



VILNIUSCODINGSCHOOL

# WEB kūrimo mokymų PRE-KURSAS

# Ivadas

Pre-kursas skirtas Vilnius Coding School studentams, pasirinkusiems kursą „**WEB kūrimo pagrindai**“.

Tai savaitės trukmės programa, skirta savarankiškam darbui namuose, prieš prasidedant intensyviems WEB kūrimo mokymams. Pre-kurso medžiagą sudaro teorinė medžiaga, praktiniai uždaviniai bei programinės įrangos instrukcijos, padėsiančios įsisavinti programavimo pradmenis bei kuo geriau pasiruošti savarankiškai iki mokymų pradžios.

Pre-kursas orientuotas į jokios programavimo patirties neturinčius studentus. Turintiems bent jau minimalią patirtį, vis tiek pravartu pasikartoti pradmenis. Labai svarbu prisijungus į pirmąją paskaitą užduoti klausimus, kurie kilo skaitant pre-kursą.

# WEB programavimo istorija

## **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 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 struktūrinė 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 kur kas labiau pradėtas 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.
- Web technologijos atsinaujina tokia didele sparta, jog neįmanoma spėti visko išmokti, todėl Web programuotojai moka pagrindines kalbas ir renkasi papildomas kalbas, kurios padėtų atlikti jų darbą.

# Kas yra WEB programavimas?

Web programavimas susideda iš 2 disciplinų – Front-End ir Back-End programavimai. Taip pat galite išgirsti trumpinį Full-Stack programavimas, kuris apjungia abi prieš tai minėtas disciplinas.

## **Front-End**

Kuomet 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 jį suprogramuoja patį puslapį. Kaip turinys atrodo kompiuterio ekrane ir telefono ekrane, taipogi priklauso nuo Front-End'io, tai tarsi savotiškas programuojamas interaktyvus maketas.

Front-End kodas yra visada matomas vartotojui, jūs laisvai galite nueiti į tinkamą [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 programavime naudojamos kalbos.

## **Back-End**

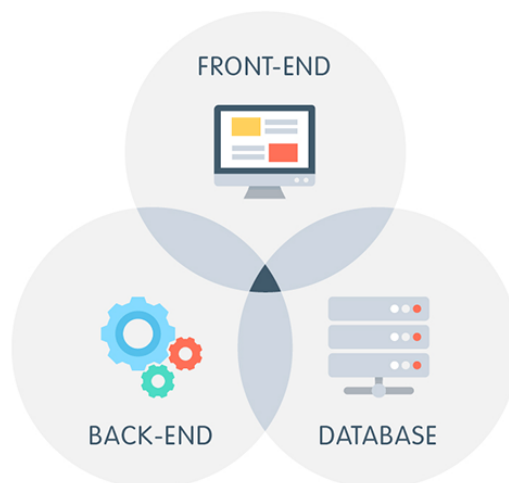
Visiška priešingybė Front-End'ui. Jis rašomas tokiomis kalbomis kaip PHP, Node.js, Python, Ruby, C#, ASP ir t.t. Toks kodas nėra matomas vartotojams, nes jis kompiliuojamas serveryje, o Front-End kodas kompiliuojamas kiekvieno vartotojo naršyklėje. Back-End programuotojai rūpinasi visais loginiais puslapio 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.

## Full-Stack

Full-stack programavimas apima matomą ir nematomą svetainės dalis. Full-stack programuotojai puikiai išmano tiek Front-end, tiek Back-End dalis.

Mokymų metu išmoksite susikurti statinį Front-End puslapį, papildysite jį interaktyviais sprendimais, pvz. nuotraukų galerija ar karusele. Taip pat išmoksite išsaugoti vartotojų duomenis duomenų bazėje, sukursite prisijungimo sistemą.

## FULL-STACK DEVELOPMENT



*Full-Stack programavimo struktūra*



# Programavimo kalbos ir technologijos

## HTML

HTML (Hyper Text Markup Language) yra puslapio struktūrizavimo kalba. Tai pirmoji pagrindinė kalba, kurią turite išmokti, norėdami būti Front-End programuotojais. HTML yra puslapio struktūra ir visas jos pamatas, todėl labai svarbu parašyti gerą ir semantišką kodą. Su šia kalba galime susidėlioti elementus mūsų puslapyje, pavyzdžiui turėsime sekciją su antrašte ir paragrafu, o po jais eis tėvinis blokas, kuriame talpinsime dar tris blokus (vaikus).

