

# WEB PROGRAMAVIMO ISTORIJA

Pirma, kas yra internetas ir kas tie puslapiai?

World Wide Web (www) – globalus kompiuterinis tinklas, šiuo metu talpinantis milijonus interneto puslapių, kuriuose patalpinta visokio pobūdžio informacija. Visą šią informaciją galima pasiekti per interneto naršykles, kurios supaprastina interneto puslapių atvaizdavimą ir įgalina kurti bei tobulinti jų dizainą.

Interneto puslapiai naudoja įvairialypės terpės technologijas, kurios leidžia atvaizduoti tekstą, grafiką, garsus, animaciją ir video informaciją. Nuo to kaip kokybiškai ir gražiai padaryta viena ar kita svetainė priklauso jos pasisekimas interneto tinkle.

Žmogui malonu lankytis tokiose interneto svetainėse, kurios stilingai apipavidalintos, neperkrautos grafikos elementais, nereikia ilgai laukti kol jas atidarys.

Šiuolaikinės technologijos leidžia kurti naujus dizaino sprendimus, kurių nebuvo įmanoma įgyvendinti net prieš keletą metų. Interneto puslapiuose atsirado daugiau dinamiškumo ir interaktyvumo. Jų kūrimu užsiima tiek pavieniai dizaineriai, tiek daugybė dizaino firmų.

Šiek tiek sausos teorijos apie tai kaip formavosi Web puslapiai:

- 1990 metais puslapiai buvo visiškai kitokie nei dabartiniai, jie buvo statiniai, ir daug panašesni į dokumentus turinčius vieną stulpelį teksto, tiesiog paprastas tekstas, kuris buvo sustruktūrintas **HTML** kalba.
- 1995 metais prisidėjo labai paprasta koncepcija - lentelės. Tai **HTML** kalboje leido sustruktūrinti kelių teksto stulpelių puslapį. Taip pat prisidėjo **JavaScript** (toliau **JS**), kuri papildė tuometinius statinius puslapius interaktyvumu.
- 1996 metais atsiradusi **Flash** technologija (jos naudojimas nutrauktas visai neseniai) papildė Web programuotojų įrankių dėžę tokiais dalykais kaip animacijos, šriftai, formos ir pan.
- 1998 metais įrankius papildė Cascading Style Sheets (toliau **CSS**) kalba, kuri labai palengvino puslapių dizaino kūrimą. Jos dėka viename faile galima aprašyti puslapio dizainą ir išvaizdą, o kitame - **HTML** kalba aprašyti visą puslapio struktūrą. Dėl to kodas tapo paprastesnis, lengviau skaitomas, taisomas ir keičiamas.
- 2000 metais **JS** tapo plačiausiai naudojama (ir vis dar tokia yra) puslapio interaktyvumo programavimo kalba. Taip pat tais metais buvo pradėtas kur kas labiau vertinti puslapių lankytojų generuojamas turinys (profiliai, komentarai, foto albumai ir pan.). Iki tol puslapiai labiau atspindėjo laikraščio formatą.
- 2007 metais įvyko labai svarbus šuolis - mobiliųjų įrenginių ekranų palaikymas, kurį įgalino tokie dalykai kaip **Bootstrap** (tai tiesiog specifinis karkasas, naudojantis tas pačias **CSS**, **HTML** ir **JS** kalbas, tačiau leidžiantis dizainui prisitaikyti prie skirtingo dydžio ekranų.)
- 2008 - dabar. Žmonės turi daugiau mobiliųjų įrenginių nei kompiuterių, todėl *responsive* dizaino palaikymas yra daugiau nei būtinas norint, kad puslapis būtų gerai vertinamas pagal naujausius standartus. Tai socialinių tinklų klestėjimo laikas, kai vietoje forumų naudojamos Facebook grupės ir panašūs dalykai.

## Kas yra Web programavimas 2016 ir su kuo jis valgomas?

Web programavimas realiai susideda iš 2 disciplinų, tai **Front-End** ir **Back-End** programavimai.

