

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

T120B162 Programų sistemų testavimas

Lab1. Testavimo planas.

Komanda

Atliko:

Aleksas Juzukonis IFF-1/8

Lukas Kuzmickas IFF-1/6

Mildaras Karvelis IFF-1/4

Priėmė:

doc. prakt. Guogis Evaldas

Turinys

Testavimo apimtis.	3
Testavimo strategijos.	6
Pradinės sąlygos.	7
Testavimo prioritetai.	7
Testavimo tikslai.	7
Testavimo technikos.	8
Rolės ir atsakomybės.	8
Rezultatai.	9
Testavimo aplinka.	9
Testų scenarijai.	13
Testų valdymas.	22
Testavimo tvarkaraštis.	22
Testavimo rizikos.	23
Nuorodos.	23

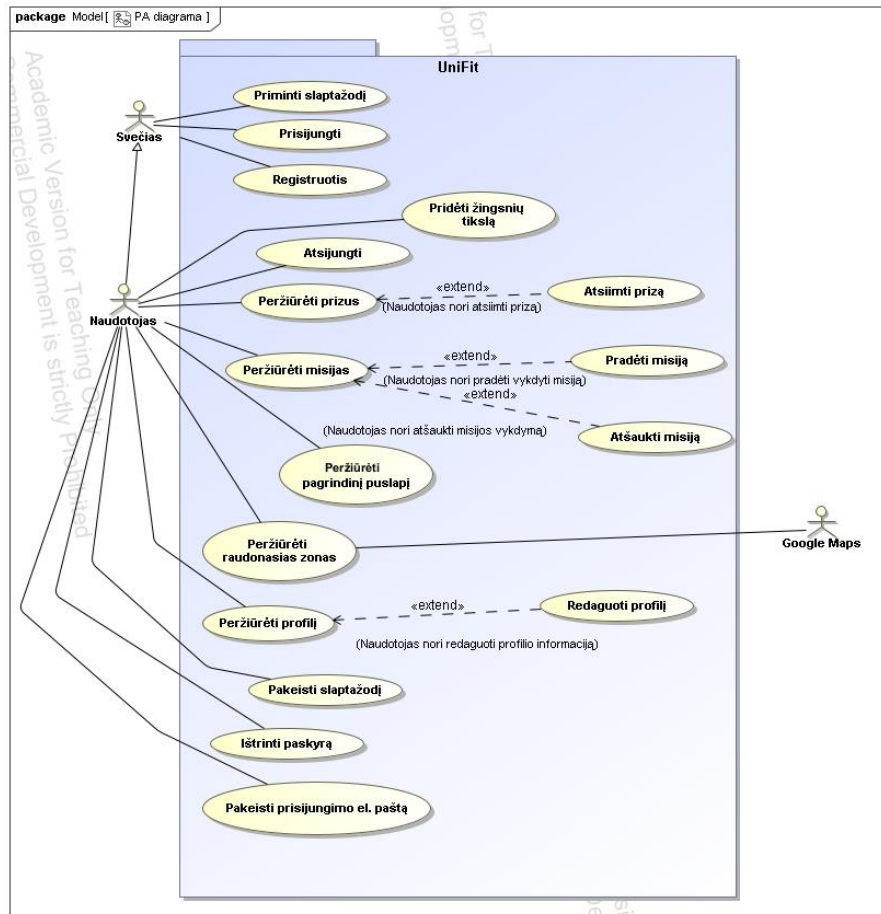
Testavimo apimtis.

- UniFit mobiliosios aplikacijos, versija v1.
- Panaudojimų atvejų modelis naudotojui.
- Naudotojo gidas.

