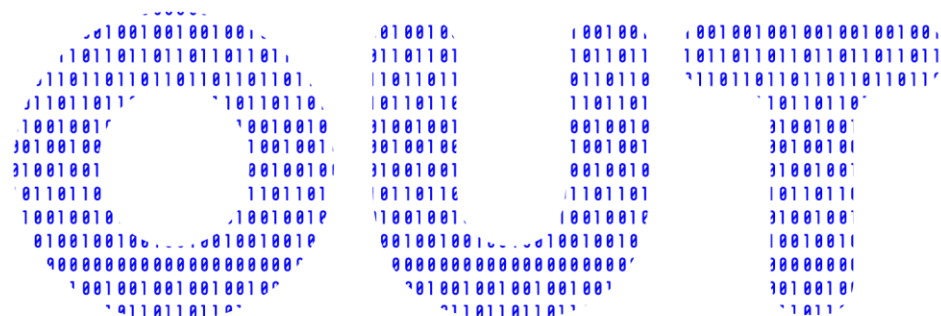




CODING-



> valuable outside tool for employment

IO4. CURSUL DE CODING PENTRU DETINUTI

MODULUL 7 DEZVOLTAREA APLICAȚIEI



principal, precum și o serie de aplicații care folosesc adesea un browser încorporat pentru a vizualiza conținut. Evident, avantajul telefonului mobil este doar: mobilitatea sa – capacitatea de a găsi lucruri în jurul dvs, de a obține indicații de orientare și de a căuta informații rapid când ieși din casă sau de la birou. Însă, din ce în ce mai mult, dispozitivele mobile sunt folosite acasă în timp ce se vizionează televizorul sau dimineața, sau când ne plictisim să așteptăm.

Mobil este cu adevărat mai puțin despre a fi în mișcare și mai mult despre a fi întotdeauna disponibil. Spațiul mobil se schimbă mai repede decât orice sector al pieței, deoarece mulți utilizatori schimbă dispozitivele pe cicluri de actualizare de 12 până la 18 luni și sute (dacă nu chiar mii) de modele noi sunt introduse în fiecare an.

Tabletă

Tabletele au existat de ani de zile, dar abia când Apple a venit împreună cu iPad-ul în 2010 au devenit mai mult un obiect de agrement decât un accesoriu pentru desktop. Multe alte dispozitive au urmat iPad-ul, tabletele bazate pe Android se descurcă deosebit de bine în gama medie de 7" și Microsoft avansând cu Windows 8. Dimensiunea lor mai mare și dependența de Wi-Fi (în multe cazuri) face tabletele mai degrabă portabile decât mobile; există într-un spațiu mai asemănător cu laptop decât un dispozitiv mobil. Majoritatea oamenilor le folosesc acasă și, desi sunt transportate, sunt de obicei scoase și folosite numai atunci când utilizatorul este staționar, mai degrabă decât să meargă pe stradă. La fel ca smartphone-urile tabletele sunt tactile, bazându-se pe introducerea tactilă în majoritatea cazurilor (desi unele acceptă și input de stil). Cercetările arată că tind să fie utilizate pentru divertisment și navigare, atunci când timpul este mai puțin problematic. Browserele de bază de pe tablete sunt, în esență, aceleași cu cele de pe dispozitivele mobile.



Contextul

Cel mai important lucru de reținut este cât de multe lucruri nu știm despre persoanele care folosesc site-urile noastre. Nu știm unde se află – nu ne referim la acest lucru din punct de vedere geographic, deoarece putem folosi GPS pentru a le localiza. Adică, dacă sunt acasă, la serviciu, în autobuz și așa mai departe. În esență, nu avem nicio idee despre contextul în care cineva folosește site-ul nostru. Presupunerea des întâlnită este că persoanele care utilizează dispozitive mobile sunt în mișcare și în grabă, folosind adesea conexiuni cu lățime de bandă redusă, dar nu întotdeauna este cazul: telefonul mobil este adesea utilizat acasă, cu o conexiune rapidă de Wi-Fi. Dar nici măcar conexiunea nu spune întreaga poveste; semnalul poate fi slab sau lățimea de bandă redusă, din cauza congestiei din cartier. În plus, prezumția este că, pentru un dispozitiv cu ecran mare, precum un televizor, spectatorii doresc un mediu grafic mai bogat, dar un ecran mare nu este egal cu un processor puternic sau cu o conexiune rapidă. Multe televizoare compatibile we, au procesoare la fel de puternice ca cele ale unui smartphone, iar viteza conexiunii este deschisă pentru aceleași capricii, ca orice alt dispozitiv. O data cu dimensiunea, intervine problema portabilității. Cu cât dispozitivul este mai portabil, cu atât putem fi mai puțin siguri de contextual în care este utilizat. De fapt, nu putem face prezumții.

