

## **JAVASCRIPT PROGRAMUOTOJO MODULINĖ PROFESINIO MOKYMO PROGRAMA**

*(Programos pavadinimas)*

Programos valstybinis kodas ir apimtis mokymosi kreditais:

P42061303, P43061303 – programa, skirta pirminiam profesiniam mokymui, 60 mokymosi kreditų  
T43061306 – programa, skirta tęstiniam profesiniam mokymui, 50 mokymosi kreditų

Kvalifikacijos pavadinimas – jaunesnysis programuotojas

Kvalifikacijos lygis pagal Lietuvos kvalifikacijų sandarą (LTKS) – IV

Minimalus reikalaujamas išsilavinimas kvalifikacijai įgyti:

P42061303 – pagrindinis išsilavinimas ir mokymasis vidurinio ugdymo programoje  
P43061303, T43061306 – vidurinis išsilavinimas

Reikalavimai profesinei patirčiai (jei taikomi) – nėra

Kiti reikalavimai: laikomas testas stojančiojo pasirengimui mokytis nustatyti

# 1. PROGRAMOS APIBŪDINIMAS

**Programos paskirtis.** JavaScript programuotojo modulinė profesinio mokymo programa skirta parengti kvalifikuotą darbuotoją, gebantį projektuoti ir kurti vartotojo kompiuteriuose, išmaniuosiuose telefonuose, planšetėse ir kt. priemonėse naudojama programinę įrangą (angl. *front-end*), parengtą jungtis prie serverių sistemų, duomenų bazių, valdyti programavimo aplinką ir kūrimo procesą.

**Būsimo darbo specifika.** Įgijęs kvalifikaciją asmuo galės dirbti jaunesniu ju JavaScript programuotoju informacinių ir ryšių technologijų sektoriaus verslo įmonėse, kitų sektorių organizacijų informacinių technologijų padaliniuose.

Dirbama komandoje ir (arba) individualiai, galimas lankstus darbo grafikas, nuotolinis darbas. Jaunesnysis JavaScript programuotojas paprastai dirba prižiūrimas vyresniojo specialisto ir vykdo jam iš anksto suformuluotas ir pateiktas užduotis, savarankiškai fiksuoja ir pats tikrina savo darbo rezultatus, dirba ir bendrauja su kolegomis (kolegomis programuotojais, sistemų analitikais, testuotojais, priežiūros specialistais, projektų vadovais), keičiantis technologijoms nuolat atnaujina ir gilina žinias, ir tobulina savo įgūdžius. Jo darbo rezultatai tikrinami prieš integruojant į bendrą komandos darbo rezultatą. Augant specialisto patirčiai jo savarankiškumo lygmuo didėja.

Jaunesniajam JavaScript programuotojui svarbios šios asmeninės savybės: kruopštumas, atsakingumas, savarankiškumas, dėmesio koncentracija.

## 2. PROGRAMOS PARAMETRAI

Valstybinis kodas	Modulio pavadinimas	LTKS lygis	Apimtis mokymosi kreditais	Kompetencijos	Kompetencijų pasiekimą iliustruojantys mokymosi rezultatai
<b>Įvadinis modulis (iš viso 1 mokymosi kreditas)*</b>					
4000005	Įvadas į profesiją	IV	1	Pažinti profesiją.	Išmanyti JavaScript programuotojo profesiją ir jos teikiamas galimybes darbo rinkoje. Suprasti JavaScript programuotojo profesinę veiklą, veiklos procesus, funkcijas ir uždavinius. Demonstruoti jau turimus, neformalioju ir (arba) savaiminiu būdu įgytus jaunesniojo JavaScript programuotojo kvalifikacijai būdingus gebėjimus.
<b>Bendrieji moduliai (iš viso 4 mokymosi kreditai)*</b>					
4102201	Saugus elgesys ekstremaliose situacijose	IV	1	Saugiai elgtis ekstremaliose situacijose.	Išmanyti ekstremalių situacijų tipus, galimus pavojus. Išmanyti saugaus elgesio ekstremaliose situacijose reikalavimus ir instrukcijas, garsinius civilinės saugos signalus.
4102105	Sąmoningas fizinio aktyvumo reguliavimas	IV	1	Reguliuoti fizinį aktyvumą.	Išmanyti fizinio aktyvumo formas. Demonstruoti asmeninį fizinį aktyvumą. Taikyti fizinio aktyvumo formas, atsižvelgiant į darbo specifiką.
4102203	Darbuotojų sauga ir sveikata	IV	2	Tausoti sveikatą ir saugiai dirbti.	Išmanyti darbuotojų saugos ir sveikatos reikalavimus, keliamus darbo vietai.
<b>Kvalifikaciją sudarančioms kompetencijoms įgyti skirti moduliai (iš viso 45 mokymosi kreditai)</b>					
<i>Privalomieji (iš viso 45 mokymosi kreditai)</i>					
406130011	Informacinių sistemų projektavimas ir kūrimas (JavaScript)	IV	20	Projektuoti ir programuoti žiniatinklio puslapių vartotojo sąsajas.	Pateikti internetinio puslapio turinį naudojant kompiuterinę žymėjimo kalbą. Apipavidalinti internetinį puslapį naudojant pakopinius stilių šablonus ir karkasus.
				Taikyti programinės įrangos kūrimui naudojamus informatikos principus ir metodus.	Suprasti skaičiavimo sistemas. Taikyti algoritmų ir logikos mokslo pagrindus programuojant. Taikyti programinio kodo dizaino modelius programuojant. Naudoti informacinių sistemų kūrimo principus ir

					metodus programinės įrangos projektavime.
				Kurti tipinę programinę įrangą.	Konfigūruoti pagrindinius įrankius, reikalingus projekto paleidimui, ir įdiegti projektui reikalingus paketus. Diegti ir konfigūruoti įrankius kodo transformavimui, pakavimui ir optimizavimui. Analizuoti ir derinti vykdomą kodą, identifikuoti problemas ir jas taisyti. Kurti JavaScript programinį kodą ir naudoti jo sintaksę. Vykdyti programinio kodo versijavimą naudojant versijavimo įrankius. Naudoti JavaScript funkcijas. Taikyti objektinio programavimo principus programuojant. Keisti ir manipuluoti HTML elementų medį (DOM). Kurti paprastas serveryje vykdomas aplikacijas naudojant JavaScript. Kurti aplikacijas naudojant React karkasą.
				Analizuoti skirtingų tipų reikalavimus, apibūdinančius kompiuterinę programą.	Vykdyti reikalavimų peržiūros procesą naudojant vartotojo pasakojimo reikalavimų programinei įrangai formatą. Naudoti funkcinis, nefunkcinis ir techninius kompiuterinės programos reikalavimus. Testuoti programinę įrangą naudojant su JavaScript programavimo kalba suderinamus testavimo įrankius ir metodus.
406130012	Nesudėtingų duomenų bazių projektavimas ir kūrimas (JavaScript)	IV	10	Projektuoti tipines reliacines ir nereliacines (NoSQL) duomenų bazes.	Pateikti reliacinės duomenų bazės schemą. Pateikti nereliacinės (NoSQL) duomenų bazės schemą.
				Programiškai įgyvendinti ir administruoti duomenų bazes.	Diegti ir valdyti duomenų bazių valdymo sistemą. Naudoti SQL kalbą duomenų bazės užpildymui ir informacijos išrinkimui.
406130013	Programavimo aplinkos ir kūrimo proceso valdymas (JavaScript)	IV	15	Naudoti tarnybinių stočių operacines sistemas.	Suprasti serverių veikimo principus ir panaudojimo galimybes. Valdyti programinius paketus.

					Naudoti Web serverio programinę įrangą HTTP bylų viešinimui.
				Taikyti aktualias programinės įrangos kūrimo metodikas.	Suprasti Scrum proceso dalis ir komandos narių atsakomybes. Analizuoti pateiktus reikalavimus, nustatant programos atitikimą reikalavimams. Naudoti projekto eigos valdymo principus.
				Valdyti savo paties ir komandos atliekamą programinio kodo kūrimą.	Diegti ir valdyti programavimo JavaScript kalba darbo aplinką. Sėti programavimo darbų vykdymą naudojant komandinio darbų planavimo sistemas.
<b>Pasirenkamieji moduliai (iš viso 5 mokymosi kreditai)*</b>					
406130014	Testavimu ir vartotojų elgsena pagrįsto programavimo metodikų taikymas (JavaScript)	IV	5	Taikyti testavimu pagrįsto programavimo metodiką.	Rašyti testus taikant testavimu pagrįsto programavimo metodiką. Panaudoti <i>JUnit</i> karkaso metodus testuojant sukurtas JavaScript taikomąsias programas.
				Taikyti vartotojo elgsena pagrįsto programavimo metodiką.	Rašyti priėmimo testą ir JavaScript kodo dalių testus pagal vartotojo pasakojimus. Taikyti reikalavimų specifیکavimą pagal pavyzdžius.
406130015	Komunikacija su serveriu naudojant GraphQL standartą	IV	5	Apibrėžti ir naudoti GraphQL pagrindinius metodus ir funkcijas.	Suprasti GraphQL veikimo ir panaudojimo principus. Naudoti GraphQL užklausas ir schemas.
				Taikyti GraphQL Apollo įrankį aplikacijų kūrime.	Apollo GraphQL server naudojimas. React aplikacijos konfigūravimas naudojant Apollo GraphQL.
<b>Baigiamasis modulis (iš viso 5 mokymosi kreditai)</b>					
4000004	Įvadas į darbo rinką	IV	5	Formuoti darbinius įgūdžius realioje darbo vietoje.	Įsivertinti ir realioje darbo vietoje demonstruoti įgytas kompetencijas. Susipažinti su būsimo darbo specifika ir adaptuotis realioje darbo vietoje. Įsivertinti asmenines integracijos į darbo rinką galimybes.

