

Full stack programuotojo praktika

Skirta baigusiems 101 studijas

Vilnius, 2020

Turinys

[Praktikos tikslas 3](#_Toc49905949)

[Vertinimo kriterijai 4](#_Toc49905950)

[Programavimo darbų eiliškumas ir laiko rėžiai 5](#_Toc49905951)

[Siūlomos temos 6](#_Toc49905952)

[Užsiėmimams ruoštis naudingos nuorodos 7](#_Toc49905953)

# Praktikos tikslas

Šios praktikos tikslas supažindinti 101 kusus baigusį programuotoją su darbine aplinka ir projekto vystymo metodais. Viskas vyksta Agile stiliui būdingą metodologija:

… **Planavimas -> Vystymas -> Testavimas -> Pristatymas** -> Planavimas -> …

Programuotojas turi pritaikyti studijų metu įgytas žinias, ir:

* Planuoti savo darbus.
* Pateikti juos vertinimui naudojant git versijavimo sistemą.
* Nuosekliai vykdyti puslapio vystymą, jei reikia, lanksčiai keisti darbų apimtis, kad žiniatinklio idėja būtų išvystyta iki numatytos praktikos baigimo datos.
* Naudoti praktikos metu nurodytas gerasias praktikas, bei taikyti rekomenduotus programavimo logikos dizaino šablonus.

Tikslas bus pasiektas praktikantui įgyvendinus darbus iki numatytos datos. Jeigu studento baigiamasis projektas bus įvertintas **9.5** balu arba daugiau, bus gaunama rekomendacija iš praktikos kuruojančio darbų vadovo/dėstytojo.

# Vertinimo kriterijai

|  |  |  |  |
| --- | --- | --- | --- |
| **Kriterijus** | **Balas** | **Kriterijus** | **Balas** |
| Idėja, darbų, bei laiko susiplanavimas. Raštu. | 0.5 | Autorizacija. Naudojant savo parašytą API. | 0.5 |
| Sudaryta duomenų struktūrų ir jų sąryšių schema. | 0.5 | Rikiavimas bent pagal 3 kriterijus | 0.5 |
| Sudaryta FE ir BE API endpont’ų lentelė. | 0.5 | Filtravimas ir/arba paieška. | 0.5 |
| Programos vaizdas korektiškai naudojant React.js bibliotekas bei jos komponentus. | 1 | Puslapiavimas | 0.5 |
| Redux bibliotekos pritaikymas aplikacijai | 1 | Duomenų trasavimas HTTP protokolo ryšiu. API užklausų suderinimas ir įgalinimas. | 1 |
| Dizainais kuris prisitaiko prie ekrano dydžio. | 0.5 | FE aplikacijai pritaikytas API serveris naudojant MVC šabloną. | 1.5 |
| Autentifikacija. | 0.5 | Korektiška duomenų bazė, duomenų struktūros ir ryšiai tarp jų. | 1 |

# Programavimo darbų eiliškumas ir laiko rėžiai

|  |  |  |
| --- | --- | --- |
| **Pamoka** | **Prieš susitikimą paruošti darbai** | **Per susitikimą užtvirtinta tema** |
| 1 | Pilna projekto idėja, pateikta raštu. Git repozitorijos pasiruošimas. Darbų planas pagal šią lentelę. | Duomenų struktūrų, bei jų ryšių išaiškinimas. Repozitorijos, bei projekto šablonas. |
| 2 | Susidaryti duomenų stūktūrų ir jų ryšių schema, kuri bus naudojame aplikacijoje. | API sąsajos sudarymo stuktūra. |
| 3 | Sudaryta API sąsajos lentelė. (API Endpoints table) | Komponentų struktūra, grupavimas, perpanaudojimas. Navigacija. |
| 4 | Sudarytas grubus aplikacijos skeletas su visų puslapių atvirai pasiekiama navigacija. | FE vystymo eiga. SCSS kintamųjų aplinka, stiliaus perpanaudojimas. |
| 5 | Pilnai įgalinti stiliai ir puslapio dizainas | Vuex metodologija |
| 6 | Pasiruošta visų norimų Vuex saugyklos struktūra, bei aprašytos kaitos funkcijos. | Komponentų sujungimas su Vuex sugykla ir saugyklos kaitos funkcijų įgalinimo pavyzdžiai. |
| 7 | Pilnai įgalintos saugyklos kaitos funkcijos | BE duomenų struktūrų sudrarymas, migracijos. |
| 8 | Suformuotos BE duomenų struktūros, sujungimas su DB. | Valdikliai. Autorizuotos ir neautorizuotos API sąsajos formavimas. Postman programa. |
| 9 | Veikianti API sąsaja pagal sudarytą API sąsajos lentelę. | BE ir FE apjungimas HTTP protokolu. Autentifikacija |
| 10 | Paruošta aplinka API panaudojimo vystymui. | API funkcionalumo iškėlimas, Axios ir autentifikacijos įgalinimas |
| 11 | Dirbtiniai duomenys pakeisti tikrais. | Algoritmai. Paieška ir filtravimas |
| 12 | Įgalintas filtravimas | Algoritmai. Rikiavimas |
| 13 | Įgalintas rikiavimas ir grupavimas | Algoritmai. Puslapiavimas ir navigavimas |
| 14 | Įgalintas puslapiavimas ir navigavimas | Konsultacijų planavimas |
| 15 -19 | Įndividualūs darbai. | Konsultacijos. |
| 20 | Pasiruošimas darbų pristatymui, prezentacija. | Darbų pristatymas, vertinimas. |