“Fast”(Rapid) este singurul context care contează

Aveți o provocare. Nu știți care sunt utilizatorii dvs, unde sunt, ce fac sau cu ce tip de dispozitiv o fac. Puteți afla unele dintre aceste informații (desi nu toate), dar contextual lor integral este complet necunoscut și variază pentru fiecare individ. Singurul lucru pe care îl puteți cunoaște în mod rezonabil este că fie vor acces la ceea ce oferiți, fie vor să afle că nu este ceea ce vor. Oricum ar fi, vor răspunsul rapid. Performanța este singurul criteriu care contează. Indiferent dacă utilizatorii sunt pe un smartphone în timpul orelor de vârf și caută informații despre următorul tren spre casă sau navighează printr-un site de cumpărături, în timp ce sunt ghemuiți pe canapea de acasă, au o sarcină pe care doresc să o îndeplinească cât mai curând posibil, iar acest lucru îl realizează utilizând cel mai



apropiat dispozitiv, fapt care îl face să se simtă mai eficient (acest lucru este cunoscut sub numele de timp găsit)

Site-ul dvs. trebuie să fie rapid - și să se miște rapid - indiferent de dispozitivul pe care este afișat. Rapid nu înseamnă numai performanță tehnică (ceea ce este incredibil de important), ci și *răspunsul/responsiveness* interfeței și cât de ușor pot naviga utilizatorii pe site și pot găsi ceea ce au nevoie pentru a finaliza sarcina care i-a adus în primul rând.



Proiectare pentru cea mai bună experiență a tuturor utilizatorilor

Această secțiune va avea mai multe detalii despre învățarea tehnicilor fundamentale care vă permit să creați site-uri web excelente, indiferent de abordarea dvs., iar cea mai importantă tehnică este cea care are în comun, atât metodele adaptative cât și cele receptive, ceva absolut esențial pentru noua lume a dispozitivelor multiple de dezvoltare: *media queries*.

Adoptarea pe scară largă a CSS permite să lăsați în urmă limitările rigide ale site-urilor web bazate pe tabel, iar interogările media fac un pas mai departe, permițându-vă să creați pagini într-un mod care să fie compatibil cu specificațiile fiecărui dispozitiv pe care sunt afișate site-urile dvs. Un *media query* este o afirmație logică: *if/* dacă logica este adevărată, se aplică regulile de stil din cadrul afirmației; *if/* dacă logica este falsă, regulile sunt omise. Parametrii declarațiilor sunt cunoscuți ca funcții media, iar cele mai frecvent utilizate astăzi se referă la dimensiunile dispozitivului sau ale ferestrei.

```
<link rel="stylesheet" href="foo.css" media="screen">
```

Acest cod apelează foaia de stil externă *foo.css* numai atunci când dispozitivul de vizualizare este un ecran — cu alte cuvinte, nu un tip de suport diferit, cum ar fi imprimarea/print. Extindeți această sintaxă prin simpla adăugare a cuvântului și includerea interogării între paranteze:

```
<link rel="stylesheet" href="foo.css" media="screen and (query)">
```

Acest cod modificat are două condiții: Tipul de suport ar trebui să fie un ecran, iar logica interogării ar trebui să fie adevărată. Dacă ambele condiții sunt îndeplinite, se aplică *foo.css*.

De asemenea, puteți utiliza interogări media pentru a include foi de stil externe din alte foi de stil, utilizând *@import* at-rule. Următorul cod are aceeași logică cu blocul de cod anterior, dar poate fi utilizat în etichetele de stil sau într-o foaie de stil externă:

```
@import url('foo.css') screen and (query);
```



În cele din urmă, puteți utiliza media queries inline, ceea ce este folositor atunci când doriți să aplicați blocuri de reguli pentru cazuri specifice, mai degrabă decât să apelați la foi de stil externe:

```
@media screen and (query) { ... }
```

Caracteristici Media bazate pe dimensiuni

Media queries sunt cel mai frecvent utilizate în zilele noastre, pentru a detecta dimensiunile agentului utilizat pentru a vizualiza conținutul dvs. și pentru a difuza regulile corespunzătoare pentru afișarea aceluia agent: text și imagini mari pentru ecrane de dimensiuni mari, text mic și un aspect cu o singură coloană pentru smartphone, și așa mai departe. Desigur, media queries implică mult mai mult decât aceste contexte simple, dar ați prins ideea generală.

Trebuie să luați în considerare două seturi de dimensiuni: în primul rând, cele ale dispozitivului în sine și, în al doilea rând, cele ale ferestrei agentului (pentru majoritatea oamenilor, acesta este un browser web, dar fereastra ar putea fi, de asemenea, o fereastră a aplicației) pe dispozitivul respectiv. O persoană vă poate vizita site-ul folosind un televizor cu ecran late norm, dar acest lucru vă preocupă foarte puțin, dacă aplicația pe care o folosește persoana respectivă pentru a vizualiza site-ul ocupă doar un sfert din ecran. Pe anumite dispozitive, cele două seturi de dimensiuni sunt aceleași – pe majoritatea smartphonurilor și a tabletelor, spre exemplu, lățimea browserului este aceeași cu lățimea dispozitivului.

Dimensiunile ecranului sunt probabil, cele mai importante și cele pe care le veți utiliza cel mai mult, iar caracteristicile media relevante pentru acestea sunt *height/înălțimea* și *width/lățimea*. Caracteristicile dimensiunii ecranului iau ca argument o valoare de lungime unică cu care se testează logica.; Dacă dimensiunea în cauză este egală cu valoarea lungimii furnizate, logica este adevărată și regulile sunt aplicate. În următoarea interogare, regula din parantezele accoladă se aplică elementului corp al ferestrei, atunci când are o lățime exactă de 480px:

```
@media screen and (width: 480px) {  
  body { background-color: #00f; }  
}
```



Am folosit o valoare px în acest exemplu, dar este permisă orice unitate de lungime. Indiferent de unitatea pe care o utilizați, o valoare exactă este probabil prea specific pentru majoritatea scopurilor. Cu toate acestea, o pereche de extensii la funcție o face mai flexibilă.