\* Šie moduliai vykstant tęstinį profesinį mokymą neįgyvendinami, o darbuotojų saugos ir sveikatos bei saugaus elgesio ekstremaliose situacijose mokymas integruojamas į kvalifikaciją sudarančioms kompetencijoms įgyti skirtus modulius.

### 3. REKOMENDUOJAMA MODULIŲ SEKA

Valstybinis kodas	Modulio pavadinimas	LTKS lygis	Apimtis mokymosi kreditais	Asmens pasirengimo mokytis modulyje reikalavimai (jei taikoma)
<b>Įvadinis modulis (iš viso 1 mokymosi kreditas)*</b>				
4000005	Įvadas į profesiją	IV	1	<i>Netaikoma.</i>
<b>Bendrieji moduliai (iš viso 4 mokymosi kreditai)*</b>				
4102201	Saugus elgesys ekstremaliose situacijose	IV	1	<i>Netaikoma.</i>
4102105	Sąmoningas fizinio aktyvumo reguliavimas	IV	1	<i>Netaikoma.</i>
4102203	Darbuotojų sauga ir sveikata	IV	2	<i>Netaikoma.</i>
<b>Kvalifikaciją sudarančioms kompetencijoms įgyti skirti moduliai (iš viso 45 mokymosi kreditai)</b>				
<i>Privalomieji (iš viso 45 mokymosi kreditai)</i>				
406130011	Informacinių sistemų projektavimas ir kūrimas (JavaScript)	IV	20	<i>Netaikoma.</i>
406130012	Nesudėtingų duomenų bazių projektavimas ir kūrimas (JavaScript)	IV	10	<i>Netaikoma.</i>
406130013	Programavimo aplinkos ir kūrimo proceso valdymas (JavaScript)	IV	15	<i>Netaikoma.</i>
<b>Pasirenkamieji moduliai (iš viso 5 mokymosi kreditai)*</b>				
406130014	Testavimu ir vartotojų elgsena pagrįsto programavimo metodikų taikymas (JavaScript)	IV	5	<i>Netaikoma.</i>
406130015	Komunikacija su serveriu naudojant GraphQL standartą	IV	5	<i>Baigti šie moduliai: Informacinių sistemų projektavimas ir kūrimas Nesudėtingų duomenų bazių projektavimas ir kūrimas</i>
<b>Baigiamasis modulis (iš viso 5 mokymosi kreditai)</b>				
4000004	Įvadas į darbo rinką	IV	5	<i>Baigti visi JavaScript programuotojo kvalifikaciją sudarantys privalomieji moduliai.</i>

\* Šie moduliai vykdant tęstinį profesinį mokymą neįgyvendinami, o darbuotojų saugos ir sveikatos bei saugaus elgesio ekstremaliose situacijose mokymas integruojamas į kvalifikaciją sudarančioms kompetencijoms įgyti skirtus modulius.

#### 4. REKOMENDACIJOS DĖL PROFESINEI VEIKLAI REIKALINGŲ BENDRŪJŲ KOMPETENCIJŲ UGDYMO

Bendrosios kompetencijos	Bendrųjų kompetencijų pasiekimą iliustruojantys mokymosi rezultatai
Raštingumo kompetencija	Rašyti gyvenimo ir profesinės patirties aprašymą, motyvacinį laišką, prašymą, ataskaitą, elektroninį laišką. Bendrauti vartojant profesinius terminus.
Daugiakalbystės kompetencija	Vartoti pagrindines profesinės terminijos sąvokas užsienio kalba. Bendrauti profesine užsienio kalba darbinėje aplinkoje žodžiu ir raštu. Skaityti ir kurti profesinę dokumentaciją užsienio kalba.
Matematinė kompetencija ir gamtos mokslų, technologijų ir inžinerijos kompetencija	Suvokti skaičiavimo sistemas, matematinės logikos principus. Taikyti matematinę logiką grįstus algoritmus ir objektinio programavimo principus. Suvokti techninės užduoties reikalavimus ir įvertinti vartotojo reikalavimus. Parengti užduoties techninę specifikaciją.
Skaitmeninė kompetencija	Suvokti kompiuterijos ir interneto veikimo principus. Naudotis interneto paieškos ir komunikavimo sistemomis, dokumentų kūrimo ir redagavimo programomis. Rinkti, apdoroti ir saugoti reikalingą darbui informaciją. Naudotis laiko planavimo ir elektroninio pašto programomis. Rengti paslaugos ir (arba) darbo pristatymą kompiuterinėmis programomis.
Asmeninė, socialinė ir mokymosi mokytis kompetencija	Įsivertinti turimas žinias ir gebėjimus. Organizuoti savo mokymąsi. Pritaikyti turimas žinias ir gebėjimus dirbant individualiai ir komandoje. Parengti profesinio tobulėjimo planą.
Pilietiškumo kompetencija	Gebėti bendrauti su klientais ir kolegomis. Valdyti savo psichologines būsenas, pojūčius ir savybes. Spręsti psichologines krizines situacijas. Gerbti save ir kitus, savo šalį ir jos tradicijas. Prisitaikyti prie tarptautinės, daugiakultūrinės aplinkos.
Verslumo kompetencija	Suprasti įmonės veiklos tikslus ir principus, verslo aplinką. Panaudojant intuiciją, kūrybiškumą ir analitinius gebėjimus atpažinti su technologijomis susijusias naujas rinkos ir vertės kūrimo galimybes. Suprasti socialiai atsakingo verslo principus. Prisiimti atsakomybę, dirbti savarankiškai ir komandoje, planuoti savo laiką.
Kultūrinio sąmoningumo ir raiškos kompetencija	Paašškinti Europos ir kitų šalių kultūrinius skirtumus, darbo su kitų kultūrų kolegomis ir (arba) klientais ypatumus. Lavinti estetinį požiūrį į aplinką.

## 5. PROGRAMOS STRUKTŪRA, VYKDANT PIRMINĮ IR TĘSTINĮ PROFESINĮ MOKYMĄ

<b>Kvalifikacija - jaunesnysis programuotojas, IV LTKS lygis</b>	
<b>Programos, skirtos pirminiam profesiniam mokymui, struktūra</b>	<b>Programos, skirtos tęstiniam profesiniam mokymui, struktūra</b>
<i>Įvadinis modulis (iš viso 1 mokymosi kreditas)</i> Įvadas į profesiją, 1 mokymosi kreditas	<i>Įvadinis modulis (0 mokymosi kreditų)</i> –
<i>Bendrieji moduliai (iš viso 4 mokymosi kreditai)</i> Saugus elgesys ekstremaliose situacijose, 1 mokymosi kreditas Sąmoningas fizinio aktyvumo reguliavimas, 1 mokymosi kreditas Darbuotojų sauga ir sveikata, 2 mokymosi kreditai	<i>Bendrieji moduliai (0 mokymosi kreditų)</i> –
<i>Kvalifikaciją sudarančioms kompetencijoms įgyti skirti moduliai (iš viso 45 mokymosi kreditai)</i> Informacinių sistemų projektavimas ir kūrimas (JavaScript), 20 mokymosi kreditų Nesudėtingų duomenų bazių projektavimas ir kūrimas (JavaScript), 10 mokymosi kreditų Programavimo aplinkos ir kūrimo proceso valdymas (JavaScript), 15 mokymosi kreditų	<i>Kvalifikaciją sudarančioms kompetencijoms įgyti skirti moduliai (45 mokymosi kreditai)</i> Informacinių sistemų projektavimas ir kūrimas (JavaScript), 20 mokymosi kreditų Nesudėtingų duomenų bazių projektavimas ir kūrimas (JavaScript), 10 mokymosi kreditų Programavimo aplinkos ir kūrimo proceso valdymas (JavaScript), 15 mokymosi kreditų
<i>Pasirenkamieji moduliai (iš viso 5 mokymosi kreditai)</i> Testavimu ir vartotojų elgsena pagrįsto programavimo metodikų taikymas (JavaScript), 5 mokymosi kreditai Komunikacija su serveriu naudojant GraphQL standartą, 5 mokymosi kreditai	<i>Pasirenkamieji moduliai (0 mokymosi kreditų)</i> –
<i>Baigiamasis modulis (iš viso 5 mokymosi kreditai)</i> Įvadas į darbo rinką, 5 mokymosi kreditai	<i>Baigiamasis modulis (5 mokymosi kreditai)</i> Įvadas į darbo rinką, 5 mokymosi kreditai

### **Pastabos**

- Vykdamas pirminį profesinį mokymą asmeniui, jaunesniam nei 16 metų ir neturinčiam pagrindinio išsilavinimo, turi būti sudaromos sąlygos mokytis pagal pagrindinio ugdymo programą (*jei taikoma*).
- Vykdamas pirminį profesinį mokymą asmeniui turi būti sudaromos sąlygos mokytis pagal vidurinio ugdymo programą (*jei taikoma*).
- Vykdamas tęstinį profesinį mokymą asmens ankstesnio mokymosi pasiekimai įskaitomi švietimo ir mokslo ministro nustatyta tvarka.
- Tęstinio profesinio mokymo programos modulius gali vesti mokytojai, įgiję andragogikos žinių ir turintys tai pagrindžiantį dokumentą arba turintys neformaliojo suaugusiųjų švietimo patirties.
- Saugaus elgesio ekstremaliose situacijose modulį vedantis mokytojas turi būti baigęs civilinės saugos mokymus pagal Priešgaisrinės apsaugos ir gelbėjimo departamento direktoriaus patvirtintą mokymo programą ir turėti tai pagrindžiantį dokumentą.



