



# INFORMATIKOS FAKULTETAS

## **T120B162 Programų sistemų testavimas**

### **1 laboratorinis darbas**

Studentas:

Ignas Jasonas IFF-6/6

Dėstytojas:

doc. Šarūnas Packevičius

KAUNAS 2019

## Turinys

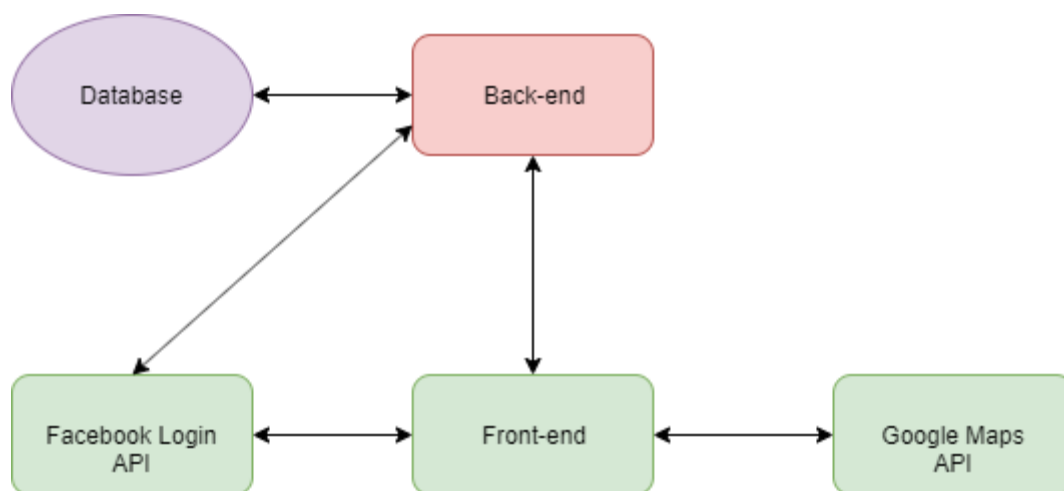
1.	Testuojama sistema.....	4
1.1.	Sistemos architektūra .....	4
1.2.	Planuojamos funkcijos .....	4
1.3.	Sistemos panaudojimo atvejai .....	5
2.	Testavimas .....	5
2.1.	Įvadas .....	5
3.	Testavimo apimtis ir strategijos.....	6
3.1.	Unit testing .....	6
3.2.	Integration testing .....	6
3.3.	Validation testing .....	6
3.4.	Interface testing.....	6
3.5.	Acceptance testing .....	6
3.6.	Performance testing.....	7
3.7.	Security testing .....	7
4.	Testavimo prielaidos.....	7
5.	Testavimo prioritetai.....	7
6.	Testavimo technikos .....	8
7.	Testavimo rolės .....	8
7.1.	Programuotojas (developer).....	8
7.2.	Testuotojas (tester).....	8
7.3.	Projekto vadovas (PM – project manager) .....	8
8.	Rezultatai .....	9
9.	Testavimo aplinka.....	9
10.	Testavimo skriptai.....	10
10.1.	Prisijungimas .....	10
10.2.	Registracija .....	10
10.3.	Grafitti peržiūra.....	10

10.4.	Vartotojo informacijos peržiūra .....	11
10.5.	Graffiti žemėlapių peržiūra .....	11
10.6.	Prisijungimas per socialinį tinklą .....	11
10.7.	Graffiti maršruto generavimas .....	12
10.8.	Slaptažodžio atkūrimas.....	12
10.9.	Profilio informacijos keitimas.....	13
10.10.	Informacijos apie graffiti kėlimas .....	13
10.11.	Graffiti vertinimas .....	13
10.12.	Graffiti informacijos redagavimas .....	14
10.13.	Vartotojų informacijos redagavimas .....	14
10.14.	Vartotojų blokavimas / atblokavimas .....	15
10.15.	Graffiti patvirtinimas .....	15
11.	Testavimo grafikas.....	15
12.	Testavimo rizikos.....	16
13.	Išvados .....	16

## 1. Testuojama sistema

Išsirinkau savo daromą projektą GraffitiTour. Tai sistema, skirta turistams, kurie prisijungę, galės kelti informaciją apie rastus grafičius. Į šią informaciją įeina grafičio lokacija, nuotrauka, komentaras. Kiti vartotojai galės, šį grafitį vertinti pagal grožį, pasiekiamumą. Sistema galės sugeneruoti kelią turistams pamatyti geriausiai įvertintus grafičius.