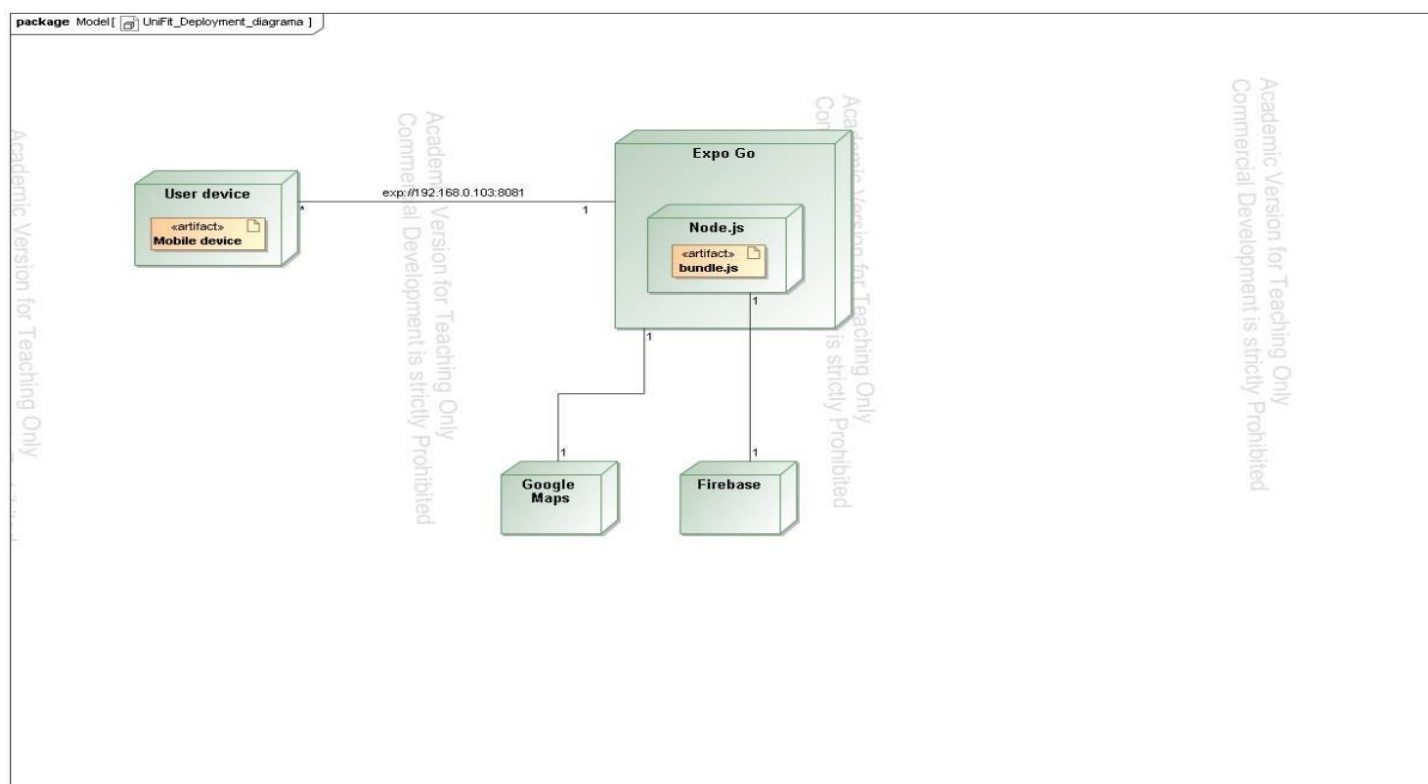
Testuojama žingsnių skaičiavimo programėlė „UniFit“. Šioje programėlėje galime rinkti taškus už kasdien suvaikščius žingsnius, turime misijas ir karštas zonas.



1 pav. „UniFit“ programėlės logotipas



2 pav. „UniFit“ programėlės PA diagrama.



3 pav. „UniFit“ programėlės Deployment diagrama

Šio testavimo tikslas – nustatyti, kaip įdiegta programėlė atitinka funkinius ir panaudojimo reikalavimus, taip pat nustatyti aptiktas problemas ir leisti jas išspręsti. Testavimas taip pat leistų pagerinti programėlės priežiūrą ir diegimą.

Testavimo strategijos.

Bus naudojami Priėmimo testai, nes tai parodo stabilų programinės įrangos veikimą ir patikimumą. Šis testavimas yra orientuojamas iš naudotojo perspektyvos, norima patikrinti, kaip programėlė bus naudojama kasdieniniuose veiklose ir ar atitinka savo kokybės reikalavimus. Tuo pačiu bus atliekami papildomi testavimai, kaip vienetų , integracijos bei saugumo ir našumo testai.

Pradinės sąlygos

Šias užduotis reikia atlikti prieš pradedant testavimo veiklą:

- Yra išsami programinės įrangos specifikacija, išreikšta atvejų ir naudojimo scenarijų modeliais;
- Veikianti ir įdiegta programinė įranga (t.y. programėlė);
- Nustatytos procedūros su problemų sprendimais testavimo metu;
- Apibrėžtas panaudojimų atvejų modelis priimtimumo testavimui;
- Sukurta testavimo aplinka;
- Pilnai paskirstyti testavimo ištekliai;
- Apibrėžti priėmimo testavimo standartai.

