VIEŠOSIOS DEBESIJOS PASLAUGŲ VALDYMO PLATFORMOS SUKŪRIMO PASLAUGŲ TECHNINĖ SPECIFIKACIJA

1. BENDRA INFORMACIJA

1.1. Valstybės skaitmeninių sprendimų agentūra (toliau – VSSA arba Perkančioji organizacija), valdanti Valstybės debesijos paslaugų teikimo infrastruktūrą bei įgyvendinanti Valstybės informacinių išteklių įstatymo ir kitų poįstatyminių teisės aktų nuostatas, skelbia šį pirkimą siekdama pasirinkti Tiekėją, kuris sukurtų Viešosios debesijos paslaugų valdymo platformą (toliau – **Debesijos valdymo platforma** arba **Sprendimas**, kuris apima vieną, kelis ar visus jį sudarančius komponentus ir/ar visų komponentų visumą). **Paslaugos bus finansuojamos projekto „Valstybės informacinių technologijų valdymo pertvarka“ lėšomis, projekto kodas Nr. 02-097-P-0001.**

1.2. Atsižvelgiant į Lietuvos Respublikos Vyriausybės 2024 m. gegužės 15 d. nutarimą Nr. 349 „Dėl Lietuvos Respublikos valstybės informacinių išteklių valdymo įstatymo įgyvendinimo“ (aktuali redakcija) bei juo patvirtintą Valstybės informacinių išteklių ir jų kopijų laikymo duomenų centruose ir šių išteklių veiklos atkūrimo iš kopijų tvarkos aprašą (Lietuvos Respublikos Vyriausybės 2024 m. spalio 30 d. nutarimo Nr. 907 redakcija) ir įvertinus kitus susijusius teisės aktų reikalavimus bei anksčiau minėtą informaciją, VSSA, vykdydama šį viešosios debesijos paslaugų valdymo platformos sukūrimo paslaugų pirkimą ir įgyvendindama pirkimo tikslus, siekia padėti valstybės informacinių sistemų ir registrų valdytojams įgyvendinti ilgalaikius uždavinius, kurie nustato strategines gaires hibridinio debesijos modelio diegimui ir vystymui.

1.3. Norėdama pasiekti pirkimo tikslus, siekiant efektyviai valdyti viešosios debesijos paslaugų gamintojų platformas (VSSA naudoja viešosios debesijos paslaugų gamintojų Google Cloud, Amazon Web Services (sutr. *AWS*) ir Azure platformas ir paslaugas) bei naudotis viešosios debesijos paslaugų gamintojų infrastruktūros technologijų privalumais, VSSA siekia sukurti ir įdiegti viešosios debesijos valdymo platformą. Įsigyjamų Debesijos platformos sukūrimo paslaugų paskirtis – automatizuoti viešosios debesijos paslaugų teikimo ir priežiūros procesus VSSA klientams, efektyviai stebėti, valdyti, optimizuoti ir automatizuoti debesų kompiuterijos išteklius, supaprastinti VSSA administratorių darbą.

1.4. VSSA, įgyvendindama šį pirkimą, užtikrina Valstybės informacinių technologijų paslaugų valdymo informacinės sistemos (toliau – VIPVIS), kuri yra realizuota Atlassian Jira sistemos (toliau – Jira) pagrindu plėtrą. Projektas **leis užtikrinti sukurtų viešosios debesijos paslaugų apskaitą, kontrolę, administravimą ir išteklių valdymą.** Neturint vieningo viešosios debesijos valdymo sprendimo, nebūtų užtikrinamas šių paslaugų tinkamas teikimas.

1.5. Debesijos valdymo platformos paskirtis:

1.5.1. Debesijos valdymo platformos sprendimas turi būti integruotas su VSSA naudojama paslaugų valdymo platforma Jira taip, kad būtų susieta su Jira sistemoje veikiančiais IT paslaugų valdymo procesais, siekiant didinti veiklos efektyvumą, mažinti klaidų tikimybę ir užtikrinti išsamius audito įrašus;

1.5.2. Pagal VIPVIS užklausoje pateiktus reikalavimus ir JSON formato duomenis, siūlomas sprendimas automatiškai turi sugeneruoti „Infrastruktūros kaip kodo“ (angl. *Infrastructure as Code, sutr. IaC*) katalogų struktūrą bei pirminius kodo failus (šablonus). Šie failai turi būti sukurti laikantis geriausių praktikų, parengti tolesniam redagavimui ir vykdymui bei suteikti galimybę automatiškai inicijuoti jų paleidimą IaC platformoje.

1.6. Debesijos valdymo platformos pagrindinės funkcijos:

1.6.1. Virtualių duomenų centrų (angl. *Landing Zones* – saugių, iš anksto sukonfigūruotų aplinkų) parengimas ir valdymas VSSA klientų organizacijoms.

1.6.2. Atskirų paslaugų (pvz., skaičiavimo, saugojimo, duomenų bazių, tinklo išteklių) diegimas, atnaujinimas, mastelio keitimas, šalinimas bei apskaitos tvarkymas VSSA naudojamose viešosios debesijos paslaugų gamintojų Google Cloud, AWS bei Azure platformose.

1.7. **Paslaugos neturi kelti grėsmės nacionaliniam saugumui, vadovaujantis LR Viešųjų pirkimų įstatymo 37 straipsnio 9 dalimi.**

1.8. **Jei pirkimo dokumentuose naudojami konkretūs modeliai ar šaltiniai, konkretūs procesai ar prekės ženklai, patentai, tipai, konkreti kilmė ar gamyba ir pan., jie gali būti pakeisti lygiaverčiais.**[1]

1.9. Apibūdinant pirkimo objektą, techninėje specifikacijoje ar kitose pirkimo dokumentuose galimai nurodytas konkretus modelis ar tiekimo šaltinis, konkretus procesas, būdingas konkretaus tiekėjo tiekiamoms prekėms ar teikiamoms paslaugoms ir/ar darbams atlikti, ar prekių ženklas, patentas, tipai, konkreti kilmė ar gamyba, sertifikatai, standartai, protokolai turi būti suprantami su žodžiais „arba lygiavertis“.

1.10. Šioje Techninėje specifikacijoje naudojami terminai „turi būti“, „turi turėti“, „turi leisti“, „turi turėti galimybę“ yra lygiaverčiai ir reiškia, kad Tiekėjas šio pirkimo apimtyje privalo užtikrinti atitinkamą funkcionalumą ar suteikti atitinkamas paslaugas. Funkcionalumas, kuris yra nurodytas būsimuoju laiku (pvz., „bus“ ir (ar) „būtų“, „leis“ ir (ar) „leistų“, „apims“ ir (ar) „apimtų“ ir pan.) nurodo siekiamą įgyvendinti būseną ir reiškia, kad Tiekėjas šio pirkimo apimtyje privalo užtikrinti atitinkamą funkcionalumą.

---

[1] Lygiaverčiu laikomas pirkimo objektas, kurio savybės nėra prastesnės (t. y. tokios pat arba geresnės) negu pirkimo dokumentuose perkamam objektui keliami reikalavimai ir siūlomą lygiavertį pirkimo objektą galima panaudoti pagal paskirtį be jokių apribojimų (įskaitant bet neapsiribojant išvardintais):

- neatliekant papildomų sąveikaujančių elementų pakeitimų;
- panaudojimas neturės įtakos sąveikaujančių elementų greitesniam susidėvėjimui, gedimams ir (ar) garantijos praradimui;
- numatytas tarnavimo laikotarpis nėra trumpesnis;
- nėra prastesnio techninio pažangumo lygio ir pan.

2. PIRKIMO OBJEKTAS IR APIMTIS

2.1. Pirkimo objektas – Debesijos valdymo platformos, užtikrinančios vieningą viešosios debesijos paslaugų valdymą ir išteklių naudojimą, kuri per integracijas papildo VIPVIS plėtrą, sukūrimas, įskaitant, bet neapsiribojant, projektavimo, programavimo, integravimo, įdiegimo, testavimo, garantinio aptarnavimo bei kitas susijusias paslaugas, numatytas šios Techninės specifikacijos 2.3 punkte.

2.2. **Debesijos valdymo platformos sukūrimui ir įdiegimui reikalingų šių pagrindinių komponentų įsigijimas:**

2.2.1. **Pirminio kodo valdymo sistema**: apima nuolatinio integravimo, pateikimo ir diegimo (CI/CD) įrankius (angl. *Source code manager with CI/CD pipelines*), saugo IaC failus, vykdo komandų grandines;

2.2.2. **Infrastruktūros kaip kodo (sutr. IaC) šablonų kūrimo sprendimas (arba lygiavertis)**: atlieka užklausų, ateinančių iš Jira orkestravimą, IaC šablonų kūrimą, IaC kodo vykdymo iniciavimą bei gyvavimo ciklo valdymą;

2.2.3. **Infrastruktūros kaip kodo (sutr. IaC) vykdymo variklis** (angl. *IaC Execution Engine*): vykdo infrastruktūros kaip kodo failus su vykdymo aplinkų valdymu;

2.2.4. **Išlaidų valdymo sprendimas** (Debesijos FinOps platforma arba „lygiavertė“, angl. *Cloud FinOps Platform*): valdo išlaidų sekimą, biudžeto paskirstymą ir atsiskaitymo integraciją;

2.2.5. **Paslapčių saugojimo sprendimas** (angl. *Secret Store*): saugo slaptažodžius, SSH raktų poras ir kitokią slaptą informaciją.

2.3. **Tiekėjo teikiamos paslaugos** (toliau – Paslaugos):

2.3.1. Sprendimo projektavimas, programavimas, testavimas, diegimas ir dokumentavimas;

2.3.2. Keitimai, apimant konfigūravimo darbus;

2.3.3. Konsultavimas;

2.3.4. Įdiegtų Sprendimo komponentų vystymas ir integravimas su kitais Perkančiosios organizacijos produktais ir sistemomis;

2.3.5. Teikiamų paslaugų automatizavimas ir susijusios konsultacijos;

2.3.6. Mokymų paslaugos pagal aktualumą;

2.3.7. Garantinis aptarnavimas;

2.3.8. Kitos tiesiogiai su pirkimo objektu susijusios paslaugos (pvz., mokymai, instruktavimai, ataskaitos ir pan.).

2.4. **Perkamų paslaugų apimtis:**

2.4.1. **Debesijos valdymo platformos sukūrimui ir įdiegimui reikalingų pagrindinių komponentų**, numatytų šios Techninės specifikacijos 2.2 punkte, **įsigijimas**. Detalūs komponentų funkciniai programiniai reikalavimai pateikiami Techninės specifikacijos 4, 5, 6, 7 ir 8 skyriuose žemiau, kaip kiekvieno Debesijos valdymo sprendimo komponento atskirai specialieji programinės įrangos funkcionalumo reikalavimai.

2.4.2. Perkančioji organizacija įsipareigoja užsakyti visus 2.4.1 papunktyje numatytus Debesijos valdymo sprendimo komponentus, tik tuo atveju, jei sutartis su Tiekėju bus sudaryta ne vėliau kaip 2026 m. kovo 31 d.

2.4.3. Jeigu pirkimo procedūrų metu paaiškėtų, kad dėl užsitęsusių pirkimo vykdymo procedūrų (pvz., Tiekėjų pasiūlymų tikslinimų ir pan.) ar kitų objektyvių priežasčių, įskaitant Debesijos valdymo platformos sukūrimo ir įdiegimo bei Tiekėjo teikiamoms paslaugoms įsigyti skirtų finansavimo laikotarpio pabaigą, finansavimo šaltinio nebuvimą pagal Projektą ar kitus su finansavimu susijusius apribojimus, **nėra galimybės sudaryti sutarties su Tiekėju iki 2026 m. kovo 31 d., Perkančioji organizacija turi teisę pirkimo procedūras nutraukti.** Paslaugos perkamos pagal projektą „Valstybės informacinių technologijų valdymo pertvarka“, projekto kodas Nr. 02-097-P-0001. Projekto įgyvendinimo pabaiga – 2026 m. balandžio 30 d., todėl tinkamų finansuoti ES lėšomis išlaidų laikotarpis taip pat baigiasi 2026 m. balandžio 30 d.

2.4.4. **Tiekėjo teikiamų paslaugų** maksimali apimtis – **7000 (septyni tūkstančiai) valandų**. Šis maksimalus valandų skaičius apima, be kita ko, ir valandas, reikalingas Tiekėjo specialistų pritraukimui Tiekėjo teikiamoms Paslaugoms teikti.

2.4.5. Tiekėjo teikiamų paslaugų trukmė – 5 (penki) mėn. nuo Sutarties įsigaliojimo datos.

2.4.6. Perkančioji organizacija neįsipareigoja užsakyti viso 2.4.4 papunktyje numatyto paslaugų valandų kiekio, tačiau įsipareigoja užsakyti ne mažiau kaip 10 (dešimt) proc. nuo maksimalaus planuojamo kiekio.

2.4.7. Tiekėjas užsakomas paslaugas turi suteikti užsakymuose numatytais terminais ir numatyta paslaugų teikimo, perdavimo ir priėmimo tvarka.

3. BENDRIEJI REIKALAVIMAI VIEŠOSIOS DEBESIJOS VALDYMO PLATFORMAI IR PASLAUGŲ TEIKIMUI

3.1. Debesijos valdymo platformos sprendimas turi būti diegiamas VSSA pateiktuose resursuose **ir/arba privačiai izoliuotoje SaaS paslaugoje, kuri atitinka LR Kibernetinio saugumo reikalavimus ir kurioje duomenys laikomi ES regione.**

3.2. VSSA, esant poreikiui, suteiks Tiekėjui galimybę naudotis šiomis jau egzistuojančiomis paslaugomis arba komponentais:

3.2.1. SMTP sietuvas: elektroninių laiškų siuntimui;

3.2.2. SMS sietuvas: SMS žinučių siuntimui;

3.2.3. Tapatybės tiekėjas: korporacinis tapatybės teikėjas su SAML arba OIDC palaikymu;

3.2.4. Jira: pagrindinė užsakymų valdymo platforma;

3.2.5. RedHat operacinė sistema, jei platformos komponentai bus diegiami VSSA resursuose, virtualiose tarnybinėse stotyse, kurias naudoja minėta sistema.

3.3. Per 5 (penkias) darbo dienas nuo Sutarties įsigaliojimo dienos privalo būti su Tiekėju pasirašyti konfidencialumo pasižadėjimai ir asmens duomenų tvarkymo susitarimas.

3.4. Tiekėjas ne vėliau kaip per 10 (dešimt) darbo dienų nuo Sutarties įsigaliojimo dienos turi parengti paslaugų teikimo reglamentą (toliau – Reglamentas) ir suderinti jį su VSSA.

3.5. Reglamentas nustato paslaugų teikimo, garantinio aptarnavimo ir valdymo tvarką bei principus. Reglamentas turi apimti, bet neapsiriboti, šias sritis:

3.5.1. Paslaugų teikimo ir vykdymo tvarka – paslaugų teikimo procesai, atsakomybės ir komunikacijos principai;

3.5.2. Paslaugų teikimo terminai ir lygiai (SLA) – reagavimo, atstatymo ir prieinamumo rodikliai;

3.5.3. Incidentų ir problemų valdymo tvarka – incidentų registravimo, klasifikavimo ir sprendimo procesai;

3.5.4. Pokyčių valdymo tvarka – pokyčių inicijavimas, vertinimas ir įgyvendinimas;

3.5.5. Saugumo ir prieigos valdymo reikalavimai – autentifikavimo, autorizacijos ir prieigos prie duomenų principai;

3.5.6. Klaidų ir defektų šalinimo tvarka – prioritetai, terminai, atsakomybės;

3.5.7. Techninės dokumentacijos ir ataskaitų teikimo tvarka – veiklos ataskaitų teikimas ir dokumentavimas;

3.5.8. Bandomosios eksploatacijos tvarka – sukurtų Debesijos valdymo platformos funkcionalumų testavimui skirta aplinka, skirta įdiegti Debesijos valdymo platformos komponentus ir jų funkcionalumo sprendimus į realią aplinką, eliminuojant klaidas;

3.5.9. Garantinio aptarnavimo tvarka – kaip Tiekėjas užtikrins tinkamą Debesijos valdymo platformos veikimą garantiniu laikotarpiu;

3.5.10. Komunikacijos ir eskalavimo procedūros – atsakingi asmenys, pranešimų kanalai ir terminai;

3.5.11. Paslaugų perdavimo ar nutraukimo tvarka – veiksmai ir atsakomybės, susijusios su paslaugų teikimo užbaigimu, perdavimu Perkančiajai organizacijai bei duomenų ir infrastruktūros tvarkymu po sutarties pabaigos.

4. SPECIALIEJI REIKALAVIMAI PIRMINIO KODO VALDYMO SISTEMAI

4.1. **Pirminio kodo valdymo sistema** – tai programinė sistema (programinė įranga), skirta programinio kodo versijų valdymui, saugojimui, kūrėjų bendradarbiavimui ir pakeitimų istorijos sekimui, užtikrinant kodo vientisumą ir saugumą.

