



Kauno technologijos universitetas
Informatikos fakultetas

Programų sistemų testavimas (P170B400)

Socialinio tinklo „Sent” testavimo planas

Darbo autoriai:

Komanda „Sourcerers“:

Benas Andriuščenka, IFF-2/5

Aleksas Balčiukynas, IFF-2/5

Mantvydas Deltuva, IFF-2/5

Justinas Teselis, IFF-2/5

Darbą priėmė:

Lekt. Eligijus Kiudys

KAUNAS, 2024

Turinys

1.	Įvadas.....	3
2.	Testavimo apimtis	3
3.	Testavimo strategijos.....	3
4.	Pradinės sąlygos	3
5.	Testavimo prioritetai.....	4
6.	Testavimo tikslai	4
7.	Testavimo technikos	4
8.	Rolės ir atsakomybės	5
9.	Rezultatai	5
10.	Testavimo aplinka	5
11.	Testų scenarijai	5
12.	Testų valdymas	8
13.	Testavimo tvarkaraštis.....	9
14.	Testavimo rizikos	9

1. Įvadas

Čia yra testavimo planas internetiniai aplikacijai – socialiniui tinklui „Sent“. Kuriant „Sent“ bus naudojamas Node.js ir jo bibliotekos kaip React, Express. Aplikacija turės daugelį funkcijų, prieinamų tiek vartotojams, tiek administratoriams. Vartotojai galės kurti, skaityti, redaguoti įrašus, komentuoti, redaguoti savo paskyrą, administratoriai galės trinti vartotojų įrašus.

2. Testavimo apimtis

Testavimas susidės iš:

1. Funkcinio testavimo naudotojo ir administratoriaus atžvilgiu; (rankiniu būdu)
2. Našumo testų; (Jmeter)
3. Vienetų testų; (Jest)
4. API testai. (Postman)

Vienetų testais turi būti padengta kiek įmanoma daugiau metodų/funkcijų.

Šio testavimo tikslas yra surasti funkcinis neatitikimus aplikacijoje „Sent“, tiek vartotojo, tiek administratoriaus teisėmis. Bus itin daug dėmesio skiriama funkciniam reikalavimui, testavimo planas padės regresiniam testavimui ateityje, ypač jei projektas atsidurtų palaikymo fazėje. Taip pat bus testuojamas našumas, kad patikrintume ar aplikaciją įmanoma patogiau naudoti esant ne vienam prijungtam vartotojui.

3. Testavimo strategijos

Kiekvienos funkcijos testų kiekis priklausys nuo funkcijos reikšmingumo, naudojimo dažnumo, klaidų rizikos dydžio.

Testavimą sudarys tiek rankinis, tiek automatizuotas testavimas, vienetų testai, komponentų testai.

Ištestavus didžiąją dalį pagrindinių funkcijų, kai programa bus funkcionali ir joje bus galima atlikti pagrindines funkcijas kaip įrašų kūrimas, peržiūra, vartotojų registracija ir prisijungimas, tada bus galima pradėti atlikinėti našumo testus.

Teksto laukų veikimą reiks ištestuoti tiek su validžiomis, tiek su nevalidžiomis reikšmėmis, validaciją ištestuoti tiek iš front-end pusės, tiek iš back-end pusės.

4. Pradinės sąlygos

1. Veikianti internetinė aplikacija (bent veikia prisijungimas ir įrašų kūrimas ir skaitymas);
2. Susitarta dėl klaidų taisymo proceso;
3. Susitarta dėl rastų klaidų aprašymo šablono;
4. Sukurti testų scenarijai priėmimo testavimui;
5. Sukurta aplinka testavimui.

5. Testavimo prioritetai

1. Funkcijos – ar funkcijos veikia kaip tikimasi;
2. Naudojamumas – ar aplikacijos dizainas yra intuityvus ir aiškus;
3. Našumas – ar aplikacija veikia pakankamu greičiu;
4. Prieinamumas – ar aplikacijoje galima naviguoti tab'ų pagalba, ar tvarkinga html struktūra.