Testavimo prioritetai

Šios testavimo veiklos yra išvardytos mažėjančio prioriteto lygiu (pirmasis turi aukščiausią prioritetą):

1. Funkcijos – ar visos apibrėžtos programėlės funkcijos veikia taip, kaip tikėtasi?
2. Naudojamumas – ar programėlė patogi vartotojui?
3. Saugumas – ar duomenys apsaugoti?
4. Našumas – ar programėlė atitinka sutartus veikimo kriterijus?

Testavimo tikslai

1. Komponentų testavimas. Visi programinės įrangos komponentai yra išbandyti.
2. Integracijos testavimas. Programinės įrangos testavimas, siekiant užtikrinti, kad komponentai tinkamai sąveikauja.
3. Patvirtinimo testavimas. Programinės įrangos testavimas emuliuotoje gamybos aplinkoje (Android), siekiant patikrinti jos funkcionalumą.
4. Priėmimo / kvalifikacijos testavimas.
5. Programinės įrangos priėmimo testavimas, siekiant užtikrinti, kad jos funkcionalumas atitiktų galutinio vartotojo lūkesčius. Priėmimo / kvalifikacijos testavimas nustato, ar sistema atitinka savo priėmimo kriterijus, paprastai tikrinant pageidaujamą sistemos elgesį pagal kliento reikalavimus. Priėmimo testavimą sudaro paskutinis testų rinkinys, kuris atliekamas prieš oficialiai paleidžiant programinės įrangos sistemą.
6. Montavimo testavimas. Dažnai, baigus sistemos ir priėmimo testavimą, programinė įranga patikrinama ją įdiegus tikslinėje aplinkoje.
7. Alfa ir beta testavimas. Prieš išleidžiant programinę įrangą, ji kartais suteikiama nedidelei pasirinktai potencialių vartotojų grupei (kas šiuo metu yra grupelė studentų) bandomajam naudojimui ir (arba) didesnei reprezentatyvių vartotojų grupei.

8. Patikimumo pasiekimas ir įvertinimas. Testavimas padidina patikimumą nustatant ir ištaisant gedimus. Be to, statistinius patikimumo matavimus galima išvesti atsitiktinai generuojant bandomuosius atvejus pagal programinės įrangos veikimo profilį.
9. Regresinis testavimas. Praktiškai siekiama parodyti, kad programinė įranga vis tiek išlaiko anksčiau išlaikytus testus bandymų rinkinyje.
10. Veikimo testavimas. Patikrina, ar programinė įranga atitinka nurodytus našumo reikalavimus, ir įvertina veikimo charakteristikas, pavyzdžiui, pajėgumą ir atsako laiką.
11. Saugumo testavimas. Saugumo testavimas patikrina sistemų ir jos duomenų konfidencialumą, vientisumą ir prieinamumą. Neigiamas testas.
12. Sąsajos testavimas. Sąsajos testavimu siekiama patikrinti, ar komponentai tinkamai susieja, kad būtų galima tinkamai keistis duomenimis ir valdymo informacija.

Testavimo technikos

Bus naudojami šie bandymo metodai:

1. Testavimas scenarijais – scenarijuose naudojami atvejai (su iš anksto nustatytais įvesties ir numatomais išvesties duomenimis).
2. Testavimo scenarijai be duomenų – testuotojai testavimo metu pasirenks įvesties duomenis.
3. Neteisėto naudojimo testai – scenarijų sudarytos veiklos, kuriomis bandoma gauti prieigą prie neleistinų duomenų programinėje įrangoje.
4. Naudojamumo kontrolinis sąrašas – veiklos, skirtos įvertinti sistemų naudojimo patogumą.
5. Veiklos statistika – veiklos charakteristikų rinkimas ir palyginimas su apibrėžtais parametrais.
6. Vienetų, integracijos, priėmimo testai.

Rolės ir atsakomybės

Apibrėžiami šie vaidmenys ir pareigos:

1. Kokybės užtikrinimo vadovas – Lukas Kuzmickas, atsakingas už testavimo proceso planavimą ir jo vykdymą.
2. Testeris – Aleksas Juzukonis, atlieka testavimo veiklą, apibrėžtą testavimo plane.

