**Documentatie modul**

**Interfata**

**Universitatea “Alexandru Ioan Cuza”**

**Facultatea de informatica**

**Disciplina: Inteligenta Artificiala**

**Componenta echipei:**

Adam Alexandru

Corduneanu Ioana

Savencu Alexandru

Stefanovici Andreea

Limbajul HTML ((HyperText Markup Language) este un limbaj care permite crearea de documente de tip hipertext, documente care vor fi utilizate si vizualizate prin paginile Web corespunzatoare.

             HTML a fost dezvoltat initial de Tim Berners-Lee la CERN in 1989. Este o implementare a limbajului SGML (Standard Generalized Markup Language), standard international ISO-8879, aprobat în 1986. HTML a fost dezvoltat initial ca o posibilitate pentru fizicienii care utilizeaza computere diferite de a realiza schimburile de informatii prin intermediul Internetului.

HTML prezinta o serie de caracteristici:

-         permite descrierea structurii documentelor fara a descrie organizarea in pagina a acestora;

-         ofera o reprezentare proprie pentru fiecare navigator;

-         permite reactualizarea documentelor HTML;

-         produsul este independent de paltforma tehnica.

     Utilizarea HTML ofera urmatoarele avantaje:

-         este usor de asimilat si de utilizat;

-         ofera un control total asupra elementelor utilizate in realizarea de pagini web;

-         permite un grad inalt de reutilizare, astfel incat pentru realizarea unei pagini web se pot folosi cu succes secvente de cod HTML de la alte pagini web realizate anterior.

Pentru realizarea unui document HTML se pot utiliza editoare clasice sau editoare de tipul WYSIWYG (What You See Is What You Get =  ceea ce vezi este ceea ce obtii). In cazul editoarelor clasice se pot folosi editoare de text ASCII, editoare HTML sau convertoare HTML. In cazul editoarelor WYSIWYG, accentul se pune pe aspectul grafic al paginii web, scrierea codului HTML realizandu-se de catre aceste editoare.

**Structura documentelor HTML**

<HTML>                              inceputul documentului

          <HEAD>                    antet

                   <TITLE>          titlu

                   </TITLE>         sfarsit de titlu

          </HEAD>                   sfarsit de antet

<BODY>                    corpul documentului

.

.

.

         </BODY>                    sfarsitul corpului documentului

</HTML>                             sfarsitul corpului documentului

**Instructiuni**

<Hi> . </Hi>   i=1,.,6      Seteaza dimensiunile caracterelor; pentru i=1 caracterele au dimensiune maxima.

<EM> . </EM>  Scoate in evidenta un caracter.

<STRONG> . </STRONG> Scoate puternic in evidenta un caracter.

<B> . </B> Bolduirea unui caracter.

<I> . </I> Scriere italica a unui caracter.

<BR> Salt la linia urmatoare

<P> . </P> Creare de paragraf nou.

< HR [SIZE=n] [WIDTH=m] [ALIGN=LEFT / CENTER / RIGHT] [NOAHADE] > Traseaza o linie orizontala de separare, cu caracteristici referitoare la lungime (SIZE), grosime (WIDTH), aliniere (stanga, dreapta, centrata) si lipsa sau aparitia umbrei pentru linie (NOSHADE).

<MULTICOL COLS=n GUTTER=p WIDTH=d>  .  </MULTICOL> Permite scrierea pe mai multe coloane, cu setarea dimensiunilor coloanelor si a spatiilor dintre coloane

<BODY BGCOLOR=b TEXT=t LINK=p VLINK=v ALINK=a> Permite setarea culorilor pentru ecran, text, legaturi.

**Liste**

Pot fi create urmatoarele tipuri de liste:

-         ordonate

-         neordonate

-         cu definitii

**Liste ordonate**

<OL TYPE=t START=s>

          <Li> text1

          <Li> text2

          <Li> text3

</OL>

t seteaza modul de numerotare al listei, iar s seteaza elementul de start.

t=1 numerotare cu numere

t=a numerotare cu litere alfabetice minuscule

t=A numerotare cu litere alfabetice minuscule

t=i numerotare cu litere romane mici

t=I numerotare cu litere romane mari

Exemplu:

<OL TYPE=1 >

          <Li> Analiza

          <Li> Engleza

          <Li> Matematica

</OL>

**Liste neordonate**

<UL TYPE=t >

          <Li> text1

          <Li> text2

          <Li> text3

</UL>

t defineste elementul din dreptul fiecarei componnete a listei.

t=square     apare un patrat

t=circle      apare un cerc

t=disc         apare un disc

Exemplu:

<UL TYPE=square >

          <Li> Analiza

          <Li> Engleza

          <Li> Matematica

</UL>

**Liste cu definitii**

<DL>

<DT> termen de definit

<DD> definitia termenului

.....

<DT> termen de definit

<DD> definitia termenului

</DL>

Exemplu.

<DL>

<DT> Autovehicul

<DD> Mijloc de transport destinat transportului de persoane sau de bunuri.

</DL>

API reprezintă acronimul în limba engleză pentru Application Programming Interface – în română – interfaţă de programare a unei aplicaţii. O librărie API poate fi privită ca un set de funcţii puse la dispoziţia programatorilor în sensul efectuării unor anumite operaţii sau sarcini.

De exemplu, există API-uri ce permit programatorilor să introducă în aplicaţiile pe care le dezvoltă facilităţi de recunoaştere facială. Programatorul respectiv nu trebuie să cunoască în detaliu mecanismele ce stau în spatele recunoaşterii faciale. El trebuie doar să ştie cum să utilizeze corect funcţiile puse la dispoziţie prin API-ul respectiv.