- Tęstinio profesinio mokymo programose darbuotojų saugos ir sveikatos mokymas integruojamas į kvalifikaciją sudarančioms kompetencijoms įgyti skirtus modulius. Darbuotojų saugos ir sveikatos mokoma pagal Mokinių, besimokančių pagal pagrindinio profesinio mokymo programas, darbuotojų saugos ir sveikatos programos aprašą, patvirtintą Lietuvos Respublikos švietimo ir mokslo ministro 2005 m. rugsėjo 28 d. įsakymu Nr. ISAK-1953 „Dėl Mokinių, besimokančių pagal pagrindinio profesinio mokymo programas, darbuotojų saugos ir sveikatos programos aprašo patvirtinimo“. Darbuotojų saugos ir sveikatos mokymą vedantis mokytojas turi būti baigęs darbuotojų saugos ir sveikatos mokymus ir turėti tai pagrindžiantį dokumentą.
- Tęstinio profesinio mokymo programose saugaus elgesio ekstremaliose situacijose mokymas integruojamas pagal poreikį į kvalifikaciją sudarančioms kompetencijoms įgyti skirtus modulius.

## 6. PROGRAMOS MODULIŲ APRAŠAI

### 6.1. ĮVADINIS MODULIS

#### Modulio pavadinimas – „Įvadas į profesiją“

Valstybinis kodas	4000005	
Modulio LTKS lygis	IV	
Apimtis mokymosi kreditais	1	
Kompetencijos	Mokymosi rezultatai	Rekomenduojamas turinys mokymosi rezultatams pasiekti
1. Pažinti profesiją.	1.1. Išmanyti JavaScript programuotojo profesiją ir jos teikiamas galimybes darbo rinkoje.	<b>Tema. <i>Programuotojo profesija, jos specifika ir galimybės darbo rinkoje</i></b> <ul style="list-style-type: none"> <li>Programuotojo profesija, jos specifika</li> <li>Programuotojo profesijai svarbios kompetencijos ir veiklos sritys</li> <li>Programuotojo vaidmuo kuriant informacinių technologijų produktus (vykdant informacinių technologijų projektus)</li> <li>Programuotojo galimybės darbo rinkoje</li> </ul>
	1.2. Suprasti JavaScript programuotojo profesinę veiklą, veiklos procesus, funkcijas ir uždavinius.	<b>Tema. <i>JavaScript programuotojo profesinės veiklos aplinka, procesai ir tikslai</i></b> <ul style="list-style-type: none"> <li>JavaScript programuotojo profesinės veiklos, aplinka, pagrindiniai veiklos procesai ir tikslai</li> <li>Veiklos, uždaviniai, kuriuos programuotojas atlieka darbo vietoje</li> <li>Gerosios praktikos taisyklės ir jų svarba darbo procesui bei rezultatų kokybei</li> </ul>
	1.3. Demonstruoti jau turimus, neformalioju ir (arba) savaiminiu būdu įgytus jaunesniojo JavaScript programuotojo kvalifikacijai būdingus gebėjimus.	<b>Tema. <i>JavaScript programuotojo modulinė profesinio mokymo programa</i></b> <ul style="list-style-type: none"> <li>Modulinės JavaScript programuotojo profesinio mokymo programos tikslai ir uždaviniai</li> <li>Mokymosi formos ir metodai, siekiami rezultatai (kompetencijos) ir jų patvirtinimas, mokymosi pasiekimų vertinimo kriterijai</li> <li>Mokymosi įgūdžių demonstravimo formos (metodai)</li> <li>Individualus mokymosi planas</li> </ul> <b>Tema. <i>Turimų gebėjimų, įgytų savaiminiu ar neformalioju būdu, vertinimas</i></b> <ul style="list-style-type: none"> <li>Savaiminiu ir (arba) neformalioju būdu įgytų JavaScript programuotojo kvalifikacijai būdingų gebėjimų demonstravimas</li> <li>Savaiminiu ir (arba) neformalioju būdu įgytų gebėjimų įsivertinimas</li> </ul>
Mokymosi pasiekimų vertinimo kriterijai	Siūlomas modulio pasiekimų įvertinimas – <i>įskaityta (neįskaityta)</i> .	
Reikalavimai mokymui	<i>Mokymo(si) medžiaga:</i>	

skirtiems metodiniams ir materialiesiems ištekliams	<ul style="list-style-type: none"> <li>• JavaScript programuotojo modulinė profesinio mokymo programa</li> <li>• Vadovėliai ir kita mokomoji medžiaga</li> </ul> <p><i>Mokymo(si) priemonės:</i></p> <ul style="list-style-type: none"> <li>• Techninės priemonės mokymo(si) medžiagai iliustruoti, vizualizuoti, pristatyti</li> </ul>
Reikalavimai teorinio ir praktinio mokymo vietai	<p>Klasė ar kita mokymuisi pritaikyta patalpa su techninėmis priemonėmis mokymo(si) medžiagai pateikti (kompiuteriu, vaizdo projektoriumi) ir kompiuteriais, skirtais mokinių darbui.</p> <p>Praktinio mokymo klasė (patalpa), aprūpinta kompiuteriais, programine įranga.</p>
Reikalavimai mokytojų dalykiniam pasirengimui (dalykinei kvalifikacijai)	<p>Modulį gali vesti mokytojas, turintis:</p> <ol style="list-style-type: none"> <li>1) Lietuvos Respublikos švietimo įstatyme ir Reikalavimų mokytojų kvalifikacijai apraše, patvirtintame Lietuvos Respublikos švietimo ir mokslo ministro 2014 m. rugpjūčio 29 d. įsakymu Nr. V-774 „Dėl Reikalavimų mokytojų kvalifikacijai aprašo patvirtinimo“, nustatytą išsilavinimą ir kvalifikaciją;</li> <li>2) Informacinių technologijų mokytojo, JavaScript programuotojo ar lygiavertę kvalifikaciją arba informatikos mokslų studijų krypties ar lygiavertį išsilavinimą, arba ne mažesnę kaip 3 metų JavaScript programuotojo profesinės veiklos patirtį.</li> </ol>

## 6.2. KVALIFIKACIJĄ SUDARANČIOMS KOMPETENCIJOMS ĮGYTI SKIRTI MODULIAI

### 6.2.1. Privalomieji moduliai

#### Modulio pavadinimas – „Informacinių sistemų projektavimas ir kūrimas (JavaScript)“

Valstybinis kodas	406130011	
Modulio LTKS lygis	IV	
Apimtis mokymosi kreditais	20	
Asmens pasirengimo mokytis modulyje reikalavimai (jei taikoma)	Netaikoma	
Kompetencijos	Mokymosi rezultatai	Rekomenduojamas turinys mokymosi rezultatams pasiekti
1. Projektuoti ir programuoti žiniatinklio puslapių vartotojo sąsajas.	1.1. Pateikti internetinio puslapio turinį naudojant kompiuterinę žymėjimo kalbą.	<p><b>Tema. Pagrindinės HTML kalbos žymės</b></p> <ul style="list-style-type: none"> <li>• Internetinis puslapis – kur jis yra, kaip jis patenka į naršyklę</li> <li>• Kas yra užklausa ir kokie failai sudaro internetinį puslapį</li> <li>• Kokie struktūriniai elementai sudaro HTML puslapį</li> <li>• Kas yra HTML žymė, kokia jos struktūra</li> <li>• Pagrindinės HTML žymės</li> </ul> <p><b>Tema. HTML kalbos turinio žymės</b></p> <ul style="list-style-type: none"> <li>• Semantinės HTML žymės</li> <li>• Puslapio kūrimas HTML kalba</li> <li>• HTML formų žymės ir jų apdorojimas</li> </ul>
	1.2. Apipavidalinti internetinį puslapį naudojant pakopinius stilių šablonus ir karkasus.	<p><b>Tema. CSS pagrindai</b></p> <ul style="list-style-type: none"> <li>• Kas yra stilius ir kaip jį naudoti</li> <li>• Kodėl stiliai kaskadiniai? Pirmumo taisyklė ir specifiškumas</li> <li>• Kas yra selektorius ir kokios yra selektorių kategorijos</li> <li>• Pseudo klasės ir elementai</li> </ul> <p><b>Tema. CSS tinklalapio maketo kūrimo technikos</b></p> <ul style="list-style-type: none"> <li>• CSS maketų kūrimo technikos ir moduliai</li> <li>• CSS3 ir Bootstrap karkasas</li> </ul>
2. Taikyti programinės įrangos kūrimui naudojamus informatikos principus ir metodus.	2.1. Suprasti skaičiavimo sistemas.	<p><b>Tema. Skaičiavimo sistemos</b></p> <ul style="list-style-type: none"> <li>• Skaičiavimo sistemų reikšmė mokslui, technikai ir skaitmeninei elektronikai</li> <li>• Įvairios skaičiavimo sistemos</li> <li>• Skaičiavimo sistemų skaičių išreiškimas ir atvirkštinis perėjimas</li> </ul>