3. Produktų vadybininkas – Mildaras Karvelis, užtikrina, kad testai būtų sėkmingai vykdomi iš vartotojo perspektyvos.

Rezultatai

Po bandymo turėtų būti prieinami šie rezultatai:

1. Testavimo planas – šis dokumentas su visais testavimo proceso metu atliktais pakeitimais.
2. Pakeitimų užklausa – dokumentas, aprašantis programėlės pakeitimus, atsiradusius dėl pasikeitusių reikalavimų arba testavimo metu aptiktų defektų.

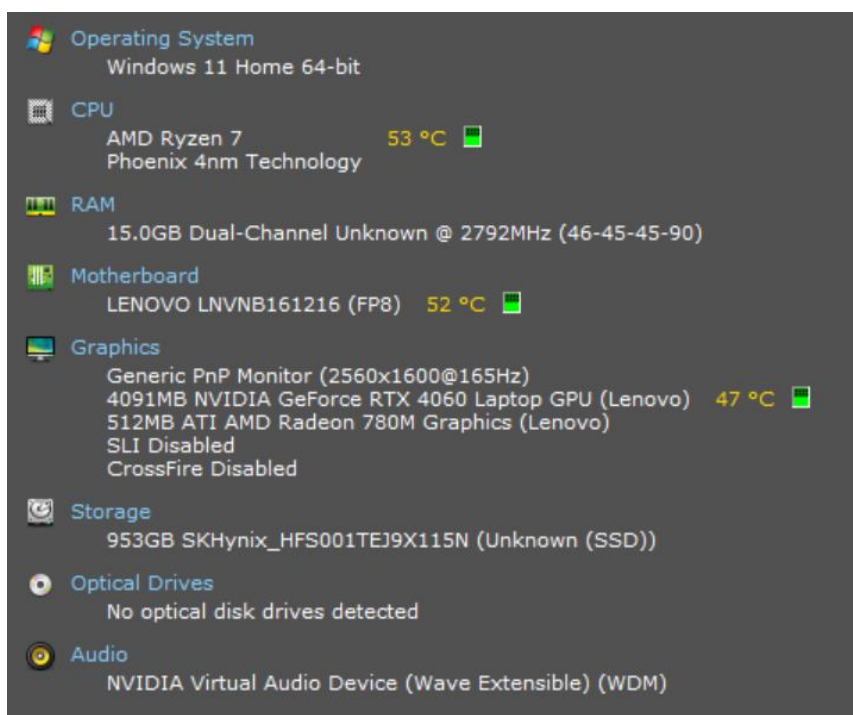
Testavimo aplinka

Kiekvienas komandos narys testuos programą ant savo įrangos, testavimui bus reikalingas kompiuteris. Testavimui galima naudoti telefoną arba emuliatorių.

Įrankiai, kurie bus naudojami:

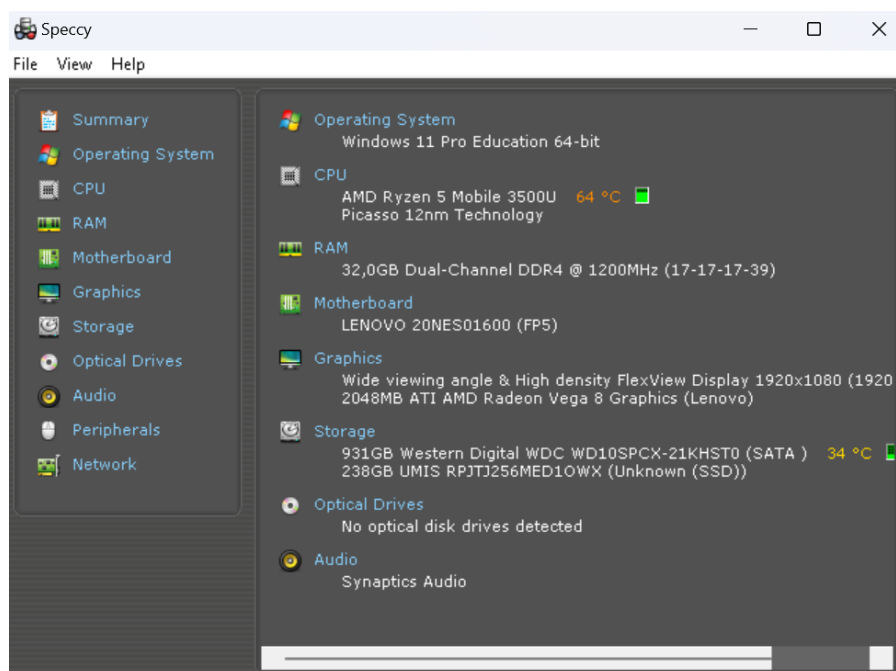
- Expo Go (Programėlė leidžianti pasileisti React-Native projektą per savo telefoną);
- Android Studio (Androido emulatorius);
- NodeJS (npm paketai);
- Visual Studio Code (kodo rašymui ir paleidimui);