**Front-End** - tai kai programuotojas atlieka programavimo darbus puslapio srityje, kuri bus matoma vartotojui. Iš esmės tai yra meniu, turinio, iššokančių langų (*pop-ups*), registracinių formų ir kitų elementų išdėstymas puslapyje. Dizainą dažniausiai pateikia dizaineris, o programuotojas pagal tai suprogramuoja patį puslapį. Kaip turinys atrodo kompiuterio ekrane ir telefono ekrane, taipogi priklauso nuo *Front-ender 'io*, tai tarsi savotiškas programuojamas interaktyvus maketavimas.

*Front-end* kodas yra visada matomas vartotojui, jūs laisvai galite nueiti į tarkim [www.15min.lt](http://www.15min.lt), ant fono paspausti dešinį pelės mygtuką ir pasirinkti "View page source". Atsidarys langas, kuriame matysite viso puslapio kodą. **HTML**, **CSS** ir **JS** yra *Front-end* programavimo kalbos.

**Back-End** yra visiška priešingybė *Front-End'ui*, jis rašomas tokiomis kalbomis kaip PHP, Node.js, Python, Ruby, C#, ASP ir t.t. Toks kodas nėra matomas vartotojams, nes jis kompiliuojamas serveryje, o *Front-End* kodas kompiliuojamas kiekvieno vartotojo naršyklėje. *Back-End* programuotojai pagrinde rūpinasi visais loginiais puslapio aspektais, pvz. vartotojų prisijungimų saugojimas, kaip vaikšto jų generuojami duomenys, kiek, kas, kada ir už ką mokėjo (jei tai mokėjimų sistema). Visa tai saugoma duomenų bazėse.

### ***Dėstomų kalbų pagrindai.***

Jau turite nemažą suvokimą kas yra tas Web programavimas, ką kas daro ir kas už ką atsakingi. Toliau pažvelgsime į kai kuriuos dalykus, kurie bus dėstomi šiame 2 savaitių kurse.

## HTML

HTML (Hyper Text Markup Language) yra puslapio struktūrizavimo kalba. Su ja galime susidėlioti kokie elementai bus mūsų puslapyje, koks elementas bus kokiam. Tarkim, kad kontaktinėje formoje bus netik vartotojo vardas ir pavardė, bet dar ir telefono numeris ir laukelis komentarui.

### HTML kodo pavyzdys:

[http://www.w3schools.com/html/tryit.asp?filename=tryhtml\\_default](http://www.w3schools.com/html/tryit.asp?filename=tryhtml_default) interaktyvus pavyzdys.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>This is a Heading</h1>
    <p>This is a paragraph.</p>
  </body>
</html>
```

<> skliausteliais apskliausti žodžiai/raidžių deriniai yra vadinami **Tag'ais**, jie bus kaip plytos statomame puslapyje. Skirtingi tagai atlieka skirtingas funkcijas, sukuria skirtingus elementus. Vieni tagai užsidaro patys, kitus reikia uždaryti. <p> yra paragrafo atidaromasis tagas, jo viduje rašomas tekstas atrodys ir bus vertinamas kaip paragrafas. Pasibaigus tekstiniam paragrafui rašomas </p> tagas, kuris nurodo, kad čia yra paragrafo pabaiga.

<!DOCTYPE html> tagas užsidaro pats, jis rašomas pačioje pradžioje ir nurodo naršyklei kokia HTML versija bus rašomas puslapis. Dabar yra nurodoma HTML 5 versija, ji plačiausiai naudojama ir pati naujausia, ją ir naudosime.

<html> </html> tagas nurodo, kad jo viduje bus rašomas HTML kodas, taigi viskas, ką rašysime tarp jų, bus traktuojama kaip HTML.

<head> </head> tago viduje yra rašomi kiti tagai kurie aprašo patį puslapį, kas į jį įeina, kokie failai yra *includinami* ir panašiai.

Jo viduje yra <title> </title> tagas, jis aprašo puslapio pavadinimą, kokį mes matome viršuje naršyklės *Tab'e*.

<body> </body> tago viduje rašomas visas kodas, kuris kuria puslapio struktūrą.

Šiuo atveju jo viduje yra <h1></h1> ir <p></p> tagai, kurie kuria turinį, matomą puslapyje.

<h1></h1> tago viduje esantis tekstas bus traktuojamas kaip antraštė, o <p></p> - paragrafas.

Daugiau apie tai tagų rūšis ir jų naudojimą galite rasti čia:

[http://www.w3schools.com/html/html\\_intro.asp](http://www.w3schools.com/html/html_intro.asp)

# CSS

## CSS kodo pavyzdys:

```
.raudonas-paragrafas {  
    color: red;  
}
```

```
#css-pavadinimas {  
    color: green;  
    text-transform: uppercase;  
}
```

```
<p class="raudonas-paragrafas">  
    <div id="css-pavadinimas">  
        CSS  
    </div>
```

(Cascading Style Sheets) – kalba, skirta nurodyti interneto naršyklėms kaip pateikti internetinėje svetainėje esantį turinį. Nuo HTML ši kalba skiriasi tuo, kad HTML nurodo KOKS turinys yra pateikiamas, o

```
<div id="css-pavadinimas">  
    CSS  
</div>
```

nurodo KAIP pateiktį tą turinį.

```
</p>
```

CSS buvo išleista 1996 metais, o nuo 1998 ši kalba tapo pripažinta kaip internetinių svetainių išvaizdos pateikimo standartas.

Vienas iš šios kalbos kūrimo tikslų buvo suteikti galimybę svetainės turinį ir išvaizdą laikyti skirtingose vietose – tai labai supaprastina svetainių kodo skaitomumą. Todėl dauguma svetainių turi .html failus, kuriuose laikomas turinys, ir .css failus, kuriuose laikomas išvaizdos aprašymas CSS kalba. CSS visada naudojama kartu su HTML. Internetinių svetainių kode dažniausiai yra matoma, kad turiniui yra suteikiamas ne tik HTML sintaksės žymėjimas, bet ir klasės arba ID atributas. Klasių ir ID atributai gali būti susieti su CSS parametrais – naršyklė nustato kokią išvaizdą suteikti elementui pagal tai, kokie išvaizdos nurodymai yra priskiriami tam tikrai klasei arba tam tikrai ID atributui. CSS (ir ne tik) sintaksėje HTML klasės yra žymimos tašku (.), o ID yra žymimi grotelėmis (#).

## JavaScript