4.2. Pirminio kodo valdymo programinės sistemos uždaviniai ir specialieji programinės įrangos funkcionalumo reikalavimai:

4.2.1. saugoti visą infrastruktūros kaip kodo konfigūraciją, užtikrinant versijavimą, atsekamumą ir centralizuotą valdymą – **vieningas tiesos šaltinis** (angl. *Single Source of Truth*);

4.2.2. vykdyti CI/CD procesus – **automatizavimo variklis** (angl. *Automation Engine*);

4.2.3. suteikti komandoms galimybę siūlyti infrastruktūros pakeitimus per suliejimo užklausas (angl. *Merge Requests*) – **bendradarbiavimo ir peržiūros platforma** (angl. *Collaboration and Review Hub*);

4.2.4. įgyvendinti saugumo ir valdymo taisykles per šakų apsaugą, privalomus patvirtinimus ir audito žurnalus, užtikrinant, kad pakeitimai atitiktų organizacijos standartus – **valdymo ir atitikties užtikrinimo įrankis** (angl. *Governance and Compliance Tool*);

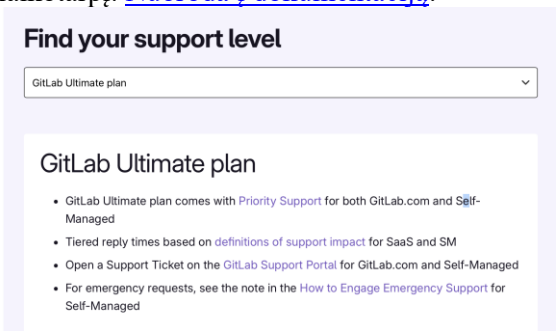
4.2.5. automatiškai skenuoti IaC kodą ieškant nesaugių konfigūracijų (angl. *IaC Scanning*) ir netyčia paliktų paslapčių (angl. *Secret Detection*) dar prieš pritaikant pakeitimus – **integruotas saugumo skydas** (angl. Integrated Security Shield);

4.2.6. užtikrinti saugią CI/CD procesų integraciją su paslapčių saugyklomis per OIDC, leidžiant procesams dinamiškai gauti trumpalaikius kredencialus be statinių raktų saugojimo – **saugus tiltas į paslapčių valdymą** (angl. *Secure Bridge to Secrets Management*);

4.2.7. valdyti CI/CD vykdytojų (angl. *Runners*) aplinkas su reikalingais resursais ir konfigūracijomis – **vykdymo aplinkos paruošimas** (angl. *Execution Environment Provisioning*).

4.3. Žemiau pateikiami detalūs specialieji reikalavimai pirminio kodo valdymo sistemos programinei įrangai ir jos funkcionalumui (žr. 1 lentelę).

1 lentelė. Reikalavimai pirminio kodo valdymo sistemos programinei įrangai

Eil. Nr.	**Charakteristikos pavadinimas**	**Reikalaujama charakteristika**	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
1.	**Bendrieji reikalavimai**		
1.1	Siūlomos programinės įrangos pavadinimas	***GitLab, Inc.,*** ***GitLab Ultimate self-managed***	
1.2	Licencijų kiekis	Tiekėjo pasiūlytos licencijos turi sudaryti galimybę pirminio kodo valdymo programine įranga naudotis ne mažiau kaip 50 (penkiasdešimt) klientų.	Tiekėjo pasiūlytos licencijos sudaro galimybę pirminio kodo valdymo programine įranga naudotis 50 (penkiasdešimt) klientų.
2.	**Reikalavimai versijų valdymui ir bendradarbiavimui**		
2.1	Diegimo modelis (savipalaikoma)	Sprendimas turi būti savipalaikomas ir diegiamas VSSA pateiktuose resursuose arba privačiai izoliuotoje SaaS paslaugoje, atitinkančioje LR Kibernetinio saugumo reikalavimus ir kurioje duomenys laikomi ES regione.	Sprendimas yra savipalaikomas ir diegiamas VSSA pateiktuose resursuose arba privačiai izoliuotoje SaaS paslaugoje, atitinkančioje LR Kibernetinio saugumo reikalavimus ir kurioje duomenys laikomi ES regione. Nuoroda į dokumentaciją.
2.2	Gamintojo palaikymas	Tiekėjas turi turėti techninį palaikymą iš gamintojo, visą prenumerata pagrįstos licencijos galiojimo laikotarpį.	Tiekėjas turi techninį palaikymą iš gamintojo, visą prenumerata pagrįstos licencijos galiojimo laikotarpį. Nuoroda į dokumentaciją. 
2.3	Git repozitorijų talpinimas	Programinė įranga turi teikti centralizuotą Git repozitorijų valdymo sprendimą su prieigos kontrole per SSH ir HTTPS protokolus.	Programinė įranga teikia centralizuotą Git repozitorijų valdymo sprendimą su prieigos kontrole per SSH ir HTTPS protokolus. Pirma nuoroda į dokumentaciją, antra nuoroda į dokumentaciją, trečia nuoroda į dokumentaciją.
2.4	Šakų apsauga	Turi būti galimybė konfigūruoti visas šakų (angl. *Branch*) apsaugos taisykles, tokias kaip pvz., apriboti operacijų „Push“ ir	Yra galimybė konfigūruoti visas šakų (angl. *Branch*) apsaugos taisykles, tokias kaip pvz., apriboti operacijų „Push“ ir „Merge“ teises, reikalauti statuso patikrų sėkmės ir pan. Nuoroda į dokumentaciją.

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
		„Merge“ teises, reikalauti statuso patikrų sėkmės ir pan.	
2.5	Suliejimo užklausos (Merge Requests)	Turi palaikyti suliejimo užklausų darbo eigą, leidžiančią atlikti kodo peržiūras, komentuoti, reikalauti patvirtinimų iš konkrečių vartotojų ar grupių.	Palaiko suliejimo užklausų darbo eigą, leidžiančią atlikti kodo peržiūras, komentuoti, reikalauti patvirtinimų iš konkrečių vartotojų ar grupių. Nuoroda į dokumentaciją.
2.6	Kodo savininkai (Code Owners)	Turi būti galimybė apibrėžti failų ar katalogų savininkus, kurie automatiškai priskiriami peržiūrai, kai atliekami pakeitimai jiems priklausančiose kodo dalyse.	Yra galimybė apibrėžti failų ar katalogų savininkus, kurie automatiškai priskiriami peržiūrai, kai atliekami pakeitimai jiems priklausančiose kodo dalyse. Nuoroda į dokumentaciją.
2.7	WEB teksto redaktorius	Sprendimas turi turėti integruotą naršyklėje veikiančią kūrimo aplinką (angl *Integrated Development Environment, IDE*), leidžiančią greitai redaguoti kodą, atlikti operaciją „commit“ ir kurti suliejimo užklausas tiesiogiai iš vartotojo sąsajos.	Sprendimas turi integruotą naršyklėje veikiančią kūrimo aplinką (angl *Integrated Development Environment, IDE*), leidžiančią greitai redaguoti kodą, atlikti operaciją „commit“ ir kurti suliejimo užklausas tiesiogiai iš vartotojo sąsajos. . Nuoroda į dokumentaciją.
2.8	Infrastruktūros pakeitimų peržiūra suliejimo užklausose	Platforma turi turėti integruotą funkciją, kuri IaC planavimo (pvz.: terraform plan) komandos išvestį atvaizduoja specializuotame valdiklyje (angl. widget) tiesiogiai suliejimo užklausos sąsajoje. Šis valdiklis turi pateikti aiškią suvestinę, nurodančią, kiek išteklių bus pridėta, pakeista ir sunaikinta.	Platforma turi integruotą funkciją, kuri IaC planavimo (pvz.: terraform plan) komandos išvestį atvaizduoja specializuotame valdiklyje (angl. widget) tiesiogiai suliejimo užklausos sąsajoje. Šis valdiklis pateikia aiškią suvestinę, nurodančią, kiek išteklių bus pridėta, pakeista ir sunaikinta. Pirma nuoroda į dokumentaciją, antra nuoroda į dokumentaciją, trečia nuoroda į dokumentaciją.
3.	**Reikalavimai CI/CD automatizavimui**		
3.1	Deklaratyvūs procesai (Pipelines)	CI/CD procesai turi būti aprašomi deklaratyviu YAML formatu (.yml) tiesiogiai repozitorijoje, leidžiant procesus valdyti kaip kodo dalį (angl. Pipelines as Code).	CI/CD procesai yra aprašomi deklaratyviu YAML formatu (.yml) tiesiogiai repozitorijoje, leidžiant procesus valdyti kaip kodo dalį (angl. Pipelines as Code). Nuoroda į dokumentaciją.
3.2	Savarankiškai talpinami vykdytojai	Platforma turi palaikyti galimybę registruoti ir valdyti nuosavus CI/CD vykdytojus (angl. runners) privačioje infrastruktūroje, užtikrinant saugų priėjimą prie vidinių sistemų.	Platforma palaiko galimybę registruoti ir valdyti nuosavus CI/CD vykdytojus (angl. runners) privačioje infrastruktūroje, užtikrinant saugų priėjimą prie vidinių sistemų. Nuoroda į dokumentaciją.
3.3	Vykdytojų grupės ir žymės (Tags)	Turi būti galimybė grupuoti visus vykdytojus ir priskirti jiems žymes, leidžiančias nukreipti konkrečias užduotis į specializuotas vykdymo aplinkas (pvz., su GPU, specifine OS ir pan.).	Yra galimybė grupuoti visus vykdytojus ir priskirti jiems žymes, leidžiančias nukreipti konkrečias užduotis į specializuotas vykdymo aplinkas (pvz., su GPU, specifine OS ir pan.). Nuoroda į dokumentaciją.
3.4	Artefaktai ir podėlis (Cache)	Turi būti palaikomas užduočių artefaktų (pvz., „terraform plan“	Yra palaikomas užduočių artefaktų (pvz., „terraform plan“ failų, ataskaitų ir pan.)

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
		failų, ataskaitų ir pan.) išsaugojimas ir perdavimas tarp procesų etapų, taip pat podėlio mechanizmas priklausomybėms paspartinti.	išsaugojimas ir perdavimas tarp procesų etapų, taip pat podėlio mechanizmas priklausomybėms paspartinti. Pirma nuoroda į dokumentaciją, antra nuoroda į dokumentaciją.
3.5	Kintamieji ir paslaptys	Turi būti suteikiama centralizuota vieta organizacijos, repozitorijos ir aplinkų lygmeniu valdyti CI/CD kintamuosius ir paslaptis, su galimybe juos apsaugoti ir apriboti prieigą.	Yra suteikiama centralizuota vieta organizacijos, repozitorijos ir aplinkų lygmeniu valdyti CI/CD kintamuosius ir paslaptis, su galimybe juos apsaugoti ir apriboti prieigą. Nuoroda į dokumentaciją.
3.6	Tėvų-vaikų procesai	Sprendimas turi palaikyti sudėtingų procesų skaidymą į atskiras, pernaudojamas darbo eigas (angl. Reusable workflows) arba dinamiškai paleidžiant kitas darbo eigas (angl. Workflow_dispatch), siekiant geresnio moduliarumo.	Sprendimas palaiko sudėtingų procesų skaidymą į atskiras, pernaudojamas darbo eigas (angl. Reusable workflows) arba dinamiškai paleidžiant kitas darbo eigas (angl. Workflow_dispatch), siekiant geresnio moduliarumo. Pirma nuoroda į dokumentaciją, antra nuoroda į dokumentaciją, trečia nuoroda į dokumentaciją.
3.7	Saugi integracija su paslapčių saugyklomis	CI/CD procesai turi palaikyti saugų, be slaptažodžių, autentifikavimą su išorinėmis paslapčių valdymo sistemomis, tokiomis kaip HashiCorp Vault arba lygiavertis, naudojant OpenID Connect OIDC standartus. Tai turi leisti procesams dinamiškai gauti trumpalaikius debesijos kredencialus vykdymo metu.	CI/CD procesai palaiko saugų, be slaptažodžių, autentifikavimą su išorinėmis paslapčių valdymo sistemomis, tokiomis kaip HashiCorp Vault arba lygiavertis, naudojant OpenID Connect OIDC standartus. Tai leidžia procesams dinamiškai gauti trumpalaikius debesijos kredencialus vykdymo metu. Nuoroda į dokumentaciją.
4.	**Reikalavimai integruotam saugumui**		
4.1	Statinė kodo analizė (SAST)	Platforma turi turėti integruotą SAST skenavimą, kuris automatiškai analizuoja išeitinį kodą ir ieško potencialių saugumo spragų CI/CD proceso metu.	Platforma turi integruotą SAST skenavimą, kuris automatiškai analizuoja išeitinį kodą ir ieško potencialių saugumo spragų CI/CD proceso metu. Nuoroda į dokumentaciją.
4.2	Priklausomybių skenavimas	Turi būti atliekamas automatinis projekto priklausomybių skenavimas, siekiant nustatyti žinomas saugumo spragas (angl. *Common Vulnerabilities and Exposures, CVE*).	Yra atliekamas automatinis projekto priklausomybių skenavimas, siekiant nustatyti žinomas saugumo spragas (angl. *Common Vulnerabilities and Exposures, CVE*). Nuoroda į dokumentaciją.
4.3	Paslapčių aptikimas	Turi būti integruotas mechanizmas, kuris kodo įkėlimo metu skenuoja repozitoriją ir aptinka netyčia paliktus kredencialus, API raktus ir kitas paslaptis.	Yra integruotas mechanizmas, kuris kodo įkėlimo metu skenuoja repozitoriją ir aptinka netyčia paliktus kredencialus, API raktus ir kitas paslaptis. Nuoroda į dokumentaciją.
4.4	Saugumo skydelis	Turi būti suteikiama centralizuota vartotojo sąsaja (angl. *Security Dashboard*) projektų ir grupių	Yra suteikiama centralizuota vartotojo sąsaja (angl. *Security Dashboard*) projektų ir grupių lygmeniu, kurioje apibendrinami visi saugumo

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
		lygmeniu, kurioje apibendrinami visi saugumo skenavimų rezultatai ir valdomos spragos.	skenavimų rezultatai ir valdomos spragos. Nuoroda į dokumentaciją.
4.5	Infrastruktūros kaip kodo (IaC) saugumo skenavimas	Platforma turi leisti integruoti trečiųjų šalių Infrastruktūros kaip kodo saugumo skenavimo įrankius (pvz., tfsec, Checkov ir pan.) į CI/CD procesą. Skenavimo rezultatai turi būti atvaizduojami suliejimo užklausoje, siekiant nustatyti nesaugias konfigūracijas prieš jas pritaikant.	Platforma leidžia integruoti trečiųjų šalių Infrastruktūros kaip kodo saugumo skenavimo įrankius (pvz., tfsec, Checkov ir pan.) į CI/CD procesą. Skenavimo rezultatai yra atvaizduojami suliejimo užklausoje, siekiant nustatyti nesaugias konfigūracijas prieš jas pritaikant. Pirma nuoroda į dokumentaciją, antra nuoroda į dokumentaciją.
5.	**Reikalavimai paketų valdymui ir registrams**		
5.1	Konteinerių registras	Platforma turi turėti integruotą, privatų OCI-suderinamą konteinerių registrą, leidžiantį saugoti, versijuoti ir platinti Docker atvaizdus.	Platforma turi integruotą, privatų OCI-suderinamą konteinerių registrą, leidžiantį saugoti, versijuoti ir platinti Docker atvaizdus. Nuoroda į dokumentaciją.
5.2	Paketų registras	Turi palaikyti įvairių tipų paketų registrus (pvz., Maven, npm, NuGet, PyPI ir pan.), leidžiančius centralizuotai valdyti projekto priklausomybes.	Palaiko įvairių tipų paketų registrus (pvz., Maven, npm, NuGet, PyPI ir pan.), leidžiančius centralizuotai valdyti projekto priklausomybes. Nuoroda į dokumentaciją.
5.3	Privatus Terraform modulių registras	Platforma turi veikti kaip privatus Terraform modulių registras, leidžiantis komandoms saugiai dalintis, versijuoti ir pernaudoti vidinius IaC modulius. Registras turi palaikyti modulio versijų valdymą ir rodyti dokumentaciją.	Platforma veikia kaip privatus Terraform modulių registras, leidžiantis komandoms saugiai dalintis, versijuoti ir pernaudoti vidinius IaC modulius. Registras palaiko modulio versijų valdymą ir rodyti dokumentaciją. Nuoroda į dokumentaciją.
5.4	Centralizuotas IaC būsenos valdymas	Platforma turi veikti kaip saugi, nuotolinė IaC būsenos failų (pvz.: Terraform terraform.tfstate) saugykla, naudojant standartinį HTTP protokolą. Turi būti palaikomas automatinis būsenos failo užrakinimas (angl. state locking) vykdant komandas, siekiant išvengti lygiagrečių pakeitimų ir būsenos sugadinimo. Prieiga prie būsenos failų turi būti valdoma per platformos prieigos kontrolės sistemą, o patys failai saugykloje turi būti šifruojami.	Platforma veikia kaip saugi, nuotolinė IaC būsenos failų (pvz.: Terraform terraform.tfstate) saugykla, naudojant standartinį HTTP protokolą. Yra palaikomas automatinis būsenos failo užrakinimas (angl. state locking) vykdant komandas, siekiant išvengti lygiagrečių pakeitimų ir būsenos sugadinimo. Prieiga prie būsenos failų yra valdoma per platformos prieigos kontrolės sistemą, o patys failai saugykloje yra šifruojami. Nuoroda į dokumentaciją.
6.	**Reikalavimai valdymui ir integracijoms**		
6.1	Vartotojų ir grupių valdymas	Sprendimas turi palaikyti smulkų, vaidmenimis grįstą prieigos valdymą vartotojams ir grupėms, taikomą projektų ir grupių lygmeniu.	Sprendimas palaiko smulkų, vaidmenimis grįstą prieigos valdymą vartotojams ir grupėms, taikomą projektų ir grupių lygmeniu. Nuoroda į dokumentaciją.
6.2	SSO ir LDAP integracija	Turi būti palaikoma integracija su įmonės tapatybės tiekėjais per	Yra palaikoma integracija su įmonės tapatybės tiekėjais per SAML 2.0 ir LDAP, leidžianti