Kompiuteris #1:



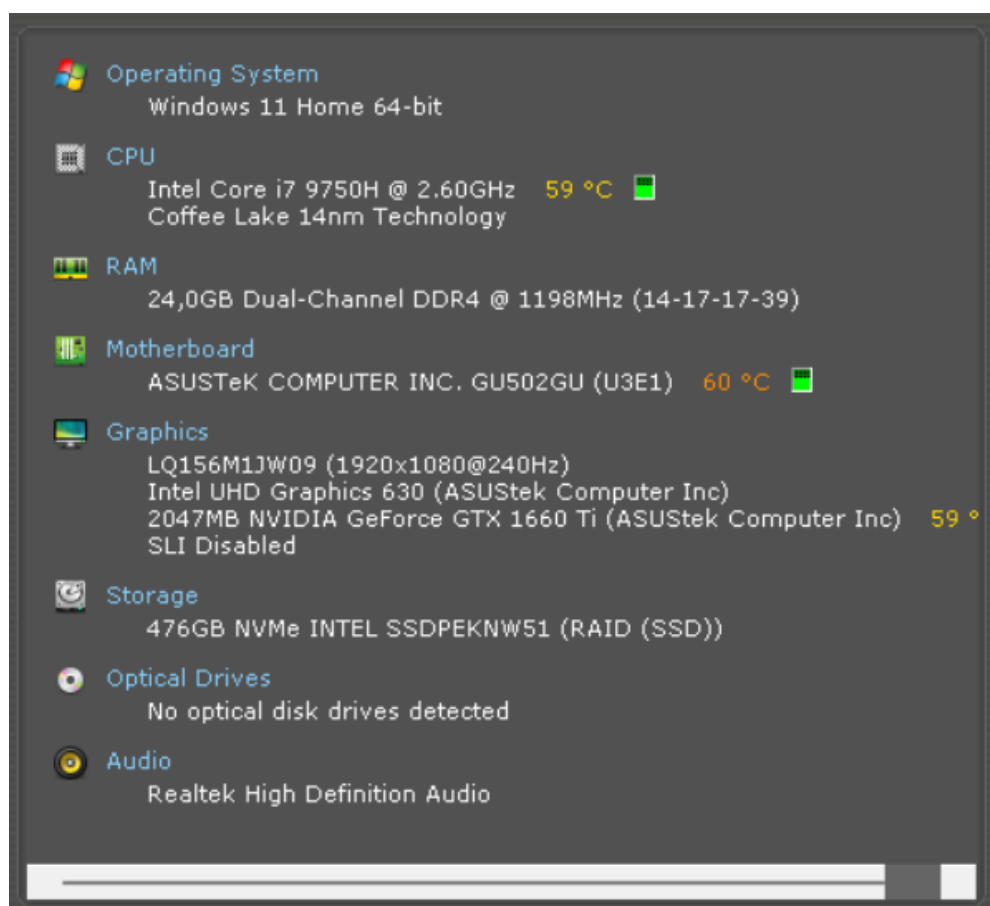
4 Pav. Kompiuterio #1 Specifikacija

Kompiuteris #2:



5 Pav. Kompiuterio #2 specifikacija

Kompiuteris #3:



6 Pav. Kompiuterio #3 specifikacija

Testų scenarijai

Bus naudojamas [gase.io](#) įrankis, testų scenarijų braižymui.