Multe caracteristici media, inclusive lățimea, permit prefixele *max-* și *min-* înainte de numele caracteristicii. Acestea reprezintă maxim, respective minim, pe care probabil nu aveți nevoie să le explicăm, și care, în practică înseamnă “nu mai mult de” și “nu mai puțin de”. De exemplu, folosind lățimea maximă, puteți aplica reguli de stil oricărui browser cu o lățime de cel mult 480px și folosind lățimea minimă pentru orice browser care are o lățime de cel puțin 480px:

```
@media screen and (max-width: 480px) { ... }
@media screen and (min-width: 480px) { ... }
```

Dacă sunteți sigur că vreți să lucrați cu dimensiunile dispozitivului, și nu cu cele din fereastră de vizualizare, Puteți utiliza caracteristicile *device-width/ lățimii dispozitivului* și *device-height/ înălțimea dispozitivului* —deși, ca și până acum, cred că *width/ lățimea* este o dimensiune mai frecvent utilizată decât *height/ înălțimea*, așa că ne vom concentra asupra former. Modul în care utilizați caracteristica de *device-width/ lățime a dispozitivului* este, pentru toate scopurile practice, același lucru ca și pentru caracteristica de lățime—doar metrica la care răspunde fiind diferită. La fel ca în cazul *width/ lățimii*, *device-width/ lățimea dispozitivului* poate fi extinsă cu prefixe *max-* și *min-* :

```
@media screen and (max-device-width: 799px) { ... }
@media screen and (device-width: 800px) { ... }
@media screen and (min-device-width: 801px) { ... }
```

Un alt mod de a difuza regulile depinde de raportul de aspect/ aspect ratio al dispozitivului sau al ferestrei. *Raportul de aspect/ aspect ratio* este raportul dintre *lățime/width* sau *înălțime/ height* (sau lățimea dispozitivului până la înălțimea dispozitivului); un monitor 1024×768, spre exemplu, are un raport de aspect de 4/3— însemnând, patru pixeli orizontali la fiecare trei pixeli verticali—cee ace este comun pentru ecranele mai vechi. Dispozitivele cu ecran plat, main oi, (precum



iPhone) tind să utilizeze un raport de 16/9. Pentru a viza un ecran cu un raport de aspect de 4/3 ar trebui să utilizați funcția dispozitiv-raport de aspect:

```
@media screen and (device-aspect-ratio: 4/3) { ... }
```

Prefixele max- și min- se aplică și acestor caracteristici, așadar, pentru a crea o interogare în care ați dori să vizati doar porturile de vizualizare —nu dispozitivele—care sunt, în prezent, într-un raport de cel puțin 8/5, ați folosi aspectul min- funcție de raport/ min-aspect-ratio:

```
@media screen and (min-aspect-ratio: 8/5) { ... }
```

O modalitate rapidă de a stabili ce raporturi de aspect sunt mai mari decât altele, este împărțirea primei cifre la a doua; un raport de 4/3 este de 1.333, în timp ce 8/5 funcționează până la 1.6. Cu cât rezultatul este mai mare, cu atât este mai mare raportul. Exemplul prezentat aici se aplică numai ecranelor pentru care acel număr este 1.6 sau mai mare. Probabil, mai utilă decât figura foarte precisă a unui raport de aspect este orientarea ecranului – indiferent dacă este în *modul portret* (înălțimea este mai mare decât lățimea) sau în *modul peisaj/landscape* (lățimea este mai mare decât înălțimea). Puteți interoga acest aspect cu ajutorul funcției de orientare. Această caracteristică este utilă, în special pentru dispozitivele portabile, așa cum sunt telefoanele și tabletele, care pot schimba orientarea des:

```
@media screen and (orientation: portrait) { ... }
```

Prefixele *max-* și *min-* nu se aplică acestei caracteristici, din motive evidente.

Combinarea și Negarea Interogărilor Media/Media Queries

Puteți adăuga o serie de enunțuri logice prin utilizarea repetată a cuvântului cheie și a cuvintelor cheie cu interogări suplimentare pentru mass-media. Când și dacă se utilizează, regulile sunt aplicate numai dacă toată logica interogării/ query este adevărată. De exemplu, pentru a testa dacă o fereastră de vizualizare este atât în modul peisaj, cât și cu o lățime de cel puțin 800em, puteți utiliza acest cod:



```
@media all and (orientation: landscape) and (min-width: 800em) { ... }
```

Defapt, puteți simplifica și mai mult: tipul de suport media este starea implicită, deci îl puteți lăsa în afara interogărilor, dacă acest lucru este, într-o primă instanță, inutil:

```
@media (orientation: landscape) and (min-width: 800em) { ... }
```

Puteți crea o serie de interogări media și puteți aplica regulile, dacă oricare dintre ele este adevărată, folosind separatorul de virgule pentru a acționa ca un operator. În acest exemplu, regulile sunt aplicate dacă dispozitivul este în modul peisaj sau lățimea ferestrei este de cel puțin 800em:

```
@media (orientation: landscape), (min-width: 800em) {}
```

În cele din urmă, puteți inversa logica unei interogări media folosind operatorul *not*; Adăugat la începutul interogării, acest operator arată că regulile sunt aplicate dacă logica interogării este falsă. În acest cod, regulile sunt aplicate oricărui dispozitiv care are un raport de aspect/aspect ratio diferit de 8/5:

```
@media not all and (device-aspect-ratio: 8/5) {}
```

Găsirea echilibrului pentru interogările media aferente site-ului dvs. poate fi un act destul de complex care implică multe variabile diferite, dar rezultatul final poate fi extrem de satisfăcător.

