les titres h1~h6 dans #touslescours
2.   catalog_item += '<li><span style="font-size: 14pt; font-weight: bold;">Les Cours</span></li>';
3.   headers.each(function () { //traverser tous les header
4.     var xheader = $(this), //l'objet de header présent
5.     v = xheader[0];
6.
7.     var text = xheader.text();
8.
9.     xheader.attr('id', 'autoid-' + l + '-' + m + '-' + n)
10.
11.     if (v.localName === 'h2') {
12.       level1 = l + '.';
13.     }
```

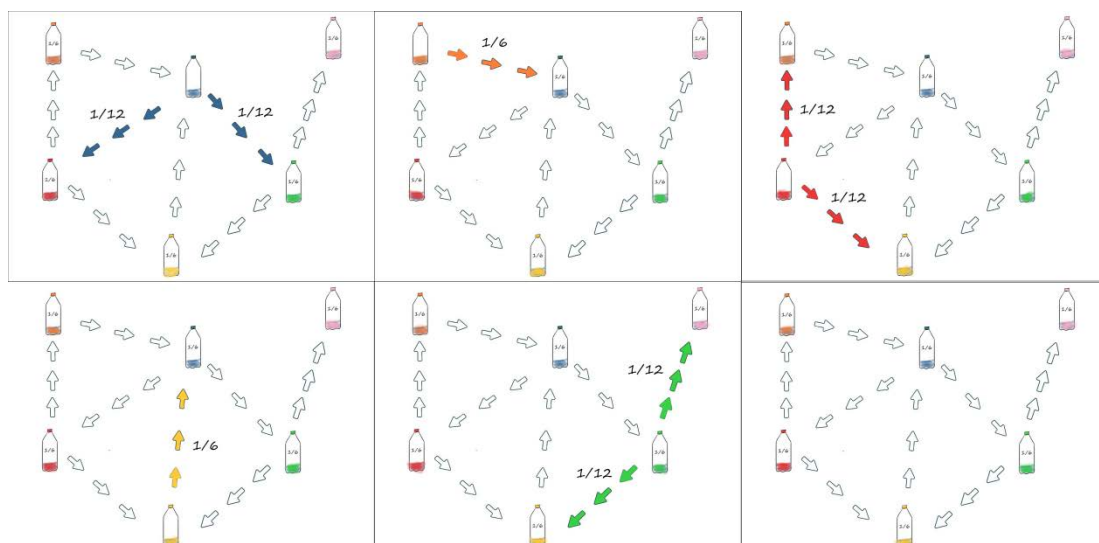
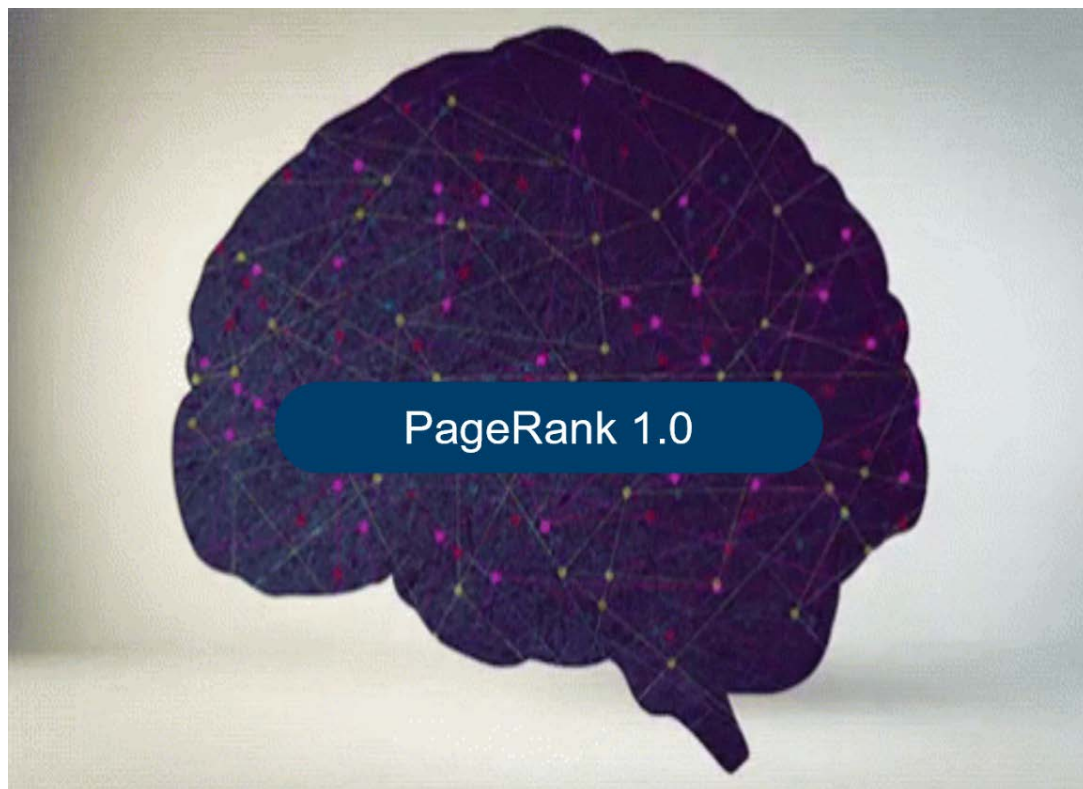
```