		<p><b>Tema. Informacijos matavimo vienetai</b></p> <ul style="list-style-type: none"> <li>• Bitas, baitas</li> <li>• Informacijos kiekio apskaičiavimas</li> <li>• Informacijos perdavimo greitis</li> </ul>
	2.2. Taikyti algoritmų ir logikos mokslo pagrindus programuojant.	<p><b>Tema. Logikos mokslo pagrindai</b></p> <ul style="list-style-type: none"> <li>• Logikos principai</li> <li>• Logikos mokslo pagrindų taikymas programuojant (Boolean algebra, De Morgano taisyklė)</li> </ul> <p><b>Tema. Algoritmai ir algoritmavimas</b></p> <ul style="list-style-type: none"> <li>• Klasikiniai algoritmų tipai (iteratyvūs, rekursyvūs)</li> <li>• Dinaminis programavimas</li> <li>• Algoritmų rašymas</li> <li>• Duomenų struktūrų pagrindai</li> </ul>
	2.3. Taikyti programinio kodo dizaino modelius programuojant.	<p><b>Tema. Dizaino šablonai</b></p> <ul style="list-style-type: none"> <li>• Dizainų šablonų rūšys</li> <li>• Dažniausiai naudojami dizaino šablonai</li> </ul> <p><b>Tema. Dizaino šablonų taikymas</b></p> <ul style="list-style-type: none"> <li>• Dizaino šablonų parinkimas</li> <li>• Dizaino šablonų taikymas programuojant</li> </ul>
	2.4. Naudoti informacinių sistemų kūrimo principus ir metodus programinės įrangos projektavime.	<p><b>Tema. Daugiasluoksnė programų architektūra ir MVC struktūra</b></p> <ul style="list-style-type: none"> <li>• Daugiasluoksnės architektūros modelis, jo panaudojimo galimybės ir savybės</li> <li>• MVC architektūros modelis ir jo taikymas kuriant programinę įrangą</li> </ul> <p><b>Tema. Pagrindiniai programavimo principai</b></p> <ul style="list-style-type: none"> <li>• Programavimo principų taikymas (SOLID, DRY, Separation of Concerns, code reuse)</li> <li>• Funkcinio programavimo principai</li> </ul>
3. Kurti tipinę programinę įrangą.	3.1. Konfigūruoti pagrindinius įrankius, reikalingus projekto paleidimui, ir įdiegti projektui reikalingus paketus.	<p><b>Tema. NodeJS ir Node Package Manager (NPM)</b></p> <ul style="list-style-type: none"> <li>• NodeJS ir NPM įrankiai</li> <li>• NodeJS aplinkos diegimas</li> <li>• Projekto sukūrimas naudojant Node Package Manager</li> </ul> <p><b>Tema. NodeJS aplinkos konfigūravimas ir valdymas</b></p> <ul style="list-style-type: none"> <li>• Projekto package.json konfigūracijos failas ir jo struktūra</li> <li>• Projekto package-lock.json failas ir jo paskirtis</li> <li>• Projekto reikalaujami susieti paketai (dependencies), jų įrašymas ir naudojimas, versijavimas</li> </ul>

<p>3.2. Diegti ir konfigūruoti įrankius kodo transformavimui, pakavimui ir optimizavimui.</p>	<ul style="list-style-type: none"> <li>• NPM komandų trumpiniai (scripts)</li> </ul> <p><b>Tema. ESLint kodo tikrinimo įrankis</b></p> <ul style="list-style-type: none"> <li>• Supažindinimas su ESLint įrankiu</li> <li>• ESLint diegimas</li> <li>• ESLint konfigūracija</li> <li>• ESLint pritaikymas darbo aplinkai (IDE)</li> </ul> <p><b>Tema. Babel įrankio diegimas ir naudojimas</b></p> <ul style="list-style-type: none"> <li>• Supažindinimas su Babel įrankiu</li> <li>• Babel diegimas</li> <li>• Babel konfigūraciniai failai ir pagrindiniai nustatymai</li> <li>• Babel paleidimas ir komandinės eilutės komandos</li> </ul> <p><b>Tema. Webpack įrankio diegimas ir naudojimas</b></p> <ul style="list-style-type: none"> <li>• Supažindinimas su Webpack įrankiu</li> <li>• Webpack diegimas</li> <li>• Webpack konfigūraciniai failai ir pagrindiniai nustatymai</li> <li>• Webpack paleidimas ir komandinės eilutės komandos</li> <li>• Kodo transformavimas naudojant Webpack</li> </ul>
<p>3.3. Analizuoti ir derinti vykdomą kodą, identifikuoti problemas ir jas taisyti.</p>	<p><b>Tema. Vykdomo kodo analizės įrankiai</b></p> <ul style="list-style-type: none"> <li>• Naudojimasis Chrome DevTools</li> <li>• HTML medžio (DOM) analizė</li> <li>• CSS koregavimas realiu laiku</li> <li>• Kodo greitaveikos analizės įrankiai (profiling, audits)</li> </ul> <p><b>Tema. Programos derinimas</b></p> <ul style="list-style-type: none"> <li>• Įvadas į kodo derinimą naudojant Chrome DevTools</li> <li>• Vykdomo kodo stabdymas ir analizavimas (breakpoints)</li> <li>• Vykdomo kodo stabdymas pagal nustatytas kondicijas</li> </ul>
<p>3.4. Kurti JavaScript programinį kodą ir naudoti jo sintaksę.</p>	<p><b>Tema. Sintaksė, gramatika, duomenų tipai</b></p> <ul style="list-style-type: none"> <li>• Esminiai sintaksės atributai ir ženklai</li> <li>• Kalbos gramatika ir raktažodžiai</li> <li>• Pagrindiniai duomenų tipai: String, Number, Boolean</li> <li>• Objekto aprašymas ir sintaksė</li> <li>• Masyvų aprašymas ir sintaksė</li> <li>• Kintamieji</li> </ul> <p><b>Tema. Operatoriai, salygos sakiniai</b></p>

		<ul style="list-style-type: none"> <li>• Priskyrimo operatoriai</li> <li>• Aritmetiniai operatoriai</li> <li>• Palyginimo operatoriai</li> <li>• Loginiai operatoriai</li> <li>• Operatoriai bitų operacijoms</li> <li>• Salygos sakiniai</li> <li>• Switch sakiniai</li> </ul> <p><b>Tema. Ciklai</b></p> <ul style="list-style-type: none"> <li>• While ciklas</li> <li>• For ciklas</li> <li>• For...in ir for...of ciklai</li> </ul> <p><b>Tema. Objektai</b></p> <ul style="list-style-type: none"> <li>• Paprastųjų objektų kūrimas</li> <li>• Prototipinių objektų sąvoka</li> <li>• Objektų išplėtimas</li> <li>• Objektų saugojimas kompiuterio atmintyje</li> <li>• Immutability</li> </ul> <p><b>Tema. ES6 standarto ypatumai</b></p> <ul style="list-style-type: none"> <li>• Skirtumai tarp let ir const raktažodžių</li> <li>• Arrow funkcijos</li> <li>• Paketų importavimas (import/export)</li> <li>• Destructors naudojimas</li> <li>• Paprastesnis objektų aprašas (Simplified Object Notations)</li> <li>• Template literals naudojimas</li> </ul>
	3.5. Vykdyti programinio kodo versijavimą naudojant versijavimo įrankius.	<p><b>Tema. Išėities kodo saugyklos</b></p> <ul style="list-style-type: none"> <li>• Išėities kodo saugyklų pagrindinės funkcijos ir panaudojimo galimybės</li> <li>• Išėities kodo saugyklos konfigūravimas</li> <li>• Bazinės komandos (Clone, Commit, Merge)</li> <li>• Šakų kūrimas ir valdymas</li> </ul> <p><b>Tema. Programinio kodo versijavimo vykdymas</b></p> <ul style="list-style-type: none"> <li>• Naujo projekto sukūrimas ir esamo projekto administravimas</li> <li>• Kodo pataisymų eksportas</li> </ul>
	3.6. Naudoti JavaScript funkcijas.	<p><b>Tema. Funkcijos</b></p> <ul style="list-style-type: none"> <li>• Funkcijos sukūrimas ir panaudojimas</li> </ul>

	<ul style="list-style-type: none"> <li>• Funkcijos sritis (scope)</li> <li>• Funkcijos kaip objektai</li> <li>• Funkcijos srities perkėlimas (binding)</li> <li>• Anoniminės funkcijos</li> <li>• IIFE (Immediately Invoked Function Expression)</li> </ul> <p><b>Tema. Standartinės JavaScript funkcijos</b></p> <ul style="list-style-type: none"> <li>• Matematinės funkcijos</li> <li>• Pagalbinės objektų funkcijos</li> </ul> <p><b>Tema. Pagrindinės masyvų funkcijos</b></p> <ul style="list-style-type: none"> <li>• Find funkcija</li> <li>• Filter funkcija</li> <li>• Sort funkcija</li> <li>• Map funkcija</li> <li>• Reduce funkcija</li> </ul>
3.7. Taikyti objektinio programavimo principus programuojant.	<p><b>Tema. Objektinis programavimas</b></p> <ul style="list-style-type: none"> <li>• Objekto sąvoka (klasė, objektas)</li> <li>• Enkapsuliacija</li> <li>• Paveldėjimas</li> <li>• Polimorfizmas</li> </ul> <p><b>Tema. Klasių instancijavimas ir metodai</b></p> <ul style="list-style-type: none"> <li>• Naujos klasės instancijos sukūrimas</li> <li>• Statiniai metodai</li> </ul>
3.8. Keisti ir manipuliuoti HTML elementų medį (DOM).	<p><b>Tema. Darbas su JavaScript Document Object Model (DOM)</b></p> <ul style="list-style-type: none"> <li>• DOM elementų ieškojimas</li> <li>• Elementų manipuliacija</li> <li>• CSS pritaikymas elementams</li> <li>• DOM elementų įvykiai (events)</li> </ul>
3.9. Kurti paprastas serveryje vykdomas aplikacijas naudojant JavaScript.	<p><b>Tema. NodeJS vykdymo aplinka</b></p> <ul style="list-style-type: none"> <li>• Supažindinimas su NodeJS</li> <li>• Standartinė įvestis/išvestis</li> <li>• Darbas su failų sistema</li> <li>• Komunikacija tinkle – serveris ir klientas</li> <li>• HTTP Web serveris</li> </ul> <p><b>Tema. Express karkasas</b></p>