O digresiune rapidă: totul despre pixeli

Pe măsură ce ne-am mutat în era multi-dispozitivelor, un aspect de luat în considerare, dincolo de dimensiunile ecranului, este rezoluția acestuia. Multe dispozitive noi, în special smartphone-uri, sunt livrate acum cu ecrane de înaltă rezoluție, așa că, dacă doriți să oferiți o experiență optimă tuturor, trebuie să luați în considerare modul în care imaginile și alte obiective vor fi afișate pe aceste ecrane. Cu toate acestea, înainte de a intra în acest subiect, trebuie să abordăm, pes



curt, tema pixelilor. Pentru a vorbi despre interogări media care depend de rezoluție, mai întâi, trebuie să definim câțiva termeni, începând cu cei referitori la pixeli (presupunem că cititorii știu ce este un pixel). *Numărul fizic de pixeli* al unui ecran este numărul de pixeli conținut în lățimea ecranului. De exemplu, un ecran de 1280×800 are un număr fizic de pixeli de 1280 de-a lungul marginii sale largi. iPhone 3GS, în poziția sa implicită (portret), are un număr fizic de pixeli de 320, în timp ce iPhone 4 și mai sus, utilizând afișajul “retină” are un număr de 640.

Densitatea de pixeli a unui ecran este măsurată, în mod traditional în *pixel per inch* (PPI)—defapt, dacă preferați sistemul metric, puteți folosi în schimb, *pixeli per centimetri* (PPC), dar voi respecta măsura imperial pentru această descriere. Măsura imperial este uneori cunoscută sub denumirea de *puncte per inch/ dots per inch* (DPI), chiar și de software-ul popular de editare a imaginilor pe desktop, iar cei doi termeni—PPI și DPI—pot fi interschimbabili. Indiferent, densitatea pixelilor este o măsură a numărului de pixeli care pot fi așezați pe o linie care se întinde pe un centimetru din ecranul dispozitivului.

Pentru marea majoritate a ecranelor utilizate până în recent, PPI este 96. Cu toate acestea, odată cu creșterea popularității smartphone-urilor, numărul PPI a crescut constant; iPhone-ul original (și toate versiunile până la 3GS) are o rezoluție de 163 ppi, în timp ce iPhone 4 și versiunile superioare au dublu față de acesta, un incredibil 326 ppi (în ciuda faptului că ecranul rămâne la aceeași dimensiune). Majoritatea ecranelor cu densitate mare au acum o unitate de pixeli virtuală suplimentară: *pixelul independent de densitate/ density-independent pixel* (DIP), denumit uneori un pixel CSS. Aceasta este o unitate relativă; un pixel fizic este egal cu orice număr de DIP-uri. Acest lucru vă permite să scalați site-urile web în sus (și în jos), în general pentru a oferi o utilizare mai bună pe ecrane mai mici.

Raportul dintre pixelii virtuali și pixelii fizici este cunoscut ca raportul pixelilor dispozitivului. Voi folosi abrevierea DPR pentru aceasta, deși este cunoscută și sub denumirea de DPPX (puncta pe pixel). Dispozitivele care nu au DIP au un DPR de 1—adică un pixel virtual la un pixel fizic. Cele mai frecvente dispozitive de înaltă rezoluție în timp ce scriu acest lucru (cum ar fi Samsung Galaxy S III sau iPhone 5) au un DPR de 2—doi pixeli virtuali la unul fizic. Figura de mai jos ilustrează acest lucru. În stânga este un bloc de 2×2 pixeli fizici; în centru, un ecran cu un DPR de



1.5 încăpe 9 DIP-uri în același spațiu; iar în dreapta, un ecran cu un DPR de 2 cuprinde 16 DIP-uri în spațiu.



Rezoluție Ecran Media Queries

După cum tocmai ați văzut, multe dispozitive sunt livrate acum cu ecrane de înaltă rezoluție. Desi, cu siguranță, este un lucru pozitiv, aceste ecrane au unul sau două dezavantaje, în principal în ceea ce privește afișarea graficii. De exemplu, JPG, PNG, ..., sunt serii de puncte colorate diferite, care sunt mai explicit legate de pixeli, care sunt salvate cu o setare PPI. Astfel, un grafic bitmap care arată bine pe un ecran cu un DPR scăzut va arăta destul de slab pe un dispozitiv cu unul ridicat. Asta înseamnă că, cel mai probabil, veți dori să utilizați hărți de biți PPI mai mari pentru ecrane cu rezoluție mai mare, motiv pentru care există funcții media pentru a face exact acest lucru. De fapt, sunt disponibile două funcții media.