14.         catalog_item += '<li><a href="#" + xheader.attr('id') + '" title
    ="' + text + '">' + text + '</a><span class="sideCatalog-
    dot pointer"></span></li>';
15.         l++;
16.     }
17. });
18. $('#sideCatalog-catalog>ul.nav').html(catalog_item);
19. $('[data-spy="scroll"]').each(function () {
20.     var $spy = $(this).scrollspy('refresh');
21. });
22. $('body').scrollspy({
23.     offset: 50,
24.     target: '.sideCatalogBg'
25. });
26.
27. $('body').on('activate.bs.scrollspy', function () {
28.     $('ul.nav li.active span').toggleClass("highlight");
29. });
30.
31. $sideCatalog = $('#sideCatalog');
32. $('#sideCatalogBtn').on('click', function () {
33.     if ($(this).hasClass('sideCatalogBtnDisable')) {
34.         $sideCatalog.css('visibility', 'hidden');
35.         $("#sideToolbar").css("z-index",0);
36.     } else {
37.         $sideCatalog.css('visibility', 'visible');
38.         $("#sideToolbar").css("z-index",999);
39.     }
40.     $(this).toggleClass('sideCatalogBtnDisable');
41. });
42.
43.
44. $('#sideToolbar-up').on('click', function () {
45.     $("html,body").animate({
46.         scrollTop: 0
47.     }, 500)
48. });

```

3. Chapitre1-Recherche WEB :

Dans ce chapitre, PageRank 1.0 est choisi comme l'algorithme présenté. L'idée principale est de distribuer de l'eau selon les flux orientées entre les bouteilles de sorte que la représentation devienne plus ludique et attractive. Pour réaliser cette idée, nous avons dessiné des photos qui contiennent chaque étape de PageRank 1.0. Et puis, en utilisant un compteur dans JQuery qui contrôle l'affichage de chaque photo, nous pouvons présenter les photos l'un par l'autre toutes les 1.5s. De plus, un bouton est cliquable pour faire commencer cette animation.



Fonction du chapitre 1

```
1. //animation dans le cours 1 et contrôler le bouton
2. $(document).ready(function(){
3.     function displayrecherche() {
4.         var li=document.getElementById('cours-
           recherche').getElementsByTagName("li");
5.         var NowFrame = 0; //définir le numéro de la photo qctuelle
6.         var MaxFrame = 9; //définir le nombre total des photos
7.         var next = 1;
8.         var timer='';
9.         //dérouler les photos chaque 1.5s
10.        timer=setInterval(show,1500);
11.        function show() {
12.            li[NowFrame].style.display = "none";
13.            li[next].style.display = "block";
14.            NowFrame = next;
15.            if (next == MaxFrame){
16.                //terminer le compteur et afficher le bouton
17.                clearInterval(timer);
18.                $("#btnrecherche").fadeIn(500);
19.            }
20.            else{
21.                next++;
22.            }
23.        }
24.    }
25.    //faire disparaître le bouton
26.    $("#btnrecherche").click(function(){
27.        $("#btnrecherche").fadeOut(100,displayrecherche);
28.    });
29. });
```

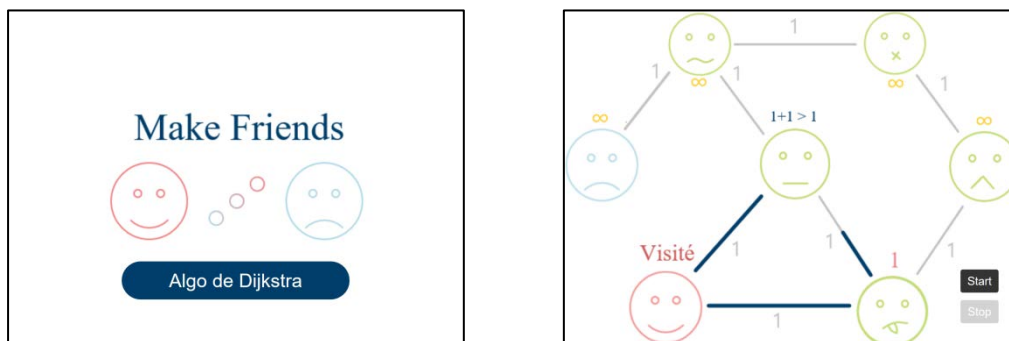
4. Chapitre2-Réseaux sociaux :

L'algorithme de dijkstra est représenté dans ce chapitre. Après avoir cliqué le bouton, vous pouvez voir continuellement des apparitions des lignes et des textes. Ces indications expliquent comment trouver les distances entre des noeuds (ici ce sont des personnages). Pour rendre plus visible, deux boutons sont disponibles à vous. Vous pouvez suspendre ou recommencer l'animation n'importe en quel moment.

La réalisation de cette animation est basé sur la balise 'canvas' comme une toile et javascript comme un pinceau. Il faut d'abord obtenir l'environnement de la balise 'canvas' pour commencer à dessiner.

Ensuite, nous pouvons dessiner les éléments comme une ligne ou un cercle etc. Par exemple, la fonction 'arc(x, y, rayon, angleInitial, angleFinal, anglehoraire)' est utilisé pour dessiner un cercle, dont (x,y) est le centre du cercle. Le principe est de positionner chaque action sur la toile.

Par ailleurs, si nous voulons agrandir ce cercle peu à peu, nous pouvons changer la constante 'rayon' x avec une formule qui contient une variable comme $x + 0.5 * b$ dont b est une variable et augment avec un compteur en utilisant la fonction 'setInterval'.



Code Javascript