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
		SAML 2.0 ir LDAP, leidžianti centralizuotą vartotojų autentifikaciją ir valdymą.	centralizuotą vartotojų autentifikaciją ir valdymą. Nuoroda į dokumentaciją.
6.3	Audito žurnalai	Platforma turi generuoti išsamius audito žurnalus apie visus svarbius įvykius (pvz., prisijungimus, pakeitimus repozitorijose, prieigos teisių keitimus ir pan.) su galimybe juos eksportuoti.	Platforma generuoja išsamius audito žurnalus apie visus svarbius įvykius (pvz., prisijungimus, pakeitimus repozitorijose, prieigos teisių keitimus ir pan.) su galimybe juos eksportuoti. Nuoroda į dokumentaciją.
6.4	REST API	Turi būti suteikiama išsami Representacinio būklės perdavimo aplikacijų programavimo sąsaja (angl. Representational State Transfer Application Programming Interface, REST API), leidžianti programiškai valdyti visus platformos aspektus: projektus, vartotojus, CI/CD procesus ir integruotus registrus.	Yra suteikiama išsami Representacinio būklės perdavimo aplikacijų programavimo sąsaja (angl. Representational State Transfer Application Programming Interface, REST API), leidžianti programiškai valdyti visus platformos aspektus: projektus, vartotojus, CI/CD procesus ir integruotus registrus. Nuoroda į dokumentaciją.
6.5	Aukštas pasiekiamumas (HA)	Savarankiškai talpinama versija turi palaikyti aukšto pasiekiamumo architektūrą (pvz., kelių mazgų diegimą ir pan.), siekiant užtikrinti sistemos atsparumą ir nepertraukiamą veikimą.	Savarankiškai talpinama versija palaiko aukšto pasiekiamumo architektūrą (pvz., kelių mazgų diegimą ir pan.), siekiant užtikrinti sistemos atsparumą ir nepertraukiamą veikimą. Nuoroda į dokumentaciją.
7.	**Reikalavimai licencijavimui ir palaikymui**		
7.1	Licencijavimo modelis	Licencijavimas turi būti pagrįstas aktyvių vartotojų skaičiumi.	Licencijavimas yra pagrįstas aktyvių vartotojų skaičiumi. Pirma nuoroda į dokumentaciją, antra nuoroda į dokumentaciją.
7.2	Veikimas pasibaigus licencijai	Turi būti užtikrinta galimybė, pasibaigus mokamos prenumeratos licencijos galiojimui, toliau naudoti programinę įrangą su nemokamos versijos (angl. Free tier) funkcionalumu. Visi pagrindiniai duomenys, tokie kaip Git repozitorijos, vartotojų paskyros, užduotys (angl. issues) ir suliejimo užklausos, turi išlikti prieinami ir valdomi.	Yra užtikrinta galimybė, pasibaigus mokamos prenumeratos licencijos galiojimui, toliau naudoti programinę įrangą su nemokamos versijos (angl. Free tier) funkcionalumu. Visi pagrindiniai duomenys, tokie kaip Git repozitorijos, vartotojų paskyros, užduotys (angl. issues) ir suliejimo užklausos, išlieka prieinami ir valdomi. Pirma nuoroda į dokumentaciją, antra nuoroda į dokumentaciją.

5. SPECIALIEJI REIKALAVIMAI INFRASTRUKTŪROS KAIP KODO ŠABLONŲ KŪRIMO SPRENDIMUI

5.1. Infrastruktūros kaip kodo (IaC) šablonų kūrimo sprendimui keliami specialieji programinės įrangos funkcionalumo reikalavimai:

5.1.1. Infrastruktūros kaip kodo šablonų kūrimo sprendimas (toliau – **Šablonų kūrimo sprendimas**), remdamasis iš Jira sistemos gautais JSON formato duomenimis, turi generuoti infrastruktūros kaip kodo išeitinio kodo katalogų struktūras bei konfigūracijos failus su pilnai arba dalinai užpildytomis kintamųjų reikšmėmis. Tikslus JSON formatas, perduodamas iš Jira sistemos, turi būti suderintas Sutarties vykdymo metu.

5.1.2. Resursų pavadinimų kūrimo taisykles pateikia Perkančioji organizacija.

5.1.3. Šablonų kūrimo sprendimas gali būti realizuotas kaip atskira programinė įranga arba kaip CI/CD procesas, veikiantis pirminio kodo valdymo sistemoje.

5.1.4. Šablonų kūrimo sprendimas turi gebėti dirbti su IaC įrankiais (pvz., Terragrunt arba lygiaverčiu), kurie padeda valdyti dideles ir sudėtingas IaC konfigūracijas (pvz.: kelias to paties projekto aplinkas).

5.1.5. Sugeneruotas kodas ir katalogų struktūra turi būti įkeliama į Git repozitoriją naudojant „git commit“ veiksmą.

5.1.6. Po kiekvieno „git commit“ veiksmo automatiškai turi būti vykdoma IaC konfigūracijos failų saugumo ir atitikties gerosioms praktikoms patikra.

5.1.7. Sukūrus IaC katalogus ir IaC failų šablonus, VSSA administratorius turi turėti galimybę redaguoti šiuos failus tiek per atskirą tekstinį redaktorių, tiek per WEB sąsają, integruotą į pirminio kodo valdymo sistemą.

5.1.8. Visų IaC failų pakeitimų išsaugojimas turi būti atliekamas naudojant Git įrankius.

5.1.9. Kai VSSA administratorius paruošia IaC failus vykdymui, turi būti aktyvuojamas „git merge“ veiksmas į atskirą Git šaką. Po šio veiksmo IaC vykdymo variklis turi pradėti vykdyti IaC failus per API sąsają.

5.1.10. Sistema privalo validuoti gaunamus struktūrizuotus JSON duomenis.

5.1.11. Generuodamas IaC konfigūracijos failus, sprendimas turi atsižvelgti į išteklių specifikacijas, tinklo konfigūraciją, saugumo politiką ir nustatytus biudžeto apribojimus.

5.1.12. Sprendimas turi efektyviai naudoti IaC vykdymo variklio darbines aplinkas.

5.2. Žemiau pateikiamas preliminarus IaC šablonų sąrašas, kuriuos reikia sukurti Sutarties vykdymo metu (žr. 2 lentelę). Jei Sutarties vykdymo laikotarpiu atsiras papildomas poreikis IaC šablonams kurti, Tiekėjui bus pateikti atskiri užsakymai. Užsakymų pateikimo ir darbų apmokėjimo tvarka numatyta šios Techninės specifikacijos 9 ir 10 skyriuose.

2 lentelė. Preliminarus IaC šablonų sąrašas

Viešosios debesijos paslaugų kategorija	Google Cloud platforma arba lygiavertė	AWS platforma arba lygiavertė	Azure platforma arba lygiavertė
Skaičiavimai	• Compute Engine; • Google Kubernetes Engine (GKE); • Cloud Functions; • Cloud Run; • Compute Engine GPU/TPU.	• EC2; • Elastic Kubernetes Service (EKS); • Lambda; • Fargate; • EC2 GPU Instances.	• Virtual Machines (VM); • Azure Kubernetes Service (AKS); • Azure Functions • Azure Container Apps; • Virtual Machines (VM) GPU.
Saugykla	• Cloud Storage; • Filestore; • Persistent Disk.	• S3; • EFS; • EBS.	• Blob Storage; • Azure Files; • Managed Disks.
Duomenų bazės	• CloudSQL; • Cloud Spanner; • Cloud Bigtable; • Firestore; • Memorystore.	• RDS; • Aurora; • DynamoDB; • ElastiCache.	• Azure SQL Database; • Cosmos DB; Table Storage.

Tinklas	• Virtual Private Cloud; • Cloud Load Balancing; • Cloud CDN; • Cloud DNS; • Cloud Armor.	• VPC; • Elastic Load Balancing; • CloudFront; • Route 53; • AWS Shield.	• Virtual Network; • Azure Load Balancer / Application Gateway; • Azure CDN; • Azure DNS; • Azure Firewall; • DDoS Protection; • Azure WAF.
Didieji duomenys ir analitika	• BigQuery; • Dataflow; • Dataproc; • Pub/Sub; • Looker Studio.	• Redshift; • Kinesis; • EMR; • MSK; • QuickSight.	• Synapse Analytics; • Stream Analytics; • HDInsight; • Event Hub; • Power BI.
Valdymas ir saugumas	• IAM; • Cloud Logging; • Cloud Monitoring; • Security Command Center; • Secret Manager; • KMS; • Organization Policies; • Billing Budgets & Alerts; • Backup & DR Service; • Certificate Manager.	• IAM; • CloudWatch; • CloudTrail; • GuardDuty; • Secrets Manager; • KMS; • Service Control Policies (SCP); • Cost Explorer, Budgets; • AWS Backup; • SNS, SES.	• Azure AD; • Monitor; • Activity Logs; • Security Center; • RBAC; • Key Vault; • Azure Policy; • Cost Management + Advisor; • Azure Backup; • App Service Certificates.

6. SPECIALIEJI REIKALAVIMAI INFRASTRUKTŪROS KAIP KODO VYKDYMO VARIKLIUI

6.1. Infrastruktūros kaip kodo (IaC) vykdymo variklio sprendimui keliami specialieji programinės įrangos funkcionalumo reikalavimai:

6.1.1. Siūlomas infrastruktūros kaip kodo (IaC) vykdymo variklio sprendimas (toliau – **Vykdymo variklio sprendimas**) turi sukurti izoliuotas vykdymo aplinkas kiekvienai VSSA debesijos paskyrai ir aplinkos kombinacijai, užtikrindamas visišką IaC būklės ir konfigūracijos failų atskyrimą.

6.1.2. Vykdymo variklio sprendimas turi palaikyti visų vykdymo aplinkų būklės failų versijų kontrolę, įgalindamas laiko taško atkūrimą ir visų infrastruktūros pakeitimų audito pėdsaką.

6.1.3. Vykdymo variklio sprendimas turi generuoti detalius vykdymo planus prieš taikant bet kokius pakeitimus, įskaitant išteklių kūrimo, modifikavimo ir šalinimo operacijas.

6.1.4. Vykdymo variklio sprendimas turi patvirtinti vykdymo planus pagal VSSA nustatytas politikas ir saugumo reikalavimus prieš tęsiant išteklių parengimą.

6.1.5. Vykdymo variklio sprendimas turi palaikyti tiek inkrementinių atnaujinimų, tiek pilno išteklių pakeitimo strategijas, atsižvelgiant į konfigūracijos pakeitimų pobūdį.

6.1.6. Vykdymo variklio sprendimas turi įgyvendinti tinkamas nepavykusių diegimų valymo procedūras, užtikrindamas, kad debesijos paskyrose nelieka našlaičių išteklių.

6.1.7. Vykdymo variklio sprendimas turi užtikrinti, kad visos operacijos yra idempotentinės, leidžiančios saugų IaC šablonų pakartotinį vykdymą be nepageidaujamų šalutinių poveikių ar dubliuotų išteklių kūrimo.

6.1.8. Vykdymo variklio sprendimas turi gauti IaC šablonus ir priklausomybių modulius iš nurodytų Git saugyklų vykdymo metu, užtikrindamas, kad visada naudojamos naujausios versijos.

6.1.9. Vykdymo variklio sprendimas turi fiksuoti visus IaC šablonų vykdymus su papildoma informacija, tokia kaip Jira užduočių nuorodos, laiko žymos ir pakeitimų santraukos.

6.1.10. Vykdymo variklio sprendimas turi palaikyti atskiras Git šakas skirtingoms aplinkoms (testų, produkcinei aplinkai) ir palaikyti „git pull request“ darbo eigas IaC šablonų pakeitimams.

6.1.11. Vykdymo variklio sprendimas turi palaikyti git webhook integraciją automatiniams IaC šablonų atnaujinimams ir patvirtinimui, kai saugyklos turinys keičiasi.

6.1.12. Vykdymo variklio sprendimas turi parengti ir valdyti išteklius per AWS, Azure ir GCP, naudojant kiekvieno teikėjo savus APIs ir SDK.

6.1.13. Vykdymo variklio sprendimas privalo periodiškai vykdyti užklausas į debesijos paslaugų teikėjų API, siekdamas palyginti realią įdiegtų išteklių būseną su IaC būsenos faile (angl. *State file*) apibrėžta norima būsena. Tiksli būsenos failų saugojimo vieta ir mechanizmas turi būti suderinti Sutarties vykdymo metu.

6.1.14. Aptikęs konfigūracijos nukrypimą (skirtumą tarp realios ir IaC apibrėžtos būsenos), Vykdymo variklio sprendimas privalo informuoti VSSA administratorius apie nustatytus neatitikimus.

6.1.15. Aptikus nukrypimą, Vykdymo variklio sprendimas turi inicijuoti automatinį jo ištaisymą, pakartotinai pritaikydamas IaC konfigūraciją. Kritinių išteklių konfigūracijos korekcijoms privalomas rankinis VSSA administratoriaus patvirtinimas.

6.2. Žemiau pateikiami detalūs specialieji reikalavimai infrastruktūros kaip kodo (IaC) Vykdymo variklio sprendimo programinei įrangai ir jos funkcionalumui (žr. 3 lentelę).

3 lentelė. Reikalavimai Vykdymo variklio programinei įrangai

Eil. Nr.	**Charakteristikos pavadinimas**	**Reikalaujama charakteristika**	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
1.	**Bendrieji reikalavimai**		
1.1	Siūlomos programinės įrangos pavadinimas	***International Business Machines Corporation (IBM), IBM Terraform Self-Managed Premium***	
1.2	Licencijų kiekis	Tiekėjas turi pateikti tiek licencijų, kad siūloma programine įranga būtų galima naudotis ne mažiau kaip 300 vnt. skirtingų vykdymo aplinkų.	Siūloma programine įranga turi galimybę naudotis ne mažiau kaip 300 vnt. skirtingų vykdymo aplinkų.
1.3	Diegimo modelis (savipalaikoma)	Sprendimas turi būti savipalaikomas arba diegiamas VSSA pateiktuose resursuose.	Sprendimas yra savipalaikomas arba diegiamas VSSA pateiktuose resursuose. Nuoroda
1.4	Gamintojo palaikymas	Turi turėti techninį palaikymą iš gamintojo, visą prenumerata pagrįstos licencijos galiojimo laikotarpį.	Sprendimas užtikrina techninį palaikymą, teikdamas gamintojo remiamą techninę pagalbą visam prenumeratos laikotarpiui. Pagalbos paslaugos nėra papildomas priedas, jos natūraliai įtrauktos į licenciją, užtikrinant ekspertų konsultacijas, saugumo pataisas ir versijų atnaujinimus viso sutarties laikotarpio metu. Nuoroda
1.5	Dokumentacija	Sprendimas turi būti pateikiamas su išsamiu, nuolat atnaujinamu ir viešai prieinamu internetiniu dokumentacijos rinkiniu, apimančiu diegimą, administravimą, naudotojų vadovus, API referencijas ir gerąsias praktikas.	„IBM HashiCorp Terraform“ sprendimas atitinka reikalavimą turėti išsamią, nuolat atnaujinamą ir viešai prieinamą internetinę dokumentaciją. „IBM HashiCorp“ teikia centralizuotą dokumentacijos portalą per „HashiCorp Developer“ platformą, kuri yra pagrindinis ir patikimas visos su „Terraform“ susijusios informacijos šaltinis. Nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
1.6	Atitiktis Terraform / OpenTofu	Sprendimas turi būti visiškai suderinamas su HashiCorp Terraform ir atvirojo kodo projekto OpenTofu (≥ 1.8) arba lygiaverčiais formatais, sintakse, būsenos (state) struktūra bei modulių valdymo mechanizmais.	Siūlomas produktas yra standartą atitinkantis sprendimas, įgyvendinantis „HashiCorp Configuration Language“ (HCL) ir būsenos valdymo protokolus, kuriais remiasi atvirojo kodo projektas „OpenTofu“ (nuo 1.8 versijos). Pirma nuoroda Antra nuoroda Trečia nuoroda
2.	**Reikalavimai pagrindiniam IaC varikliui ir kalbai**		
2.1	Deklaratyvi kalba	Sprendimas turi naudoti aukšto lygio deklaratyvią konfigūracijos kalbą, kuri yra suprantama žmogui ir vykdoma mašinos, apibrėžiant infrastruktūrą.	Sprendimas naudoja „HashiCorp Configuration Language“ (HCL) – specialiai sukurtą aukšto lygio deklaratyvią konfigūravimo kalbą, skirtą būtent debesijos infrastruktūrai. Nuoroda
2.2	Dinaminiai blokai	Konfigūracijos kalba turi palaikyti dinaminius blokus, kurie iteratyviai generuoja įdėtus konfigūracijos blokus, tokius kaip ugniasienės taisyklės ar nustatymai, iš reikšmių kolekcijos.	Sprendimas turi dinaminių blokų palaikymą „HashiCorp Configuration Language“ (HCL) kalboje. Ši funkcija leidžia programiškai ir iteratyviai generuoti įdėtinius konfigūracijos blokus, remiantis pateikta reikšmių kolekcija (pavyzdžiui ugniasienės taisyklės ar nustatymai, iš reikšmių kolekcijos). Nuoroda
2.3	Iteracijos konstrukcijos	Turi būti palaikomos iteracijų konstrukcijos (angl. count ir for_each), leidžiančios sukurti kelias išteklių arba modulių instancijas iš vieno konfigūracijos bloko, mažinant kodo dubliavimą.	Sprendimas palaiko iteracijas naudodamas „count“ ir „for_each“ meta-argumentus. Šie mechanizmai leidžia administratoriams valdyti kelis panašius infrastruktūros objektus (resursus ar modulius) nedubliuojant kodo ir taip užtikrinti „DRY“ (angl. Don’t Repeat Yourself) architektūros principų laikymąsi. Pirma nuoroda Antra nuoroda
2.4	Įvesties kintamųjų patvirtinimas	Kalba turi leisti apibrėžti pasirinktines įvesties kintamųjų patvirtinimo taisykles, leidžiančias autoriams nurodyti prielaidas ir klaidų pranešimus, padedančius jų konfigūracijų vartotojams.	Sprendimas palaiko pasirinktines įvesties kintamųjų validavimo taisykles. Ši funkcija leidžia konfigūracijų autoriams apibrėžti konkrečius apribojimus, prielaidas ir individualizuotus klaidų pranešimus, taip užtikrinant, kad į infrastruktūros diegimo procesą būtų perduodami tik galiojantys duomenys. Nuoroda
2.5	Įmontuotos funkcijos	Kalba turi apimti turtingą įmontuotų funkcijų biblioteką, skirtą tekstų manipuliavimui, skaičiavimams, datos/laiko operacijoms, kolekcijų	Sprendimas turi išsamią, integruotų funkcijų biblioteką, prieinamą „HashiCorp Configuration Language“ (HCL) kalboje. Šios funkcijos yra integruotos į sprendimo variklį ir leidžia atlikti sudėtingą duomenų