Primul și cel mai utilizat, ca urmare a popularității, WebKit pe dispozitivele mobile, este o caracteristică proprietară numită – `webkit-device-pixel-ratio`. Această caracteristică – împreună cu prefixele `max` și `min` aferente – vă permite să vizați DPR-ul unui dispozitiv. De exemplu, iată cum puteți adăuga o imagine de înaltă rezoluție pe dispozitivele cu un DPR de cel puțin 2:

```
E { background-image: url('foo.png'); }
@media screen and (-webkit-min-device-resolution: 2) {
  E { background-image: url('foo-hi-res.png'); }
}
```

Toate par destul de simple, nu? Greșit. `-webkit-device-pixel-ratio` este, după cum a fost menționat, o extensie CSS parte WebKit și nu face parte din specificațiile Media Queries. Specificația conține defapt, o interogare similară, `resolution/ rezoluție`.



Această interogare îmbunătățește imediat raportul *-webkit-device-pixel-ratio*, fiind mai flexibilă în valorile pe care le acceptă; de exemplu, puteți utiliza unitatea dpi. Amintiți-vă, o rezoluție standard a monitorului are un DPI (sau PPI) de 96; situație în care, puteți viza acei monitoare cu acest cod:

```
@media screen and (resolution: 96dpi) {
  E { background-image: url('foo.png'); }
}
```

Puteți viza și ecrane cu rezoluție mai mare, înmulțind numărul standard, 96, cu DPR. Spre exemplu, următoarea interogare se aplică dispozitivelor cu un DPR de 2 sau mai mare, la fel ca în exemplul *-webkit-device-pixel-ratio* de acum câteva paragrafe:

```
@media screen and (min-resolution: 192dpi) {
  E { background-image: url('foo-hires.png'); }
}
```

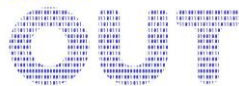
Puteți face acest lucru într-un mod și mai bun, totuși, utilizând unitatea dppx. Această unitate este echivalentă cu DPR, deci o versiune actualizată a exemplului anterior folosește acest cod:

```
@media screen and (min-resolution: 2dppx) {
  E { background-image: url('foo-hires.png'); }
}
```

-webkit-device-pixel-ratio a existat de când a fost lansat iPhone – ul în 2006, iar o mulțime de dispozitive vechi sunt disponibile, așa că, această extensie nu merge nicăieri în grabă. În acest caz, probabil că va trebui să testați pentru ambele prin extinderea interogării media cu o logică suplimentară. După cum s-a menționat anterior, utilizarea unei liste de interogări media separate prin virgule, înseamnă că regulile sunt aplicate dacă una dintre ele este adevărată:

```
@media screen and (min-resolution: 2dppx),
  screen and (-webkit-min-device-resolution: 2) {
  E { background-image: url('foo-hires.png'); }
}
```

Această logică testează dacă dispozitivul are un DPR de cel puțin 2, dar funcționează atât în browserele WebKit vechi, cât și în cele care sunt conforme cu specificațiile.



Adaptarea Dispozitivului

Browsersle mobile sunt capabile să scaleze site-urile web pentru o mai bună prezentare pe mici, dar presupun că majoritatea site-urilor web sunt destinate browserelor desktop. Această prezumție înseamnă că, în mod implicit, acestea afișează un site micșorat/ scalat pentru a-l găzdui (sau cât mai mult posibil) pe ecran. Aceasta este în general, cunoscută sub numele de fereastră de vizualizare a aspectului, iar dezavantajul vizualizării de aspect este că vizualizarea mărită poate lăsa într-adevăr foarte mici site-urile optimizate pentru mobil.

Soluția este de a seta parametrii ferestrei în CSS. Faceți acest lucru cu @viewport at-rule. În cazul în care meta tag-ul viewport utilizează parametri diferiți în atributul de conținut, @viewport folosește sintaxa CSS pentru a îndeplini aceleași funcții cu un subset de proprietăți de specialitate. Pentru a ilustra, am repetat exemplul anterior:

```
@viewport { width: 480px; }
```

Motivul pentru care această soluție este mai potrivită, este că puteți combina @viewport cu interogări media pentru a avea parametri diferiți în viewport, în funcție de dispozitivul de vizualizare. Luați în considerare acest exemplu:

```
@media screen and (max-width: 480px) {
  @viewport {
    width: 480px;
  }
}
@media screen and (min-width: 481px) {
  @viewport {
    width: device-width;
  }
}
```

Pentru ecranele cu o lățime de 480px sau mai mica, va fi afișată fereastra de vizualizare la 480px și pe ecrane mai mari, la dimensiunea lor naturală. Această abordare este cunoscută sub numele de *device adaptation/ adaptare a dispozitivului* și are avantajul de a face modificările la nivel de site mult mai ușoare prin necesitatea modificării unei singure reguli CSS, mai degrabă decât a unui număr necunoscut de pagini individuale.



Modulul 7. Subcapitolul 7.2

Design&Utilitate

Timp: 02h00m

Proiectare pentru Utilitate

Deși există o anumită libertate artistică la crearea unui site web, există și “bune practici” printre web designerii profesioniști. Ați vizitat vreodată o pagină web greu de navigat, dificil de înțeles sau pur și simplu urâtă? Un bun web designer poate privi aceste pagini și poate oferi sugestii de îmbunătățire.