6. Testavimo tikslai

1. Vartotojo sąsajos testavimas:

Tikslas: įsitikinti, kad aplikacija yra intuityvi, lengva naudoti galutiniam vartotojui, patogi, nekelia nusivylimo.

2. Naršyklių suderinamumo testavimas:

Tikslas: įsitikinti, kad aplikacija veikia teisingai su keliomis skirtingomis naršyklėmis, kad kuo daugiau naudotojų galėtų naudotis aplikacija.

3. Testavimas mobiliuosiuose įrenginiuose:

Tikslas: įsitikinti, kad naudojant mobiliuosius įrenginius aplikacija yra tvarkinga, lengvai naudojama, viskas padidėję iki pakankamo dydžio.

4. Našumo testavimas:

Tikslas: Įvertinti aplikacijos greitį ir pajėgumą sutalpinti tam tikrą kiekį žmonių, kad būtų galima pritaikyti serverių apimtis ar optimizuoti programą pagal tikimąsi vartotojų kiekį.

5. Vartotojo autentifikacijos testavimas:

Tikslas: Validuoti, ar vartotojo prisijungimo mechanizmas veikia sklandžiai ir saugiai.

6. Funkcijų testavimas:

Tikslas: Ištestuoti, ar kiekviena aplikacijos funkcija veikia sklandžiai, visas funkcionalumas atitinka reikalavimus ir veikia visais atvejais.

7. Testavimo technikos

Testų scenarijai – naudojimo scenarijai su iš anksto numatytais duomenimis.

Regresijos testavimas – automatizuoti kai kurie naudojimo atvejai, kad užtikrintume jog po pakeitimo viskas vis dar veikia kaip tikimasi.

Ribinių reikšmių analizė – įsitikinti, kad ribiniais atvejais funkcijos veikia kaip tikimasi, jeigu reikalavimai kaip elgiasi funkcija ribiniais atvejais nenumatyti – juos aptarti su komanda ir išsiaiškinti iki galo.

API testavimas – įsitikinti, kad iš back-end pusės viskas veikia tvarkingai, duomenys yra validuojami ir gražinami tinkamu formatu, nėra perteklinės informacijos.

8. Rolės ir atsakomybės

Benas Andriuščenka – Product manager

Aleksas Balčiukynas – Tester

Mantvydas Deltuva – Test support

Justinas Teselis – Quality assurance lead

9. Rezultatai

Testavimo planas – dokumentas, kur dokumentuotas testavimo procesas

Metrikos – aktualios metrikos apie aplikaciją ir testavimą, kaip klaidų dažnumas, kodo eilučių padengimas, kelių padengimas, našumo metrikos

10. Testavimo aplinka

Testavimui bus naudojama lokali bei esant galimybėm host'inta aplinka. Bus testuojama su Windows 10/Windows 11 operacinėmis sistemomis, Firefox bei Google Chrome naršyklėmis. Taip pat naudojant „developer tools“ bus dirbtinai sulėtintas internetas bei procesoriaus greitaveika.

11. Testų scenarijai

Aprašas: ištestuoti prisijungimo funkciją.

Pradiniai duomenys: Sukurtas validus vartotojas. Vietoj „Valid username“, „Valid email“, „Valid password“ atitinkamai įdėti sukurto validaus vartotojo duomenis.

Žingsniai:

- 1) Nueiti į prisijungimo puslapį;
- 2) Suvesti žemiau nurodytus duomenis (kadangi dabar nėra duomenų sistemoje, nurodyti placeholderiai vietoj kurių reikės įsidėti validžius arba nevalidžius duomenis savo nuožiūra).

Username	Email	Password	Expected result
-	Valid email	Valid password	Logged in
Valid username	-	Valid password	Logged in

Invalid username	-	Invalid password	Not logged in
-	Invalid email	Invalid password	Not logged in
Valid username	-	Invalid password	Not logged in
-	Valid email	Invalid password	Not logged in

Aprašas: Ištestuoti atsijungimo funkciją

Pradiniai duomenys: pradinė duomenų bazė.