# Siūlomos temos

|  |  |
| --- | --- |
| **Tema** | **Programavimo darbų apytikslis santykis** |
| Pramogų ar užsiėmimų rezervavimo sistema. | FE 30% | BE 70% |
| Chat applikacija | FE 40% | BE 60% |
| Internetinė parduotuvė. | FE 50% | BE 50% |
| Socialinis puslapis turiniui dalintis. „Blog“. | FE 60% | BE 40% |
| Javacript žaidimo puslapis | FE 70% | BE 30% |

Galima rinktis ir savo temą. Pasirinkus savo temą būtina suderinti su darbų vadovu/dėstytoju jo apimtį, tiksliai apsibrėžti darbus ir jų eigą.Užsiėmimams ruoštis naudingos nuorodos

***Super duper efektyvaus mokymosi patarimas !!!***

*Mokytis temas, kurios bus užtvirtinamos per pamoką, reikia PRIEŠ PAMOKĄ savarankiškai. Tokiu būdų turėsite bendrą suvokimą apie tai kas aiškina, galėsite užduoti taiklius klausimus, taip pilnai susitruktūrizuosite informaciją ir geriau ją įsisavinsite.*

1. Git, agile, relation schema.

* Git game <https://learngitbranching.js.org/>
* Github <https://www.youtube.com/watch?v=r50BKIFGCI0&list=PLB5jA40tNf3v1wdyYfxQXgdjPgQvP7Xzg>
* Agile <https://www.youtube.com/watch?v=jSayJF0epJs>
* Duomenų struktūrų ryšių schema <https://www.youtube.com/watch?v=c0_9Y8QAstg>

1. API sąsaja, HTTP request
   * API - <https://www.youtube.com/watch?v=s7wmiS2mSXY>
   * HTTP protocol - <https://www.youtube.com/watch?v=eesqK59rhGA>
   * Endpoints - <https://www.youtube.com/watch?v=Q-BpqyOT3a8>
2. Komponentų struktūra, grupavimas, perpanaudojimas. Navigacija.
   * Reusable componets <https://www.youtube.com/watch?v=HVzYir_0Kgg>
   * Router <https://www.youtube.com/watch?v=UIu-7qOdki4>
3. Stilizavimas, scss
   * Scss funkcionalumas <https://www.youtube.com/watch?v=St5B7hnMLjg&list=PL4cUxeGkcC9iEwigam3gTjU_7IA3W2WZA&index=1>
4. Vuex
   * Dokumentacija <https://vuex.vuejs.org/>
   * Idėja ir panaudojimo tikslas <https://www.youtube.com/watch?v=wbp_ro-eWwQ>
5. Vuex
   * Pavyzdžiai <https://www.youtube.com/watch?v=BGAu__J4xoc&list=PL4cUxeGkcC9i371QO_Rtkl26MwtiJ30P2>
6. BE duomenų modeliai (Entity models)
   * Models and relationships <https://www.youtube.com/watch?v=42l4nHl_aUM&t=3s>
7. Valdikliai, navigavimas, tarpinis kodo įterpimas (controllers, navigation, middleware)
   * Valdikliai - <https://www.youtube.com/watch?v=sLFNVXY0APk>
   * Middleware - <https://www.youtube.com/watch?v=quiUytHXutM>
8. Autentifikacija <https://www.youtube.com/watch?v=Vd1YwbHrY1Q&list=PLfdtiltiRHWF1jqLcNO_2jWJXj9RuSDvY&index=1>
9. API sąsają ir axio biblioteka <https://www.youtube.com/watch?v=6LyagkoRWYA>
10. Filtrai
    * Search filter: <https://www.youtube.com/watch?v=3NG8zy0ywIk>
    * Filter by prop <https://www.youtube.com/watch?v=_BHCiR2L660>
11. Rikiavimas, iteracinės funkcijos
    * Sort <https://www.youtube.com/watch?v=MWD-iKzR2c8>
    * Iteration functions <https://www.youtube.com/watch?v=R8rmfD9Y5-c>
12. Puslapiavimas
    * Puslapiavimas <https://www.youtube.com/watch?v=y8Y6oe5CKXs>
    * Paieška ir puslapiavimas <https://www.youtube.com/watch?v=FJ1MaNtjPDs>