În această secțiune veți învăța câteva tehnici pentru ca site-ul dvs. web să fie cât mai ușor de utilizat. Făcând site-ul dvs web mai ușor și distractiv pentru navigarea vizitatorilor, creșteți timpul pe care oamenii îl vor petrece pe site-ul dvs, precum și de câte ori vor reveni.

CSS oferă, de asemenea, metode pentru controlul modului în care documentele vor fi prezentate în alte context decât browserul desktop tradițional, cum ar fi în tipar și pe dispozitive cu lățimi mici ale ecranului. De asemenea, are reguli pentru specificarea prezentării non-vizuale a documentelor, cum ar fi modul în care acestea vor suna atunci când sunt redată de un cititor de ecran (deși acestea nu sunt bine acceptate). Foile de stil sunt, de asemenea, un instrument excelent pentru automatizarea producției, deoarece puteți schimba modul în care un element arată pe toate paginile de pe site-ul dvs., editând un singur document de foaie de stil. Foile de stil sunt acceptate într-o oarecare măsură de toate browserele moderne

Înțelegerea Utilității

Utilitatea se referă la experiența pe care o au vizitatorii atunci când vă accesează site-ul web. Include aceste calități, rezumate după cum urmează:

- *Ușurința de a învăța*, Cât de repede înțeleg oamenii cum funcționează navigarea pe site? Oare oamenii care nu au mai văzut până acum interfața



site-ului, pot să o înțeleagă suficient de bine pentru a se descurca să o folosească fără o curbă de învățare abruptă?

- *Ușurința în utilizare*, după ce o persoană a înțeles cum să navigheze pe site, cât de ușor le este să găsească efectiv informațiile de care au nevoie? Un site care are o mare ușurință în utilizare pune informațiile la îndemâna utilizatorilor, cu instrumente de căutare și navigare flexibile și puternice.
- *Memorabilitate*, cât de mult își va aminti un vizitator tipic despre site-ul dvs? Un site extrem de utilizabil rămâne în mintea vizitatorilor.
- *Gestionarea erorilor*, cât de des greșesc utilizatorii în navigarea pe site-ul dvs. și cât de ușor pot să revină la ceea ce caută? Un site extrem de utilizabil oferă mesaje de eroare utile atunci când apare o problemă, completate cu hyperlinkuri care îi ajută pe utilizatori să facă ceea ce intenționează.
- *Satisfacție subiectivă*, cât de mult le place utilizatorilor să vă viziteze site-ul? Un site extrem de utilizabil este distractiv de explorat.

Utilitatea reprezintă un aspect extrem de important în generarea unor vizitatori fideli și repetați pe site-ul dvs. O campanie publicitară poate atrage vizitatori pe site-ul dvs, însă, dacă site-ul nu este ușor de utilizat, marea majoritate a vizitatorilor nu îl vor mai accesa niciodată.

Proiectare pentru Utilitate

Proiectarea utilității ar trebui să înceapă înainte de a crea prima pagină a site-ului dvs. web, deoarece răspunsurile dvs la întrebările cheie despre scop și audiență, vor forma structura generală și aspectul site-ului dvs.

În primul rând, de ce ai nevoie de un site web? Ce speri să obții cu acesta? Un site web conceput pentru vânzarea produselor online, va arăta probabil, foarte diferit de unul care este destinat, în principal, schimbului de informații. Clarificarea obiectivelor dvs. înainte de a începe proiectarea site-ului dvs. facilitează definirea și crearea site-ului pe care vi-l doriți.

Încercați să vă stabiliți niște obiective specifice pe care să le legați de direcțiile de acțiune ale afacerii sau organizației (ex. de obiective: reducerea contactelor telefonice din partea clienților care au nevoie de sprijin, pentru a face vizitatorii să



se simtă mai încrezători în achiziționarea de produse și/ sau servicii, ...). Apoi, care este audiența pe care o vizați? “Toată lumea” este un răspuns slab la această întrebare. Nu puteți mulțumi pe toată lumea și, dacă încerci, o să sfârșiți prin a nu mulțumi pe nimeni. Gândiți-vă la caracteristicile persoanelor pe care le vizați și la ceea ce se caută într-un website.

Schițarea organizării site-ului

Următorul pas este să schițați o diagramă care arate modul în care utilizatorii vor accesa conținutul, începând cu pagina dvs. de pornire (pagina de start) în partea de sus. Orice pagină care va fi direct accesibilă din pagina de pornire va apărea la primul nivel, iar paginile care sunt subordonate acestora vor apărea la niveluri inferioare.

Iată câteva sfaturi pentru planificarea organizării site-ului:

- Decideți ce legături vor fi în bara de navigare. Aranjați legăturile în ordinea importanței, de sus în jos (sau de la stânga la dreapta). Pagina de pornire ar trebui să fie întotdeauna link-ul cel mai sus sau cel mai stâng al barei de navigare.
- Decideți ce conținut veți livra pe fiecare pagină. Eliminați orice pagini pe care vizitatorii nu le vor sau care nu furnizează informații care susțin obiectivul companiei sau organizației dvs. pentru site.
- Planificați modalități de a reduce numărul de clicuri pe care trebuie să le facă utilizatorul mediu. O modalitate de a face acest lucru este să puneți un link direct pe pagina principală către cel mai popular conținut.