### 1.1. Sistemos architektūra



*Pav. 1. Sistemos architektūra*

Sistemos architektūra yra gana paprasta. Ji yra sudaryta iš:

- Front-end – Next.js + React + Redux. Front-end dalis tiesiogiai komunikuoja su Google Maps API, Facebook API ir Back-end'u
- Back-end – Express (nodejs) + MySQL (duomenų bazė)
  - Pagrindinis servisas – duomenų bazės valdymas, vartotojų autentifikacija;
  - Pagalbinis servisas – Google Maps API – vartotojų lokacijos gavimui, grafičių vietų tag'inimui reikalingas servisas. Facebook Login API – vartotojų autentifikavimui per Facebook

### 1.2. Planuojamos funkcijos

- Žemėlapiu su grafičių vietomis peržiūra
- Grafičio lokacijos kūrimas, į kurią įeina nuotraukų kėlimas, komentavimas, tag'inimas žemėlapyje
- Statistikos apie grafičius peržiūra (kiek aplankė, įvertinimai)
- Prisijungimas per Facebook
- Share funkcija įvairiuose socialiniuose tinkluose (Facebook, Twitter, LinkedIn)
- Maršruto pro geriausiai įvertintus grafičius generavimas

### 1.3. Sistemos panaudojimo atvejai



Pav. 2. Sistemos panaudojimo atvejai

## 2. Testavimas

### 2.1. Įvadas

Šis skyrius aprašo šiai sistemai ištestuoti reikalingas testavimo tipus, kurie yra vykdomi išvardinta tvarka:

1. Unit testing – mažiausių sistemos komponentų, funkcijų testavimas
2. Integration testing – testavimas ar komponentai vienas su kitu teisingai komunikuoja
3. Validation testing – realios aplinkos testavimas, formų duomenų validacijų testavimas

4. Interface testing – dizaino, vartojimo patogumo testavimas
5. Acceptance testing – vartotojų pasiekiamų funkcijų testavimas ir lyginimas ar tai atitinka suplanuotas sistemos funkcijas
6. Performance testing – greitaveikos, sistemos užkrovimo testavimas
7. Security testing – privačių duomenų saugojimo, pasiekiamumo tretiesiems asmenims saugumo testavimas

### 3. Testavimo apimtis ir strategijos

#### 3.1. Unit testing

Į unit testus įeina visos galimos ištestuoti (paduodant tam tikrus parametrus ir pagal tuos parametrus žinant rezultatą) GraffitiTour sistemos funkcijos. Tai būtų pagalbinės, skaičiavimo, duomenų apdorojimo ir kitos funkcijos.

#### 3.2. Integration testing

Į integracijos testavimą įeina tam tikrų komponentų, kurie bendrauja tarpusavyje, aplinkos sukūrimas ir tų komponentų komunikacijos ištestavimas. Tai galėtų būti duomenų lygio komponento, kuris parsisiunčia duomenis iš serverio ir komponento, kuris gauna bei atvaizduoja tuos duomenis komunikacijos testavimas įvairiais scenarijais, kaip neteisingų duomenų gavimo arba serverio klaidos atvejų testavimas

#### 3.3. Validation testing

Į validacijos testavimą įeina sistemos formų kuriose vartotojas gali įvesti duomenis, validacijos tikrinimas. Tai galėtų būti formos laukų į kuriuos galima įvesti apribotą kiekį simbolių tikrinimas, įvedant per daug arba per mažai simbolių ir tikrinant sistemos reakciją į tai (ar rodomas klaidos pranešimas, ar serveris taip pat validuoja tuos duomenis).

#### 3.4. Interface testing

Į UI testavimą įeina dizaino, patogumo testavimas. Testuojama kaip greitai galima naviguoti po sistemą, ar tai yra patogiu ir patraukliu akiai.

#### 3.5. Acceptance testing

Į acceptance testavimą įeina sistemos apibrėžtų funkcijų testavimas ir palyginimas su tuo kas yra įgyvendinta. Testuojama kiekvienos rolės funkcijų pasiekiamumas ir ar to funkcijos atlieka tai ką ir privalo. Testuojamos use case funkcijos (Pav. 2)

### 3.6. Performance testing

Į performance testing įeina sistemos greitaveikos testavimas. Kaip sistema greitai užkrauna parsųstus duomenis ir atvaizduoja juos vartotojui. Kaip greitai atliekamos numatytos sistemos funkcijos.