		<ul style="list-style-type: none"> <li>• Supažindinimas su Express karkasu</li> <li>• Kelių sukūrimas (routing)</li> <li>• Užklausų apdorojimas</li> <li>• Užklausų validacija</li> </ul> <p><b>Tema. Duomenų bazės</b></p> <ul style="list-style-type: none"> <li>• Susipažinimas su duomenų bazėmis ir jų tipais</li> <li>• ORM pristatymas ir apžvalga</li> <li>• Susipažinimas su MongoDB</li> <li>• Duomenų saugojimas ir gavimas naudojant Mongoose įrankį</li> </ul>
	3.10. Kurti aplikacijas naudojant React karkasą.	<p><b>Tema. Populiarių karkasų palyginimas</b></p> <ul style="list-style-type: none"> <li>• React</li> <li>• Angular</li> <li>• Vue.js</li> </ul> <p><b>Tema. React karkasas</b></p> <ul style="list-style-type: none"> <li>• Supažindinimas su React karkasu</li> <li>• Darbo aplinkos paruošimas</li> <li>• React komponento sąvoka</li> <li>• Komponento kūrimo sintaksė</li> <li>• JSX sintaksė ir panašumai su HTML</li> <li>• Duomenų perdavimas su props</li> <li>• Aplikacijos puslapiavimas (Routing)</li> <li>• React lifecycle metodai</li> <li>• React komponentų duomenų saugykla (State)</li> <li>• DOM įvykiai (events) ir darbas su jais</li> <li>• Stateless Functional Components ir Higher-Order Components (<i>HOC</i>) naudojimas</li> </ul>
4. Analizuoti skirtingų tipų reikalavimus, apibūdinančius kompiuterinę programą.	4.1. Vykdyti reikalavimų peržiūros procesą naudojant vartotojo pasakojimo reikalavimų programinei įrangai formatą.	<p><b>Tema. Reikalavimų programinei įrangai formatai</b></p> <ul style="list-style-type: none"> <li>• Vartotojo pasakojimo reikalavimų formatas, jo panaudojimo sritys, galimybės ir apribojimai</li> <li>• Vartojimo atvejų formatas</li> </ul> <p><b>Tema. Reikalavimų peržiūros procesas</b></p> <ul style="list-style-type: none"> <li>• Reikalavimų peržiūros žingsniai</li> <li>• Reikalavimų paruošimo rezultatai. Ready ir done kriterijai</li> <li>• Reikalavimuose apibūdintos programinės įrangos integravimas į ją naudojančios įmonės verslo procesus</li> </ul>

	4.2. Naudoti funkcinis, nefunkcinis ir techninius kompiuterinės programos reikalavimus.	<p><b>Tema. Funkciniai reikalavimai</b></p> <ul style="list-style-type: none"> <li>• Funkciniai kompiuterinės programos reikalavimai</li> <li>• Atitikimo funkciniams reikalavimams nustatymas</li> </ul> <p><b>Tema. Nefunkciniai reikalavimai</b></p> <ul style="list-style-type: none"> <li>• Nefunkciniai (techniniai, saugos) kompiuterinės programos reikalavimai</li> <li>• Atitikimo nefunkciniams reikalavimams nustatymas</li> </ul>
	4.3. Testuoti programinę įrangą naudojant su JavaScript programavimo kalba suderinamus testavimo įrankius ir metodus.	<p><b>Tema. Testavimo principai</b></p> <ul style="list-style-type: none"> <li>• Testavimo svarba ir nauda kuriant programinę įrangą</li> <li>• Test Driven Development (TDD) principai</li> <li>• Testų rūšys ir jų palyginimas</li> </ul> <p><b>Tema. Modulių (unit) testavimas</b></p> <ul style="list-style-type: none"> <li>• Modulių testavimo įrankių diegimas ir konfigūracija</li> <li>• Įrankių valdymas ir komandinė eilutė</li> <li>• Įrankių funkcijos ir panaudojimas (API)</li> <li>• Testų rašymas</li> </ul> <p><b>Tema. Pilnas (End-to-end) testavimas</b></p> <ul style="list-style-type: none"> <li>• Pilno testavimo įrankių diegimas ir konfigūracija</li> <li>• Įrankių valdymas ir komandinė eilutė</li> <li>• Įrankių funkcijos ir panaudojimas (API)</li> <li>• Testų rašymas</li> </ul>
Mokymosi pasiekimų vertinimo kriterijai	<p>Sukurtas žiniatinklio “vieno puslapio” aplikacija (angl. <i>single page app</i>) naudojant NodeJs, duomenų bazę, (nereliacinę, pvz. MongoDB arba reliacinę, pvz. Postgres, MySQL ar pan), React arba alternatyvų karkasą (Vue.js, Angular ir pan). Vartotoja sąsaja yra interaktyvi, panaudotas aplikacijos puslapiavimas. Pakartotinio panaudojimo komponentai išskirti į atskirą modulį (sub-modulį). Vartotojo sąsaja padengta automatiniais testais. Serverio dalyje yra sukurti interfeisai (API) duomenų gavimui ir saugojimui. Duomenys yra saugomi išorinėje duomenų bazėje ir serverį perkrovus išlieka. Sukurtos programos kodas yra tvarkingas ir parašytas naudojantis kodavimo standartais.</p> <p>Darbo aplinkoje sukonfigūruota integracija su <i>TFS</i>, <i>Git</i> kodo saugykla. Panaudotas išeities kodo pasiėmimui iš saugyklos veiksmas, atlikti kodo pakeitimai pasinaudojant <i>Git</i> versijavimo principais (<i>pull request</i>, <i>merge</i>).</p>	
Reikalavimai mokymui skirtiems metodiniams ir materialiesiems ištekliams	<p><i>Mokymo(si) medžiaga:</i></p> <ul style="list-style-type: none"> <li>• Testas turimiems gebėjimams vertinti</li> <li>• Vadovėliai ir kita mokomoji medžiaga</li> </ul> <p><i>Mokymo(si) priemonės:</i></p> <p>Techninės priemonės mokymo(si) medžiagai iliustruoti, vizualizuoti, pristatyti.</p>	
Reikalavimai teorinio ir	Klasė ar kita mokymuisi pritaikyta patalpa su techninėmis priemonėmis mokymo(si) medžiagai pateikti (kompiuteriu, vaizdo	

praktinio mokymo vietai	projektoriumi) ir kompiuteriais, skirtais mokinių darbui. Praktinio mokymo klasė (patalpa), aprūpinta kompiuteriais, programine įranga (naršykle, tarnybinės stoties operacine sistema, Web serveriu, programavimo karkasais, SQL (NoSQL) DBVS, išeities kodo saugykla, komandinio darbų planavimo sistema).
Reikalavimai mokytojų dalykiniam pasirengimui (dalykinei kvalifikacijai)	Modulį gali vesti mokytojas, turintis: 1) Lietuvos Respublikos švietimo įstatyme ir Reikalavimų mokytojų kvalifikacijai apraše, patvirtintame Lietuvos Respublikos švietimo ir mokslo ministro 2014 m. rugpjūčio 29 d. įsakymu Nr. V-774 „Dėl Reikalavimų mokytojų kvalifikacijai aprašo patvirtinimo“, nustatytą išsilavinimą ir kvalifikaciją; 2) Informacinių technologijų mokytojo, baigusio atitinkamą kvalifikacijos tobulinimo programą, arba JavaScript programuotojo ar lygiavertę kvalifikaciją arba informatikos mokslų studijų krypties ar lygiavertį išsilavinimą, arba ne mažesnę kaip 3 metų JavaScript programuotojo profesinės veiklos patirtį.

#### Modulio pavadinimas – „Nesudėtingų duomenų bazių projektavimas ir kūrimas (JavaScript)“

Valstybinis kodas	406130012	
Modulio LTKS lygis	IV	
Apimtis mokymosi kreditais	10	
Asmens pasirengimo mokyti modulyje reikalavimai (jei taikoma)	Netaikoma	
Kompetencijos	Mokymosi rezultatai	Rekomenduojamas turinys mokymosi rezultatams pasiekti
1. Projektuoti tipines reliacines ir nereliacines (NoSQL) duomenų bazes.	1.1. Pateikti reliacinės duomenų bazės schemą.	<b>Tema. Įvadas į DBVS ir SQL kalbą</b> <ul style="list-style-type: none"> <li>DBVS sąvokos bei taikymo galimybės (reliacinės duomenų bazės, SQL kalba, SQL sakinių tipai)</li> <li>Koreguoti duomenų bazę naudojant komandas <i>insert</i>, <i>select</i>, <i>update</i> ir <i>delete</i></li> </ul> <b>Tema. Duomenų bazių projektavimas (CREATE TABLE sakiny, duomenų normalizavimas)</b> <ul style="list-style-type: none"> <li>Reliacinis duomenų modeliavimas, norminės formos (pirma, antra, trečia, Boyce-Codd, ketvirta)</li> <li>CREATE TABLE sakiny, pagrindiniai duomenų tipai. Pirminis ir išorinis raktai, indeksai. Unique indeksai</li> </ul>
	1.2. Pateikti nereliacinės (NoSQL) duomenų bazės schemą.	<b>Tema. ACID suderinamumas ir palyginimas su reliacinėmis DB</b> <ul style="list-style-type: none"> <li>ACID ir BASE transakcijos</li> <li>Skirtumai tarp NoSQL ir reliacinių duomenų bazių</li> </ul> <b>Tema. NoSQL duomenų bazių tipai ir jų savybės</b> <ul style="list-style-type: none"> <li>Key-Value DB tipas</li> </ul>