```
1. var canvas = document.getElementById('canvas'), //obtenir 'canvas'
2.   context = canvas.getContext('2d'),           //obtenir la toile '2d' pour de
   ssiner
3. var step = 0;
4. //ouvrir un compteur
5. var timer1='';
6. var turn=true;
7.   timer1 = setInterval(beginframe,50);
8.
```



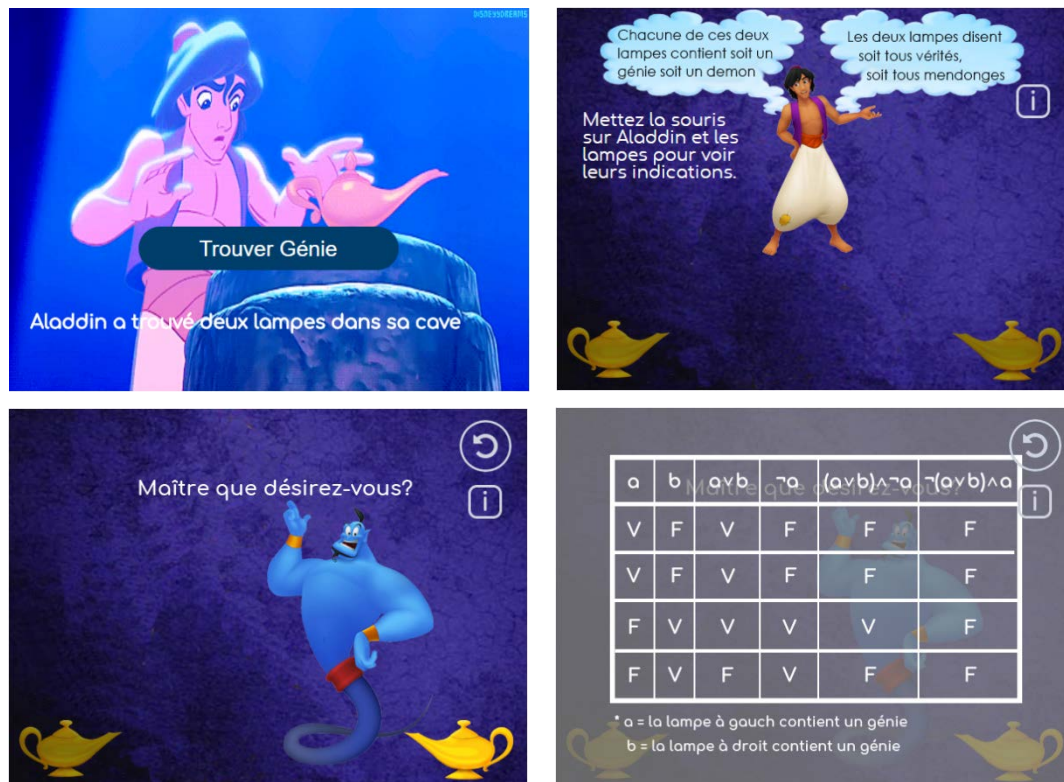
```