### 3.7. Security testing

Į security testavimą įeina vartotojų privačių duomenų pasiekiamumo tretiesiems asmenims pasiekiamumas, sistemos funkcijų pasiekiamumą pagal vartotojo roles tikrinimas.

## 4. Testavimo prielaidos

Testavimo prielaidos - sąlygos, kurios yra būtinos įvykdyti, kad testavimo planas galėtų būti pradėtas vykdyti.

- 1) Funkciniai reikalavimai turi būti išreikšti panaudos atvejų diagrama bei veiklos diagramomis
- 2) Turi būti numatyta sistemos testavimo strategija (kokios sistemos vietos, funkcionalumas turi būti testuojamos pirmiausia)
- 3) Remiantis veiklos diagramomis turi būti sudaryti sistemos testavimo scenarijai
- 4) Prieš kiekvieną konkretų funkcionalumo testavimą, sistemoje privalo būti realizuotas atitinkamas funkcionalumas
- 5) Turi būti įdiegta bei tinkamai sukonfigūruota testavimo aplinka (pvz. testavimo duomenų bazė atskirta nuo realiosios)

## 5. Testavimo prioritetai

Testai turi skirtingą svarbumo lygį, pagal kurį yra skiriamas galimų resursų kiekis. Šie testai yra išvardinti iš eilės pagal prioriteto lygį (pirmi testai sąrašė yra svarbiausi, tolesni – mažiau)

- 1) Unit testing – ar visos funkcijos atlieka tai ką ir turi. Gražina rezultatus kurių tikimasi
- 2) Integration testing – ar komponentai teisingai komunikuoja vienas su kitu
- 3) Acceptance testing – ar funkcijos atitinka specifikaciją
- 4) Validation testing – ar įvedami duomenys yra validuojami
- 5) Security testing – ar privatūs duomenys yra gerai apsaugoti ir pasiekiami tik tam tikriems vartotojams
- 6) Interface testing – ar UI sąsaja atitinka dizaino kriterijus ir yra patogi naudotis
- 7) Performance testing – ar sistema veikia pakankamai greitai

## 6. Testavimo technikos

Šios testavimo technikos bus panaudotos testuojamoje sistemoje:

- 1) Testų skriptai – bibliotekos pagalba parašyti testai bus vykdomi kiekvieną kartą kompiliuojant programą. Taip yra testuojami unit testai
- 2) Automatizuoti testai – išorine programine įranga paruošti testai kurie testuos sistemos funkcijas, paduodant tam tikrus duomenis ir gaunant rezultatą kurio tikimasi
- 3) Neautorizuoti testavimo scenarijai – testų skriptai, kurių pagalba bandoma pasiekti resursus, kuriems neturėtų turėti prieigos.
- 4) Aptiktų klaidų sąrašas – rankinio testavimo metu pastebėti pagal sistemos dokumentaciją neapibrėžti neatitikimai

## 7. Testavimo rolės

Šiame skyriuje pateikiami testavimo procese dalyvaujantys asmenys bei jų rolės/pareigos.

### 7.1. Programuotojas (developer)

- 1) Sistemos kūrimas (funkcijų realizavimas);
- 2) Sistemos panaudos atvejų, nefunkcinių reikalavimų išpildymas/realizavimas;
- 3) Unit ir Integration testų realizavimas sistemoje;
- 4) Priėmimo testų palaikymas.

### 7.2. Testuotojas (tester)

- 1) Testavimo scenarijų vykdymas remiantis testų scenarijais bei testavimo planu;
- 2) Naujų klaidų sistemoje identifikavimas;
- 3) Naudotojo vadovo sudarymas.

### 7.3. Projekto vadovas (PM – project manager)

- 1) Skirtingų rolių darbuotojų (programuotojų, testuotojų) darbų koordinavimas, remiantis sudarytu testavimo grafiku;
- 2) Darbų užbaigtumo užtikrinimas (ar atlikti visi numatyti reikiami darbai);
- 3) Bendravimas su galutiniu klientu;
- 4) Sistemos specifikacijos sudarymas, jei yra galimybė, bendraujant su galutiniu klientu;
- 5) Sistemos kūrimo resursų paskirstymas;
- 6) Komandos informavimas (susirinkimų organizavimas, kliento naujienų/naujų reikalavimų/pakeitimų transliavimas komandai).



