

FACULTATEA CALCULATOARE, INFORMATICA SI MICROELECTRONICA

UNIVERSITATEA TEHNICA A MOLDOVEI

MEDII INTERACTIVE DE DEZVOLTARE A PRODUSELOR SOFT

LUCRAREA DE LABORATOR#4

Web development

Autor:

Victor Talpa

lector asistent:

Irina COJANU

lector superior:

Svetlana COJOCARU

Laboratory work #4

1 Scopul lucrării de laborator

Studierea tehnologiilor folosite in Web development

2 Obiective

1. Realizarea unui simplu Web Site personal
2. Familiarizarea cu HTML si CSS
3. Interactiuni Javascript

3 Laboratory work implementation

3.1 Tasks and Points

1. Realizeaza un mini site cu 3 pagini statice
2. Site-ul trebuie sa pastreze toata informatia intr-o baza de date
3. Site-ul trebuie sa contina AJAX Requests.
Implimentarea XHR sau JSON responses. Careva din informatie trebuie sa fie dinamic incarcata pe pagina.

3.2 Analiza lucrarii de laborator

Link spre repozitoriu: <https://github.com/VictorTalpa/MIDPS/tree/master/Lab%234>

1. Initial am elaborat 4 pagini statice
2. Am lucrat la afisarea corecta a imaginii background si mentinerea pozitiei si dimensiunilor corecte pe diferite monitoare
3. Am ales un font si am afisat un text
4. Background-ul si textul au fost formate cu ajutorul CSS
5. Am implementat bara de meniuri, cu linkuri spre paginile website-ului cit si stilizarea acestora cu culori transparente si color fade in si fade out la mouse hover
6. Am adaugat cateva functii Javascript pentru autoscrolling la apasarea unui buton si schimbarea textului la apasarea acestuia
7. Am instalat node.js dupa care folosind comanda "npm install express" am instalat modulul express.js pentru pornirea unui server pe localhost
8. Am elaborat un javascript server.js care porneste serverul si creeaza referinta catre pagina locala .html
9. Tot aici am folosit app.use pentru a crea si o referinta catre resursele necesare
10. Am creat un fisier de tip JSON ce contine lista studentilor grupei
11. Am folosit AJAX request pentru a deschide JSON-ul, apoi am folosit JSON.parse pentru a putea lucra cu informatia din JSON in Javascript, astfel informatia fiind dinamic incarcata pe pagina.

3.3 Imagini

Folosind terminalul, navigam in directoria serverului si executam comanda "node server.js" pentru a porni serverul folosind node.js, dupa care folosind console.log, pentru comoditate am afisat adresa, portul si mesajul pentru oprirea serverului



```
Server — node server.js — 81x15

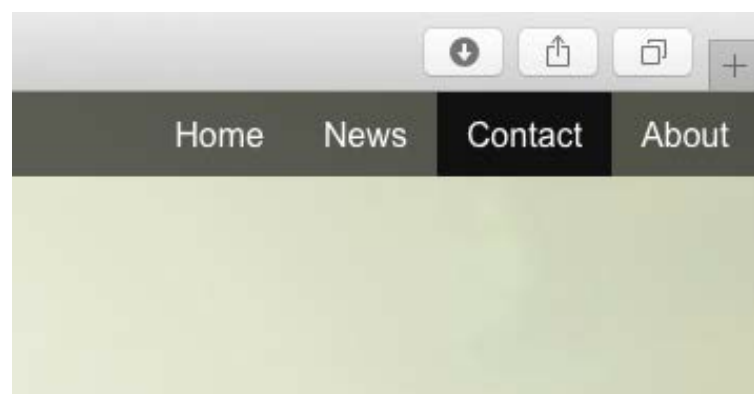
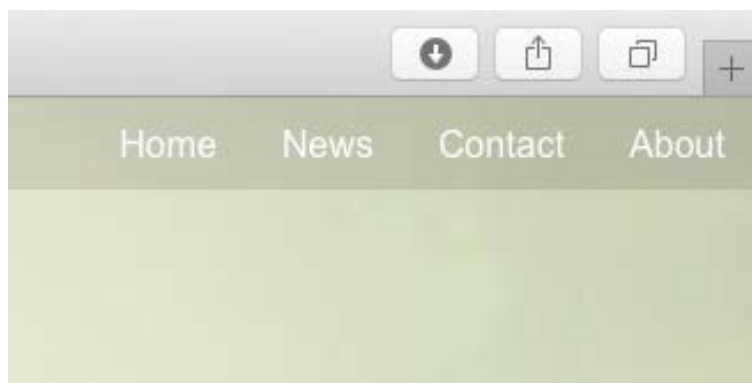
[MacBook-Pro:Server Viktor$ node server.js
Server running at http://localhost:8080 Ctrl+C to stop
```