9.  function beginframe(){
10.    //effacer la toile
11.    context.clearRect(0, 0, canvas.width, canvas.height);
12.
13.    connect(context,step);
14.
15.    step += 0.02;
16.    if(step>2){
17.        step=0;
18.    }
19. }
20. //l'animation de connection
21. function connect(ctx,speed){
22.    //faire tourner la toile
23.    ctx.save();
24.    ctx.translate(300,250);
25.    ctx.beginPath();
26.    ctx.lineWidth=2;
27.    //définir le couleur
28.    var gradient = ctx.createLinearGradient(-45,-10,45,10);
29.    gradient.addColorStop(0, '#F08080');
30.    gradient.addColorStop(1, '#ADD8E6');
31.    ctx.strokeStyle = gradient;
32.    //dessiner le circle
33.    ctx.rotate(Math.PI*speed);
34.    ctx.arc(-35,0,(2.5*Math.sin(Math.PI*speed))+7.5,0,Math.PI*2,false);
35.    ctx.moveTo((2.5*Math.sin(Math.PI*speed))+7.5,0);
36.    ctx.arc(0,0,(2.5*Math.sin(Math.PI*speed))+7.5,0,Math.PI*2,false);
37.    ctx.moveTo((2.5*Math.sin(Math.PI*speed))+42.5,0);
38.    ctx.arc(35,0,(2.5*Math.sin(Math.PI*speed))+7.5,0,Math.PI*2,false);
39.    ctx.stroke();
40.    ctx.restore();
41. }

```

5. Chapitre3-Logique et WEB sémantique I:

On a choisi *Le langage de la logique propositionnelle* pour ce cours. Pour montrer l'utilisation de ce langage, on donne à l'utilisateur un petit jeu « Trouver le génie ». L'utilisateur fait son choix d'une lampe en cliquant sur ce dernier selon les indications données et son propre jugement. En cliquant sur l'icône « info », l'utilisateur peut savoir la principe de ce jeu. L'utilisateur peut refaire le choix en cliquant sur le bouton « retour ». Dans la partie Code JQuery, nous avons utilisé les fonctions concernant les actions de la souris et 'fadeTo' afin de contrôler chaque élément et de rendre l'interface plus interactive.



```

1. $("#lampel").mouseover(function(){
2.     if (end===false){
3.         $("#lampel").stop();
4.         $("#lampelHover").stop();
5.         $("#dialogL").stop();
6.         $("#lampel").fadeTo(0,0);
7.         $("#lampelHover").fadeTo(0,1);
8.         $("#dialogL").fadeTo(500,1);
9.         $("#dialogL").css("z-index",100);
10.        lampeleft=true;
11.    }
12. });

```

```

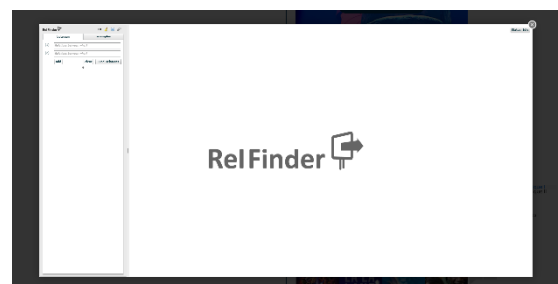
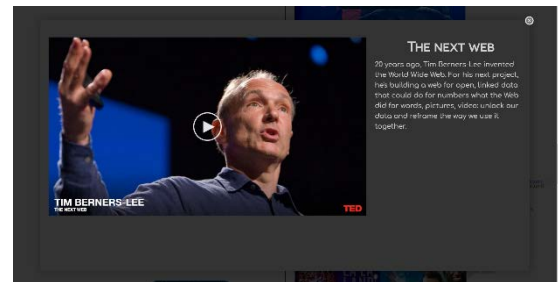
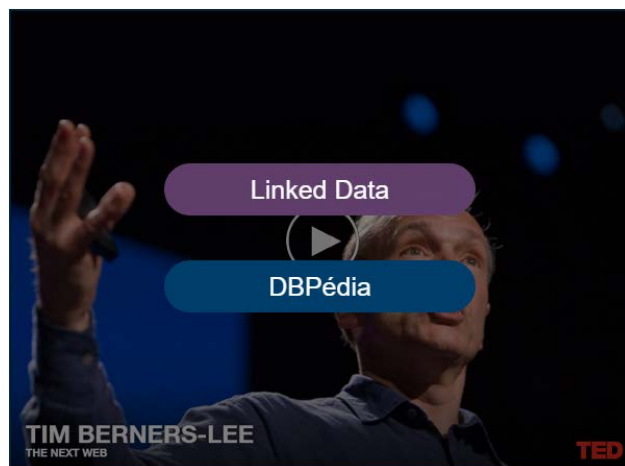
13. $("#lampel").mouseout(function(){
14.     if (end===false){
15.         $("#lampel").stop();
16.         $("#lampelHover").stop();
17.         $("#dialogL").stop();
18.         $("#lampel").fadeTo(0,1);
19.         $("#lampelHover").fadeTo(0,0);
20.         $("#dialogL").fadeTo(0,0);
21.         $("#dialogL").css("z-index",0);
22.     }
23.     changeIndication();
24. });
25. $("#lampel").mousedown(function(){
26.     if (end===false){
27.         $("#lampel").stop();
28.         $("#lampelHover").stop();
29.         $("#lampel").fadeTo(0,1);
30.         $("#lampelHover").fadeTo(0,0);
31.     }
32. });
33. $("#lampel").mouseup(function(){
34.     if (end===false){
35.         $("#lampel").stop();
36.         $("#lampelHover").stop();
37.         $("#lampel").fadeTo(0,0);
38.         $("#lampelHover").fadeTo(0,1);
39.         if ((lampeleft===true && lamperight===true && aladdin===true)||back=
==true){
40.             $("#demon").fadeTo(500,1);
41.             $("#demon").css("z-index",100);
42.             $("#dialogL").fadeTo(0,0);
43.             $("#return").fadeTo(500,1);
44.             $("#icon").fadeTo(500,1);
45.             $("#return").css("z-index",300);
46.             $("#Aladdin").fadeTo(0,0);
47.             $("#AladdinHover").fadeTo(0,0);
48.             $("#indication2").hide();
49.             $("#indication1").hide();
50.             end=true;
51.         }
52.     }
53. });

```

6. Chapitre4-Logique et WEB sémantique II :

Dans cette partie, l'utilisateur va comprendre la définition de linked-data et comprendre qu'il existe des relations différentes et complexes entre les données.

Pour expliquer ces concepts, on montre à l'utilisateur un vidéo de TED et le site de DBPédia, en fonction d'un plugin qui s'appelle *layer*. Quand l'utilisateur clique sur chacun de ces deux boutons, la couche extensive ouvre avec soit un vidéo soit un site.