Proiectarea unui șablon de pagină consecvent

Pentru o navigare mai ușoară, întregul site ar trebui să aibă un aspect coerent, cu elemente comune precum bara de navigare și titlul paginii în același loc, pe fiecare pagină. Cel mai simplu mod de a realiza acest lucru este să creați o pagină șablon și apoi să bazați toate celelalte pagini pe ea. Pagina șablon poate utiliza tabele, cadre sau diviziuni pentru aspect.



Iată câteva sfaturi pentru aranjarea paginii șablon:

- Plasați un masthead în partea de sus a paginii, conținând sigla și numele organizației dvs.
- Plasați bara de navigare în dreapta, în stânga sau în partea de sus a paginii. Barele din stânga și din partea de sus suntr cele mai frecvente, însă mulți experți în utilitate afirmă că bara de navigare așezată în dreapta, este de fapt, mai intuitivă pentru un vizitator.
- Dacă aveți un site bogat în informații, vă recomandăm să aveți mai multe bare de navigare—una în partea de sus a paginii, pentru a include categoriile principale generale și una în stânga sau în dreapta, cu o listă mai lungă de subcategorii.
- Asigurați-vă ca bara de navigare să iasă în evidență. Spre exemplu, poate fi de altă culoare, cu un fundal diferit sau înconjurat de un chenar.
- Plasați o bară de navigare text în partea de jos a paginii, astfel încât oamenii să nu fie nevoiți să deruleze pagina în sus pentru a naviga spre alte pagini.
- Dacă decideți să utilizați diferite cadre, fiți foarte atenți. Este foarte ușor să creați un set de cadre în care unele să fie pre amici și să taie conținutul plasat în el. Asigurați-vă că fiecare cadru este dimensionat corespunzător nu numai pentru conținutul implicit cu care începe, ci și pentru fiecare pagină care ar putea apărea în fiecare cadru pe parcursul întregii vizite a utilizatorului pe site-ul dvs.
- Când este posibil, faceți dimensiunea paginii flexibilă (de exemplu, lăsând o coloană de tabel sau o diviziune vertical pentru a umple spațiul rămas în fereastră). Dacă specificați o lățime fixă pentru conținutul paginii, setați-l să nu depășească 800 pixeli. În acest fel, chiar și persoanele a căror dispozitive au ecrane cu rezoluție scăzută îl vor putea vizualiza fără derulare.
- Selectați culori care să reflecte conținutul și identitatea site-ului. Nuanțele de roșu și galben creează emoție; albastru și verde emană liniște. Corpul de text trebuie să fie litere întunecate, pe un fundal deschis la culoare.
- Adaptați opțiunile de culoare la grupul țintă. Cercetătoarea Natalia Khouw relatează, spre exemplu, că bărbații preferă albastrul și portocaliul, în timp ce femeile prefer galbenul și roșul. Tinerilor le plac culorile primare



strălucitoare; persoanelor de vârstă mijlocie și de o vârstă mai înaintată, le plac culorile modeste, cum ar fi argintiul, gri – albastrui și galben pal.

- Selectați ca implicit un font simplu, lizibil, așa cum este Arial (Helvetica, Sans Serif), la o dimensiune adecvată pentru publicul dvs.

Proiectarea conținutului paginilor individuale

După crearea șablonului care va forma structura fiecărei pagini, începeți să vă gândiți la conținutul unic pentru fiecare pagină. Iată câteva sfaturi pentru crearea de pagini web eficiente:

- Folosiți propoziții scurte (20 de cuvinte sau mai puțin) și paragrafe scurte (5 propoziții sau chiar mai puține).
- Asigurați-vă că există un spațiu vertical între fiecare paragraph. În mod implicit, eticheta <p> lasă o cantitate bună de spațiu, însă unele persoane elimină sau micșorează spațiul vertical modificând stilul.
- Ori de câte ori este posibil, împărțiți informațiile în liste cu marcatori sau numerotare pentru o mai mare ușurință în utilizare.
- Potrivii lungimea paginii cu scopul acesteia. Paginile care sumarizează sau oferă posibilitatea de navighare ar trebui să fie scurte; paginile care furnizează informații detaliate despre un subiect pot fi cât de lungi este necesar.
- Păstrați articolele pe o singură pagină. Nu împărțiți textul unui articol pe mai multe pagini doar pt că o pagină pare prea lungă. Vizitatorii care doresc să tipărească articolul le va fi mult mai ușor să facă acest lucru, dacă totul este pe o singură pagină și vor aprecia că nu trebuie să facă clic pe un link pentru a vedea restul articolului.
- Împărțiți articolele lungi folosind multe titluri descriptive. Dacă articolul are mai mult de câteva pagini, includeți hyperlinkuri de marcaje în partea de sus a paginii care indică titlurile principale.
- Dacă conținutul are o adâncime de peste un nivel pe site-ul dvs., utilizați "breadcrumbs" pentru a ajuta utilizatorii să își găsească drumul înapoi către pagina de start. "Breadcrumbs" sunt o pistă de hyperlinkuri care permit



utilizatorului să facă backup la unul sau mai multe niveluri în structură, astfel: *Home > Jazz > John Coltrane*

- Limitați dimensiunea fișierelor grafice pe care le utilizați pe o pagină, astfel încât pagina să nu necesite un timp îndelungat pentru a descărca și nici să nu permit o conexiune lentă. Dimensiunea totală a fișierului pentru toate elementele grafice aferente unei paginii ar trebui să nu depășească în mod ideal 30KB. Dacă trebuie să afișați imagini mai mari, luați în considerare utilizarea de thumbnails .
- Căutați, în permanență, modalități de a reduce spațiul neutilizat. Pe o pagină care are un spațiu generos în partea dreaptă, spre exemplu, să aveți în vedere adăugarea unei casete de text care să cuprindă informații. O modalitate de a face acest lucru este utilizarea unei diviziuni cu poziționare absolută.