## 8. Rezultatai

Po testavimo šie rezultatai yra norimi gauti:

- 1) Testavimo planas – atnaujintas dokumentas po testavimo
- 2) Atrastų neatitikimų sąrašas – per testavimo procesą pastebėti neatitikimai ir klaidos turėtų būti surašyti šitame dokumente
- 3) Kliento pasirašytas dokumentas – dokumentas, patvirtinantis funkcijų atitikimą ir gerai, pagal kliento norus, realizuotą sistemą.

## 9. Testavimo aplinka

Ši įranga su pateiktais minimaliais nustatymais turėtų būti tenkinama testavimo metu:

Serveris:

- 1) Xeon 4GHz, 8GB RAM, 512GB HDD, 256GB SSD,
- 2) Ubuntu Linux 18.04,
- 3) Nginx Web Server,
- 4) NodeJS 10.16,
- 5) MySQL duombazė,
- 6) Veikianti sistemos GraffitiTour versija.

Testavimo (vartotojo) kompiuteriai:

Chrome, Firefox bei Edge naršyklių testavimui:

- 1) AMD 2GHz, 2GB RAM, 512GB HDD
- 2) Microsoft Windows 10
- 3) Chrome v. 70
- 4) Firefox v. 75
- 5) Microsoft Edge v.40

Safari naršyklės testavimui:

- 1) MacBook Pro
- 2) 8GB RAM, 128 GB SSD, 512 GB HDD
- 3) 1.4 GHz quad-core Intel Core i5
- 4) Safari 11

Taip pat veikianti GraffitiTour kurioje būtų pridėti tokių rolių vartotojai:

- 1) Sistemos administratorius
- 2) Prisiregistravęs vartotojas 1
- 3) Prisiregistravęs vartotojas 2
- 4) Nepriregistravęs vartotojas

## 10. Testavimo skriptai

### 10.1. Prisijungimas

Aprašymas: neprisijungęs vartotojas turi galėti prisijungti prie sistemos

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais ir rolėmis

Testavimo žingsniai:

- 1) Vartotojas nueina į prisijungimo puslapį
- 2) Įvedami duomenys į prisijungimo formą
- 3) Paspaudžiamas prisijungimo mygtukas

Testavimo atvejai:

El. paštas	Slaptažodis	Norimas rezultatas
admin@admin.local	admin	Prisijungta
admin@admin.local	qweqwe	Prisijungimo klaida
Client1@client.local	Client1	Prisijungta
Client1@client.local	qweqwe	Prisijungimo klaida
Client2@client.local	Client2	Prisijungta
Client2@client.local	qweqwe	Prisijungimo klaida

### 10.2. Registracija

Aprašymas: neprisiregistravęs vartotojas turi galėti prisiregistruoti prie sistemos

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais ir rolėmis

Testavimo žingsniai:

- 1) Vartotojas nueina į registracijos puslapį
- 2) Įvedami duomenys į registracijos formą
- 3) Paspaudžiamas registracijos mygtukas

Testavimo atvejai:

Vartotojas	Slaptažodis	El. paštas	Norimas rezultatas
Administratorius	admin	admin@admin.local	Toks el. paštas jau egzistuoja
Administratorius	qweqwe	admin@admin1.local	Toks vartotojas jau egzistuoja
Vartotojas3	Client3	client@client.local	Registracija sėkminga

### 10.3. Graffiti peržiūra

Aprašymas: vartotojas turi turėti prieigą prie graffiti peržiūros

Pradiniai duomenys: duombazė su bent vienu graffiti informacija

Testavimo žingsniai:

- 1) Vartotojas nueina į žemėlapių puslapį
- 2) Pasirenkamas graffiti
- 3) Matoma graffiti informacija

#### 10.4. Vartotojo informacijos peržiūra

Aprašymas: vartotojas turi galėti peržiūrėti prisiregistravusio vartotojo informaciją

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais ir rolėmis

Testavimo žingsniai:

- 1) Vartotojas nueina į vartotojų statistikos puslapį
- 2) Pasirenkamas vartotojas
- 3) Matoma tam tikra vartotojo informacija

#### 10.5. Graffiti žemėlapių peržiūra

Aprašymas: vartotojas turi pasiekti graffiti žemėlapių informaciją

Pradiniai duomenys: -

Testavimo žingsniai:

- 1) Vartotojas nueina į pagrindinį
- 2) Turi būti matomas graffiti

#### 10.6. Prisijungimas per socialinį tinklą

Aprašymas: vartotojas turi galėti prisijungti prie sistemos per socialinį tinklą

Pradiniai duomenys: priregistruotas vartotojas kuris turi savo el. Paštą priregistruotą facebook paskyrą

