**Įvadas**

Šis dokumentas aprašo testavimo planą automatizuojant serverio užsakymo procesą iš vartotojo pusės. Šio plano tikslas yra užtikrinti, kad serverio užsakymas veiktų su įvairiais parametrų rinkiniais. Dokumentas apima testavimo plano apimtį, testavimo tikslus, testavimo strategiją, testavimo rezultatų pateikimą, testavimo aplinką ir testavimo planą.

**Apimtis**

* Testuoti API taško serverio užsakymui veikimą siunčiant užklausas su atsitiktiniais parametrais.
* Testuoti API taško serverio užsakymui veikimą siunčiant užklausas su norimais parametrais (regionas, serveris, OS šablonas).
* Testuoti API taško gebėjimus apdoroti klaidingas arba netikėtas užklausas.
* Patikrinti ar atsakymai į užklausas pateikia tinkamus duomenis bei HTTP statuso kodus.

**Testavimo tikslai**

* Įsitikinti, kad API gali apdoroti netikėtas arba netinkamas užklausas nesukeliant sistemos trikdžių.
* Patikrinti iš API gautus rezultatus.
* Registruoti užklausų ir atsakymų duomenis ir suteikti galimybę atlikti rezultatų analizę.

**Testavimo strategija**

Naudojama ,,Juodos dėžės“ testavimo metodika API testavimui, nežinant vidinio API kodo ir veikimo. Testavimas atliekamas automatiškai per terminalą paleidus programą ,,beždžionę“ tam tikru laiko intervalu. ,,Beždžionė“ bando užsakyti serverį atsitiktinai parenkant galimus parametrus arba paleidimo metu uždedant tam tikrus filtrus (pvz., norimas regionas). Programos veikimo aprašymas:

1. Per API taškus surenkami galimi duomenys užklausai formuoti:

* Regionų sąrašas.
* Planų sąrašas.
* Kiekvieno plano OS šablonų sąrašas.

1. Suformuojama užklausa:
   1. Visus parametrus parenkant atsitiktinai
   2. Parametrus atrenkant pritaikius įvestus filtrus ir parametrus per CLI
      * –timeout – nurodomas maksimalus sveikas skaičius laiko limitui minutėmis
      * –testcount – nurodomas atliekamų testų skaičius
      * –random – užsakyti serverį parenkant visiškai atsitiktinius parametrus. Šis parametras nesuderinamas su kitais 4 žemiau išvardytais filtrais
      * –region pavadinimas – užsakyti serverius tik nurodytame regione
      * –plan pavadinimas – užsakyti konkretų produktą
      * –image pavadinimas – užsakyti serverį su nurodytu OS šablonu
      * –category pavadinimas – užsakyti serverį iš nurodytos kategorijos

Šiuos 4 filtrus galima kombinuoti. Nenurodyti parametrai bus parenkami atsitiktinai iš galimų variantų pritaikius jau nurodytus filtrus ant visų planų sąrašo.

* 1. Naudojant standartinius parametrus

1. Vykdomas kreipimasis į API tašką siunčiant sugeneruotą užklausą.
2. Gaunamas teigiamas arba neigiamas rezultatas ir informacija apie jį išsaugoma duomenų bazėje.

**Testavimo rezultatai**

Gauti testavimo rezultatai bus saugomi MySQL duomenų bazėje. Testo rezultatuose bus nurodyta kiekvieno testo būsena, vykdymo laikas, unikalus ID ir išsaugoti užklausos bei gauto atsakymo duomenys.

**Testavimo aplinka**

Testavimas atliekamas su realiu produkcijoje veikiančiu API. Programos ,,beždžionės“ rašymui naudojama C# programavimo kalba ir jos naudojamos bibliotekos bei karkasai.

Beždžionės paleidimui per CLI:

* .NET 7.0
* RestSharp 110.2.0
* Newtonsoft.Json 13.0.3
* Microsoft.Extensions.Configuration.Json 7.0.0
* Microsoft.Extensions.FileProviders.Abstractions
* Microsoft.Extensions.FileProviders.Physical
* Microsoft.Extensions.Primitives
* Microsoft.Extensions.Configuration.FileExtensions
* NUnit 3.13.3
* CommandLineParser 2.9.1

Papildomai projekto unit testų paleidimui iš Visual Studio aplinkos:

* NUnit3TestAdapter 4.4.2
* NUnit.Analyzers 3.6.1
* Microsoft.NET.Test.Sdk 17.5.0
* Coverlet.collector 3.1.2

Parašyta programa yra konsolės tipo ir sukompiliuotas vykdomasis failas gali būti paleidžiamas skirtingose operacinėse sistemose, kuriose įdiegtas tinkamas .NET Runtime su reikalingais bibliotekų dll failais.