*Nuorodos susipažinimui:*

- [https://www.w3schools.com/html/html\\_intro.asp](https://www.w3schools.com/html/html_intro.asp)
- <https://www.youtube.com/playlist?list=PL4cUxeGkcC9ibZ2TSBaGGNrgh4ZgYE6Cc>

## Schema

Schema yra HTML struktūros papildymas suteikiantis daugiau semantikos ir daugiau informacijos paieškos vorams. Google, Bing ir kiti paieškos varikliai turi vadinamuosius vorus, kurie naršo po jūsų puslapį ir indeksuoja jame esančią informaciją. Schemas pagalba galime šiems vorams konkrečiai nurodyti kur ir kokia informacija yra pateikta mūsų puslapyje.

*Keletas nuorodų susipažinimui:*

- <http://schema.org/>
- <http://schema.org/docs/gs.html>
- <http://schema.org/Article>

## CSS

Antroji pagrindinė kalba yra CSS (Cascading Style Sheets). CSS išleista 1996 metais, o nuo 1998 pripažinta interneto puslapių stilizavimo standartu.

Jei su HTML kalba kuriama puslapio struktūra, tai su CSS toji struktūra yra stilizuojama, t.y. nustatoma teksto spalva, teksto dydis, tarpai tarp eilučių, tarpai tarp raidžių, fono spalva, rėmeliai ir t.t. Be CSS mūsų puslapis būtų juodai baltas, ne itin patrauklus ir greičiausiai prarastume visus atėjusius lankytojus.

*Kelios nuorodos susipažinimui:*

- [https://www.w3schools.com/css/css\\_intro.asp](https://www.w3schools.com/css/css_intro.asp)
- <https://www.youtube.com/playlist?list=PL4cUxeGkcC9gQeDH6xYhmO-db2mhoTSrT>



*CSS kalbos logotipas*

## **Bootstrap**

Rašydami CSS taip pat projektuose panaudosime Bootstrap framework'ą - karkasą, kurio pagalba galime lengvai sukurti puslapio išsidėstymą, mygtukus ir t.t.

Bootstrap dėka lengvai susikursime 3, 4 ar net 12 stulpelių vienoje eilutėje, tuo pačiu prisitaikdami prie mobilių įrenginių, kuriuose šis stulpelių skaičius keisis.

*Nuoroda į oficialų puslapį:*

- <http://getbootstrap.com/>



## Sass

Sass trumpai tariant yra CSS preprocesorius. Sass leidžia daug paprasčiau ir greičiau rašyti CSS kodą, papildo jį naujais dalykais - kintamaisiais, mixin'ais, funkcijomis ir t.t. Sass naudojamas tik rašant kodą ir yra iš karto kompiliuojamas į CSS kodą. Šiam kompiliavimui atlikti mes naudosime programą Gulp, kurią įdiegsime ir susiesime su teksto editoriaus programa jau kurso metu.

*Kelios nuorodos susipažinimui:*

- <https://sass-lang.com/>
- <https://gulpjs.com/>

## JavaScript

JavaScript yra populiariausia programavimo kalba pasaulyje, interneto puslapiuose naudojama interaktyvumui kurti ir įvairius apskaičiavimus atlikti. Išmokus šios kalbos programavimo principus, nesunkiai galima imtis kitos programavimo kalbos, kadangi ji atitinka kitose programavimo kalbose esančius standartus – kintamuosius, funkcijas, ciklus ir t.t.

Tai yra trečioji iš pagrindinių kalbų, kurias būtina mokėti norint įvardinti save kaip Front-End programuotoją. JavaScript yra labai plati, turi daugybę bibliotekų ir framework'ų, bet prieš naudojant juos - būtina išmokti standartinę JavaScript kalbą. Kurso metu susipažinsime su pamatinėmis JavaScript funkcijomis, lavinsime loginį mąstymą, kuris yra būtinas kiekvienam programuotojui.