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
		manipuliavimui, kodavimui ir failų sistemų prieigai.	apdorojimą be jokių išorinių papildinių ar plėtinių. Nuoroda
2.6	Pagrindinė darbo eiga	Pagrindinė darbo eiga turi būti standartizuota į tris atskiras stadijas: Write (konfigūracijos apibrėžimas), Plan (pakeitimų peržiūra) ir Apply (pakeitimų vykdymas).	Sprendimas standartizuoja infrastruktūros valdymo gyvavimo ciklą į tris aiškiai apibrėžtus etapus: „Write“, „Plan“ ir „Apply“. Ši darbo eiga užtikrina, kad pakeitimai būtų aiškiai aprašyti, iš anksto peržiūrėti saugumo ir poveikio požiūriu bei įgyvendinti nuspėjamai. Nuoroda
2.7	Vykdymo planavimas	„Plan” stadija turi sukurti spekuliatyvų vykdymo planą, kuriame nurodoma, kurie ištekliai bus sukurti, atnaujinti ar sunaikinti. Šis planas turi turėti galimybes būti išsaugotas faile ir taikomas vėliau.	Sprendimo etape „Plan“ yra sugeneruojamas spekuliatyvus vykdymo planas, kuriame detaliai nurodoma, kurie resursai bus sukurti, atnaujinti ar pašalinti. Šis planas suteikia išsamią planuojamų pakeitimų peržiūrą, pagrįstą skirtumu tarp esamos infrastruktūros būsenos ir pageidaujamos konfigūracijos. Nuoroda
2.8	Išteklių naikinimas	Darbo eiga turi apimti specialią trynimo (angl. Destroy) operaciją, užtikrinančią, jog visi konfigūracijos valdomi ištekliai būtų tinkamai pašalinti ir išvalyti, kai jie nebebus reikalingi.	Sprendimas kaip esminę standartizuotos darbo eigos dalį turi atskirą „Destroy“ operaciją. Ši operacija užtikrina tvarkingą, nuoseklų ir pilną visų konkrečios konfigūracijos valdomų resursų pašalinimą, kai jie tampa nebereikalingi. Nuoroda
2.9	Būklės valdymas	Sprendimas turi palaikyti būklės failą, kuris priskiria realaus pasaulio išteklius konfigūracijai, suteikdamas infrastruktūros būklės tikrovės šaltinį.	Sprendimas naudoja būklės failą („terraform.tfstate“) kaip pagrindinį mechanizmą, leidžiantį sekti ryšį tarp aukšto lygio konfigūracijos ir realiai valdomų resursų Nuoroda
2.10	Nuotolinio būklės saugojimo backend	Platforma turi užtikrinti natūralų ir saugų nuotolinį būklės saugojimą serverio pusėje (angl. *Backend*) su šifravimu, versijų valdymu ir role pagrįsta prieiga prie būklės failų.	Sprendimas užtikrina natyvią ir saugią nuotolinę būklės saugyklą per valdomą platformą („HCP Terraform“) arba savarankiškai talpinamą sprendimą („Terraform Enterprise“). Ši specializuota saugykla pašalina su vietiniais būsenos failais susijusias rizikas ir suteikia patikimą saugumo sistemą. Nuoroda
2.11	Būklės užrakinimas	Platforma turi suteikti automatinį būklės užrakinimą, siekiant išvengti lygiagretumo modifikacijų ir galimų būklės korupcijos situacijų, kai keli	Sprendimas turi automatinę būsenos užrakinimo funkciją, skirtą užkirsti kelią vienu metu atliekamiems pakeitimams. Ši apsauga užtikrina, kad į būsenos failą vienu metu gali rašyti tik vienas procesas, taip pašalinant lenktynių sąlygų, konfliktuojančių

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	Siūloma charakteristika *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
		vartotojai veikia toje pačioje infrastruktūroje.	atnaujinimų ar būsenos sugadinimo riziką daugiafunkcėje vartotojų aplinkoje. Nuoroda
2.12	Komandinė eilutė ir komandos	Turi būti įdiegta pilna komandinės eilutės sąsaja (angl. *Command Line Interface - CLI*) vietos vystymui ir integracijai, įskaitant komandas validavimui (validate), formatavimui (fmt) ir interaktyviai naršyklei (console).	Sprendimas turi patikimą, kelių platformų komandų eilutės sąsają (CLI), suteikiančią visus reikalingus įrankius vietiniam vystymui ir darbo eigos integracijai. Joje yra specialios komandos komandas validavimui (validate), formatavimui (fmt) ir interaktyviai naršyklei (console). Nuoroda
2.13	Mašiniškai skaitomas išėjimas	CLI turi generuoti struktūruotą JSON išvestį planams ir būklės failams, leidžiant programinį tikrinimą ir integraciją su pasirinktine automatizacija.	Sprendimo komandų eilutės sąsaja (CLI) sukurta automatizacijos pirmumo aplinkoms ir natyviai palaiko struktūrizuotą JSON išvestį. Tai leidžia programiškai tikrinti vykdymo planus, būsenos failus ir komandų rezultatus naudojant pasirinktinius scenarijus arba išorines automatizacijos platformas. Nuoroda
3.	**Reikalavimai bendradarbiavimo ir valdymo platformai**		
3.1	Vykdymo aplinkos	Platforma turi suteikti „vykdymo aplinkas“ infrastruktūros organizavimui, kurios yra kolekcijos viso, ko reikia konfigūracijos vykdymui: kodo, kintamųjų ir būklės duomenų.	Sprendimas suteikia specializuotas vykdymo aplinkas, vadinamas „Workspaces“. Nuoroda
3.2	Vykdymo aplinkų organizavimas	Turi būti palaikomas vykdymo aplinkų naudojimas monolitinių konfigūracijų skaidymui, pokyčių poveikio srities ribojimui ir atitikčiai programų ar grupių organizavimui.	Sprendimas palaiko „Workspaces“ naudojimą kaip izoliuotas vykdymo aplinkas, leidžiančias monolitines konfigūracijas suskaidyti į modulinį „mikroinfrastruktūros“ modelį. Nuoroda
3.3	Projektai	Platforma turi palaikyti vykdymo aplinkų grupavimą į "projektus" organizavimo ir valdymo mastu, taikant nuoseklias grupės teises ir politiką vykdymo aplinkų grupėms.	Sprendimas palaiko vykdymo aplinkų („Workspaces“) grupavimą į projektus („Projects“). Ši funkcija leidžia organizacijos administratoriams valdyti infrastruktūros rinkinius kaip vienetą, taikant nuoseklų vaidmenimis pagrįstą prieigos valdymą (RBAC), kintamųjų rinkinius ir valdymo (governance) politiką visose projekto aplinkose. Nuoroda
3.4	Nuotolinis vykdymas	Sprendimas turi suteikti nuoseklią nuotolinę vykdymo aplinką, naudojant vienkartinius, metamus agentus visiems parengimo procesams, siekiant užtikrinti patikimumą ir saugumą.	Sprendimas užtikrina patikimumą ir saugumą naudodama laikinas vykdymo agentūras visiems įrenginių (provisioning) procesams. Kiekviena „Plan“ ir „Apply“ operacija vykdoma naujoje, izoliuotoje aplinkoje, kuri iš karto sunaikinama po užduoties atlikimo. Nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
3.5	Vykdymo paleidimo užtaisai	Turi būti palaikoma trigerio funkcija (angl. *Run triggers*), leidžianti sukurti priklausomybę tarp vykdymo aplinkų, kad sėkmingas vienos vykdymo aplinkos taikymas automatiškai pradėtų naują vykdymą kitose vykdymo aplinkose.	Sprendimas palaiko trigerio funkciją (angl. *Run triggers*), kuri leidžia automatizuotai sukurti priklausomybes tarp skirtingų vykdymo aplinkų („Workspaces“). Tai užtikrina, kad sėkmingi pakeitimai pagrindinėje infrastruktūroje būtų automatiškai perduodami priklausomoms sluoksniams be rankinės intervencijos. Nuoroda
3.6	Automatinis grandininis taikymas	Vykdymo paleidimo užtaisai turi palaikyti nustatymą, leidžiantį automatiškai taikyti pokyčius sekimo vykdymo aplinkose, jei vykdymas buvo sėkmingas, leidžiant visiškai automatizuotus, grandininius diegimus.	Sprendimas siūlo „Auto-apply“ nustatymą, leidžiantį pereiti nuo sėkmingo plano („Plan“) prie vykdymo („Apply“) be rankinio patvirtinimo. Naudojant šią funkciją kartu su „Run Triggers“, sukuriama visiškai automatizuota diegimo (deployment) linija priklausomose vykdymo aplinkose („Workspaces“). Nuoroda
3.7	Grupių valdymas ir RBAC	Platforma turi turėti funkcijas vartotojų organizavimui grupėse ir smulkiam role pagrįstam prieigos valdymui (RBAC) organizacijos, projekto ir vykdymo aplinkos lygiuose.	Sprendimas turi išsamų vaidmenimis pagrįsto prieigos valdymo (RBAC) sistemą, sukurtą įmonių masto valdymui. Ji leidžia administratoriams taikyti „mažiausių privilegijų“ principą, grupuojant vartotojus į komandas ir suteikiant jiems detalias teises kiekviename infrastruktūros hierarchijos lygyje. Pirma nuoroda Antra nuoroda Trečia nuoroda
3.8	Pasirinktinių leidimų palaikymas	Turi palaikyti pasirinktinių leidimų rinkinių kūrimą norint valdyti smulkius veiksmus, tokius kaip vykdymų peržiūra, planų vykdymas, pokyčių taikymas, kintamųjų valdymas ir politikų viršijimas.	Sprendimas suteikia galimybę pritaikyti individualizuotą vaidmenimis pagrįstą prieigos valdymą (RBAC) darbo sričių („Workspaces“) ir projektų („Projects“) lygiuose, leidžiant administratoriams pasirinkti konkrečias teises. Nuoroda
3.9	Politikos kaip kodas	Platforma turi turėti integruotą politikos kaip kodo įrankį, skirtą saugumui, atitikčiai ir operacijų kontrolės mechanizmams užtikrinti prieš infrastruktūros parengimą.	Sprendimas apima integruotą „Policy-as-Code“ (PaC) sistemą, kuri privalomai taikoma įrenginių (provisioning) darbo eigoje, užtikrinant saugumą, atitiktį reglamentams ir operacinius apribojimus. Ši sistema įvertina infrastruktūros planus pagal užkoduotas taisykles ir blokuoja nesuderinamus pakeitimus dar prieš juos pasiekiant debesijos paslaugų tiekėją. Pirma nuoroda Antra nuoroda Trečia nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	Siūloma charakteristika *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
3.10	Daugiakarkasių politikų palaikymas	Politikos karkasas turi palaikyti tiek vietinę politikos kalbą (pvz., Sentinel ir pan.), tiek atvirą standartą Open Policy Agent (OPA ir pan.).	Sprendimas suteikia dvigubą politikos vykdymo variklių sistemą, kuri natyviai integruoja „Sentinel“ ir „Open Policy Agent“ (OPA). Ši lankstumo funkcija leidžia organizacijoms naudoti tiek specialiai sukurtas, aukšto našumo politikas, tiek universalius atviro standarto sprendimus infrastruktūros valdymui. Nuoroda
3.11	Politikos rinkiniai ir versijų valdymas	Turi leisti grupuoti politikas į "politikos rinkinius", kuriuos būtų galima valdyti versijomis kodo repozitorijoje ir taikyti globaliai, specifiniams projektams arba individualioms vykdymo aplinkoms.	Sprendimas leidžia administratoriams sugrupuoti atskiras politikas į „politikos rinkinius“ (Policy Sets). Šie rinkiniai veikia kaip valdymo sluoksnis, jungiantis užkoduotas verslo taisykles su infrastruktūros aplinkomis, palaikant tiek globalų valdymą, tiek tikslinį įgyvendinimą projektų lygyje. Nuoroda
3.12	Politikos vertinimas pagal poreikį	Platforma turi suteikti galimybę vykdyti *ad hoc* politikos vertinimus prieš paskutinį vykdymo aplinkos plano įvertinimą, leidžiant saugiai testuoti politikos pakeitimus be poveikio aktyviems vykdymams.	Sprendimas suteikia specialų vietinį ir nuotolinį testavimo procesą. Tai užtikrina, kad politikų autoriai galėtų patikrinti naujas saugumo ar atitikties taisykles realiomis sąlygomis, nedarydami įtakos veikiančioms produkcinėms vykdymo aplinkoms ar aktyviems diegimo procesams. Pirma nuoroda Antra nuoroda
4.	**Reikalavimai ekosistemai ir integracijoms**		
4.1	Teikėjų ekosistema	Sprendimas turi palaikyti plačią ekosistemą, kuri apimtų daugiau nei 4000 "teikėjų", veikiančių kaip įskiepiai valdantys išteklius daugelyje viešųjų debesų, privačioje infrastruktūroje ir SaaS paslaugose.	Sprendimą palaiko plati ekosistema, kurioje yra daugiau nei 4 000 „teikėjų“. Šie tiekėjai veikia kaip įskiepiai, verčiantys sprendimo kodą į konkrečius API kvietimus, leidžiant vienu nuosekliu darbo procesu valdyti resursus įvairiose platformose. Pirma nuoroda Antra nuoroda
4.2	Privati modulių registracija	Platforma turi turėti privatų registrą vidiniam infrastruktūros modulių dalijimuisi, atradimui ir valdymui per versijų valdymą gamintojų/vartotojų režimu.	Sprendimas turi natyvų privačių modulių registrą („Private Module Registry“), veikiantį kaip centralizuotas „Vidinis paslaugų katalogas“ patvirtintiems infrastruktūros komponentams. Šis registras leidžia organizacijoms pereiti nuo susiskaidžiusių, ad hoc scenarijų prie valdomos gamintojų / vartotojų ekosistemos. Nuoroda
4.3	Modulių testavimo karkasas	Sprendimas turi pateikti vietinį testavimo karkasą vienetiniam ir integraciniam modulių testavimui	Sprendimas turi testavimo karkasą /sistemą. Ši sistema leidžia kurti automatizuotus testų rinkinius tiek vienetiniams, tiek integracijos testams, naudojant tą pačią konfigūracijos