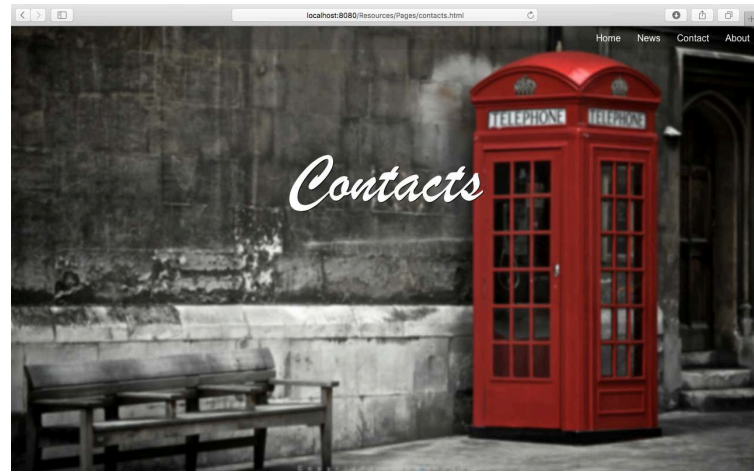
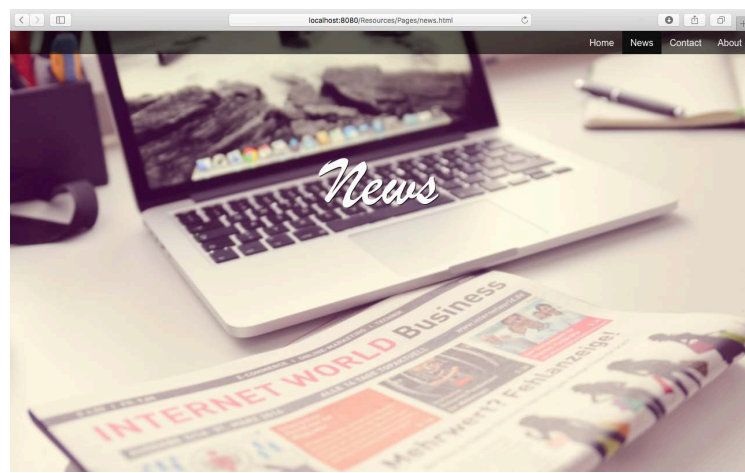
```
1  var express = require('express');
2  var app = express();
3  const PORT = 8080;
4
5  app.use(express.static(__dirname + '/HTML'));
6
7  app.get('/', function (req, res) {
8    res.sendFile(__dirname + "/HTML/main.html");
9  });
10
11 var server = app.listen(PORT);
12 console.log("Server running at http://localhost:%s Ctrl+C to stop", PORT);
```

Acum putem accesa adresa propusa pentru a fi directionati catre html-ul nostru



Folosim bara de meniuri pentru a accesa alte pagini





Aici se afiseaza lista studentilor extrasa din JSON prin intermediul unei cereri AJAX

Lista studentilor TI-144

No.	Name
1.	Capcelea Nicolae
2.	Cherdivara Artiom
3.	Gorduz Daniel
4.	Melninceanu Rodion
5.	Mitu Rostislav
6.	Nanii Gheorge
7.	Nirones Cristian
8.	Novac Mihai
9.	Olaru Cristian
10.	Pascan Ion
11.	Pascari Ion
12.	Petrenco Mihai
13.	Plotnicu Rodica
14.	Prisacari Ion
15.	Puhalschi Alexandru
16.	Rosca Andrei
17.	Rusu Dragos
18.	Saca Marin
19.	Soltan Alexandru
20.	Sova Cristian
21.	Stratan Ion
22.	Susorov Vladislav
23.	Talpa Victor
24.	Vasilic Iurie

```

var xhttp = new XMLHttpRequest();

xhttp.onreadystatechange = function()
{
    if (xhttp.readyState == 4 && xhttp.status == 200)
    {
        var list = JSON.parse(xhttp.responseText);
        printJson(list);
    }
}

xhttp.open("GET", "Resources/Data/jData.json", true);
xhttp.send();

```

Concluzie

În urma efectuării acestei lucrări de laborator, am studiat metodele de creare a unei pagini web de la fundament ne avind nimic facut. Am studiat sintaxa si folosirea concomitenta a HTML-ului cu CSS si posibilitatile acestora pentru a obtine un design dupa placul oricarui developer. Folosind CSS putem formata orice element folosit si afisat in HTML. La fel CSS ne permite crearea unor elemente dinamice cum ar fi color fade folosit in aceasta lucrare. Javascript la rindul sau ne permite manipularea cu elementele afisate pe pagina cum ar fi schimbarea continutului, unele elemente dinamice si de fapt poate fi folosit pentru schimbari radicale pe pagina in timp real. Un detaliu important in Javascript este ca el se aseamana cu limbaje ca C/C++ initial studiate si are o sintaxa familiarizata ceea ce il face sa fie usor de invatat. La fel am obtinut cunostiinte despre JSON sau JavaScript Object Notation, care a devenit concurent formatului XML si defapt permite realizarea scopurilor fisierelor XML folosind mai putin text, deci JSON-ul pur si simplu este mai simplu si continind mai putin text decit XML-ul il face sa ocupe memorie fizica mai putina ceea ce il face mai lightweight ce spre sfirsit se observa in crestere de performanta. AJAX este tehnologia care ne permite sa creem un ciclu de cerere-raspuns (request-response) care in momente necesare incarca informatia dintr-o baza de date pe pagina, fara necesitatea de a face refresh paginii. Folosind toate aceste tehnologii, putem implementa pagini atit cu un design modern cit si implementa functionalitate si performanta in pagina noastra.

References

1 TutorialsPoint

<http://www.tutorialspoint.com/index.htm>

2 W3Schools

<http://www.w3schools.com/>

3 Stack Overflow Community <http://stackoverflow.com/>