*Kelios nuorodos susipažinimui:*

- <https://www.w3schools.com/js/default.asp>
- <https://www.youtube.com/playlist?list=PL4cUxeGkcC9i9Ae2D9Ee1RvylH38dKuET>

## SQL

Antroji kurso pusė yra skirta Back-End programavimui. Pirmiausia, susipažinsime su SQL (Structured Query Language), kuri yra duomenų bazių valdymo kalba. Šio kurso metu dirbsime su MySQL duomenų baze, kuri naudoja SQL programavimo kalbą. Trumpai tariant SQL kalba yra skirta rašyti užklausas serveriui ir tokiu būdu kurti, trinti ir atnaujinti duomenis esančius duomenų bazėse.

*Nuoroda susipažinimui:*

- <https://www.w3schools.com/sql/default.asp>

## PHP

PHP yra programavimo kalba veikianti serveryje, t.y. PHP kalbos pagalba parašytas kodas yra siunčiamas į serverį ir ten apdorojamas. PHP yra viena iš populiariausių Back-End programavimo kalbų. Didžiąją dalį programavimo principų galėsite naudoti iš JavaScript kalbos (PHP kalboje skirstis tik sintaksė, tačiau logika išliks ta pati).

Šios kalbos pagalba HTML dokumente esančią struktūrą padarysime dinamine, kai informacija bus pateikiama ne iš HTML dokumento, o iš duomenų bazėje esančių duomenų.

*Kelios nuorodos susipažinimui:*

- <http://php.net/manual/en/intro-what-is.php>
- <https://www.w3schools.com/php/default.asp>

## Wordpress

Wordpress (dar kitaip vadinama tiesiog WP) yra populiariausia turinio valdymo sistema (TVS, angliškai - CMS). Ji yra lengvai suprantama ir yra labai plačiai modifikuojama. Šiame kurse išmoksime savo sukurtą statinį puslapį „užkelti“ ant šios valdymo sistemos naudodami PHP programavimo kalbą bei sparčiai sukurti tinklalapius panaudojant esamus šablonus.

*Kelios nuorodos susipažinimui:*

- <https://wordpress.org/>
- <https://wordpress.org/about/>
- [https://codex.wordpress.org/Theme\\_Development](https://codex.wordpress.org/Theme_Development)

## **Git**

Kodo versijavimas išpopuliarėjo dar visai neseniai, tačiau dabar laikomas standartu ir naudojamas itin plačiai. Dažniausiai versijavimo sistemas naudoja tuomet kai dirbama komandoje, t.y. kad du ar daugiau programuotojų, galėtų dirbti prie to pačio projekto, tų pačių failų, o versijavimo sistema jiems įkeliant atnaujinimus sujungtų abiejų programuotojų kodus į vieną ir nepanaikintų nei vieno iš jų kodo.

Tačiau kartais versijavimas naudojamas ir vieno programuotojo. Paprastai tai itin patogiu, jeigu po kelių savaičių suprantame, kad ankščiau sprendimą buvome atlikę geriau ir galime grįžti prie buvusio sprendimo.

Pati populiariausia kodo verijavimo sistema yra Git.

*Nuoroda susipažinimui:*

- <https://www.atlassian.com/git/tutorials/what-is-git>

# Programinė įranga

## Kodo redaktorius (instaliuoti prieš kursą)

Kurso metu naudosime populiariausią kodo redaktorių, Visual Studio Code - <https://code.visualstudio.com/download>