Modulul 7. Subcapitolul 7.3

Învățarea în regim individual

Timp: 07h00m

Exerciții

Porniți procesul vizualizând următoarea [web page](#) și corespondentul [CSS](#) la o dimensiune mica a ferestrei pentru a simula cum ar putea arăta pe un dispozitiv mobil. Ajustați o parte din CSS pentru a regla fin modul de afișare a paginii.

- Măriți dimensiunea fontului
- Măriți dimensiunea fontului h1
- Creșteți poziția de "leading"
- Adăugați o cantitate mică de margine la extremitățile paginii

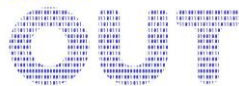
În browser (ex. Google Chrome) puteți afla cu ușurință ce dimensiune are fereastra browserului dvs. utilizând Web Inspector. Deschideți web inspector-ul în Chrome și redimensionați fereastra browserului în timp ce urmăriți colțul din dreapta sus. Dimensiunile pixelilor trebuie să apară pe măsură ce redimensionați.

Începeți să extindeți lățimea ferestrei browserului. Faceți acest lucru până ajungeți într-un punct în care designul nu mai funcționează. De exemplu, lungimea liniei poate deveni prea lungă pentru ca textul să fie ușor de citit. Sau poate pagina devine suficient de largă încât să nu mai aibă sens ca imaginile să fie afișate într-un teanc.

Înregistrați lățimea ferestrei browserului în acest moment. Acesta va fi primul tău *breakpoint*. Un breakpoint este pur și simplu un punct în care definim c ear trebui să se schimbe la design.

Creați o nouă media query în browserul dvs. utilizând dimensiunea pixelilor pe care ați înregistrat-o ca test de caracteristică media cu *lățime minimă/ min-width*.

În cadrul acestui breakpoint, modificați CSS pentru a putea face următoarele:



- Schimbați culoarea de fundal (pentru a vă ajuta să vedeți când are effect un media query)
- Ajustați dimensiunea fontului, dacă este necesar
- Ajustați marginile paginii
- Afișați fotografiile în două coloane

Redimensionați fereastra browserului pentru a găsi un alt breakpoint. Adăugați un media query în CSS și ajustați următoarele reguli:

- Schimbați culoarea de background
- Ajustați dimensiunea fontului dacă acest lucru este necesar
- Ajustați marginile paginii
- Afișați fotografiile în trei coloane
- Adăugați o lățime maximă la pagină pentru a vă asigura să lungimea liniei nu devine niciodată prea lungă.

(o posibilă soluție: [HTML](#) și [CSS](#))

Replicați procesul menționat mai sus în portofoliul personal și într-un formular de web page.

Exerciții Online

https://www.w3schools.com/css/css_rwd_intro.asp



Modulul 6. Subcapitolul 6.4

Hai să petrecem!

Time: 04h00m

Teamwork and Role-playing Activity

Participanții sunt împărțiți în echipe a câte 2 sau 3 membrii.

Fiecare membru al echipei trebuie să-i valideze colegului de echipă HTML și CSS implementat în exercițiile de auto-învățare, oferind soluții și/ sau sugestii de îmbunătățire, iar apoi, fiecare membru al echipei trebuie să își actualizeze paginile în consecință.

În cele din urmă, fiecare participant își poate prezenta propriile pagini și le poate discuta *inter-pares* (cu ceilalți participanți).



Referințe pentru dezvoltarea conținutului:

Robbins, Jennifer (2012), *“Learning web design : a beginner's guide to HTML, CSS, Javascript, and web graphics”*. Sebastopol, CA: O'Reilly.

Wempen, Faithe. (2011), *“HTML5 : step by step.”*, Sebastopol, CA: Microsoft O'Reilly distributor.

Gasston, Peter (2013), *“The modern Web : multi-device Web development with HTML5, CSS3, and JavaScript”*. San Francisco, CA: No Starch Press.



Referințe pentru dezvoltarea activității:

Robbins, Jennifer (2012), *“Learning web design : a beginner's guide to HTML, CSS, Javascript, and web graphics”* . Sebastopol, CA: O'Reilly.

Wempen, Faithe. (2011), *“HTML5 : step by step.”*, Sebastopol, CA: Microsoft O'Reilly distributor.

Gasston, Peter (2013), *“The modern Web : multi-device Web development with HTML5, CSS3, and JavaScript”*. San Francisco, CA: No Starch Press.

<http://web.simmons.edu/~grovesd/comm328/>



CODING-



> valuable outside tool for employment

IO4. CURSUL DE CODING PENTRU DETINUTI

MODULUL 7 DEZVOLTAREA APLICAȚIEI