Testavimo žingsniai:

- 1) Vartotojas nueina į prisijungimo puslapį
- 2) Paspaudžiamas facebook prisijungimo mygtukas
- 3) Įvedami facebook prisijungimo duomenys
- 4) Spaudžiamas prisijungimo prie facebook mygtukas

Testavimo atvejai:

Facebook El. paštas	Slaptažodis	Norimas rezultatas
admin@admin.local	admin	Prisijungimo klaida
Client3@gmail.com	qweqwe	Prisijungta

### 10.7. Graffiti maršruto generavimas

Aprašymas: vartotojas turi galėti sugeneruoti graffiti taškų maršrutą

Pradiniai duomenys: Arti vienas kito įdėti graffiti taškai

Testavimo žingsniai:

- 1) Vartotojas nueina į graffiti žemėlapi
- 2) Paspaudžiamas generavimo mygtukas
- 3) Gaunamas maršrutas

### 10.8. Slaptažodžio atkūrimas

Aprašymas: vartotojas turi galėti prisijungti prie sistemos per socialinį tinklą

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais ir rolėmis

Testavimo žingsniai:

- 1) Vartotojas nueina į prisijungimo puslapį
- 2) Paspaudžiamas slaptažodžio atkūrimo mygtukas
- 3) Įvedamas el.paštas
- 4) Spaudžiamas atkūrimo mygtukas
- 5) (el. pašto platformoje) paspaudžiama gauto el. pašto nuoroda
- 6) Atkūrimo puslapyje suvedamas naujas slaptažodis
- 7) Prisijungimo puslapyje įvedami nauji prisijungimo duomenys
- 8) Spaudžiamas prisijungimo mygtukas

Testavimo atvejai:

El. paštas	Naujas slaptažodis	Įvedamas el. paštas	Įvedamas slaptažodis	Norimas rezultatas
admin@admin.local	qweqwe	admin@admin.local	admin	Prisijungimo klaida
admin@admin.local	qweqwe	admin@admin.local	qweqwe	Prisijungta
admin@admin.local	qweqwe	admin1@admin.local	qweqwe	Prisijungimo klaida

## 10.9. Profilio informacijos keitimas

Aprašymas: vartotojas turi galėti pakeisti savo profilio informaciją

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais ir rolėmis

Testavimo žingsniai:

- 1) Prisijungęs vartotojas nueina į savo profilio puslapį
- 2) Paspaudžiamas profilio informacijos redagavimo mygtukas
- 3) Pakeičiamas vartotojo vardas
- 4) Spaudžiamas išsaugojimo mygtukas

Testavimo atvejai:

Senas vardas	vartotojo	Naujas vartotojo vardas	Norimas rezultatas
Administratorius		Administratorius1	Informacija pakeista
Administratorius		Vartotojas2	Klaida

## 10.10. Informacijos apie graffiti kėlimas

Aprašymas: vartotojas turi galėti įkelti graffiti informaciją

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais ir rolėmis

Testavimo žingsniai:

- 1) Prisijungęs vartotojas nueina į graffiti kėlimo puslapį
- 2) Suvedami duomenys
- 3) Spaudžiamas kėlimo mygtukas

Testavimo atvejai:

Graffiti pavadinimas	Nuotrauka	Pažymėta žemėlapyje	vieta	Norimas rezultatas
qweqwe	Įkelta	pažymėta		Įkelta
qweqwe	Neįkelta	pažymėta		Kėlimo klaida
(nėra)	Įkelta	pažymėta		Kėlimo klaida
(nėra)	Neįkelta	pažymėta		Kėlimo klaida

## 10.11. Graffiti vertinimas

Aprašymas: vartotojas turi galėti įvertinti graffiti

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais ir rolėmis ir įkeltas bent vienas graffiti

Testavimo žingsniai:

- 1) Prisijungęs vartotojas nueina į graffiti žemėlapių puslapį
- 2) Iš žemėlapių pasirenkamas graffiti
- 3) Graffiti informacijos puslapyje spaudžiamas vertinimo mygtukas
- 4) Suvedami vertinimo duomenys
- 5) Spaudžiamas patvirtinimo mygtukas

Testavimo atvejai:

Graffiti vertinimas	Komentaras	Norimas rezultatas
10/10	(nėra)	Įvertinta
(nepažymėta)	(nėra)	Klaida
(nepažymėta)	Patiko	Klaida