Testavimo scenarijus 01.0.1 Prisijungimas naudotojui (Užpildomi el. pašto ir slaptažodžio laukai)

UNIFIT-1

Autorizacija 01.0.1

EditCloneDeleteAI Test Case

GeneralRun historyChange historyDefectsComments

Description

Registruoti naudotojai gali prisijungti prie programėlės

Pre-conditions

Naudotojas turi būti užsiregistravęs

Post-conditions

Not set

Steps

	Action	Data	Expected result
1	Nueiti į prisijungimo puslapį		Prisijungimo puslapis yra matomas
2	Užpildyti prisijungimo formą su el. paštu "testas@gmail.com" ir slaptažodžiu "testas123"		Užpildoma prisijungimo forma
3	Paspausti "Prisijungti" mygtuką		Prisijungimo forma yra užpildyta duomenimis ir naudotojas yra nukreipiamas į pagrindinį programėlės puslapį

7 Pav. Testavimo scenarijus 01.0.1

Testavimo scenarijus 01.0.2 Prisijungimo klaida (Užpildomi el. pašto ir slaptažodžio laukai)

UNIFIT-2

Autorizacija 01.0.2

EditCloneDeleteAI Test Case

GeneralRun historyChange historyDefectsComments

Description

Nėra galimybės prisijungti prie programėlės neregistruotam naudotojui

Pre-conditions

Not set

Post-conditions

Not set

Steps

	Action	Data	Expected result
○ 1	Nueiti į prisijungimo puslapį		Prisijungimo puslapis yra matomas
○ 2	Užpildyti prisijungimo formą el. paštu "klaida@gmail.com" ir slaptažodžiu "klaida123"		Užpildoma prisijungimo forma
○ 3	Paspausti "Prisijungti" mygtuką		Išmetamas klaidos pranešimas apie neegzistuojančią naudotojo paskyrą arba blogai įvestus duomenis

8 Pav. Testavimo scenarijus 01.0.2

Testavimo scenarijus 01.0.3 Registracija (Užpildomi el. pašto ir slaptažodžio laukai)

UNIFIT-3

Autorizacija 01.0.3

EditCloneDeleteAI Test Case

GeneralRun historyChange historyDefectsComments

Description

Naudotojas negali prisiregistruoti jeigu jau turi paskyrą

Pre-conditions

Naudotojas jau turi priregistravęs paskyrą su nurodytu el. paštu

Post-conditions

Not set

Steps

	Action	Data	Expected result
○ 1	Nueiti į registracijos puslapį		Registracijos puslapis yra matomas
○ 2	Užpildyti registracijos formą su el. paštu "testas@gmail.com" ir slaptažodžiu "testas123"		Užpildoma registracijos forma
○ 3	Paspausti "Registruotis" mygtuką		Išmetamas klaidos pranešimas apie jau esamą naudotoją

9 Pav. Testavimo scenarijus 01.0.3

Testavimo scenarijus 01.0.4 Slaptažodžio pakeitimas (Užpildomas el. pašto laukas)

UNIFIT-4

Autorizacija 01.0.4

EditCloneDeleteAI Test Case

GeneralRun historyChange historyDefectsComments

Description

Naudotojas gali gauti slaptažodžio pakeitimo nuorodą į savo paskyros el. paštą

Pre-conditions

Naudotojas turi paskyrą

Post-conditions

Not set

Steps

	Action	Data	Expected result
○ 1	Nueti į "Užmiršau slaptažodį" puslapį		"Užmiršau slaptažodį" puslapis yra matomas
○ 2	Užpildyti slaptažodžio užmiršimo formą su el. paštu "testas@gmail.com"		Užpildoma slaptažodžio užmiršimo forma
○ 3	Paspausti "Pakeisti slaptažodį" mygtuką		Rodomas pranešimas apie išsiųstą žinutę į nurodytą el. paštą, jeigu naudotojas turi paskyrą programėlėje

10 Pav. Testavimo scenarijus 01.0.4

Testavimo scenarijus 01.0.5 Atsijungimas (Naudotojas jau prisijungęs prie programėlės)

UNIFIT-5

Autorizacija 01.0.5

EditCloneDeleteAI Test Case

GeneralRun historyChange historyDefectsComments

Description

Naudotojas gali atsijungti nuo programėlės

Pre-conditions

Naudotojas turi būti prisijungęs prie programėlės

Post-conditions

Not set

Steps

	Action	Data	Expected result
○ 1	Nueiti į naudotojo puslapį		Matomas naudotojo informacijos puslapis
○ 2	Paspausti "Atsijungti" mygtuką		Paspaudžiamas atsijungti mygtukas ir naudotojas nukreipiamas į prisijungimo puslapį

11 Pav. Testavimo scenarijus 01.0.5

Testavimo scenarijus 01.0.6 Registracija (Užpildomi el. pašto ir slaptažodžio laukai)