```
function papasakoti_apie_js() {  
    alert("JavaScript – interpretuojamoji programavimo kalba, skirta atlikti logines operacijas  
    ir turinio manipuliacijas interneto naršyklėje. Pirmoji JavaScript versija buvo sukurta vos  
    per 10 dienų 1995 metais. Ši kalba buvo pristatyta visuomenei kaip paprastesnė Java  
    programavimo kalbos versija, skirta platesnei auditorijai ir alternatyva Microsoft  
    sukurtai Visual Basic kalbai. Tačiau pavadinimas JavaScript šiai kalbai duotas tik dėl  
    marketingo tikslų (pradinėse versijose ši kalba vadinosi LiveScript), o iš techninės pusės  
    su Java programavimo kalba JavaScript turi mažai ką bendro. JavaScript funkcijos, kaip  
    ir CSS, gali būti rašomos ir atskiruose nuo HTML failuose (.js failuose), ir pačiuose  
    HTML failuose. Kaip ir tradicinės programavimo kalbos, ši kalba leidžia rašyti funkcijas,  
    kurios atlieka nustatytus skaičiavimus arba veiksmus ir gražina vartotojui (arba svetainei)  
    rezultatą. Ši kalba palaiko visus standartinius turinio formatus, objektinio programavimo  
    principus, imperatyvinį programavimą bei įprastąsias išraiškas.");  
    return true;  
}
```

## PAVYZDŽIAI

<http://www.vilniuscoding.lt>

- HTML

- Visi tekstai
- Visos nuotraukos
- Visos nuorodos
- Logotipas
- Antraštės
- Formos

- CSS

- Visos spalvos
- Atstumai tarp paragrafų ir svetainės dalių
- Mirksintis stulpelis puslapio viršuje
- Barzdotos dėdė, kurio nuotrauka nejuda kai važiuojame žemyn (tai vadinama „parallax“ efektu)
- Viršutinis meniu
- Mobilio svetainės išvaizda (pabandykite sumažinti naršyklės dydį – svetainė pagalvos, kad naršote mažesniame ekrane ir prisitaikys prie naršyklės dydžio)

- JavaScript

- DUK puslapyje paspaudus ant klausimo – atsakymas išslenka
- Google maps atvaizdavimas
- Teksto pabrėžimas užvedus pele ant meniu punkto

<http://www.delfi.lt>

- HTML

- Visi tekstai
- Visos nuotraukos
- Visos nuorodos

- CSS

- Nuotraukų dydžiai
- Atstumai tarp elementų
- Spalvos
- Tekstų dydžiai

- JavaScript

- Naujienų slankikliai (*slideriai*)

## Kas „kieto“ sukurta su HTML/CSS/JavaScript:

<http://www.overthetinyhills.com/>  
<https://babadum.com/>  
<http://histography.io/>  
<http://www.fritzo.org/keys>  
<http://archeologie1418.culture.fr/en>  
<http://alfredservice.com/>  
<http://www.reuters.tv/>  
<http://www.natgeoeat.com/>

### Labai vertingi ir informatyvūs video apie web programavimą:

<https://youtu.be/pB0WvcxTbCA> nupasakoja kokie įrankiai yra prieinami *Front-end'e* ir *Back-end'e* (nesijaudinkite, jums visko žinoti nereikia).

<https://github.com/dypsilon/frontend-dev-bookmarks> labai daug vertingų nuorodų apie tai, kaip išmokti būti *Front-end* programuotoju.

### Praktinė užduotis.

Užduoties tikslas – ištaisyti mažas klaidėles, kurios yra paliktos mūsų atsiųstame puslapio kode. Kad paleistumėte užduoties puslapį – atidarykite index.html failą, kurį rasite atsiųstame aplanke, per savo naršyklę. Tai galite padaryti su „Open with...“ funkcija (dešinio pelės mygtuko paspaudimas ant index.html failo -> Open with funkcija -> pasirinkite savo naršyklę). Arba galite atsidaryti naują naršyklės langą ir pele užtempti index.html failą į jį.

Kad galėtumėte koreguoti .html .css ir .js failus – atsidarykite juos su bet kokia teksto redagavimo programa. Tačiau rekomenduočiau Sublime Text 3 arba Notepad++ - tai programavimui skirtos nedidelės programėlės, kurios padės lengviau susigaudyti programų failuose.

Taisytini dalykai puslapyje:

1. Puslapio fonas keistokas. Geriau būtų padaryti, kad nebūtų jokio fono.
2. Antraštės „Labas“ spalva keistoka – galbūt geriau būtų uždėti labiau „standartinę“.
3. Sąrašė esantis tekstas labai didelis. Tikriausiai reikėtų jį sumažinti.
4. Ne visi sąrašo elementai apdoroti tinkamai. Įdomu kas ne taip?
5. Puslapyje esanti nuoroda kažkodėl tik ant vienos raidės – reikėtų daugiau (gal ne viso teksto).
6. Po lentelės esančio paragrafo šriftas gana mažas – reikėtų padidinti.
7. Po lentelės esantis paragrafas lyg ir nematomas – kaip pataisyti?

Puslapyje įdėjome JavaScript failą, kuris patikrins ar visi punktai sutvarkyti. Jeigu kas nors bus nesutvarkyti – jums apie tai praneš. Jeigu viskas bus sutaisyta – taip pat sužinosite.

BEJE – jauniems programišiams: jeigu kur nors užstrigote ar tiesiog nenorite taisyti visų punktų – galbūt yra kiti būdai kaip apeiti klaidų tikrinimo mechanizmą? Pabandykite.

**Sėkmės.**