```
1. //ouvrir une couche (layer)
2. //les paramètre entrées sont le type et le contenu du layer
3. //typeL:(1, adapté pour DOM, str; 2, adapté pour frame),
   contenu(type=1:${'xxx'}); type=2:'xxxxxxx')
4. function layershow(typeL,contenu){
5.     layer.ready(function(){
6.         layer.open({
7.             type: typeL,
8.             title: 0,
9.             maxmin: false,
10.            shade: 0.8,
11.            closeBtn:1,
12.            shadeClose: true,
13.            area: ['90%', '90%'],
14.            content:contenu,
15.        });
16.    });
17. }
```

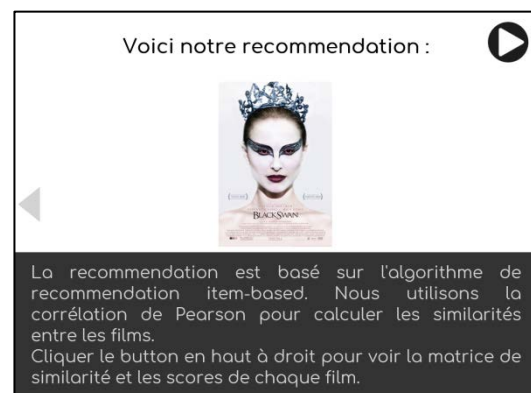
7. Chapitre5-Intelligence collective apprentissage :

Dans cette partie, nous présentons l'algorithme de recommandation item-based en recommandant d'un film qu'il peut être aimé aussi en fonction des préférences de l'utilisateur.

D'abord, l'utilisateur doit évaluer 4 films. La notation est faite en marquant le nombre d'étoiles (1 à 5). Dans cette page, nous enregistrons la note de l'utilisateur pour chaque film, et après avoir calculé des similarités de Pearsons entre chaque films notés et chaque films qui vont être recommandés, nous mettons les films préparés en ordre et recommandons l'utilisateur le premier film dans la deuxième page.

Dans la deuxième, en cliquant les différents boutons, l'utilisateur peut retourner à la première pour changer des notes, ou passer à la dernière page pour consulter la théorie de ce algorithme.

Nous avons utilisé la JQuery méthode '.attr()' afin de restituer la valeur que nous allions affecté dans 'id' de la balise 'div'. Nous pouvons donc obtenir le score que l'utilisateur donne pour chaque film.



	Evaluation	Coco	Les Misérables	Black Swan	Star War
Avenger 3	4	-0.42	-0.03	0.62	0.2
La la land	2	0.22	0.54	0.95	0.92
Lion king	1	-0.13	0.52	0.67	0.06
Intouchable	5	0.53	0.59	0.25	0.38
Normalisé		-0.68	0.04	0.83	0.70

```

1. //éclair les stars
2. $(function () {
3.     var objs = $(".stars a");
4.     $(objs).mouseover(function () {
5.         var ix = $(this).index();
6.         sets(ix, this);
7.     });
8.
9.     $(objs).mouseout(function () {
10.        var id=$(this).parent().attr("id");
11.        var idn=Number(id.substr(1));
12.        var ix = starcliq[idn];
13.        if (ix == undefined || ix == 0)
14.            ix = -1;
15.        sets(ix-1, this);
16.    });
17.
18.    $(objs).click(function () {
19.        var ix = $(this).index();
20.        var id=$(this).parent().attr("id");
21.        var idn=Number(id.substr(1));
22.        film[idn]=ix +1;
23.        starcliq[idn]=ix + 1;
24.        sets(ix, this);
25.        $(this).parent().next("p").html(film[idn] + '.0');
26.    });
27. });
28.
29. function sets(ix, obj) {
30.     $(obj).parent().children().each(function (ik) {
31.         if (ik <= ix) {
32.             $(this).css("backgroundPosition", '0 0');
33.         } else {
34.             $(this).css("backgroundPosition", '0 -20px');
35.         }
36.     });
37. }