UNIFIT-6

Autorizacija 01.0.6

EditCloneDeleteAI Test Case

GeneralRun historyChange historyDefectsComments

Description

Naudotojas gali užsiregistruoti ir pradėti naudoti programėlę

Pre-conditions

Not set

Post-conditions

Not set

Steps

▼	Action	Data	Expected result
○ 1	Nueiti į registracijos puslapį		Matomas registracijos puslapis
○ 2	Užpildyti registracijos formą el. paštu "testas123@gmail.com" ir slaptažodžiu "testas123"		Užpildoma registracijos forma
○ 3	Paspausti "Registruotis" mygtuką		Rodoma žinutė apie sėkmingą registraciją, prašoma patvirtinti naudotojo el. paštą su kuriuo buvoma užsiregistruota, naudotojas nukreipiamas į prisijungimo puslapį

12 Pav. Testavimo scenarijus 01.0.6

UNIFIT-8

Profilis 01.0.8

EditCloneDeleteAI Test Case

GeneralRun historyChange historyDefectsComments

Description

Naudotojas gali keisti savo naudotojo duomenis, kaip svoris ir ūgis

Pre-conditions

Naudotojas yra prisijungęs prie programėlės

Post-conditions

Not set

Steps

Shown as table

▼	Action	Data	Expected result	Attachments
○ 1	Esmė profilio puslapyje arba nueiname link jo		Matomas naudotojo profilio puslapis	Screenshot_1728727866.png
○ 2	Užpildome svorio ir ūgio formas naujais duomenimis	Weight=78; Height=178	Užpildome svorio ir ūgio formas	Screenshot_1728728105.png
○ 3	Spaudžiame "Save" išsaugojimo mygtuką		Yra matomas sėkmės pranešimas apie atnaujintus svorio ir ūgio duomenis	Screenshot_1728728187.png
○ 4	Esmė profilio puslapyje arba nueiname link jo		Matomas naudotojo profilio puslapis su įvestomis reikšmėmis	Screenshot_1728728253.png

13 Pav. Testavimo scenarijus 01.0.8

UNIFIT-9

Profilis 01.0.9

EditCloneDeleteAI Test Case

GeneralRun historyChange historyDefectsComments

Description

Naudotojas gali pakeisti savo slaptažodį

Pre-conditions









Naudotojas turi būti prisijungęs prie programėlės

Post-conditions

Not set

Steps

Shown as table

	Action	Data	Expected result	Attachments
1	Nuename link naudotojo profilio puslapio		Naudotojo profilio puslapis yra matomas	 Screenshot_1728728444.png
2	Spaudžiame šalia "Password" slaptažodžio dalies, šalia esanti užrašą "Change"		Yra matomas slaptažodžio keitimo formos langas	 Screenshot_1728728538.png
3	Užpildome slaptažodžio keitimo formą	new_password="slaptazodis123"; confirm_new_password="slaptazodis123"	Užpildoma slaptažodžio keitimo forma	 Screenshot_1728728671.png
4	Spaudžiame "Change" mygtuką		Yra matomas slaptažodžio pakeitimo sėkmės pranešimas	 Screenshot_1728728852.png
5	Nuename link naudotojo profilio puslapio ir spaudžiame "Log Out" (atsijungimo) mygtuką		Yra matomas prisijungimo langas	 Screenshot_1728728896.png  Screenshot_1728728945.png
6	Užpildome prisijungimo formą	email="test_case@gmail.com"; password="slaptazodis123";	Užpildoma prisijungimo forma	 Screenshot_1728729120.png
7	Spaudžiame "Login" mygtuką		Yra matomas pagrindinis puslapis	 Screenshot_1728729168.png

14 Pav. Testavimo scenarijus 01.0.09

UNIFIT-10

Missions 01.0.1

EditCloneDeleteAI Test Case

GeneralRun historyChange historyDefectsComments

Description

Naudotojas gali pradėti kalorijų deginimo misiją

Pre-conditions

Nauditijas yra prisijungęs prie programėlės

Post-conditions

Not set

Steps

Shown as table

	Action	Data	Expected result	Attachments
1	Nueiti iki misijų puslapio		Matome misijų puslapį	
2	Paspausti ant "Burn 6 "Big Mac's" mygtuko		Rodoma kiek kalorijų sudegino nuo misijos pradžios	

15 Pav. Testavimo scenarijus 01.0.10

UNIFIT-11

Missions 01.0.2

Edit

Clone

Delete

AI Test Case

General

Run history

Change history

Defects

Comments

Description

Atšaukti misija

Pre-conditions

Naudotojas turi būti prisijungęs ir pradėjas misiją

Post-conditions

Not set

Steps

Shown as table

	Action	Data	Expected result	Attachments
1	Nueiti iki misijų puslapio		Matome misijų puslapį	
2	Paspausti "cancel" mygtuką		Nebematome sudegintų kalorijų ir galime per naują pradėti misiją	

16 Pav. Testavimo scenarijus 01.0.11

UNIFIT-12

Parduotuvė 01.0.12

Edit

Clone

Delete

AI Test Case

General

Run history

Change history

Defects

Comments

Description

Naudotojas gali nusipirkti prekes iš parduotuvės.