Žingsniai:

- 1) Prisijungti prie vartotojo
- 2) Paspausti ant atsijungimo mygtuko. Turėtume atsidurti prisijungimo lange
- 3) Paspausti naršyklėje grįžimo atgal mygtuką. Turėtume vėl atsidurti prisijungimo lange, o ne pagrindiniame puslapyje.

Aprašas: ištestuoti registracijos funkciją. Vartotojo vardo ilgis neturėtų viršyti 20 simbolių

Pradiniai duomenys: pradinė duomenų bazė.

Žingsniai:

- 1) Nueiti į registracijos puslapį;
- 2) Suvesti žemiau nurodytus duomenis.

Username	Password	Email	Expected result
Valid username	Valid password	Valid email	Registered successfully
-	Valid password	Valid email	Registration unsuccessful
Valid username	-	Valid email	Registration unsuccessful
Valid username	Valid password	-	Registration unsuccessful
Valid username	Valid password	Non-existent email	Registration unsuccessful

Aprašas: ištestuoti įrašų kūrimo ir redagavimo funkciją. Antraštės maksimalus ilgis turi būti 50 simbolių, turinio 280.

Pradiniai duomenys: pradinė duomenų bazė

(visus žingsnius kartoti tiek su kūrimu, tiek su redagavimu)

Žingsniai:

- 1) Nueiti į pagrindinį puslapį;
- 2) Spausti ant įrašų sukūrimo mygtuko (arba redaguoti pasirinktą įrašą);
- 3) Suvesti žemiau nurodytus duomenis.

Title	Content	Expected result
„Valid title“	„Valid content“	Post created successfully
-	„Valid content“	Post creation unsuccessful
„Valid title“	-	Post creation unsuccessful
-	-	Post creation unsuccessful

„Valid title“	281 characters of content	Post creation unsuccessful
„Valid title“	280 characters of content	Post created successfully
51 characters title	„Valid content“	Post creation unsuccessful
50 characters title	„Valid content“	Post created successfully

Aprašas: ištutuoti įrašų trinimo funkciją.

Pradiniai duomenys: pradinė duomenų bazė.

Žingsniai:

- 1) Nueiti į pagrindinį puslapį;
- 2) Sukurti įrašą su validžiais duomenimis;
- 3) Surasti jį pagrindiniame puslapyje;
- 4) Spausti trinimo mygtuką (turėtų iššokti patvirtinimo mygtukas);
- 5) Sutikti su triniu.

„Expected result“: įrašas yra ištrintas, perkrovus puslapį jis vis dar ištrintas. (Jeigu yra galimybė, patikrinti duomenų bazę ir pažiūrėti ar įrašo nebėra ir ten).

Aprašas: ištutuoti komentarų kūrimo funkciją. Simbolių limitas turi būti 280.

Pradiniai duomenys: pradinė duomenų bazė.

Žingsniai:

- 1) Nueiti į pagrindinį puslapį;
- 2) Spausti ant įrašo;
- 3) Ištutuoti žemiau esančius atvejus:

Comment	Expected result
„Valid comment“	Comment created successfully
-	Comment created unsuccessfully
281 characters comment	Comment created unsuccessfully
280 characters comment	Comment created successfully

Aprašas: Ištutuoti naudotojų sekimo funkciją. Turėtų būti galima sėkmingai pridėti ir šalinti sekamus žmones.

Pradiniai duomenys: pradinė duomenų bazė.

Žingsniai:

- 1) Jeigu dar nėra kitų vartotojų sukurtu turinio, tai prisijungti prie kito vartotojo ir sukurti įrašą
- 2) Grįžus atgal į pradinį vartotoją turėtų matyti įrašą ir vartotojo, paskelbusio jį username. Reikia spustelt ant jo
- 3) Paspausti „follow“ mygtuką. Po šio veiksmo mygtukas turėtų pavirsti į unfollow mygtuką.
- 4) Nueiti į savo paskyrą ir paspausti ant „following“ mygtuko. Turėtų iššokti sąrašas, kuriame turėtų matytis ką tik follow'into vartotojo ikona ir username bei šalia mygtukas

unfollow. Sąrašas turėtų būti surūšiuotas abėcėlės tvarka pagal sekamų vartotojų usernames.