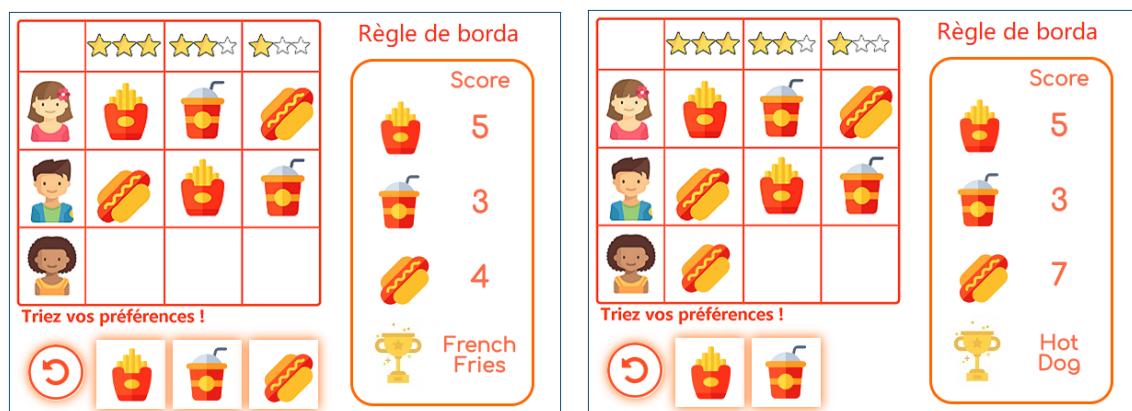
```

8. Chapitre6-Décisions collectives :

Le règle de Borda a été choisi pour montrer ce chapitre. L'animation de cette partie utilise JavaScript pour interagir. L'utilisateur peut faire glisser les trois blocs du bas et les déposer dans les trois cases vides au-dessus.

Les utilisateurs peuvent librement glisser et déposer n'importe quel bloc, soit sur les cases vides, soit lors du placement initial. Chaque fois le bloc est déposé sur la case vide, le score et le vainqueur dans le tableau de score à droite changeront en conséquence. Si l'utilisateur veut initialiser tous les blocs et les scores, il peut cliquer sur le bouton retour du bas.

Nous avons défini 'draggable' attribut 'true' dans les trois images en bas et puis 'ondragstart' attribut appelle la fonction 'drag(event)', qui spécifie les images à faire glisser. Après 'ondragover' spécifie où les images glissées peuvent être déposés. Par défaut, nous ne pouvons pas insérer aucun élément dans un DOM existant directement sur l'interface. La résolution est d'appliquer la fonction event.preventDefault(). En fin, 'ondrop' attribut appelle la fonction 'drop(event)', qui nous permet d'appendre les images dans les cases vides ou le placement initial.



Exemple de code :

```

1. <!--la position de Trois blocs de flotteur dans la table de socre-->
2. <div id="trash">
3. <div id="drop1" ondrop="drop(event)" ondragover="testDrop(event)"></div>
4. <div id="drop2" ondrop="drop(event)" ondragover="testDrop(event)"></div>
5. <div id="drop3" ondrop="drop(event)" ondragover="testDrop(event)"></div>
6. </div>
7. <!--la position de Trois blocs de flotteur-->
8. <div id="source" ondrop="drop(event)" ondragover="allowDrop(event)">
9. 

```