		<ul style="list-style-type: none"> <li>• <i>Document DB</i> tipas</li> <li>• <i>Column family</i></li> <li>• <i>Graph DB</i> tipas</li> </ul> <b>Tema. NoSQL duomenų bazių valdymas</b> <ul style="list-style-type: none"> <li>• Užklausų vykdymas</li> <li>• Optimizacijos technikos</li> </ul>
2. Programiškai įgyvendinti ir administruoti duomenų bazes.	2.1. Diegti ir valdyti duomenų bazių valdymo sistemą.	<b>Tema. DBVS diegimas</b> <ul style="list-style-type: none"> <li>• DBVS (H2, MySQL, T-SQL) diegimas tarnybinėje stotyje</li> </ul> <b>Tema. DBVS administravimas</b> <ul style="list-style-type: none"> <li>• DBVS (H2, MySQL, T-SQL) administravimas naudojant pagrindines jos funkcijas</li> </ul>
	2.2. Naudoti SQL kalbą duomenų bazės užpildymui ir informacijos išrinkimui.	<b>Tema. Duomenų išrinkimas naudojant SQL select sakinį ir pagrindinius select elementus</b> <ul style="list-style-type: none"> <li>• Duomenų išrinkimas pagal nurodytas sąlygas, įskaitant sudėtingesnius sąlyginio išrinkimo (where) atvejus (and, or)</li> <li>• Distinct funkcija</li> <li>• Duomenų rikiavimas panaudojant <i>order by</i></li> <li>• Duomenų agregavimas panaudojant funkcijas min, max, sum, avg, count</li> <li>• Duomenų grupavimas, naudojant <i>group by</i></li> <li>• Sakinių kūrimas naudojant <i>having</i></li> </ul> <b>Tema. Duomenų išrinkimas naudojant sąryšius (SQL select su join)</b> <ul style="list-style-type: none"> <li>• Lentelių duomenų jungimo būdai</li> <li>• Paprasta Dekarto sandauga</li> <li>• <i>Join</i> sakinio variantai</li> </ul>
Mokymosi pasiekimų vertinimo kriterijai	Išrinkti, pakeisti, ištrinti duomenys naudojant bazinius SQL sakinius ir funkcijas duotoje duomenų bazėje. Sukurtas duomenų bazės projektas: sukurtos pagal nurodytas sąlygas duomenų lentelės, lentelės laukams parinkti tinkami duomenų tipai, atlikti veiksmai užklausų optimizavimui: sukurti indeksai, pirminiai ir išoriniai raktai, panaudota automatiškai didinamos reikšmės galimybė, suprojektuota duomenų bazė yra suderinta iki reikiamos norminės formos. Pagal pateiktą užduotį atlikta duomenų transformacija ir sukurti DTO objektai.	
Reikalavimai mokymui skirtiems metodiniams ir materialiesiems ištekliams	<i>Mokymo(si) medžiaga:</i> <ul style="list-style-type: none"> <li>• Testas turimiems gebėjimams vertinti</li> <li>• Vadovėliai ir kita mokomoji medžiaga</li> </ul> <i>Mokymo(si) priemonės:</i> <ul style="list-style-type: none"> <li>• Techninės priemonės mokymo(si) medžiagai iliustruoti, vizualizuoti, pristatyti</li> </ul>	
Reikalavimai teorinio ir	Klasė ar kita mokymuisi pritaikyta patalpa su techninėmis priemonėmis mokymo(si) medžiagai pateikti (kompiuteris, vaizdo	

praktinio mokymo vietai	projektorius) ir kompiuteriais, skirtais mokinių darbui (praktinių užduočių sprendimui). Praktinio mokymo klasė (patalpa), aprūpinta kompiuteriais, programine įranga (naršykle, tarnybinės stoties operacine sistema, SQL DBVS).
Reikalavimai mokytojų dalykiniam pasirengimui (dalykinei kvalifikacijai)	Modulį gali vesti mokytojas, turintis: 1) Lietuvos Respublikos švietimo įstatyme ir Reikalavimų mokytojų kvalifikacijai apraše, patvirtintame Lietuvos Respublikos švietimo ir mokslo ministro 2014 m. rugpjūčio 29 d. įsakymu Nr. V-774 „Dėl Reikalavimų mokytojų kvalifikacijai aprašo patvirtinimo“, nustatytą išsilavinimą ir kvalifikaciją; 2) Informacinių technologijų mokytojo ar programuotojo, ar lygiavertę kvalifikaciją arba informatikos mokslų studijų krypties ar lygiavertį išsilavinimą, arba ne mažesnę kaip 3 metų programuotojo profesinės veiklos patirtį.

### Modulio pavadinimas – „Programavimo aplinkos ir kūrimo proceso valdymas (JavaScript)“

Valstybinis kodas	406130013	
Modulio LTKS lygis	IV	
Apimtis mokymosi kreditais	15	
Asmens pasirengimo mokytis modulyje reikalavimai (jei taikoma)	Netaikoma	
Kompetencijos	Mokymosi rezultatai	Rekomenduojamas turinys mokymosi rezultatams pasiekti
1. Naudoti tarnybinių stočių operacines sistemas.	1.1. Suprasti serverių veikimo principus ir panaudojimo galimybes.	<b>Tema. Serveriai ir jų operacinės sistemos</b> <ul style="list-style-type: none"> <li>• Serverių pagrindinės funkcijos, paskirtis ir panaudojimo galimybės</li> <li>• Baziniai operacinės sistemos veikimo principai bei panaudojimo galimybės</li> </ul> <b>Tema. Serverio operacinės sistemos diegimas</b> <ul style="list-style-type: none"> <li>• Virtualizacijos tipai, naudojimas</li> <li>• Virtualizacijos įrankiai</li> <li>• Operacinės sistemos diegimas ir konfigūravimas virtualioje aplinkoje</li> </ul> <b>Tema. Bazinis serverio operacinės sistemos funkcionalumas</b> <ul style="list-style-type: none"> <li>• Bylų sistema</li> <li>• Tinklo resursai</li> <li>• Pagrindiniai operacinės sistemos katalogai ir jų turinio administravimas</li> <li>• Saugumo grupių ir vartotojų administravimas</li> </ul>
	1.2. Valdyti programinius paketus.	<b>Tema. Programiniai paketai</b> <ul style="list-style-type: none"> <li>• Programinių paketų koncepcija</li> <li>• Programinių paketų valdymo sistemos</li> </ul>

		<ul style="list-style-type: none"> <li>• Programinių paketų priklausomybės</li> <li>• Programinių paketų diegimas, konfigūravimas ir valdymas</li> </ul> <b>Tema. Programinių paketų repozitorijos</b> <ul style="list-style-type: none"> <li>• Programinių paketų repozitorijos</li> <li>• Programinių paketų repozitorijų konfigūravimas ir valdymas</li> <li>• Programinių paketų papildomų repozitorijų konfigūravimas ir valdymas</li> </ul>
	1.3. Naudoti Web serverio programinę įrangą HTTP bylų viešinimui.	<b>Tema. Web serverio aplinka operacinėje sistemoje</b> <ul style="list-style-type: none"> <li>• Web serverio pagrindinės funkcijos ir savybės</li> <li>• Web serverio diegimas ir konfigūravimas</li> <li>• Web serverio apsauga</li> </ul> <b>Tema. Web serverio naudojimas</b> <ul style="list-style-type: none"> <li>• Statinių tinklapių viešinimas naudojant Web serverį</li> <li>• Dinaminių tinklapių viešinimas naudojant Web serverį</li> </ul>
2. Taikyti aktualias programinės įrangos kūrimo metodikas.	2.1. Suprasti Scrum proceso dalis ir komandos narių atsakomybes.	<b>Tema. Scrum procesas</b> <ul style="list-style-type: none"> <li>• Scrum metodologija</li> <li>• Scrum proceso vaidmenys</li> <li>• Scrum proceso dokumentai</li> <li>• Scrum proceso fazės ir iteracijos</li> </ul> <b>Tema. Scrum proceso komandos narių atsakomybės</b> <ul style="list-style-type: none"> <li>• Scrum procesas ir jo poveikis dirbant komandoje</li> <li>• Scrum projekto planavimas naudojant fazes ir iteracijas</li> <li>• Scrum komandos atsakomybės vykdant projektą</li> </ul>
	2.2. Analizuoti pateiktus reikalavimus, nustatant programos atitikimą reikalavimams.	<b>Tema. Programinės įrangos reikalavimų analizė</b> <ul style="list-style-type: none"> <li>• Vartotojo poreikių nustatymas ir analizė</li> <li>• Vartotojo reikalavimų analizė kuriant iteracijos užduotis</li> <li>• Vartotojo poreikių ir reikalavimų analizavimas taikant baigtumo kriterijus</li> </ul> <b>Tema. Programinės įrangos atitikimas reikalavimams</b> <ul style="list-style-type: none"> <li>• Programinės įrangos atitikimo reikalavimams nustatymas</li> <li>• Programinės įrangos atitikimo reikalavimams vertinimas</li> </ul>
	2.3. Naudoti projekto eigos valdymo principus.	<b>Tema. Projekto eiga</b> <ul style="list-style-type: none"> <li>• Projekto planavimas</li> <li>• Projekto darbų apimtys</li> <li>• Projekto fazės ir iteracijos</li> <li>• Projekto užbaigimas</li> </ul>