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	Siūloma charakteristika *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
		naudojant tą pačią konfigūracijos kalbą.	kalbą (HCL), kuria apibrėžiama pati infrastruktūra. Nuoroda
4.4	Modulių imitavimo testavimas	Testavimo karkasas turi palaikyti teikėjo API atsakų imitavimą, leidžiantį greitai vykdyti be kredencialų vienetinius testus modulių logikai.	Sprendimas turi API atsakų imitavimą („Mocking Framework“), integruotą į komandą *terraform test.* Ši funkcija leidžia tikrųjų debesijos tiekėjų („cloud providers“) vietoje naudoti „mock“ tiekėjus, kurie imituoja tikrųjų paslaugų schemą ir elgseną. Tai suteikia galimybę greitai, saugiai ir be prisijungimo duomenų vykdyti vienetinius testus sudėtingai modulio logikai. Nuoroda
4.5	Integruotas testavimas ir publikavimas	Platforma turi palaikyti darbo eigą automatiniams modulių testams vykdyti ir sėkmei patvirtinti prieš leidžiant juos į privatų registrą.	Sprendimas palaiko automatizuotą testavimo integruotą darbo eigą („Test-Integrated Workflow“) privačiam modulių registrui. Ši funkcija užtikrina, kad kiekviena modulio versija būtų kruopščiai patikrinta prieš ją padarant prieinamą organizacijos naudotojams, efektyviai užkertant kelią sugadinto ar nesuderinamo infrastruktūros kodo platinimui. Nuoroda
4.6	Versijų valdymo sistemos integracija (VCS)	Platforma turi siūlyti gilias, natūralias integracijas su visomis pagrindinėmis versijų valdymo sistemomis, tokiomis kaip GitHub, GitLab, Bitbucket ir Azure DevOps arba lygiavertėmis, palaikančias debesijos ir savipalaikomas instancijas.	Sprendimas turi pirmos klasės integracijas su visomis pagrindinėmis versijų valdymo sistemomis. Ji palaiko tiek debesijoje talpinamas (SaaS), tiek savarankiškai diegiamas (on-premises) šių sistemų instancijas, užtikrindama, kad organizacijos galėtų išlaikyti esamus saugumo ir architektūros nustatymus, automatizuojant infrastruktūros tiekimą. Nuoroda
4.7	Privati VCS jungtis	Sprendimas turi palaikyti jungtis su savipalaikomomis VCS instancijomis privačiuose tinkluose per savipalaikomą agentą be įeinančios interneto prieigos.	Sprendimas palaiko saugų ryšį su savarankiškai talpinamomis versijų valdymo sistemomis (VCS), esančiomis privačiuose tinkluose, naudojant „HCP Terraform Agents“ su užklausų persiuntimu (Request Forwarding). Ši pull tipo architektūra pašalina poreikį atidaryti įeinančius ugniasienės taisykles arba viešai eksponuoti jūsų vidinius kodo saugyklos resursus internete. Nuoroda
4.8	API prieiga ir vykdymo užduotys	Platforma turi teikti išsamų REST API ir palaikyti užduočių vykdymo funkciją (angl. *Run tasks*) integruoti trečiųjų šalių partnerių ir pasirinktines paslaugas į	Sprendimas sukurtas kaip išplečiama orkestravimo sistema. Ji teikia išsamią REST API (v2) programiniam valdymui ir natyvią „Run Tasks“ sistemą, leidžiančią integruoti trečiųjų šalių paslaugas arba pasirinktinius

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
		parengimo darbo eigą skirtingais etapais (pvz., prieš planą, po plano ir pan.).	vidinius įrankius konkrečiuose, smulkiai valdomuose įrenginių (provisioning) proceso etapuose. Pirma nuoroda Antra nuoroda
4.9	Privatūs vykdymo užduočių vykdytojai	Turi būti galimybė vykdymo užduočių integracijas vykdyti per savipalaikomą agentą, leidžiant platformai jungtis prie trečiųjų šalių priemonių, esančių privačiame tinkle.	Sprendimas plečia „Run Tasks“ sistemą į privačias aplinkas, naudojant savarankiškai talpinamus agentus („Self-Hosted Agents“) su įjungtu užklausų persiuntimu (Request Forwarding). Tai leidžia platformai inicijuoti užduočių užklausas iš saugaus tinklo viduje, užtikrinant, kad jautri metaduomenų informacija ir vykdymo planai būtų siunčiami tik patikimiems vidiniams galutiniams taškams. Nuoroda
4.10	Atvaizdų atitikties integracija	Sprendimas turi turėti natūralią vykdymo užduočių integraciją su įmonių valdomais atvaizdų sprendimais (pvz., HCP Packer arba lygiaverčiais ir pan.), užtikrinančią infrastruktūros diegimą naudojant patvirtintus ir suderintus mašinos atvaizdus.	Sprendimas integruojasi su „HCP Packer“, kad sukurtų saugią ir automatizuotą „Golden Image“ darbo eigą. Tai užtikrina, kad infrastruktūra būtų diegiama tik naudojant mašinų vaizdus (AMIs, VMDKs ir pan.), kurie buvo sukurti, patikrinti ir patvirtinti centralizuotų platformos ar saugumo komandų. Nuoroda
4.11	ITSM ir paslaugų katalogas	Turi būti oficialus, sertifikuotas integravimas su ServiceNow, leidžiantis savitarnos parengimą ir CMDB atnaujinimus. Sprendimas taip pat turi palaikyti AWS Service Catalog arba lygiavertę integraciją.	Sprendimas apima dvi atskiras, sertifikuotas programas, prieinamas „ServiceNow Store“, kurias kuria ir palaiko „HashiCorp“, siekiant padengti visą savitarnos ir turto sekimo gyvavimo ciklą. Sprendimas teikia oficialią integraciją su „AWS Service Catalog“, leidžiančią AWS administratoriams tvarkyti ir valdyti Terraform produktus pažįstamoje AWS sąsajoje. Pirma nuoroda Antra nuoroda
4.12	Kubernetes operatorius	Turi būti galimybė valdyti Kubernetes operatorių platformos išteklius (pvz., vykdymo aplinkos, agentų grupės ir pan.) bei inicijuoti naudojant Kubernetes Custom Resource Definitions (CRDs) arba lygiavertį sprendimą.	Sprendimas teikia pirmos klasės „Kubernetes“ operatorių, kuris leidžia platformos resursus pateikti kaip „Kubernetes“ objektus. Tai suteikia galimybę platformos komandoms valdyti HCP Terraform naudodami standartines *kubectl* komandas ir GitOps įrankius (pvz., ArgoCD ar Flux), traktuojant infrastruktūros automatizavimo resursus kaip kodą „Kubernetes“ valdymo lape. Nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
4.13	Kūrėjų platformos integracija	Turi palaikyti integraciją su kūrėjų platformomis (pvz., HCP Waypoint ar pan.) arba lygiaverčiais sprendimais, suteikiančią standartizuotą, geriausių praktikų darbo eigą diegiant programas virš valdomos infrastruktūros.	Sprendimas sukurtas būti „Infrastruktūros varikliu“ bet kuriai moderniai vidinei kūrėjų platformai (Internal Developer Platform, IDP). Ji suteikia galingą API rinkinį ir natyvius tiekėjus (providers), leidžiančius trečiųjų šalių kūrėjų portalams diegti, valdyti ir vizualizuoti infrastruktūrą naudojant standartizuotą, geriausiomis praktikomis pagrįstą darbo eigą. Pirma nuoroda Antra nuoroda Trečia nuoroda
5.	**Reikalavimai saugumui ir atitikčiai**		
5.1	Dinaminiai teikėjų kredencialai	Platforma turi palaikyti OIDC pagrindu sukurtus dinamiškus teikėjų kredencialus trumpalaikiams (angl. *just-in-time*) (JIT) prieigos tokenams generuoti autentifikacijai prie pagrindinių debesijos teikėjų arba Kubernetes.	Sprendimas turi natyvią OIDC pagrindu veikiančią tapatybės mainų sistemą, vadinamą „Workload Identity“. Ši sistema leidžia Terraform veikti kaip tapatybės teikėjas, užmezgant pasitikėjimo santykį su išorinėmis platformomis ir keičiant trumpalaikį OIDC žetoną į laikinas debesijos paskyras (cloud credentials). Nuoroda
5.2	Smulkūs JIT role	Dinaminiai kredencialai turi palaikyti skirtingų IAM rolių priskyrimą plano ir vykdymo etapams, leidžiant taikyti mažiausių privilegijų principą.	Sprendimas palaiko fazėms pritaikytas dinamines prieigos teises („Phase-Specific Dynamic Credentials“). Tai leidžia priskirti skirtingus IAM vaidmenis „Plan“ ir „Apply“ etapams vienoje vykdymo operacijoje, užtikrinant, kad planavimo etape būtų suteikta tik skaitymo teisė, o vykdymo etape – reikalingos rašymo teisės. Pirma nuoroda Antra nuoroda
5.3	Vieningas prisijungimas (SSO)	Platforma turi palaikyti SAML 2.0 integraciją su įmonės tapatybės teikėju viengubam prisijungimui (SSO), įskaitant tapatybės teikėjo grupių atvaizdavimą į platformos grupes.	Sprendimas suteikia natyvią SAML 2.0 integraciją su automatizuotu komandos narių priskyrimo palaikymu (Automated Team Membership Mapping). Tai leidžia centralizuotai valdyti vartotojų teises jūsų Tapatybės teikėjo (Identity Provider, IdP) sistemoje, o teisių sinchronizacija vyksta automatiškai prisijungimo metu. Nuoroda
5.4	Audito žurnalai	Turi būti generuojami išsamūs auditų žurnalai visiems vartotojų veiksmams ir sistemos įvykiams. Žurnalai turi turėti galimybę būti transliuojamiems į išorines SIEM sistemas, tokias kaip Splunk.	Sprendimas suteikia natyvią audito žurnalų srautų funkciją („Native Audit Log Streaming“). Tai leidžia fiksuoti kiekvieną vartotojo veiksmą ir sistemos įvykį realiuoju laiku bei siųsti juos tiesiogiai į išorines SIEM sistemas, tokias kaip Splunk, Datadog ar Amazon CloudWatch. Nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	Siūloma charakteristika *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
5.5	Saugūs kintamieji	Platforma turi suteikti saugią kintamųjų saugyklą, kurioje visos reikšmės turi būti šifruojamos prieš įrašymą.	Sprendimas suteikia saugią duomenų saugyklą (Secret Storage System), pagrįstą „Vault“ tranzito šifravimu (transit encryption). Ši architektūra užtikrina, kad jautrūs kintamieji būtų užšifruoti dar prieš pasiekiant nuolatinę saugyklą ir išliktų užšifruoti saugykloje. Nuoroda
5.6	Pakartotinai naudojamos kintamųjų grupės	Turi būti leidžiama grupuoti bendrus kintamuosius į kintamųjų grupes, kurias galima būtų naudoti keliuose vykdymo aplinkose, mažinant dubliavimą.	Sprendimas palaiko kintamųjų rinkinius („Variable Sets“), kurie veikia kaip pakartotinai naudojami konteineriai tiek „Terraform“ įvesties kintamiesiems, tiek aplinkos kintamiesiems. Šis mechanizmas leidžia platformos komandoms vieną kartą apibrėžti standartines konfigūracijas ar prisijungimo duomenis ir pritaikyti juos bet kuriam skaičiui vykdymo aplinkų. Nuoroda
5.7	Prioritetinės kintamųjų grupės	Platforma turi suteikti prioritetines kintamųjų grupes, leidžiančias platformos administratoriams apibrėžti specifines kintamųjų reikšmes, kurių negalima būtų keisti atskirų vykdymo aplinkų nustatymuose.	Sprendimas naudoja prioritetines kintamųjų grupes („Priority Variable Sets“). Ši funkcija leidžia platformos administratoriams apibrėžti privalomus kintamuosius, kurie turi aukščiausią prioritetą prieš visas kitas reikšmes ir negali būti pakeisti ar perrašyti naudotojų atskirose darbo srityse („workspaces“). Nuoroda
5.8	Centralizuotas tokenų valdymas	Platforma turi suteikti centralizuotą vartotojo sąsają administratoriams valdyti ir atšaukti visas grupės API žymas, įskaitant informaciją apie paskutinį naudojimą ir galiojimo datas.	Sprendimas turi centralizuotą saugumo ir API žetonų („Security & API Tokens“) valdymo sąsają. Ši valdymo sritis suteikia platformos administratoriams visišką matomumą į programinių prisijungimo duomenų gyvavimo ciklą, įskaitant komandų (grupių) lygmens žetonus, ir užtikrina, kad visa prieiga atitiktų organizacijos saugumo politiką. Nuoroda
6.	**Reikalavimai gyvavimo ciklo valdymui ir antrinei peržiūrai**		
6.1	Nukrypimų aptikimas	Platforma turi suteikti automatizuotą, nuolatinį nukrypimų aptikimą, kuris identifikuoja skirtumus tarp realios infrastruktūros būklės ir apibrėžtos konfigūracijos, su konfigūruojamais perspėjimais apie aptiktus nukrypimus.	Sprendimas turi natyvų, automatizuotą konfigūracijos nukrypimų (drift) aptikimą, kuris užtikrina nuolatinį jūsų daugiadebesės infrastruktūros būsenos matomumą. Vykdydama foninius būklės patikrinimus, platforma nedelsiant aptinka bet kokius „už juostos ribų“ („out-of-band“) pakeitimus – tiek rankinius pakeitimus debesijos valdymo konsolėse, tiek atnaujinimus iš išorinių automatizavimo įrankių – ir pateikia juos šalinimui. Nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	Siūloma charakteristika *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
6.2	Nuolatinė validacija	Sprendimas turi palaikyti nuolatinę parengtų išteklių patikrą pagal pasirinktines sąlygas ir patikras (pvz., atvaizdų panaikinimo ar galiojančių sertifikatų tikrinimą ir pan.).	Sprendimas turi nuolatinę parengtų išteklių patikrą („Continuous Validation“) – automatizuotą testavimo sistemą, kuri reguliariais intervalais vykdo individualiai apibrėžtą logiką. Tai užtikrina, kad jūsų daugiadebesė aplinka nuolat atitiktų funkcinius ir saugumo reikalavimus, pavyzdžiui, kad naudojamas mašinos vaizdas nebūtų atšauktas arba TLS sertifikatas išliktų galiojantis, nereikalaujant rankinio „Terraform“ vykdymo. Nuoroda
6.3	Konfigūracijos pagrindu importavimas	Turi būti deklaratyvus, konfigūracija pagrįstas importavimo procesas, leidžiantis perimti esamą infrastruktūrą į valdymą, su planavimo ir peržiūros ciklu prieš būklės modifikaciją.	Sprendimas palaiko deklaratyvų, konfigūracija pagrįstą importavimo procesą, leidžiantį komandoms perkelti anksčiau nevaldomą infrastruktūrą į „Terraform“ valdymą naudojant tuos pačius tarpusavio peržiūros (peer review) ir planavimo ciklus, kaip ir kuriant naujus resursus. Nuoroda
6.4	Community Edition migracija	Turi būti pateikiama komandinės eilutės priemonė automatizuoti ir supaprastinti atviro kodo konfigūracijų bei būklės failų migraciją į verslo platformą.	Sprendimas teikia komandų eilutės įrankį tf-migrate. Šis specialiai sukurtas CLI įrankis skirtas automatizuoti perėjimą iš atviro kodo konfigūracijų bei būklės failų migraciją į verslo platformą Nuoroda
6.5	Ephemerinės vykdymo aplinkos	Platforma turi palaikyti automatinius išteklių sunaikinimus vykdymo aplinkose pagal konfigūruojamą gyvavimo laiką (TTL) ar neveiklumo periodą.	Sprendimas suteikia laikinąsias darbo sritis („Ephemeral Workspaces“). Ši funkcija leidžia platformos komandoms taikyti „Time-to-Live“ (TTL) politikas, kurios automatiškai inicijuoja infrastruktūros sunaikinimo („Destroy“) vykdymą, kai darbo sritis pasiekia nustatytą galiojimo datą arba apibrėžtą neaktyvumo laikotarpį. Nuoroda
6.6	Projektinis valymas	Turi būti galimybė nustatyti automatinio sunaikinimo politikas projekto lygiu, užtikrinant nuoseklų valymą visose projekto vykdymo aplinkose.	Sprendimas palaiko projekto lygmens automatinio sunaikinimo nustatymus („Project-Scoped Auto-Destroy Settings“). Ši valdymo (governance) funkcija leidžia platformos administratoriams apibrėžti numatytąjį „Time-to-Live“ (TTL) visoms projekto darbo sritims („workspaces“), užtikrinant, kad laikinoji infrastruktūra būtų automatiškai išjungta ir pašalinta visoje projekto aplinkoje. Nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
6.7	Centralizuotas matomumas	Sprendimas turi apimti centrinį valdymo skydą, suteikiantį organizacijos masto matomumą visoms vykdymo aplinkoms, jų būklės būsenoms, naudojamų modulių/teikėjų versijoms ir nukrypimų būsenai.	Sprendimas turi centralizuotą valdymo skydelį, vadinamą „Explorer“. Ši specializuota sąsaja suteikia išsamų, aukšto lygio vaizdą apie visas darbo sritis („workspaces“), jų būklės rodiklius ir visoje organizacijoje naudojamas tarpusavio priklausomybes. Nuoroda
6.8	Be kodo parengimas	Platforma turi suteikti be kodo parengimo darbo eigą, leidžiančią galutiniams vartotojams atrasti ir diegti infrastruktūrą iš patvirtintų modulių registre be konfigūracijos kodo rašymo.	Sprendimas suteikia be kodo („No-Code“) infrastruktūros diegimo galimybę. Ši funkcija leidžia atrasti ir diegti iš anksto patvirtintus infrastruktūros šablonus tiesiogiai iš privataus registrų („Private Registry“). Nuoroda
6.9	Modulių gyvavimo ciklo valdymas	Turi būti suteikiama sisteminė darbo eiga modulių visam gyvavimo ciklui valdyti, įskaitant naudojimo stebėseną ir pasenusių versijų atšaukimą rizikos mažinimui.	Sprendimas suteikia išsamų modulio gyvavimo ciklo valdymo („Module Lifecycle Management“) sistemą. Ši sistema leidžia platformos komandoms valdyti modulius nuo jų pradinio išleidimo iki stebėsenos, pasenimo („deprecation“) ir galutinio atšaukimo, taip mažinant pasenusių arba nesuderinamų infrastruktūros komponentų keliamą riziką. Nuoroda
6.10	Įvykių pranešimai	Platforma turi suteikti pritaikomas pranešimų galimybes plačiam vykdymo aplinkos būklės ir paleidimo įvykių spektrui, leidžiančias siųsti pranešimus el. paštu, Slack, Teams ir webhook.	Sprendimas suteikia patikimą darbo sričių („Workspace“) pranešimų sistemą. Ši funkcija leidžia administratoriams konfigūruoti realaus laiko pranešimus visam vykdymo („run“) gyvavimo ciklui – įskaitant „plan“, „apply“ ir būklės vertinimo („health assessment“) etapus – kurie siunčiami per natyvią integraciją su Slack, Microsoft Teams, el. paštu arba universaliais („Generic“) webhooks. Nuoroda
7.	**Reikalavimai pažangiam orkestravimo varikliui**		
7.1	Sudėtiniai diegimai	Sprendimas turi suteikti pažangų variklį kurti ir valdyti sudėtingus infrastruktūros diegimus iš kelių atskirų komponentų (pvz., tinklų, duomenų bazių, skaičiavimo ir pan.).	Sprendimas suteikia „Terraform Stacks“. Šis pažangus orkestravimo variklis leidžia apibrėžti ir diegti visą aplinką – derinant tinklus, duomenų bazes, skaičiavimo išteklius ir kitus komponentus – kaip vieningą, sinchronizuotą vienetą, žymiai pranokstant nepriklausomų darbo sričių („workspaces“) apribojimus. Nuoroda
7.2	Deklaratyvus orkestravimas	Variklis turi leisti valdyti visą kelių komponentų infrastruktūros gyvavimo ciklą ir įvairias diegimo aplinkas (pvz., Staging, QA,	Sprendimo orkestravimo variklis yra sukurtas valdyti daugiasluoksnę infrastruktūrą ir jos skirtingas diegimo aplinkas (pvz., Staging,