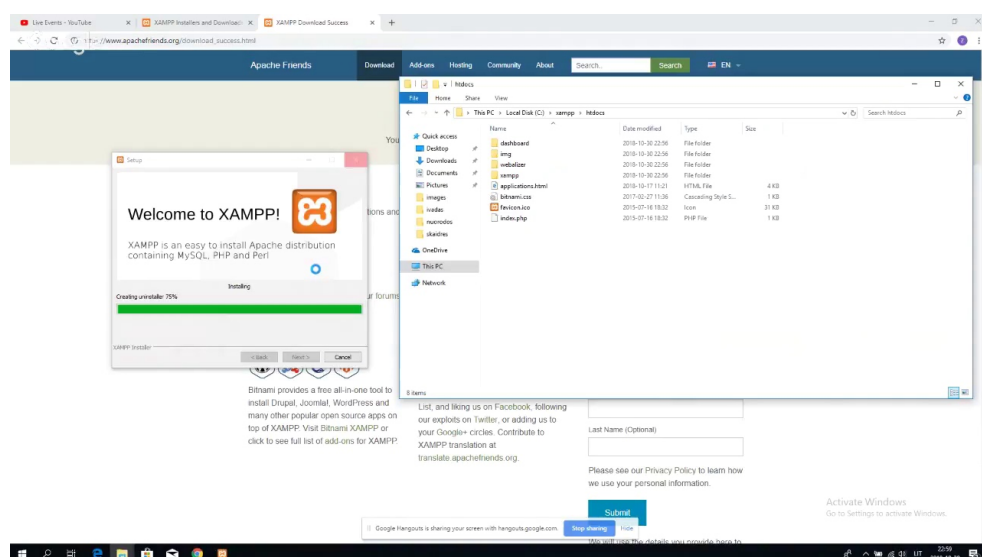
Visual Studio Code programoje rašysime ir redaguosime visą savo kodą, parašytą su HTML, CSS, JavaScript, SQL ar PHP.

## Lokalus serveris (instaliuoti prieš kursą)

Front-End kalbos veikia naršyklėje ir joms nereikalingas serveris, tačiau Back-End programavimui mums būtinas serveris, kurį galime nusipirkti iš šias paslaugas teikiančių įmonių arba susikurti patys.

Dažniausiai programuotojai savo kompiuteriuose turi susikūrę lokalų serverį, kuriame saugo visus duomenis. Populiariausi „viskas viename“ serveriai:

- <https://www.apachefriends.org/index.html> - XAMPP, veikiantis WIN, MAC ir Linux operacinėse sistemose. Paruošėme instaliavimo instrukcijas:



- <https://www.mamp.info/en/> - MAMP, veikiantis WIN ir MAC operacinėse sistemose, patogesnis pradedantiesiems vartotojams, lengviau valdomas nei XAMPP.

## Versijavimas

Versijavimui savo kompiuteriuose įsidiekite Git programą. Siųstis iš <https://git-scm.com/>

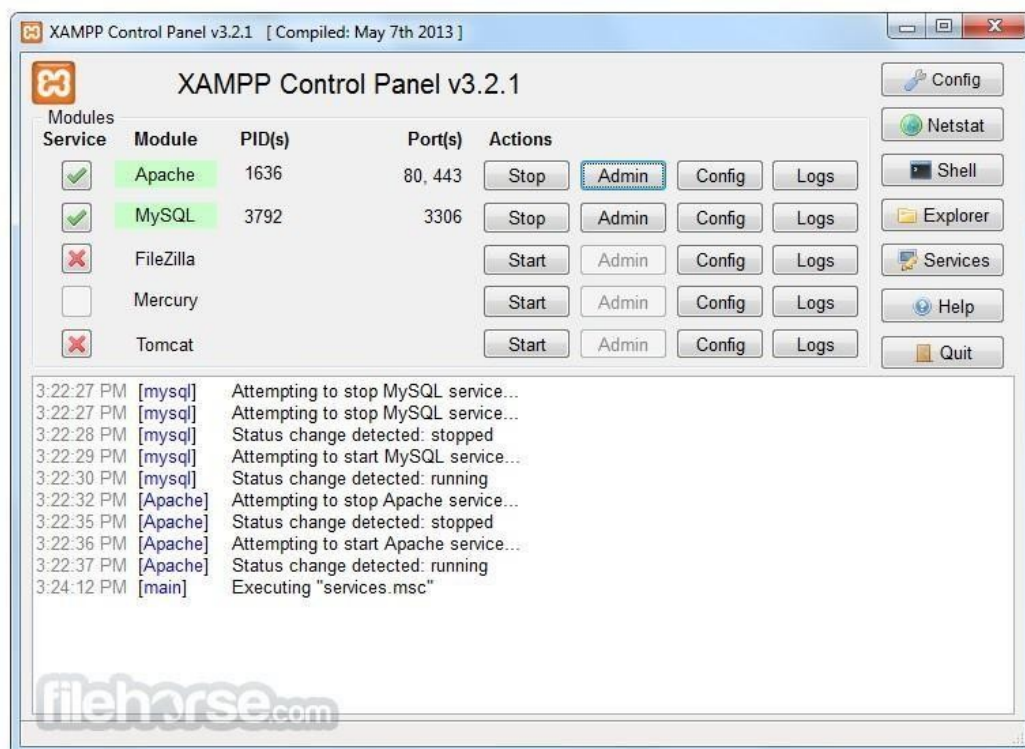
Jei naudojate Mac kompiuterį, greičiausiai jau turite Git instaliuotą. Atsidarius Terminal programą parašykite: `git --version` ir paspauskite Enter. Jei Git įdiegtas, rodys jo versiją. Tuo atveju jei Git trūksta, iššoks atskiras langas, kuris leis jums suinstaliuoti naujausią versiją.