Un API este o colecţie de funcţii conţinute în librării statice sau dinamice, ce pot fi folosite la un moment dat într-o aplicaţie pentru a efectua diverse sarcini.

Librăriile API conţin fişierele cu definiţiile funcţiilor utilizate de producătorul respectiv, librăriile statice necesare în momentul compilării sau librăriile dinamice necesare în monentul rulării. API-urile conţin de asemenea instrucţiuni necesare programatorilor în ceea ce priveşte apelul funcţiilor sau compilarea codului sursă

De exemplu, dacă doriţi ca o aplicaţie Windows să deschidă o casetă de dialog pentru salvarea unui fişier, atunci trebuie să apelaţi o funcţie conţinută în librăria API de la Microsoft pentru utilizarea funcţiilor sistem.

Acelaşi lucru este valabil pentru cei ce doresc să includă facilităţi Adobe, Google sau Facebook în aplicaţiile lor – trebuiesc utilizate API-urile respective. Funcţiile specifice sistemelor de operare pot fi apelate numai dacă se cunoaşte sintaxa acestora, mai precis argumentele ce se transmit funcţiilor respective.

Librăriile API permit potenţialilor atacatori să aibă acces la detaliile de implementare şi funcţionare ale unui sistem de operare sau a unei aplicaţii software. Din acest motiv, librăriile API sunt puse la dispoziţie contra cost sau pe bază de înscriere pe site-ul producătorului respectiv.

Api-ul pentru animatia de redare vizuala a sunetului din acest proiect se numeste WaveSurfer.

Instructiuni de folosire:

In primul rand, va trebui sa inseram libraria wavesurfer.js in fisierul nostru html. Ultima versiune o putem gasi la link-ul: [unpkg.com](https://unpkg.com/wavesurfer.js).

<**script** src="https://unpkg.com/wavesurfer.js"></**script**>

Vom crea un container acolo unde vrem sa apara animatia wavesurfer:

<**div** id="waveform"></**div**>

Apoi, in codul javascript vom creea o instanta globala a obiectului WaveSurfer.

**var** wavesurfer = WaveSurfer.create({

container: '#waveform'

});

Container-ul va fi singurul parametru de care vom avea nevoie.

In orice caz, putem adauga oricate optiuni dorim. De exemplu, sa implementam functia de scroll in cadrul animatiei.

**var** wavesurfer = WaveSurfer.create({

container: '#waveform',

scrollParent: true

});

After creating an instance, you may want to load an audio track and draw its waveform. You can load files either from the same domain:

Dupa crearea unei instante, va trebui sa adaugam un fisier audio pentru a genera forma animatiei.

wavesurfer.load('../audio/song.mp3');

Wavesurfer.js va incarca fisierul, il va decoda si va afisa animatia. Cand este gata pentru afisarem va declansa un eveniment ready.

Wavesurfer.js are mai multe evenimente din care putem alege. Eventimentul ready specificat mai sus poate fi folosit in felul urmator:

wavesurfer.on('ready', **function** () {

wavesurfer.play();

});

Alte evenimente:

wavesurfer.pause(), wavesurfer.skipForward(), wavesurfer.toggleMute() etc.

**CSS** *(Cascading Style Sheets)* este un standard pentru formatarea elementelor unui document [HTML](https://ro.wikipedia.org/wiki/HTML). Stilurile se pot atașa elementelor HTML prin intermediul unor fișiere externe sau în cadrul documentului, prin elementul <style>și/sau atributul style. CSS se poate utiliza și pentru formatarea elementelor [XHTML](https://ro.wikipedia.org/wiki/XHTML), [XML](https://ro.wikipedia.org/wiki/XML) și [SVGL](https://ro.wikipedia.org/w/index.php?title=SVGL&action=edit&redlink=1).

**SS3 reprezintă un upgrade ce aduce c**â**teva atribute noi și ajută la dezvoltarea noilor concepte in webdesign.**

Unele dintre cele mai importante segmente (*module*) noi adăugate acestui standard pentru formatarea elementelor HTML aduc un plus considerabil in dezvoltarea activități webdesign.

Mai jos sunt prezente in listă cele mai importante modulele adăugate in CSS3:

* Selectors
* Box Model
* Backgrounds and Borders
* Image Values and Replaced Content
* Text Effects
* 2D/3D Transformations
* Animations
* Multiple Column Layout
* User Interface

Deși au apărut unele deficiente de compatibilitate intre browsere, majoritatea proprietăților CSS3 au fost implementate cu succes in variantele browserelor noi.

**CSS3 - Borduri**

Acum CSS3 oferă posibilitatea de a crea borduri cu colțurile rotunjite fară a folosi elemente grafice de fundal așa cum se folosea anterior acestui upgrade.

Proprietatea CSS3 border-radius definește prin valorile exprimate in pixeli cat de rotunjite vor fi colțurile unui element HTML sau unei imagini. Fiecare colț poate avea o alta valoare exprimată in pixeli diferită de un alt colț al aceluiași element. Prin urmare putem folosi pana la 4 valori diferite atribuite unui element HTML sau imagine.

Exemplu:

border-radius: 5px ;

- definește valoarea de 5px radius pentru toate cele 4 colțuri ale elementului.

border-radius: 5px 7px 12px 4px;

- aceste valori multiple definesc cat de mult vor fi rotunjite colțurile elementului HTML, iar pentru fiecare colt este specificata valorarea. Colțul stanga-sus are valoarea border-radius de 5px, colțul dreapta-sus are valoarea border-radius de 7px, colțul dreapta-jos al elementului HTML are valoarea de 12px iar colțul din stanga-jos are valoarea de 4px.