**Testavimo rezultatai**

Kiekvieno testo rezultatuose saugoma informacija:

1. Testo vykdymo laikas.
2. Testo rezultatas – sėkmė arba nesėkmė.
3. Siunčiamos užklausos parametrai:
   * Regionas
   * Planas
   * OS šablonas
4. Užklausos atsakymo HTTP statuso kodas.
5. Užklausos rezultatas:
   * Sėkmingai įvykdžius testą, grąžinami serverio duomenys (ID, pavadinimas, regionas, planas, įdiegtas OS šablonas.
   * Neįdiegus serverio per tam tikrą laiko limitą tiesiog fiksuojamas timeout.
   * Nepavykus pradėti diegimo išsaugomas atsakymo pranešimas, nurodantis priežastį, kodėl nepavyko pradėti diegimo.

Rezultatų duomenims saugoti naudojama MySQL duomenų bazės valdymo sistema.

**Testavimo planas**

Testavimo scenarijai – užsakomas serveris:

* su standartiniais parametrais
* su atsitiktiniais parametrais
* su klaidingais parametrais
* nurodant regioną
* nurodant serverį
* nurodant OS šabloną
* nurodant regioną ir OS šabloną
* nurodant regioną ir serverio planą
* nurodant serverį ir OS šabloną
* nurodant regioną, planą ir OS šabloną
* nurodant regioną ir kategoriją
* nurodant OS šabloną ir kategoriją
* nurodant tipo kategoriją

Testavimo scenarijus – viršytas laikas:

* Nepavyko įdiegti tam tikro serverio per norimą laiko limitą.

Testavimo laiko limitai:

* Virtualūs serveriai – 3min
* Dedikuoti serveriai – 15 min
* Standartinis limitas – 10 min
* CLI įvedamas limitas, mažiausiai 1 min.

Viršijus laiko limitą testas laikomas nesėkmingu.

Testavimo atvejai:

|  |  |
| --- | --- |
| Įvestis | Numatomas rezultatas |
| Užklausa su klaidingais parametrais | HTTP status code 4xx su pranešimu, kodėl nepavyko pradėti diegimo, pvz.: blogi parametrai, nepakankamas kreditas paskyroje, planas išparduotas arba filtrų parametrai tarpusavyje nesuderinami. |
| Užklausa su standartiniais parametrais – pirmas regionas, pirmas planas, pirmas OS šablonas | HTTP status code 201, užsakytas ir įdiegtas serveris iš pirmo galimo regiono su pirmu planu ir pirmu OS atvaizdu, pagal tai, ką grąžino API. |
| Užklausa su atsitiktiniais galimais parametrais. | HTTP status code 201, užsakytas ir įdiegtas serveris su galima atsitiktine parametrų kombinacija. |
| Užklausa su iš anksto nurodytu regionu. | HTTP status code 201, užsakytas ir įdiegtas serveris iš nurodyto regiono atsitiktiniu planu ir OS atvaizdu. |
| Užklausa su iš anksto nurodytu serverio planu. | HTTP status code 201, užsakytas ir įdiegtas serveris atsitiktinai parenkant nurodyto plano regioną ir OS šablonu. |
| Užklausa su iš anksto nurodytu OS šablonu. | HTTP status code 201, užsakytas ir įdiegtas serveris su nurodytu OS šablonu, atsitiktinai parenkant regioną ir planą. |
| Užklausa su iš anksto nurodytu regionu ir OS šablonu | HTTP status code 201, užsakytas ir įdiegtas serveris atsitiktinai parenkant nurodyto regiono planą, kuris turi nurodytą OS šabloną. |
| Užklausa su iš anksto nurodytu regionu ir planu | HTTP status code 201, užsakytas ir įdiegtas tam tikro plano serveris nurodytame regione atsitiktinai parenkant tik OS šabloną. |
| Užklausa su iš anksto nurodytu planu ir OS šablonu | HTTP status code 201, užsakytas ir įdiegtas tam tikro plano serveris su OS atvaizdu, o regionas atsitiktinai parinktas iš tų kurie turi nurodytą planą. |
| Užklausa, kai užsakomas konkretus serveris nurodant regioną, planą ir OS šabloną | HTTP status code 201, užsakytas ir įdiegtas konkretus norimas serveris pagal tris duotus parametrus. |
| Užklausa su iš anksto nurodytu regionu ir kategorija | HTTP status code 201, užsakytas ir įdiegtas norimos kategorijos serveris iš nurodyto regiono, atsitiktinai parenkant planą ir OS šabloną. |
| Užklausa, kai nurodoma kategorija | HTTP status code 201, užsakytas ir įdiegtas norimos kategorijos serveris, atsitiktinai parenkant regioną, planą ir diegiamą OS. |
| Viršytas leistinas laiko limitas diegiant serverį. | Užfiksuotas limito viršijimas, sėkmingai nutrauktas serverio diegimas ir serveris pašalintas. |