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
		Production ir pan.) kaip vieną kodifikuotą vienetą.	QA ir Production) kaip vieningą, užkoduotą valdymo vienetą. Nuoroda
7.3	Atidėto pakeitimų valdymas	Variklis turi generuoti dalinį planą, kai susiduria su nežinomomis reikšmėmis dėl priklausomybių. Jis turi atidėti planavimą ir automatiškai vykdyti tolesnius planus, kai reikšmės tampa žinomos, taip išvengdamas rankinių įsikišimų.	Sprendimas sprendžia nežinomų reikšmių („unknown value“) problemą. Kai konfigūracijoje yra resursų, kurie priklauso nuo reikšmių, prieinamų tik po fizinio diegimo (pvz., Kubernetes manifestui reikalingas klasterio galinis taškas), variklis sukuria dalinį planą ir atideda tolesnius veiksmus tol, kol šios reikšmės bus patvirtintos. Nuoroda
7.4	Orkestravimo taisyklės	Variklis turi palaikyti orkestravimo taisykles, automatizuojančias rankinius diegimo veiksmus, įskaitant automatinį planų patvirtinimą, kai atitinka saugumo kriterijus (pvz., nėra destruktyvių pakeitimų ir pan.).	Sprendimas naudoja orkestravimo taisykles („Orchestration Rules“) „Terraform Stacks“ variklyje. Ši funkcija leidžia administratoriams pakeisti rankinius patvirtinimo žingsnius deklaratyvia „auto-approve“ logika, pagrįsta tam tikrais kriterijais, pavyzdžiui, destruktyvių pakeitimų nebuvimu arba sėkmingu saugumo patikrinimų atlikimu. Nuoroda
7.5	VCS darbo eigos integracija	Pažangus orkestravimo variklis turi integruotis su standartine, VCS pagrindu valdoma darbo eiga, leidžiančia planuoti sudėtinę infrastruktūrą iš kodo repozitorijos kreipinių (angl. *Commit, pull request*).	Sprendimas suteikia VCS valdomą darbo eigą („VCS-driven workflow“), kuri yra pilnai integruota su „Terraform Stacks“. Ši integracija leidžia inicijuoti sudėtingus, daugiasluoksnius infrastruktūros planus ir diegimus tiesiogiai iš saugyklos įvykių, tokių kaip „commit“ ar „pull request“. Nuoroda

7. SPECIALIEJI REKALAVIMAI IŠLAIDŲ VALDYMO SPRENDIMUI

7.1. Išlaidų valdymo sprendimui keliami specialieji programinės įrangos funkcionalumo reikalavimai:

7.1.1. Išlaidų valdymo sprendimas turi rinkti kasdienes išlaidų ir naudojimo ataskaitas iš VSSA naudojamų viešosios debesijos paslaugų teikėjų suteikiamų įrankių: AWS Cost and Billing Reports, Azure Cost Management APIs ir GCP Billing Export.

7.1.2. Išlaidų valdymo sprendimas turi normalizuoti išlaidų duomenis iš skirtingų debesijos teikėjų į suvienytą formatą su nuoseklia valiuta, laiko zonomis ir išlaidų kategorijomis.

7.1.3. Išlaidų valdymo sprendimas turi paskirstyti išlaidas klientams, projektams ir aplinkoms naudojant išsamias žymėmis pagrįstas atvaizdavimo taisykles, konfigūruotas išteklių parengimo metu.

7.1.4. Išlaidų valdymo sprendimas turi palaikyti hierarchinį išlaidų paskirstymą, įgalinantį išlaidų suvestinę nuo atskirų išteklių iki projekto ir organizacijos lygių.

7.1.5. Išlaidų valdymo sprendimas turi sekti išlaidų tendencijas laiko eigoje ir identifikuoti išlaidų anomalijas ar netikėtus išlaidų padidėjimus proaktyviam perspėjimui.

7.1.6. Išlaidų valdymo sprendimas turi teikti išlaidų prognozavimo galimybes remiantis istoriniais naudojimo šablonais ir esamomis išteklių konfigūracijomis.

7.1.7. Išlaidų valdymo sprendimas turi saugoti numatytuosius biudžeto IaC šablonus, išlaidų centrų atvaizdavimus ir perspėjimų ribas kaip kodą Git saugyklose versijų kontrolei ir auditavimui.

7.1.8. Išlaidų valdymo sprendimas turi siųsti el. pašto pranešimus nurodytiems kontaktams, kai išlaidos pasiekia 80 % ir 100 % biudžeto ribų.

7.1.9. Išlaidų valdymo sprendimas turi palaikyti kelis biudžeto tipus, įskaitant mėnesio limitus, projekto gyvavimo ciklo limitus ir ištekliui specifinius apribojimus.

7.1.10. Išlaidų valdymo sprendimas turi teikti žiniatinklio vartotojo sąsają, rodančią išlaidų informaciją dabartiniam mėnesiui ir istoriniams duomenims, t. y. ankstesnių mėnesių išlaidoms.

7.1.11. Išlaidų valdymo sprendimas turi atskleisti REST API galutinį tašką Įmonės atsiskaitymo sistemos integracijai su standartizuotais duomenų formatais ir puslapių numeracijos palaikymu.

7.1.12. Išlaidų valdymo sprendimas turi teikti išlaidų duomenis su privalomais laukais: record_id, organisation_id, cost_center, cloud_provider, billing_period_start, billing_period_end, service_name, usage_quantity, cost_amount, currency, budget_id, tags, generated_at.

7.2. Žemiau pateikiami detalūs specialieji reikalavimai išlaidų valdymo sprendimo programinei įrangai ir jos funkcionalumui (žr. 4 lentelę).

4 lentelė. Reikalavimai išlaidų valdymo sprendimo programinei įrangai

Eil. Nr.	**Charakteristikos pavadinimas**	**Reikalaujama charakteristika**	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
1.	**Bendrieji reikalavimai**		
1.1	Siūlomos programinės įrangos pavadinimas	***International Business Machines Corporation (IBM), IBM Cloudability Standard***	
1.2	Licencijų kiekis	Tiekėjas turi pateikti tiek licencijų, kad siūloma programine įranga būtų galima valdyti ne mažiau 300 vnt. klientų viešosios debesijos platformose naudojamus resursus. Planuojamus valdyti resursus klientai naudoja AWS, Azure ir GCP viešosios debesijos platformose. Turi būti užtikrinta bendra planuojamų valdyti resursų apimtis – klientų išlaidos debesijos platformų gamintojams bus ne didesnės kaip 1 mln. Eur be PVM.	Siūloma licencija suteikia galimybę valdyti ne mažiau 300 vnt. klientų viešosios debesijos platformose naudojamus resursus, užtikrinant reikalaujamą bendrą planuojamų valdyti resursų apimtį AWS, Azure ir GCP viešosios debesijos platformose – klientų išlaidos debesijos platformų gamintojams bus ne didesnės kaip 1 mln. Eur be PVM Nuoroda
1.3	Gamintojo palaikymas	Turi turėti techninį palaikymą iš gamintojo, visą prenumerata pagrįstos licencijos galiojimo laikotarpį.	Siekiant užtikrinti nuolatinį techninį palaikymą iš gamintojo, tvirtiname, kad siūlomas sprendimas yra valdomas pagal standartinius IBM programinės įrangos prenumeratos ir palaikymo (Software Subscription and Support, S&S) bei SaaS palaikymo (SaaS Support) mechanizmus. Nuoroda
2.	**Reikalavimai vartotojų ir paskyrų valdymui**		
2.1	SSO palaikymas (SAML/WS-Fed)	Sprendimas turi palaikyti Vieningo prisijungimo (SSO) autentifikavimą ir autorizavimą per tapatybės federacijos standartus, palaikydamas SAML 1.1, SAML 2.0 ir WS-Federation. Integracija turi leisti VSSA daugiapakopės autentifikacijos naudojimą.	Sprendimas suteikia patikimas vieningo prisijungimo (SSO) ir tapatybių federacijos galimybes. Sprendimas deleguoja autentifikavimą organizacijos tapatybės teikėjui. Tai užtikrina, kad prieiga būtų valdoma pagal įmonės saugumo politikas, įskaitant daugiapakopės autentifikavimą (MFA) ir sąlygines prieigos taisykles.

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
			Pirma nuoroda Antra nuoroda
2.2	Vaidmenimis pagrįsta prieigos kontrolė (RBAC)	Sprendimas turi užtikrinti pilną vaidmenimis pagrįstos prieigos kontrolę (RBAC). Visoms vartotojų paskyroms turi būti priskirtas vaidmuo su atitinkamais leidimais.	Sprendimas suteikia išsamų vaidmenimis pagrįstą prieigos valdymą (RBAC), kuris atitinka reikalavimą dėl privalomo vaidmenų priskyrimo ir tinkamų prieigos teisių suteikimo. Nuoroda
2.3	Iš anksto apibrėžti ir pasirinktiniai vaidmenys	Sprendimas turi apimti iš anksto nustatytų standartinių vaidmenų rinkinį su dažnai naudojamais leidimais ir suteikti galimybę kurti naujus, labai pritaikytus vaidmenis su smulkiomis teisėmis.	Sprendimas įgyvendina reikalavimą per centralizuotą vaidmenų ir prieigos valdymo sistemą, kuri naudoja „Access Administration“ ir leidžia priskirti standartinius vaidmenis bei valdyti teises vartotojams. Ši sistema suteikia biblioteką iš anksto apibrėžtų vaidmenų, kurių kiekvienas turi konkrečias privilegijas, ir palaiko tiek standartinių vaidmenų, tiek jų pritaikymų valdymą per identiteto ir prieigos kontrolės portalą. Nuoroda
2.4	Kelių vaidmenų priskyrimas	Sprendimas turi leisti vienam vartotojui priskirti daugiau nei vieną vaidmenį.	Sprendimas leidžia vienam vartotojui priskirti daugiau nei vieną vaidmenį. Kai vartotojas turi kelis vaidmenis, platforma taiko prieigos teisių „pridėjimo“ modelį (additive permission model), kurio metu vartotojui suteikiamos visos teises iš visų priskirtų vaidmenų, o ne griežtai apribojant prieigą tik iki mažiausiai reikalingų teisių. Nuoroda
2.5	Mažiausių privilegijų principas	Sprendimas turi taikyti mažiausių privilegijų principą, leisdamas prieigos teises (pvz., skaitymas, rašymas, prieiga prie konkrečių duomenų ir pan.) suteikti pagal konkrečias vartotojo pareigas.	Sprendimas įgyvendina mažiausių privilegijų principą (Principle of Least Privilege, PoLP). Jis leidžia administratoriams riboti prieigos teises ir duomenų matomumą taip, kad vartotojai turėtų tik tas teises, kurios yra būtinos jų konkrečioms darbo funkcijoms atlikti. Pirma nuoroda Antra nuoroda
2.6	Iš anksto apibrėžti administratorių/vartotojų vaidmenys	Sprendimas turi apimti bent šiuos iš anksto apibrėžtus vaidmenis: bendrą “Administratorių” vartotojų valdymui (pridėti, redaguoti, ištrinti), sprendimui specifinį „Administratorių“ įrankio vaizdų valdymui ir „Vartotoją“ su bendrąja prieiga prie programos.	Sprendimas atitinka iš anksto apibrėžtų vaidmenų reikalavimą: bendrą “Administratorių” vartotojų valdymui (pridėti, redaguoti, ištrinti), sprendimui specifinį „Administratorių“ įrankio vaizdų valdymui ir „Vartotoją“ su bendrąja prieiga prie programos.