Pre-conditions

Naudotojas yra prisijungęs prie programėlės ir turi pakankamai taškų.

Post-conditions

Not set

Steps

Shown as table

	Action	Data	Expected result	Attachments
1	Esame parduotuvės puslapyje arba nuolainam link jo.		Matomos parduotuvės.	Screenshot_1728847014.png
2	Pasirenkame vieną iš parduotuvių.		Matomos pasirinktos parduotuvės prekės.	Screenshot_1728847108.png
3	Pasirenkame vieną iš prekių.		Matomas patvirtinimo langas.	Screenshot_1728847185.png
4	Paspaudžiame "ok" atsidariusiame patvirtinimo lange.		Matomas sėkmingo pranešimo langas.	Screenshot_1728847240.png
5	Pakartojame 1-3 žingsnius.		Matome, jog taškų kiekis sumažėjo, o prekės būseną pasikeitė į "Already Redeemed"	Screenshot_1728847342.png Screenshot_1728847345.png

17 Pav. Testavimo scenarijus 01.0.12

UNIFIT-13

Parduotuvė 01.0.13

EditCloneDeleteAI Test Case

GeneralRun historyChange historyDefectsComments

Description

Naudotojas negali nusipirkti prekės iš parduotuvės.

Pre-conditions

Vartotojas yra prisijungęs ir neturi pakankamai taškų prekėms.

Post-conditions

Not set

Steps

Shown as table

	Action	Data	Expected result	Attachments
1	Esame parduotuvės puslapyje arba nuiname iki jo.		Matomos parduotuvės.	Screenshot_1728847508.png
2	Pasirenkame vieną iš parduotuvių.		Matomos parduotuvės prekės.	Screenshot_1728847543.png
3	Pasirenkame vieną iš prekių.		Prekės būseną yra pilkame fone su žodžiu "Redeem", mygtuko paspaudimas neveikia.	Screenshot_1728847622.png

18 Pav. Testavimo scenarijus 01.0.13

Testų valdymas

Testai bus registruojami testavimo plane ir qase.io aplinkoje.

Registruojant testą reikės nurodyti:

1. Testo pavadinimą
2. Testavimo tikslą
3. Testavimo scenarijų aprašymą
4. Būseną
5. Prioritetą
6. Tipą

Rasti defektai taip pat bus registruojami testavimo plane.

Registruojant defektą reikės nurodyti:

1. Defekto pavadinimą ir aprašymą
2. Defekto atsiradimo sąlygas
3. Nurodytus testų scenarijus, kurie defektą aptiko
4. Defekto būseną

Atliekant testus bus stebima:

1. Kiek testų įvykdytą
2. Kokie testų rezultatai
3. Rastų defektų skaičių ir jų būklę

Testavimo tvarkaraštis

Testavimo užduotis	Pradžia	Terminas
Vienetų testai	2024-10-01	2024-10-25
Integracijos testai	2024-11-01	2024-11-25
Naudotojo sąsajos testai	2024-12-01	2024-12-25
Sistemos testai	2025-01-01	2025-01-25

Testavimo rizikos

Galima rizika, galinti turėti įtakos testavimo procesui:

Rizika	Aprašymas	Poveikio sumažinimas
Duomenų konfidencialumas	Firebase duomenų bazė gali neužtikrinti duomenų konfidencialumo	Naudoti netikrus duomenis testavimo metu, iki kol konfidencialumas bus užtikrintas
Testavimo ištekliai	Testuojama programa gali labai lėtai dirbti arba nulūžti, dėl prastų resursų išnaudojimo	Testuoti atskiras programos dalis, stebėti programos naudojamus resursus.

Nuorodos

1. IEEE 829-2008. IEEE Standard for Software and System Test Documentation.
2. <https://specflow.org/learn/gherkin/>
3. <https://qase.io/>