```

10. 
11. 
12. </div>

```

```

1. function testDrop(ev)
2. {
3.     dropid=ev.target.id;
4.     if (dropid.indexOf("drag")!=0 && document.getElementById(dropid).childE
        lementCount==0){
5.         ev.preventDefault();
6.     }
7. }
8.
9. function drag(ev)
10. {
11.     targetid = ev.target.id;
12.     document.body.ondrop = function(event){
13.         event.preventDefault();
14.         event.stopPropagation();
15.     };
16.     ev.dataTransfer.setData("Image",targetid);
17. }
18.
19. function drop(ev)
20. {
21.     var data=ev.dataTransfer.getData("Image");
22.     document.body.ondrop = function(event){
23.         event.preventDefault();
24.         event.stopPropagation();
25.     };
26.     ev.target.appendChild(document.getElementById(data));
27.     dropid=ev.target.id;
28.     calcul();
29.     winner();
30. }

```

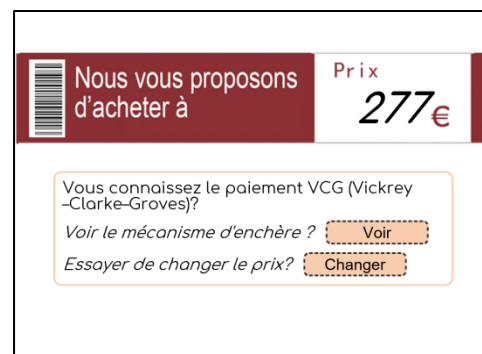
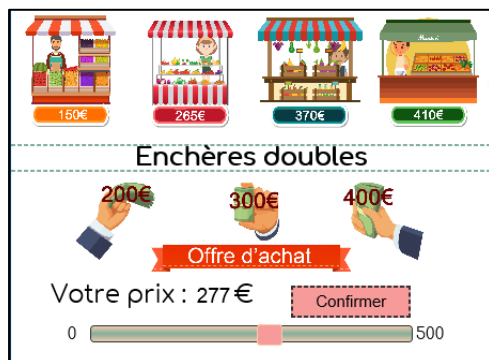

9. Chapitre7-Mécanismes d'enchères:

On a choisi l'enchère double, qui fixe un bien unique et compose par des vendeurs multiples et des acheteurs multiples, pour montrer la règle de paiement VCG.

Dans une marche libre, 4 vendeurs proposent leurs offres du prix de vente, ainsi que 3 acheteurs existant proposent des offres du prix d'achat. L'utilisateur peut proposer son offre d'achat entre 0€ à 500€ en déplacer le bloc coulissant.

Après avoir confirmé en cliquant le bouton *Confirmer*, le prix final est proposé. Si l'utilisateur veut connaître le paiement VCG et voir le mécanisme d'enchère, en cliquant le bouton Voir, le règle de paiement selon la situation s'affiche sur le site. Sinon, l'utilisateur peut revenir au marché en cliquant le bouton *Changer*.

Dans ce cours, nous avons utilisé la propriété pour renvoyer la coordonnée horizontale (selon la zone client) du pointeur de la souris lorsqu'un événement de souris a été déclenchée. Donc, nous pouvons obtenir la distance horizontale que le bloc coulissant déplacé.



1. Classement des prix

Prix vente	150	265	370	410
Prix achat	400	300	277	200

2. Relier au maximum K valeur (Achat > Vente) $k = 2$

3. Acheteur paie 277€. Vendeur reçoit 300€.

	$a^k < v^{k+1}$	$a^k \geq v^{k+1}$
$a^{k+1} < v^k$	(v^k, a^k)	(v^k, v^{k+1})
$a^{k+1} \geq v^k$	(a^{k+1}, a^k)	(v^{k+1}, a^{k+1})

Fermer

```
1. var prix = document.getElementsByClassName("move_prix")[0];
2. var input = document.getElementsByTagName("input")[0];
```

```

3. prix.style.left = 0;
4. prix.onmousedown = function (evt) {
5.     var obj = this;
6.     var oldX = evt.clientX;
7.     var left = parseInt(obj.style.left);
8.
9.     document.onmousemove = function(evt) {
10.        //la position du bloc
11.        var x = evt.clientX - oldX;
12.        obj.style.left = left + x + "px";
13.
14.        var res=parseInt(obj.style.left);
15.
16.        if ( res < 0) {
17.            obj.style.left = 0;
18.            Res = 0;
19.        }
20.        if ( res > 400) {
21.            obj.style.left = 400 + "px";
22.            Res = 400;
23.        }
24.        //montrer la position
25.        input.value = Math.ceil(res / 400 * 500);
26.        inputprix = input.value;
27.    }
28.
29.    document.onmouseup = function (evt) {
30.        document.onmouseup = null;
31.        document.onmousemove = null;
32.    }
33. }

```

10. Chapitre 8 - Allocation des ressources:

Le problème du sac à dos est choisi pour montrer l'algorithme de programmation par contraintes.

L'utilisateur doit essayer d'optimiser la valeur de la combinaison d'objets, qui peuvent être choisis par des checkbox, sans dépasser le poids max du sac à dos. Le poids restant s'affiche après des choix d'objets. L'utilisateur peut vérifier sa réponse en cliquant le bouton *Vérifier*. Après le succès de trouver le meilleur choix, la valeur et le poids d'objets vont changer automatiquement.


Problème de sac à dos


Valeur Total : 915 \$


Vérifier réponse


Poids total :


342 g restant


☒ 18g 88\$


☐ 300g 185\$

☐ 350g 270\$


☒ 140g 112\$

☐ 485g 210\$

☐ 600g 450\$

☒ 1000g 715\$

Essayer
d'optimiser la valeur
de la combinaison
d'objets sans
dépasser le poids
max : 1500g



```
1. var sumid=0,
2.   sumvalue=0;
3. $(' .ressource').click(function() {
4.   sumid=0;
5.   $(' .ressource').find('input:checked').each(function(){
6.     sumid+=Number($(this).attr("id"));
7.   });
8.   sumvalue=0;
9.   $(' .ressource').find('input:checked').each(function(){
10.    sumvalue+=Number($(this).attr("value"));
```

```
11.         });
12.
13.         if(sumid>1500){
14.             alert("C'est déjà hors limite. Essayez une autre fois.");
15.             $('.ressource').find('input:checked').prop("checked",false);
16.             $(".sumid").text(0);
17.             $(".pourcentage").animate({width:"0"});
18.             $(".sumvalue").text(0);
19.         }
20.         else{
21.             $(".sumid").text(1500-sumid);
22.             $(".pourcentage").animate({width:sumid/30*4},500);
23.             $(".sumvalue").text(sumvalue);
24.         }
25.
26.     });
```

11. Fonction supplémentaire

Tous les textes sont invisibles quand l'utilisateur ouvre ce site, ils se font apparaître pendant que l'utilisateur glisse vers le bas.