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
			Pirma nuoroda Antra nuoroda
3.	**Reikalavimai administravimui ir sąsajai**		
3.1	Vieninga žiniatinklio sąsaja	Sprendimas turi suteikti žiniatinklio pagrindu sukurtą, intuityvią ir draugišką naudotojui sąsają, kuri suteiktų "vieno lango" vaizdą į visas kelių debesijų išlaidas.	Sprendimas suteikia žiniatinklio (web) pagrindu veikiančią „vieno lango" („single pane of glass") sąsają, specialiai sukurtą daugiadebesijos (multi-cloud) kaštams konsoliduoti. Nuoroda
3.2	Pritaikomi valdymo skydeliai ir ataskaitos	Naudotojo sąsaja turi turėti platų įvairių ataskaitų pasirinkimą, pateiktų grafikuose, diagramose ir lentelėse. Visi valdymo skydeliai ir ataskaitos turi būti visiškai pritaikomi.	Sprendimas turi išsamų ataskaitų ir valdymo skydelių (dashboard) kūrimo modulį. Platforma siūlo platų vizualizacijų pasirinkimą, įskaitant pažangius grafikus, diagramas ir lenteles, kurios yra pritaikomos pagal skirtingų šalių specifinius poreikius. Pirma nuoroda Antra nuoroda Trečia nuoroda
3.3	Pažangus filtravimas	Kiekviena ataskaita ir valdymo skydelio valdiklis turi turėti filtravimo galimybes, suteikdamas tikslų, kontekstui pritaikytą debesijos išlaidų vaizdą.	Sprendimas suteikia platų ataskaitų pasirinkimą ir visiškai pritaikomus valdymo skydelius. Sąsaja sukurta taip, kad sudėtingus daugiadebesijos (multi-cloud) sąskaitų duomenis paverstų vizualiai aiškiomis ir praktiškai panaudojamomis įžvalgomis, naudojant įvairius atvaizdavimo formatus. Nuoroda
3.4	Daugiavaliutis ir kalbų palaikymas	Sprendimas turi palaikyti kelias valiutas visuose išlaidų valdymo, ataskaitų ir analizės funkcionalumuose. Pagrindinė naudotojo sąsajos kalba turi būti anglų.	Sprendimas pilnai palaiko kelių valiutų (multi-currency) valdymą ir naudoja anglų kalbą kaip pagrindinę naudotojo sąsajos kalbą. Nuoroda
4.	**Reikalavimai išlaidų valdymui**		
4.1	Kelių debesijų duomenų įsisavinimas	Sprendimas turi turėti galimybę įsisavinti, apdoroti ir pateikti išlaidų ir naudojimo duomenis iš Amazon Web Services (AWS), Microsoft Azure ir Google Cloud Platform (GCP) viename vaizde.	Sprendimas yra specialiai sukurtas rinkti, apdoroti ir pateikti AWS, „Azure" ir GCP duomenis viename, suvienodintame vaizde. Platforma veikia kaip „vieno lango" („single pane of glass") sprendimas, suvienodindama skirtingus šių debesijos tiekėjų sąskaitų formatus ir duomenų struktūras į bendrą „kalbą", leidžiančią valdyti daugiadebesijos ekosistemą kaip vieną vientisą visumą. Pirma nuoroda Antra nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
4.2	Valandinė išlaidų detalizacija	Sprendimas turi teikti išlaidų duomenis su bent valandine detalizacija.	Sprendimas yra sukurtas taip, kad galėtų apdoroti didelį šiuolaikinių debesijos sąskaitų detalumo lygį. Sprendimas surenka ir apdoroja valandinius duomenis, patikrina įsipareigojimų (commitment) padengimą ir analizuoja vieneto ekonomikos rodiklius. Nuoroda
4.3	Lankstus išlaidų paskirstymas	Sprendimas turi siūlyti lanksčias išlaidų paskirstymo galimybes, įskaitant paskirstymą pagal teikėjo žymes, paskyros/prenumeratos struktūras bei vartotojo apibrėžtus verslo grupavimus (verslo atvaizdavimus).	Sprendimas suteikia plačias ir lanksčias sąnaudų paskirstymo galimybes, leidžiančias organizacijoje paskirstyti 100 % debesijos išlaidų. Platforma palaiko paskirstymą pagal tiekėjų žymas („provider tags“), sąskaitų hierarchijas ir sudėtingą vartotojo apibrėžtą verslo logiką. Nuoroda
4.4	Kubernetes išlaidų paskirstymas	Sprendimas turi turėti galimybę automatiškai atvaizduoti ir paskirstyti išlaidas, susijusias su Kubernetes (K8s) klasteriais.	Sprendimas turi specialų „Container Insights“ modulį. Šis įrankis automatizuoja klasterių, veikiančių AWS (EKS), „Azure“ (AKS), GCP (GKE) ir vietiniuose („on-premise“, pvz., OpenShift) aplinkose, sąnaudų atradimą ir paskirstymą, susiedamas resursų naudojimo metrikas su sąskaitų duomenimis. Nuoroda
4.5	Rodymo/grąžinimo ataskaitos	Sprendimas turi teikti rodymo ir grąžinimo ataskaitų funkcionalumą.	Sprendimas palaiko tiek rodymo („showback“), tiek grąžinimo („chargeback“) ataskaitas. Platforma suteikia reikiamus įrankius, leidžiančius pereiti nuo paprasto debesijos išlaidų matomumo prie subrendusio „FinOps“ modelio, kuriame sąnaudos pagrįstai priskiriamos jomis besinaudojančioms verslo vienetams. Nuoroda
4.6	Išlaidų amortizacija	Sprendimas turi turėti galimybę amortizuoti vienkartines išlaidas (pvz., išankstinius rezervavimo mokesčius ir pan.) per jų gyvavimo laikotarpį ir atspindėti tai ataskaitose.	Sprendimas suteikia pažangius sąnaudų amortizacijos (Cost Amortization) rodiklius, skirtus kaupimo (accrual-based) finansinės atskaitomybės palaikymui. Naudodama „Cost (Amortized)“ metriką, platforma automatiškai paskirsto vienkartines išlaidas – pavyzdžiui, AWS „Savings Plans“ pradinio mokesčio ar Azure „Reserved Instance“ (RI) rezervavimo mokesčius – per visą įsipareigojimo laikotarpį. Nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
4.7	Pilnų išlaidų integracija	Sprendimas turi palaikyti pilnų išlaidų integraciją, leidžiančią taikyti sudėtingas pasirinktines kainodaros taisykles ir tikslų nuolaidų atspindėjimą iš AWS Savings Plans/Reserved Instances ir kitų įsipareigojimų modelių.	Sprendimas palaiko pilnų išlaidų integraciją, skirtą debesijos sąskaitų duomenims suvienodinti, taikyti individualius kainų koregavimus ir tiksliai paskirstyti įsipareigojimais pagrįstų nuolaidų (pvz., Savings Plans ir Reserved Instances) naudą organizacijoje. Pirma nuoroda Antra nuoroda
4.8	Automatinis išlaidų paskirstymo variklis	Sprendimas turi turėti automatizuojamą išlaidų paskirstymo variklį.	Sprendimas naudoja pažangų taisyklių pagrindu veikiantį variklį, vadinamą „Business Mapping“. Ši funkcija automatizuoja kiekvienos sąskaitos eilutės kategorijas – įskaitant sudėtingus, bendrus arba nepažymėtus resursus – užtikrinant, kad 100 % debesijos išlaidų būtų tiksliai priskirta teisingam verslo kontekstui. Nuoroda
5.	**Reikalavimai išlaidų optimizavimui**		
5.1	Skaičiavimo išteklių dydžio pritaikymas	Sprendimas turi teikti veiksmingas skaičiavimo išteklių (pvz., EC2, VM ar lygiavertes) dydžio pritaikymo rekomendacijas VSSA naudojamiems AWS, Azure ir Google Cloud paslaugoms.	Sprendimas suteikia automatizuotą „Rightsizing“ variklį. Šis modulis nuolat analizuoja jūsų debesijos skaičiavimo išteklių – įskaitant AWS EC2, Azure VM ir Google Cloud GCE – našumą ir naudojimą, teikdamas duomenimis pagrįstas rekomendacijas, kurios subalansuoja sąnaudų taupymą ir programų našumą. Nuoroda
5.2	Konteinerių dydžio pritaikymas	Sprendimas turi teikti rekomendacijas konteinerių dydžio pritaikymui.	Sprendimas suteikia automatizuotas „Rightsizing“ rekomendacijas Kubernetes konteineriams. Ši funkcija nustato konkrečias galimybes sumažinti sąnaudas, suderinant konteinerių resursų užklausas ir limitus su faktiniais istorinių naudojimo duomenimis. Nuoroda
5.3	Nenaudojamų išteklių identifikavimas	Sprendimas turi identifikuoti ir pranešti apie nenaudojamus arba nepakankamai išnaudojamus išteklius.	Sprendimas turi automatinį integruotą variklį, leidžiantį aptikti ir pranešti apie nereikalingas išlaidas debesijos paslaugose AWS, Azure ir Google Cloud. Nuoroda
5.4	Atjungtų saugyklų identifikavimas	Sprendimas turi identifikuoti ir pranešti apie atjungtas saugojimo apimtis (pvz., atjungtas EBS apimtis ir pan.).	Sprendimas identifikuoja ir praneša apie neprijungtus (atidėtus) saugojimo diskus debesijos aplinkose (AWS, Azure ir Google Cloud). Ši funkcija yra pagrindinė platformos išlaidų švaistymo nustatymo ir optimizavimo įrankių rinkinio dalis.

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
			Pirma nuoroda Antra nuoroda
5.5	Vieninga įsipareigojimų planavimo sistema	Sprendimas turi teikti vieningą įsipareigojimų planavimą, atsižvelgdamas tiek į rezervuotas instancijas (RI), tiek į taupymo planus (SP), siekdamas pateikti optimalius pirkimo rekomendacijas.	Sprendimas apima vieningą „Commitment Manager“ ir „Commitment Recommendations“ variklį. Ši sistema suteikia visapusišką įsipareigojimų portfelio vaizdą – tuo pačiu metu analizuodama Reserved Instances (RI) ir Savings Plans (SP) – ir generuoja optimalias pirkimo bei keitimo rekomendacijas, pagrįstas istoriniu naudojimu. Nuoroda
6.	**Reikalavimai duomenų integracijai, išgavimui ir saugojimui**		
6.1	Natūrali debesijos atsiskaitymo integracija	Sprendimas turi integruotis su natūraliais debesijos atsiskaitymo duomenų šaltiniais, įskaitant AWS Cost and Usage Report (CUR), Azure Enterprise Agreement (EA) / Microsoft Customer Agreement (MCA) atsiskaitymo duomenis ir GCP Billing Export į BigQuery arba lygiaverčiais.	Sprendimas sukurtas tiesioginės, API pagrindu veikiančios integracijos su pagrindiniais viešaisiais debesų paslaugų tiekėjais modeliu. Sprendimas natūraliai surenka aukščiausios tikslumo sąskaitų duomenų rinkinius – pvz., AWS CUR, Azure MCA ir GCP integraciją (BigQuery) – užtikrindamas, kad jūsų ataskaitos remtųsi ta pačia „tiesos šaltinio“ informacija. Pirma nuoroda Antra nuoroda Trečia nuoroda
6.2	Artimo realaus laiko duomenų atnaujinimas	Sprendimas turi palaikyti duomenų atnaujinimą kelis kartus per dieną, suteikdamas artimo realaus laiko duomenis.	Sprendimas sukurtas suteikti realaus laiko matomumą, nuolat tikrinant debesijos sąskaitų duomenis. Kadangi debesijos tiekėjai (AWS, Azure ir GCP) atnaujina savo sąskaitų failus visos dienos metu, sprendimas automatiškai surenka, suvienodina ir atspindi šiuos pokyčius valdymo skydeliuose. Nuoroda
6.3	Dvipusis API	Sprendimas turi teikti visapusišką dvipusį API (skaitymo ir rašymo galimybes), leidžiantį automatizuoti ir integruoti su kitomis sistemomis.	Sprendimas sukurtas pagal API-pirmiau architektūrą. Ji suteikia išsamų RESTful API (v3), palaikantį tiek skaitymo, tiek rašymo (CRUD) operacijas, leidžiantį sklandžiai integruotis su IT paslaugų valdymo (ITSM) įrankiais, CI/CD procesais ir vidinėmis finansų sistemomis. Pirma nuoroda Antra nuoroda Trečia nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
6.4	Neribotas duomenų saugojimas	Sprendimas turi siūlyti pilną įsisavintų duomenų saugojimą be galiojimo pabaigos.	Sprendimas siūlo pilną įsisavintų duomenų saugojimą be galiojimo pabaigos per istorinių duomenų eksporto („historical data export“) funkcionalumą. Nuoroda
6.5	CSV/PDF eksportas	Sprendimas turi suteikti galimybę eksportuoti bet kurią ataskaitą ar vaizdą į CSV ir PDF formatus.	Sprendimas palaiko visų sąnaudų ir naudojimo ataskaitų bei valdymo skydelių vizualizacijų eksportą į standartinius CSV ir PDF formatus. Pirma nuoroda Antra nuoroda
7.	**Reikalavimai stebėsenai ir ataskaitoms**		
7.1	Pritaikomi valdymo skydeliai	Sprendimas turi leisti vartotojams kurti ir pritaikyti kelis valdymo skydelius, vizualizuojančius išlaidų ir naudojimo duomenis pagal jų specifines roles ar verslo vienetus.	Sprendimas suteikia itin lanksčią „Custom Dashboard“ sistemą. Ji leidžia vartotojams kurti, išsaugoti ir dalintis neribotu kiekiu vaidmeniui pritaikytų vizualizacijų, suderinančių debesijos išlaidų duomenis su jų unikalia verslo aplinka. Nuoroda
7.2	Anomalijų aptikimas ir perspėjimai	Sprendimas turi teikti pažangų anomalijų aptikimą, identifikuojantį neįprastus naudojimo modelius ar išlaidų šuolius ir siunčiantį automatinius perspėjimus.	Sprendimas naudoja dirbtiniu intelektu pagrįstą anomalijų aptikimo („Anomaly Detection“) variklį. Ši funkcija automatiškai nustato bazinius išlaidų modelius kiekvienam jūsų debesijos aplinkos segmentui ir iškart siunčia įspėjimus, kai aptinkami nukrypimai, pvz., sąnaudų šuoliai ar neįprastas naudojimas. Nuoroda
7.3	Individualių ataskaitų kūrimas	Vartotojai turi turėti galimybę kurti ir saugoti individualias ataskaitas.	Sprendimas suteikia galingą ataskaitų variklį, leidžiantį vartotojams kurti, pritaikyti ir išsaugoti neribotą kiekį individualių ataskaitų. Nuoroda
7.4	Suplanuotos el. pašto ataskaitos	Sprendimas turi palaikyti ataskaitų planavimą ir pristatymą el. paštu pagal pasikartojantį grafiką.	Sprendimas suteikia automatizuotą ataskaitų planavimą ir pristatymą el. paštu pagal pasikartojantį grafiką per savo „Subscription“ funkciją. Nuoroda
7.5	Ataskaitų ir valdymo skydelių dalijimasis	Sprendimas turi suteikti mechanizmą vartotojams dalintis ataskaitomis ir valdymo skydeliais su kitais platformos vartotojais.	Sprendimas suteikia mechanizmą vartotojams dalintis ataskaitomis ir valdymo skydeliais su kitais platformos vartotojais. Nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
8.	**Reikalavimai biudžetavimui ir prognozavimui**		
8.1	Aprėpties biudžeto kūrimas	Sprendimas turi leisti kurti biudžetus įvairioms aprėptims, pvz., komandai, programai ar projektui.	Sprendimas leidžia kurti ir valdyti kelis biudžetus skirtingiems aprėpties lygiams (komandoms, programoms ar projektams) per savo biudžetų architektūrą, pagrįstą „Views“. Nuoroda
8.2	Biudžeto perspėjimai	Sprendimas turi teikti perspėjimų galimybes, kai išlaidos prognozuojamos viršyti arba artėti prie biudžeto limito.	Sprendimas turi pažangų biudžetų ir prognozavimo („Budgets and Forecasting“) variklį. Ši sistema naudoja dirbtiniu intelektu pagrįstus modelius, prognozuojančius būsimą išlaidų lygį pagal istorinius modelius, ir automatiškai siunčia įspėjimus, kai išlaidos tikėtina viršys arba jau viršijo nustatytas ribas. Pirma nuoroda Antra nuoroda Trečia nuoroda
8.3	ML pagrįstas prognozavimas	Sprendimas turi teikti prognozavimo galimybes, remiantis istoriniais naudojimo šablonais ir mašininio mokymosi modeliais.	Sprendimas turi pažangų biudžetų ir prognozavimo („Budgets and Forecasting“) variklį. Ši sistema naudoja dirbtiniu intelektu pagrįstus modelius, prognozuojančius būsimą išlaidų lygį pagal istorinius modelius, ir automatiškai siunčia įspėjimus, kai išlaidos tikėtina viršys arba jau viršijo nustatytas ribas. Pirma nuoroda Antra nuoroda Trečia nuoroda
8.4	"Kas jeigu" scenarijų analizė	Sprendimas turi palaikyti „kas jeigu“ scenarijų analizę, leidžiančią modeliuoti būsimų infrastruktūros pakeitimų finansinį poveikį.	Sprendimas palaiko „what-if“ scenarijų analizę. Tai daugiausia realizuojama per „Commitment Manager“ finansiniam modeliui ir „Rightsizing Explorer“ infrastruktūros optimizavimui, leidžiant simuliuoti įvairių operacinių ir pirkimo sprendimų sąnaudų poveikį prieš jų įgyvendinimą. Pirma nuoroda Antra nuoroda
9.	**Reikalavimai mastelio nustatymui ir našumui**		
9.1	Didelių duomenų rinkinių masteliavimas	Sprendimas turi būti suprojektuotas tvarkyti labai didelius ir sudėtingus kelių debesijų duomenų rinkinius be našumo pablogėjimo.	Sprendimas yra specialiai sukurtas apdoroti itin didelius ir sudėtingus debesijos duomenų rinkinius. Sprendimas naudoja didelio našumo "Cloud Data Ingestion" (CDI) variklį, kuris normalizuoja duomenis į vieningą schemą ir užtikrina pastovų veikimą nepriklausomai nuo duomenų rinkinio dydžio ar sudėtingumo.

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
			Nuoroda
10.	**Reikalavimai mokymams**		
10.1	Išsamus švietimo prenumeratos modelis	Tiekėjas turi pasiūlyti metinę prenumeratą (pvz., All Access Education Pass atba lygiavertę), skirtą sprendimo administratoriams, suteikiant išsamius švietimo išteklius.	Tiekėjas teikia specialią edukacinę prenumeratą, vadinamą „All Access Education Pass“ (AAEP). Ši prenumerata suteikia administratoriams neribotą prieigą prie viso profesinių mokymų katalogo, įskaitant pagrindinius kursus, pažangius administravimo seminarus ir oficialius pramonėje pripažintus sertifikavimo kelius sprendimo platformai. Nuoroda
10.2	Platus kursų katalogas	Prenumerata turi suteikti neribotą prieigą prie daugiau nei 200 valandų švietimo turinio, įskaitant savarankiškus ir instruktorių vedamus kursus.	Prenumerata suteikia neribotą prieigą prie daugiau nei 200 valandų švietimo turinio. Nuoroda
10.3	Vadovaujami praktiniai užsiėmimai	Mokymo programa turi apimti galimybes vadovaujamai praktikai, siekiant užtikrinti praktinių įgūdžių su sprendimu vystymą.	Mokymo programa apima galimybes vadovaujamai praktikai, siekiant užtikrinti praktinių įgūdžių su sprendimu vystymą. Nuoroda
10.4	Vardinio naudotojo licencijos modelis	Mokymo prenumerata turi būti suteikiama vardinio naudotojo pagrindu, kai prieiga priskiriama konkrečiam asmeniui. Prisijungimo duomenų dalijimasis turi būti draudžiamas.	Mokymo prenumerata suteikiama vardinio naudotojo pagrindu, kai prieiga priskiriama konkrečiam asmeniui. Prisijungimo duomenų dalijimasis turi būti draudžiamas. Nuoroda
10.5	Išsami mokymo programa	Programa turi apimti platų temų spektrą nuo pagrindinių iki pažangių funkcijų, siekiant užtikrinti nuodugnų produkto supratimą ir palaikyti nuolatinį mokymąsi.	Programa apima platų temų spektrą nuo pagrindinių iki pažangių funkcijų, siekiant užtikrinti nuodugnų produkto supratimą ir palaikyti nuolatinį mokymąsi. Nuoroda
10.6	Ekspertų konsultacijų prieiga	Prenumerata turi suteikti galimybę naudotojams kreiptis į sprendimo ekspertus dėl gairių ir palaikymo mokymosi procese.	Prenumerata suteikia galimybę naudotojams kreiptis į sprendimo ekspertus dėl gairių ir palaikymo mokymosi procese. Nuoroda

8. SPECIALIEJI REIKALAVIMAI PASLAPČIŲ SAUGOJIMO SPRENDIMUI

8.1. Paslapčių saugojimo sprendimui keliami specialieji programinės įrangos funkcionalumo reikalavimai:

8.1.1. Paslapčių saugojimo sprendimas turi užtikrinti centralizuoto pasitikėjimo šaltinio (angl. *Platform's Centralized Root of Trust*) funkcijas: saugoti ir teikti pačius svarbiausius, aukščiausių privilegijų kredencialus.