		<p><b>Tema. Projekto eigos valdymas</b></p> <ul style="list-style-type: none"> <li>• Projekto darbų apimties nustatymas ir planavimas</li> <li>• Projektų darbų apimties skaičiavimas</li> <li>• Projekto darbų eigos sekimas ir valdymas</li> <li>• Projekto dinamika</li> </ul>
3. Valdyti savo paties ir komandos atliekamą programinio kodo kūrimą.	3.1. Diegti ir valdyti programavimo JavaScript kalba darbo aplinką.	<p><b>Tema. Javascript programavimo aplinka</b></p> <ul style="list-style-type: none"> <li>• JavaScript IDE programavimo aplinkos funkcijos ir panaudojimo galimybės</li> <li>• IDE aplinkų diegimas</li> <li>• Pagrindinės IDE aplinkos funkcijos</li> </ul> <p><b>Tema. IDE naudojimas projektuose</b></p> <ul style="list-style-type: none"> <li>• Naujo projekto kūrimas naudojant IDE</li> <li>• Komandinis darbas naudojant IDE</li> </ul>
	3.2. Sekti programavimo darbų vykdymą naudojant komandinio darbų planavimo sistemas.	<p><b>Tema. Komandinio darbų planavimo sistemos</b></p> <ul style="list-style-type: none"> <li>• Pagrindiniai programinės įrangos kūrimo etapai</li> <li>• Sistemos funkcionalumas ir jos panaudojimo galimybės</li> <li>• Pagrindiniai sistemos aplinkos elementai</li> </ul> <p><b>Tema. Programavimo darbų vykdymo sekimas</b></p> <ul style="list-style-type: none"> <li>• Projekto struktūros elementai (projektas, darbai, nuorodos)</li> <li>• Užduotys ir jų elementai (darbų sukūrimas, planavimas, apimties nurodymas, sunaudoto ir likusio laiko užrašymas)</li> <li>• Darbų priskyrimas atskiriems vartotojams, komentarų kūrimas, kitų sukurtų komentarų ir kodo pakeitimų peržiūra</li> <li>• Darbų užbaigimo valdymas</li> <li>• Paieška darbų planavimo sistemoje</li> </ul>
Mokymosi pasiekimų vertinimo kriterijai	<p>Virtualioje aplinkoje įdiegta bazinė OS distribucija. Sukurtos naujos skaitmeninės bylos operacinės sistemos bylų sistemoje, saugumo grupės, saugumo grupėms priskirti vartotojai, pakeistas pagrindinių direktorių turinys, atlikti kompleksiški pakeitimai skaitmeninių bylų sistemoje. Įdiegti programiniai paketai, sukurtos ir sukonfigūruotos papildomos repozitorijos. Įdiegtas ir parengtas darbui <i>HTTP</i> serveris; naudojant <i>HTTP</i> serverį įkeltos, pakoreguotos ir paviešintos <i>HTTP</i> bylos. Paašškintos <i>Scrum</i> proceso ir vaidmenų sąvokos, jų svarba dirbant komandoje. Pademonstruotas supratimas ir gebėjimas naudoti skirtingus <i>Scrum</i> proceso elementus (vaidmenys, įvykiai, artefaktai, taisyklės), suprantama jų paskirtis ir tikslai. Apibūdintas projekto darbų sąrašo peržiūra (angl. <i>grooming</i>), planavimas, demonstracija, retrospektyva, ir progreso sekimas. Nurodytos projekto komandos ir jos narių atsakomybės. Suplanuotas projektas panaudojant įvykius ir iteracijos (angl. <i>sprint</i>) planus. Išanalizuoti reikalavimai, pagal juos parengtos iteracijos užduotys, teisingai įvertintas programinės įrangos atitikimas reikalavimams. Nustatyta darbų apimtis naudojant pasakojimo taškus, planavimo pokerį bei idealias ir realias darbo valandas.</p>	

	<p>Parodytas gebėjimas sekti projekto eigą stebint projekto ir iteracijos darbų sąrašus, likusių darbų apimtis ir jų dinamiką. Paskaičiuotas darbų vykdymo greitis.</p> <p>Užrašyti programos atitikimo reikalavimams kriterijai. <i>Jira</i> sistemoje atlikti įrašai: priskirtas darbas, užrašytas ir apskaitytas darbo laikas, įrašyti komentarai, pranešimai kitiems sistemos vartotojams.</p>
Reikalavimai mokymui skirtiems metodiniams ir materialiesiems ištekliams	<p><i>Mokymo(si) medžiaga:</i></p> <ul style="list-style-type: none"> <li>• Testas turimiems gebėjimams vertinti</li> <li>• Vadovėliai ir kita mokomoji medžiaga</li> </ul> <p><i>Mokymo(si) priemonės:</i></p> <ul style="list-style-type: none"> <li>• Techninės priemonės mokymo(si) medžiagai iliustruoti, vizualizuoti, pristatyti</li> </ul>
Reikalavimai teorinio ir praktinio mokymo vietai	<p>Klasė ar kita mokymuisi pritaikyta patalpa su techninėmis priemonėmis mokymo(si) medžiagai pateikti (kompiuteriu, vaizdo projektoriumi) ir kompiuteriais, skirtais mokinių darbui.</p> <p>Praktinio mokymo klasė (patalpa), aprūpinta kompiuteriais, programine įranga (naršykle, tarnybinės stoties operacine sistema, Web serveriu).</p>
Reikalavimai mokytojų dalykiniam pasirengimui (dalykinei kvalifikacijai)	<p>Modulį gali vesti mokytojas, turintis:</p> <ol style="list-style-type: none"> <li>1) Lietuvos Respublikos švietimo įstatyme ir Reikalavimų mokytojų kvalifikacijai apraše, patvirtintame Lietuvos Respublikos švietimo ir mokslo ministro 2014 m. rugpjūčio 29 d. įsakymu Nr. V-774 „Dėl Reikalavimų mokytojų kvalifikacijai aprašo patvirtinimo“, nustatytą išsilavinimą ir kvalifikaciją;</li> <li>2) Informacinių technologijų mokytojo, baigusio atitinkamą kvalifikacijos tobulinimo programą, arba JavaScript programuotojo ar lygiavertę kvalifikaciją arba informatikos mokslų studijų krypties ar lygiavertį išsilavinimą, arba ne mažesnę kaip 3 metų JavaScript programuotojo profesinės veiklos patirtį.</li> </ol>



### 6.3. PASIRENKAMIEJI MODULIAI

#### Modulio pavadinimas – „Testavimu ir vartotojų elgsena pagrįsto programavimo metodikų taikymas (JavaScript)“

Valstybinis kodas	406130014	
Modulio LTKS lygis	IV	
Apimtis mokymosi kreditais	5	
Asmens pasirengimo mokytis modulyje reikalavimai (jei taikoma)	Netaikoma	
Kompetencijos	Mokymosi rezultatai	Rekomenduojamas turinys mokymosi rezultatams pasiekti
1. Taikyti testavimu pagrįsto programavimo metodiką.	1.1. Rašyti testus taikant testavimu pagrįsto programavimo metodiką.	<b>Tema. Testavimu pagrįstas programavimas (TDD)</b> <ul style="list-style-type: none"> <li>• Testavimu pagrįsto programavimo principai</li> <li>• Testavimu pagrįsto programavimo privalumai ir trūkumai</li> <li>• Testavimu pagrįsto programavimo ciklo taikymas</li> </ul> <b>Tema. Testavimu pagrįsto programavimo taikymas</b> <ul style="list-style-type: none"> <li>• Gerosios testavimu pagrįsto programavimo praktikos</li> <li>• Gerųjų testavimu pagrįstų programavimo praktikų taikymas</li> </ul>
	1.2. Panaudoti <i>JUnit</i> karkaso metodus testuojant sukurtas JavaScript taikomąsias programas.	<b>Tema. <i>JUnit</i> karkasas</b> <ul style="list-style-type: none"> <li>• Testavimo progreso sekimas panaudojant <i>JUnit</i></li> <li>• Testavimo rezultatų analizė</li> <li>• Integravimas su JavaScript IDE</li> </ul> <b>Tema. <i>JUnit</i> metodai</b> <ul style="list-style-type: none"> <li>• Anotacijos</li> <li>• Išplėstinės galimybės (parametrai, taisyklės, kategorijos)</li> </ul>
2. Taikyti vartotojo elgsena pagrįsto programavimo metodiką.	2.1. Rašyti priėmimo testą ir JavaScript kodo dalių testus pagal vartotojo pasakojimus.	<b>Tema. Vartotojų elgsena pagrįstas programavimas (BDD)</b> <ul style="list-style-type: none"> <li>• Vartotojų elgsena pagrįsto programavimo principai</li> <li>• Vartotojų elgsena pagrįsto programavimo privalumai ir trūkumai</li> </ul> <b>Tema. Vartotojų elgsena pagrįsto programavimo ciklo taikymas</b> <ul style="list-style-type: none"> <li>• TDD ir BDD skirtumai</li> <li>• Gerųjų testavimu pagrįstų programavimo praktikų taikymo pavyzdžiai</li> </ul>
	2.2. Taikyti reikalavimų specifikavimą pagal pavyzdžius.	<b>Tema. Vartotojų elgsena pagrįsto programavime taikomi reikalavimų specifikavimo metodai</b> <ul style="list-style-type: none"> <li>• Specifikavimas pagal pavyzdžius išskiriant <i>given</i>, <i>when</i> ir <i>then</i> dalis</li> <li>• Ryšys su vartotojo pasakojimais</li> </ul>