#### 10.12. Graffiti informacijos redagavimas

Aprašymas: administratorius turi galėti keisti graffiti informaciją

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais ir rolėmis ir įkeltas bent vienas graffiti

Testavimo žingsniai:

- 1) Prisijungęs administratorius nueina į graffiti sąrašo puslapį
- 2) Iš sąrašo pasirenkamas graffiti
- 3) Įvedama / pakeičiama informacija
- 4) Spaudžiamas išsaugojimo mygtukas

Testavimo atvejai:

Senas pavadinimas	Naujas pavadinimas	Senas nuotrauka	Nauja nuotrauka	Norimas rezultatas
qweqwe	(nėra)	Įkelta	Įkelta	Klaida
qweqwe	Qweqwe123	Įkelta	Įkelta	Pakeista
qweqwe	qweqwe	Įkelta	Įkelta nauja	Pakeista

#### 10.13. Vartotojų informacijos redagavimas

Aprašymas: administratorius turi galėti keisti vartotojų informaciją

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais

Testavimo žingsniai:

- 1) Prisijungęs administratorius nueina į vartotojų sąrašo puslapį
- 2) Iš sąrašo pasirenkamas vartotojas
- 3) Įvedama / pakeičiama informacija

#### 4) Spaudžiamas išsaugojimo mygtukas

Testavimo atvejai:

Senas vardas	vartotojo	Naujas vartotojo vardas	Norimas rezultatas
Vartotojas2		Vartotojas1	Klaida (toks jau egzistuoja)
Vartotojas2		Vartotojas2	Klaida (toks pats negali būti)
Vartotojas2		Vartotojas0	Išsaugota

#### 10.14. Vartotojų blokavimas / atblokavimas

Aprašymas: administratorius turi galėti blokuoti ir atblokuoti vartotojus

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais

Testavimo žingsniai:

- 1) Prisijungęs administratorius nueina į vartotojų sąrašo puslapį
- 2) Iš sąrašo pasirenkamas vartotojas
- 3) Spaudžiamas blokavimo / atblokavimo mygtukas

#### 10.15. Graffiti patvirtinimas

Aprašymas: administratorius turi galėti patvirtinti nepatvirtintus graffičius

Pradiniai duomenys: sukurta duombazė su nurodytais vartotojais ir su įkeltu bent vienu nepatvirtintu graffiti

Testavimo žingsniai:

- 1) Prisijungęs administratorius nueina į laukiančių patvirtinimo graffiti sąrašą
- 2) Iš sąrašo pasirenkamas graffiti
- 3) Spaudžiamas patvirtinimo mygtukas

### 11. Testavimo grafikas

Lentelėje žemiau yra pateikiamas testavimo grafikas, kuriomis datomis koks testavimas turi būti atliekamas

Testavimas	Nuo	Iki
------------	-----	-----

Unit testing	2019-10-05	2019-12-20
Integration testing	2019-10-05	2019-12-20
Interface testing	2019-12-20	2020-01-28
Validation testing	2019-12-20	2020-01-28
Acceptance testing	2020-01-28	2020-03-15
Security testing	2020-03-15	2020-04-01
Performance testing	2020-04-01	2020-04-20

## 12. Testavimo rizikos

Lentelėje žemiau yra pateikiamas testavimo rizikos

Rizika	Aprašymas	Rizikos išvengimas
Nepankamas testuotojų kiekis	Testuotojai gali būti priskirti kitiems projektams testuoti	Užtikrinti šio projekto testavimo svarbumą
Nepankamas testavimo prietaisų kiekis	Kiti testuotojai gali naudotis tais pačiais testavimo prietaisais, todėl gali atsirasti trūkumas	Sudaryti testavimo prietaisų naudojimo grafiką
Sugedęs testavimo prietaisas	Testavimo metu galimas prietaiso sugedimas	Užtikrinti atsargų naudojimą, parūpinti atsarginių prietaisų

Šiame skyriuje pateikiami testavimo scenarijai, kurie bus vykdomi testuojant kuriamą sistemą. Testavimo scenarijai aprašyti naudojant Gherkin testų rašymo kalbą.

## 13. Išvados

Laboratorinio darbo metu buvo išsirinkta testuojama sistema, ji išanalizuota, taip pat susipažinta su testavimo planu, jo struktūra. Parašytas testavimo planas pritaikytas išsirinktai testuojamai sistemai