```
1. $(document).ready(function(){
2.     //L'événement de défilement
3.     $(window).scroll(function(){
4.         //obtenir la hauteur où la page a descendu
5.         var scroll_top = $(window).scrollTop();
6.         $('p, h1, h2, .description img, li, .titleframe').each(function(){
7.             var height = $(this).offset().top;
8.             if (scroll_top >= height-1000 ){
9.                 $(this).fadeTo(1000,1);
10.            }
11.        });
12.    });
13. });
```

V- Problèmes rencontrés

Dans le processus de développement de page web, nous avons principalement rencontré des difficultés à la fois en matière de développement et de coopération.

Programmation

lorsque nous utilisons JavaScript et Canvas pour dessiner, nous devons trouver la position exacte de chaque texte et ligne. En raison de différents ordinateurs et différentes résolutions, nous devons constamment tester et ajuster, ce qui est une lourde charge de travail et très gênant. Cependant, nous n'avons pas trouvé de bonne solution.

En outre, lors de l'utilisation de jQuery glisser-déposer (drag and drop), une partie du code HTML affectée change en conséquence. Une autre partie du code HTML va changer en fonction de ses changements. Nous essayons de l'implémenter avec un bouton (cliquez pour déclencher), c'est-à-dire, faites glisser et déposer d'abord et cliquez après. Toutefois, lorsque la page n'est pas actualisée, la fonction déclenchée par un clic ne peut pas lire le code modifié de la première partie, nous n'avons donc pas réussi. Enfin, nous utilisons la fonction de fin de glisser-déposer pour déclencher.

Coopération

Depuis que nous travaillons dans une division du travail, certains utilisent des frameworks externes et des codes externes qui peuvent fonctionner correctement sur leurs ordinateurs respectifs. Mais lorsque nous les intégrons dans la même page web, nous ne pouvons pas tous fonctionner correctement. Il existe également des différences dans la dénomination des chemins de fichier, ce qui entraîne également des problèmes d'intégration. Cependant, nous avons découvert et rectifié à temps, donc nous n'avons pas perdu trop de temps.

VI- Conclusion

L'adresse de la page Web finalisée: <http://etu-web2.ut-capitole.fr/~21611938/JSProjet/Index.html>

Et notre code source et tous les certificats de copyright de photo obtenus sont postés sur GitHub: <https://github.com/liaoxiyue/JS-Projet2018-UT1>

Merci encore pour notre tuteur.

Dans la production de page web, nous avons utilisé ce que nous avons appris. En même temps, nous avons également fait beaucoup de recherche sur internet et fait beaucoup des tentatives pour atteindre nos objectifs. Ce n'est pas un processus facile, mais nous avons pris assez de patience et en fait, nous avons le résultat que nous voulons.

Afin de garantir la convivialité de site web, nous adaptons notre site web dans différents navigateurs :

	Chrome	Firefox	IE	Microsoft Edge	Opera
Problème	X	X	HTML drag and drop ne fonctionne que sur la version 9.0	HTML drag and drop ne fonctionne que sur la version 9.0	X

La division du travail a aussi eu un très bon effet, de sorte que nous pouvons respecter le délais de réalisation. Ce projet a été vraiment intéressant et nous sommes fiers de notre résultat.

VII- Référence

1. WIKI- Notepad++ : <https://fr.wikipedia.org/wiki/Notepad%2B%2B>
2. JQueryui : <https://jqueryui.com/droppable/#photo-manager>
3. JavaScript Array : http://www.w3school.com.cn/js/js_obj_array.asp
4. JavaScript Calcul : https://blog.csdn.net/ye_nianfeng/article/details/70767268
5. JS getElements : <https://zhidao.baidu.com/question/1540833269142888387.html>
6. Icon : <https://codepen.io/nourabusoud/pen/ypZzMM>
7. Canvas tutorial : https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API/Tutorial
8. Template Site web : <https://www.wix.com/website-template/view/html/2145/?siteId=65d42569-1b1b-4289-a341-b795551f1a25&metaSiteId=24ef4d84-33fe-49eb-84ed-2cdf3e8a1fbd&originUrl=https%3A%2F%2Ffr.wix.com%2Fwebsite%2Ftemplates%2Fhtml%2Fcreative-arts>
9. Carrousel : <http://juniortour.net/demo/standard-js-carousel/loop-theory-demonstration.html>
10. Répertoire latéral : <https://www.cnblogs.com/dreamkeeper/p/7634044.html>
11. Slider : <https://www.cnblogs.com/11lang/p/6819142.html>