		<p><b>Tema. Pokalbiai kaip priemonė reikalavimų specifikavime</b></p> <ul style="list-style-type: none"> <li>• Pokalbių tarp srities ekspertų ir programavimo komandos taikymas</li> <li>• Pokalbių pavertimas scenarijais</li> </ul>
Mokymosi pasiekimų vertinimo kriterijai	<p>Parašytas išbaigtas testų rinkinys bent vienai funkcijai, pagrįstas jo išbaigtumas suskaldžius galimus parametrus į režius ir ištestavus kertinius taškus.</p> <p>Parašytas išbaigtas testų rinkinys bent vienam vartotojo elgsenos scenarijui, pagrįstas jo išbaigtumas suskaldžius galimus parametrus į režius ir ištestavus kertinius taškus.</p>	
Reikalavimai mokymui skirtiems metodiniams ir materialiesiems ištekliams	<p><i>Mokymo(si) medžiaga:</i></p> <ul style="list-style-type: none"> <li>• Testas turimiems gebėjimams vertinti</li> <li>• Vadovėliai ir kita mokomoji medžiaga</li> </ul> <p><i>Mokymo(si) priemonės:</i></p> <ul style="list-style-type: none"> <li>• Techninės priemonės mokymo(si) medžiagai iliustruoti, vizualizuoti, pristatyti</li> </ul>	
Reikalavimai teorinio ir praktinio mokymo vietai	<p>Klasė ar kita mokymuisi pritaikyta patalpa su techninėmis priemonėmis mokymo(si) medžiagai pateikti (kompiuteriu, vaizdo projektoriumi) ir kompiuteriais, skirtais mokinių darbui.</p> <p>Praktinio mokymo klasė (patalpa), aprūpinta kompiuteriais, programine įranga (naršykle, tarnybinės stoties operacine sistema, Web serveriu, testavimo ir programavimo karkasais, SQL DBVS, išeities kodo saugykla, komandinio darbų planavimo sistema).</p>	
Reikalavimai mokytojų dalykiniam pasirengimui (dalykinei kvalifikacijai)	<p>Modulį gali vesti mokytojas, turintis:</p> <ol style="list-style-type: none"> <li>1) Lietuvos Respublikos švietimo įstatyme ir Reikalavimų mokytojų kvalifikacijai apraše, patvirtintame Lietuvos Respublikos švietimo ir mokslo ministro 2014 m. rugpjūčio 29 d. įsakymu Nr. V-774 „Dėl Reikalavimų mokytojų kvalifikacijai aprašo patvirtinimo“, nustatytą išsilavinimą ir kvalifikaciją;</li> <li>2) Informacinių technologijų mokytojo, baigusio atitinkamą kvalifikacijos tobulinimo programą, arba JavaScript programuotojo ar lygiavertę kvalifikaciją arba informatikos mokslų studijų krypties ar lygiavertį išsilavinimą, arba ne mažesnę kaip 3 metų JavaScript programuotojo arba testuotojo profesinės veiklos patirtį.</li> </ol>	

### Modulio pavadinimas – „Komunikacija su serveriu naudojant GraphQL standartą“

Valstybinis kodas	406130015	
Modulio LTKS lygis	IV	
Apimtis mokymosi kreditais	5	
Asmens pasirengimo mokyti modulyje reikalavimai (jei taikoma)	<i>Baigti šie moduliai:</i> Informacinių sistemų projektavimas ir kūrimas (JavaScript)	
Kompetencijos	Mokymosi rezultatai	Rekomenduojamas turinys mokymosi rezultatams pasiekti
1. Apibrėžti ir naudoti GraphQL pagrindinius	1.1. Suprasti GraphQL veikimo ir panaudojimo principus.	<p><b>Tema. Kas yra GraphQL</b></p> <ul style="list-style-type: none"> <li>• GraphQL užklausų kalbos standarto paskirtis ir istorija</li> </ul>

metodus ir funkcijas		<ul style="list-style-type: none"> <li>GraphQL pranašumai kuriant API užklaudas</li> </ul> <b>Tema. Darbas su GraphQL</b> <ul style="list-style-type: none"> <li>Paprastų GraphQL užklausių rašymas</li> <li>Įvadas į GraphQL duomenų tipus</li> </ul>
	1.2. Naudoti GraphQL užklaudas ir schemas.	<b>Tema. GraphQL užklausių kalba</b> <ul style="list-style-type: none"> <li>Dokumentai ir operacijos</li> <li>Laukai</li> <li>Mutacijos</li> </ul> <b>Tema. GraphQL schema</b> <ul style="list-style-type: none"> <li>Schemas objektas</li> <li>Tipų sistema</li> <li>Resolve metodas</li> <li>Validacija</li> </ul>
2. Taikyti GraphQL Apollo įrankį aplikacijų kūrime	2.1. Apollo GraphQL server naudojimas.	<b>Tema: Kas yra Apollo</b> <ul style="list-style-type: none"> <li>Apollo įrankio apžvalga</li> <li>Apollo server galimybės</li> <li>Apollo client galimybės</li> <li>Programavimo aplinkos konfigūravimas</li> </ul> <b>Tema: Serverio dalies kūrimas</b> <ul style="list-style-type: none"> <li>Schemas aprašymas</li> <li>Pajungimas prie duomenų bazės</li> <li>GraphQL “resolve” funkcijų rašymas</li> </ul>
	2.2. React aplikacijos konfigūravimas naudojant Apollo GraphQL.	<b>Tema: Apollo kliento naudojimas</b> <ul style="list-style-type: none"> <li>Apollo kliento dalies konfigūravimas</li> <li>GraphQL užklausių vykdymas naudojant Apollo klientą</li> </ul> <b>Tema: Apollo kliento integravimas į React karkasą</b> <ul style="list-style-type: none"> <li>Integracija naudojant “Apollo provider”</li> <li>Query komponentų rašymas</li> <li>Query komponentų naudojimas React aplinkoje</li> <li>Duomenų keitimas naudojant mutacijas.</li> </ul>
Mokymosi pasiekimų vertinimo kriterijai	Sukurtas paprasta (angl. <i>single page</i> ) aplikacija naudojant <i>React</i> , <i>Apollo client</i> ir <i>Apollo server</i> įrankius. Duomenys gaunami iš serverio naudojant GraphQL užklausių kalbos standartą. Naudojamos mutacijos duomenų keitimui. Sukurtos programos kodas yra tvarkingas ir parašytas naudojant kodavimo standartus.	
Reikalavimai mokymui	<i>Mokymo(si) medžiaga:</i>	

skirtiems metodiniams ir materialiesiems ištekliams	<ul style="list-style-type: none"> <li>• Testas turimiems gebėjimams vertinti</li> <li>• Vadovėliai ir kita mokomoji medžiaga</li> </ul> <p><i>Mokymo(si) priemonės:</i></p> <ul style="list-style-type: none"> <li>• Techninės priemonės mokymo(si) medžiagai iliustruoti, vizualizuoti, pristatyti</li> </ul>
Reikalavimai teorinio ir praktinio mokymo vietai	<p>Klasė ar kita mokymuisi pritaikyta patalpa su techninėmis priemonėmis mokymo(si) medžiagai pateikti (kompiuteriu, vaizdo projektoriumi) ir kompiuteriais, skirtais mokinių darbui.</p> <p>Praktinio mokymo klasė (patalpa), aprūpinta kompiuteriais, programine įranga (naršykle, tarnybinės stoties operacine sistema, Web serveriu, programavimo karkasais, DBVS, išeities kodo saugykla, komandinio darbų planavimo sistema).</p>
Reikalavimai mokytojų dalykiniam pasirengimui (dalykinei kvalifikacijai)	<p>Modulį gali vesti mokytojas, turintis:</p> <ol style="list-style-type: none"> <li>1) Lietuvos Respublikos švietimo įstatyme ir Reikalavimų mokytojų kvalifikacijai apraše, patvirtintame Lietuvos Respublikos švietimo ir mokslo ministro 2014 m. rugpjūčio 29 d. įsakymu Nr. V-774 „Dėl Reikalavimų mokytojų kvalifikacijai aprašo patvirtinimo“, nustatytą išsilavinimą ir kvalifikaciją;</li> <li>2) Informacinių technologijų mokytojo, JavaScript programuotojo ar lygiavertę kvalifikaciją arba informatikos mokslų studijų krypties ar lygiavertį išsilavinimą, arba ne mažesnę kaip 3 metų JavaScript programuotojo profesinės veiklos patirtį.</li> </ol>

## 6.4. BAIGIAMASIS MODULIS

### Modulio pavadinimas – „Įvadas į darbo rinką“

Valstybinis kodas	4000004
Modulio LTKS lygis	IV
Apimtis mokymosi kreditais	5
Kompetencijos	Mokymosi rezultatai
1. Formuoti darbinius įgūdžius realioje darbo vietoje	1.1. Įsivertinti ir realioje darbo vietoje demonstruoti įgytas kompetencijas. 1.2. Susipažinti su būsimo darbo specifiška ir adaptuotis realioje darbo vietoje. 1.3. Įsivertinti asmenines integracijos į darbo rinką galimybes.
Mokymosi pasiekimų vertinimo kriterijai	Siūlomas modulio pasiekimų įvertinimas – <i>atlikta (neatlikta)</i> .
Reikalavimai mokymui skirtiems metodiniams ir materialiesiems ištekliams	<i>Nėra.</i>
Reikalavimai teorinio ir praktinio mokymo vietai	Darbo vieta, leidžianti įtvirtinti įgytas jaunesniojo JavaScript programuotojo kvalifikaciją sudarančias kompetencijas.
Reikalavimai mokytojo dalykiniam pasirengimui (dalykinei kvalifikacijai)	Mokinio mokymuisi modulio metu vadovauja mokytojas, turintis: 1) Lietuvos Respublikos švietimo įstatyme ir Reikalavimų mokytojų kvalifikacijai apraše, patvirtintame Lietuvos Respublikos švietimo ir mokslo ministro 2014 m. rugpjūčio 29 d. įsakymu Nr. V-774 „Dėl Reikalavimų mokytojų kvalifikacijai aprašo patvirtinimo“, nustatytą išsilavinimą ir kvalifikaciją; 2) Informacinių technologijų mokytojo, JavaScript programuotojo ar lygiavertę kvalifikaciją arba informatikos mokslų studijų krypties ar lygiavertį išsilavinimą, arba ne mažesnę kaip 3 metų JavaScript programuotojo profesinės veiklos patirtį. Mokinio mokymuisi realioje darbo vietoje vadovaujantis praktikos vadovas turi turėti ne mažesnę kaip 3 metų profesinės veiklos JavaScript programavimo srityje patirtį.