## Užduotys

### Patikrinkite ar veikia serveris

Ne tik įrašykite lokalų serverį į savo kompiuterį, bet ir patikrinkite ar jis veikia.

Jeigu pasirinkote Xampp, įsijungę programos langą pirma įjunkite Apache ir tuomet MySQL. Jeigu ties abiem moduliais matote žalią šviesą – serveris veikia.



Jei instaliavote MAMP, paspauskite Start mygtuką. Ties Apache ir MySQL turėtumėte matyti žalios spalvos rutuliukus - viskas veikia.

Nuėjus į C://Xampp arba C://MAMP aplanką savo kompiuteryje, rasite aplanką pavadintą „htdocs“, kurį taip pat galite rasti nuėję į failų tvarkyklę, ten kur įrašėte programą arba Xampp programos lange paspaudus „Explorer“ mygtuką.

Šioje htdocs direktorijoje susikurkite „test.php“ failiuką ir jo viduje įrašykite tokį kodą:

```
1 <?php echo "Hello World!"; ?>
```

Naršyklėje įveskite adresą localhost/test.php



Jeigu PHP failas jums išveda „Hello World!“, vadinasi serveris veikia. Kitu atveju kažkur yra palikta klaida.

# Projektas

Visą mokymų metu kursite savo projektą, kurį kurso pabaigoje teks pristatyti. Norime, jog šis procesas Jums būtų smagus ir suteiktų kaip įmanoma daugiau motyvacijos, todėl rekomenduojame žemiau pateiktus metodus, galvojant apie projekto mintį ir dizainą:

- Jeigu jūs jau turite darbdavį - paprašykite darbdavio jų kurto puslapio Figma dizaino failo (baigiasi su galūne .fig). Tokiu būdu ir jums bus lengviau naujoje darbovietėje ir darbdavys žinos kokias sritis po kursų reikėtų tobulinti labiausiai.
- Jeigu neturite darbdavio arba darbdavys nenori Jums siųsti jokio dizaino failo, pagalvokite kokį projektą kursite, kokia bus pagrindinė jo paskirtis. Jeigu visgi nepavyksta sugalvoti – paklauskite jums artimų žmonių, gal jie norėtų turėti kokį nors puslapį. Kai sugalvosite puslapio pagrindinę mintį ir kas bus tame puslapyje susiraskite jums artimiausią .fig dizaino failą (nuoroda pateikta apačioje).
- Jeigu visgi nesugalvojate projekto kurį norite kurti – susikurkite savo asmeninį puslapį/CV. Tokiu būdu po kursų turėsite savo CV internete ir galėsite šį puslapį tobulinti toliau.

Galite išsirinkti ir kelis variantus. Pirmąją dieną galėsime prisėsti kartu ir išsirinkti, kuris variantas yra geresnis Jums. Dizainus rinkitės iš šio tinklalapio:

<https://www.freefigmatemplates.com/website-designs-and-apps>

Iki pirmosios paskaitos įsirašykite Visual Studio Code, Git, XAMPP arba MAMP bei patikrinkite ar Jūsų serveris yra veikiantis. Taip pat susiraskite dizainą kurį kursite ateinančias keturias savaites.

Linkime sėkmės mokymuose!