- 5) Paspausti šiame sąraše mygtuką unfollow. Jis turėtų pavirsti į mygtuką follow, bet iš sąrašo be puslapio perkrovimo ar išėjimo ir grįžimo į sąrašą vartotojas neturėtų dingti iš sąrašo.
- 6) Perkrauti puslapį. Unfollow'intas vartotojas turėtų būti dingęs iš sekamų vartotojų sąrašo.

Aprašas: Ištestuoti profilio redagavimo funkcionalumą. Vartotojas turėtų galėti pakeisti savo icon, display name, description.

Pradiniai duomenys: pradinė duomenų bazė.

Žingsniai:

- 1) Nunaviguoti iki savo profilio
- 2) Paspausti ant redagavimo mygtuko
- 3) Pakeisti display name į kažką kito ir išsaugoti. Display name turėtų būti sėkmingai pasikeitęs.
- 4) Vėl spausti ant redagavimo mygtuko.
- 5) Redaguoti profilio nuotrauką
- 6) Įkelti nuotrauką iš kompiuterio
- 7) Išsaugoti pakeitimus
- 8) Profilio nuotrauka turėtų būti sėkmingai pasikeitus

Aprašas: Ištestuoti pranešimų funkcionalumą. Vartotojas turėtų gauti pranešimus kai kažkas komentuoja ant jo įrašo, kai kažkas uždeda patiktuką ant jo įrašo, kai kažkas „tag‘ina“ tavo įrašą, kai gauni naują sekėją.

Pradiniai duomenys: pradinė duomenų bazė, turėti 2 vartotojus

Žingsniai:

- 1) Prisijungti prie pirmo vartotojo paskyros.
- 2) Pakomentuoti ant kito vartotojo įrašo, uždėti ant jo patiktuką, sukurti įrašą, kur tag‘inamas tas vartotojas (@vartotojo_username), sekti kitą vartotoją
- 3) Nuėjus ant kito vartotojo profilio prie pranešimų mygtuko turėtų matytis skaičius 4. Paspaudus ant pranešimų mygtuko (varpelio ikona), turėtų matytis visi 4 pranešimai.
- 4) Paspausti ant komentaro pranešimo. Turėtų nunaviguoti iki to komentaro.
- 5) Paspausti ant patikto pranešimo. Turėtų nunaviguoti iki to įrašo, ant kurio buvo uždėtas patiktukas
- 6) Paspausti ant tag‘inimo pranešimo. Turėtų nunaviguoti iki įrašo, kur buvau patag‘intas
- 7) Paspausti ant sekėjo pranešimo. Turėtų nunaviguoti iki sekančio žmogaus profilio.

12. Testų valdymas

1. Testų paleidimas;
2. Testų rezultatų aptarimas;
3. Defektų taisymas;
4. Testavimo aplinkos konfigūravimas;
5. Automatinių testų paleidimo atvejų sudarymas.

13. Testavimo tvarkaraštis

Testavimo užduotis	Pradžia	Pabaiga
Vienetų testai	2024-03-15	2024-05-10
Integravimo testai	2024-04-01	2024-05-10
Vartotojo sąsajos testai	2024-04-01	2024-05-14
Sistemos testavimas	2024-04-15	2024-05-14
Priėmimo testavimas	2024-05-01	2024-05-20

14. Testavimo rizikos

Našumo testavimui gali sutrukdyti mažas biudžetas serveriams, jeigu vienintelis būdas pasileisti serverį taptų lokalus, našumo testavimo atlikti iš vis nepavyks. Naudojant nemokamus serverius, aplikacijos našumas gali būti itin mažas ir tada našumo testavimas būtų bereikšmis.

Funkcinio testavimo rizikos apima reikalavimų trūkumą, dviprasmybės pačių testų scenarijų apraše, skirtumai tarp naršyklių ir aparatūros testuojant.

Kai kuriom funkcijom/metodam gali būti itin sunku sukurti vienetų testus. Tokiais atvejais gali būti verta pergaltvoti metodų/funkcijų struktūrą