8.1.2. Paslapčių saugojimo sprendimas turi atlikti dinaminių kredencialų generatoriaus debesijos prieigai (angl. *Dynamic Credential Generator for Cloud Access*) funkcijas.

8.1.3. Paslapčių saugojimo sprendimas turi atlikti saugaus autentifikatoriaus automatizuotoms darbo eigoms (angl. *Secure Authenticator for Automated Workflows*) funkcijas: teikti saugius, mašina-mašinai (angl. *machine-to-machine*) autentifikacijos metodus.

8.1.4. Paslapčių saugojimo sprendimas turi užtikrinti nuo Debesijos platformos nepriklausomą paslapčių API: suteikti vieningą ir nuoseklų API sąsają paslaptims pasiekti.

8.1.5. Paslapčių saugojimo sprendimas turi atlikti centralizuoto ir nekintamo audito žurnalo funkcijas: registruoti kiekvieną autentifikacijos bandymą ir kiekvieną kreipinį į paslaptį vienoje, centralizuotoje vietoje.

8.1.6. Paslapčių saugojimo sprendimas turi atlikti šifravimo kaip paslaugos variklio (angl. *Encryption-as-a-Service Engine*) funkcijas: teikti kriptografines paslaugas, leidžiančias automatizavimo procesams užšifruoti jautrius duomenis, neatskleidžiant naudojamų šifravimo raktų.

8.1.7. Paslapčių saugojimo sprendimas turi užtikrinti centralizuotų politikų vykdymo taško funkcijas: įgyvendinti smulkiagrūdę prieigos kontrolę per deklaratyvias politikas, apibrėžiančias, kuris CI/CD procesas, vartotojas ar sistema gali pasiekti konkrečias paslaptis.

8.2. Žemiau pateikiami detalūs specialieji reikalavimai paslapčių saugojimo programinei įrangai ir jos funkcionalumui (žr. 5 lentelę).

5 lentelė. Reikalavimai paslapčių saugojimo sprendimo programinei įrangai

Eil. Nr.	**Charakteristikos pavadinimas**	**Reikalaujama charakteristika**	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
1.	**Bendrieji reikalavimai**		
1.1	Siūlomos programinės įrangos pavadinimas	***International Business Machines Corporation (IBM), IBM Vault Self-Managed Premium***	
1.2	Licencijų kiekis	Tiekėjas turi pateikti reikiamą kiekį licencijų, kurios leistų siūloma programine įranga naudotis ne mažiau kaip 50 (penkiasdešimt) vnt. klientų.	Siūloma programinė įranga turi galimybe naudotis ne mažiau kaip 50 (penkiasdešimt) vnt. klientų.
1.3	Diegimo modelis (savipalaikoma)	Sprendimas turi būti savipalaikomas arba diegiamas VSSA pateiktuose resursuose.	Sprendimas yra savipalaikomas arba diegiamas VSSA pateiktuose resursuose. Nuoroda
1.4	Gamintojo palaikymas	Turi turėti techninį palaikymą iš gamintojo, visą prenumerata pagrįstos licencijos galiojimo laikotarpį.	Sprendimas užtikrina techninį palaikymą, teikdamas gamintojo remiamą techninę pagalbą visam prenumeratos laikotarpiui. Pagalbos paslaugos nėra papildomas priedas, jos natūraliai įtrauktos į licenciją, užtikrinant ekspertų konsultacijas, saugumo pataisas ir versijų atnaujinimus viso sutarties laikotarpio metu. Nuoroda
1.5	Centralizuota saugykla	Platforma turi veikti kaip centralizuota, API valdoma saugykla statinėms paslaptims (API raktams, slaptažodžiams, sertifikatams) saugoti, naudojant raktas – reikšmė (angl. *Key-Value*) paslapčių variklį, užtikrinant jų šifravimą saugojimo vietoje (angl. *encryption at rest*).	Sprendimas naudoja raktų ir reikšmių („Key-Value“, KV) slaptųjų duomenų variklį („Secrets Engine“). Šis variklis yra pagrindinis mechanizmas saugoti įvairius jautrius duomenis – pavyzdžiui, API raktus, slaptažodžius ar sertifikatus – užtikrinant, kad jie būtų apsaugoti tvirtu

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
			šifravimu saugykloje ir valdomi per išsamų REST API. Pirma nuoroda Antra nuoroda
1.6	Dinamiškų paslapčių generavimas	Turi palaikyti dinamiškų, trumpalaikių kredencialų generavimą įvairioms sistemoms, tokioms kaip debesijos tiekėjai (AWS, Azure, GCP arba lygiavertės) ir duomenų bazės (pvz., PostgreSQL, MySQL arba lygiavertės).	Sprendimas turi dinaminių slaptųjų duomenų variklį („Dynamic Secrets Engine“). Šis variklis generuoja unikalius, trumpalaikius prisijungimo duomenis pagal poreikį debesijos platformoms ir duomenų bazėms, automatiškai juos atšaukiant, kai užduotis yra baigta. Pirma nuoroda Antra nuoroda Trečia nuoroda
1.7	Kriptografijos paslauga	Sprendimas turi veikti kaip kriptografijos paslauga (angl. *Encryption as a Service*), leidžianti aplikacijoms šifruoti ir dešifruoti duomenis, nelaikant šifravimo raktų pačiame aplikacijos kode.	Sprendimas suteikia specialų „Transit Secrets Engine“. Šis variklis veikia kaip „Encryption as a Service“ (EaaS) platforma, leidžianti programoms atlikti aukšto našumo kriptografines operacijas, neprivalant pačioms tvarkyti, saugoti ar net matyti pagrindinių šifravimo raktų. Nuoroda
1.8	Duomenų šifravimas	Visos saugomos paslaptys turi būti šifruojamos saugojimo vietoje. Turi būti palaikomas ryšio šifravimas tarp kliento ir serverio.	Sprendimas užtikrina visapusišką jautrių duomenų apsaugą, privalomai taikant šifravimą tiek saugykloje („at rest“), tiek duomenų perdavimo metu („in transit“). Platforma remiasi nulinio pasitikėjimo („zero-trust“) tinklo principu ir užtikrina, kad duomenys niekada nebūtų prieinami atviru tekstu už Vault proceso ribų. Nuoroda
1.9	Centralizuotas autentifikavimas	Turi palaikyti įvairius autentifikavimo metodus (angl. *Auth Methods*), skirtus tiek vartotojams, tiek mašinoms. Turi būti palaikoma integracija su LDAP, SAML, OIDC/JWT ir specializuotais metodais, tokiais kaip AppRole.	Sprendimas palaiko išsamų autentifikavimo („Auth Methods“) metodų spektrą, skirtą tiek vartotojams, tiek mašinų tapatybėms. Sprendimas integruojasi su standartų protokolais – įskaitant LDAP, SAML ir OIDC/JWT – bei specializuotais mašinų tarpusavio autentifikavimo metodais, tokiais kaip AppRole. Pirma nuoroda Antra nuoroda Trečia nuoroda Ketvirta nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
1.10	Detali autorizacija (Policy as Code)	Prieiga prie paslapčių turi būti valdoma naudojant deklaratyvias politikas (angl. *Policy as Code*), parašytas HCL formatu arba lygiaverčiu. Politikos turi leisti smulkmeniškai apibrėžti leidimus (pvz., kurti, skaityti, atnaujinti, trinti) konkretiems keliams (angl. *paths*).	Sprendimas naudoja „Policy-as-Code“ sistemą. Prieiga valdomai per žmogui suprantamas, versijų valdomas politikas, rašomas HCL kalba, kurios tiksliai nustato leidimus kiekvienam unikaliam sistemos keliui (angl. *paths*). Nuoroda
1.11	Paslapčių versijavimas	Kiekviena paslaptis, saugoma raktas – reikšmė saugykloje, turi būti versijuojama, suteikiant galimybę atkurti senesnes versijas arba atlikti pakeitimų auditą.	Sprendimo KV-V2 slaptųjų duomenų variklis („Secrets Engine“) suteikia natyvią versijavimo ir metaduomenų sekimo funkciją visiems saugomiems statiniams slaptiesiems duomenims. Ši sistema užtikrina duomenų patikimumą ir atsiskaitymo galimybę, išlaikydama konfigūruojamą kiekvieno pakeitimo istoriją, leidžiančią lengvai atkurti netyčia perrašytus duomenis ir užtikrinančią pilną slaptųjų duomenų gyvavimo ciklo matomumą. Nuoroda
1.12	Audito žurnalai	Visi bandymai autentifikuotis ir visos operacijos su paslaptimis (skaitymas, rašymas) turi būti detaliai registruojamos audito žurnaluose, kurie gali būti siunčiami į Saugumo informacijos ir įvykių valdymas (angl. *Security Information and Event Management, SIEM*) sistemas.	Sprendimas turi audito įrenginių („Audit Device“) sistemą. Ši sistema užtikrina visų sąveikų įrašų fiksavimą ir yra specialiai sukurta siųsti šiuos įvykius į SIEM platformas, tokias kaip Splunk, QRadar, Datadog ar Google SecOps. Nuoroda
1.13	Grafinė sąsaja, CLI ir API	Sprendimas turi suteikti kelis sąveikos būdus: grafinę vartotojo sąsają, komandinės eilutės įrankį ir išsamią REST API programinei integracijai.	Sprendimas suteikia tris būdus sąveikai su platforma: grafinę sąsają („Web UI“), komandų eilutės sąsają („CLI“) ir išsamų REST API. Visi šie sąsajų tipai bendrauja su vieningu, centralizuotu *backend'u*. Pirma nuoroda Antra nuoroda Trečia nuoroda
2.	**Reikalavimai dinaminėms paslaptims**		
2.1	"Just-in-Time" kredencialai	Platforma turi gebėti generuoti kredencialus „pagal pareikalavimą“ (angl. *just-in-time*) su apibrėžtu gyvavimo laiku (angl. T*ime to Live, TTL*). Pasibaigus TTL, kredencialai automatiškai atšaukiami.	Sprendimas palaiko „pagal poreikį“ („on-demand“) prisijungimo duomenų generavimą per Dinaminių slaptųjų duomenų („Dynamic Secrets“) architektūrą. Ši funkcija programiškai generuoja unikalius, laikui apribotus prisijungimo duomenis, kuriuos Vault automatiškai atšaukia pasibaigus jų galiojimo laikui (TTL). Pirma nuoroda Antra nuoroda Trečia nuoroda

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
2.2	Debesijos platformų palaikymas	Turi būti palaikomi dinaminiai paslapčių varikliai pagrindinėms Debesijos platformoms (pvz., AWS, Azure, GCP arba lygiavertėms), leidžiantys generuoti trumpalaikius IAM vartotojus ar servisų paskyras.	Sprendimas turi specializuotus slaptųjų duomenų variklius („Secrets Engines“) visiems pagrindiniams debesijos tiekėjams (AWS, Azure ir GCP). Šie varikliai yra specialiai sukurti generuoti „just-in-time“ trumpalaikius IAM vartotojus, paslaugų principalus arba prieigos žetonus, užtikrinant, kad infrastruktūros prieiga visada būtų laikina ir griežtai ribota. Pirma nuoroda Antra nuoroda Trečia nuoroda
2.3	Duomenų bazių palaikymas	Turi palaikyti dinaminių kredencialų generavimą populiarioms duomenų bazių sistemoms (pvz., PostgreSQL, MySQL, MS SQL arba lygiavertėms), suteikiant aplikacijoms unikalius prisijungimus.	Sprendimas naudoja prijungiamą („pluggable“) duomenų bazių slaptųjų duomenų variklį („Database Secrets Engine“). Šis variklis veikia kaip patikimas tarpininkas, sąveikaujantis su jūsų duomenų bazėmis – įskaitant PostgreSQL, MySQL, MariaDB, MS SQL Server (MSSQL), Oracle, MongoDB, Snowflake ir Cassandra – ir generuojantis unikalius, trumpalaikius vartotojus kiekvienam programos užklausimui, užtikrinant, kad prisijungimo duomenys būtų automatiškai pašalinami po naudojimo. Nuoroda
2.4	Automatinis atšaukimas	Sprendimas turi automatiškai valdyti dinaminių paslapčių gyvavimo ciklą ir užtikrinti, kad pasibaigus jų galiojimo laikui, prieiga būtų nedelsiant panaikinta.	Sprendimas turi vidinį galiojimo valdymo modulį („Expiration Manager“). Ši komponentė veikia kaip centrinė sistema, sekanti kiekvieną dinaminių paslaptį ir užtikrinanti, kad jų gyvavimo ciklas – nuo sukūrimo iki privalomo atšaukimo – būtų valdomas automatiškai, be žmogaus įsikišimo. Nuoroda
3.	**Reikalavimai duomenų šifravimui (Encryption-as-a-Service)**		
3.1	Tranzitinio šifravimo variklis	Sprendimas turi teikti šifravimas kaip paslauga (angl. *Encryption-as-a-Service*) funkciją per tranzitinį variklį, leidžiantį aplikacijoms šifruoti ir dešifruoti duomenis, neatskleidžiant šifravimo raktų.	Sprendimas turi specialų „Transit Secrets Engine“. Šis variklis veikia kaip paslauga („Encryption-as-a-Service“) platforma, leidžianti programoms šifruoti ir dešifruoti jautrius duomenis per API, niekada netvarkant ar neatskleidžiant pagrindinių šifravimo raktų. Nuoroda
3.2	Centralizuotas raktų valdymas	Platforma turi centralizuotai valdyti šifravimo raktus, palaikyti jų	Sprendimas turi „Transit Secrets Engine“. Šis variklis veikia kaip specializuota raktų

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
		versijavimą ir rotaciją be aplikacijų kodo pakeitimų.	valdymo paslauga (Key Management Service, KMS), centralizuojanti visą šifravimo raktų gyvavimo ciklą – įskaitant generavimą, versijavimą ir keitimą. Tai leidžia programoms atlikti kriptografines operacijas per API, niekada nematant pačių šifravimo raktų. Nuoroda
3.3	Įvairių tipų raktai	Turi palaikyti įvairius šifravimo algoritmus ir raktų tipus (pvz., AES-GCM, RSA, ECDSA ir pan.), skirtus tiek simetriniam, tiek asimetriniam šifravimui bei skaitmeniniams parašams.	Sprendimas palaiko platų standartų algoritmų spektrą (įskaitant AES-GCM, RSA, ECDSA). Jis veikia kaip vieninga kriptografinė sąsaja simetriniam ir asimetriniam šifravimui, taip pat skaitmeniniam pasirašymui ir patikros operacijoms. Nuoroda
4.	**Reikalavimai tapatybės valdymui ir autentifikavimui**		
4.1	Įvairūs autentifikavimo metodai	Platforma turi palaikyti platų aplikacijų ir vartotojų autentifikavimo metodų spektrą, įskaitant AppRole, JWT/OIDC, LDAP, ir debesijos IAM.	Sprendimas naudoja autentifikavimo architektūrą, integruojamą su esamais patikimais tapatybės teikėjais. Tai leidžia organizacijoms centralizuoti prieigos valdymą be naujų atskirų prisijungimo duomenų kūrimo, naudojant pažįstamus protokolus, tokius kaip LDAP, OIDC ir *platform-native* debesijos IAM sprendimai. Pirma nuoroda Antra nuoroda Trečia nuoroda Ketvirta nuoroda Penkta nuoroda
4.2	Tapatybės valdymas	Turi būti palaikomas tapatybės (angl. *Identity*) modulis, leidžiantis susieti autentifikuotus subjektus iš skirtingų sistemų su viena vidine "esybės" (angl. *Entity*) sąvoka ir taikyti bendras politikas.	Sprendimas turi „Identity Secrets Engine“. Šis modulis veikia kaip centrinis „tiesos šaltinis“ Vault sistemoje, leidžiantis sujungti įvairius autentifikavimo aliasus (pvz., vartotojo LDAP paskyrą ir jų GitHub ID) į vieną kanoninę „Entity“, užtikrinant, kad bendros politikos būtų taikomos nepriklausomai nuo to, kaip vartotojas ar mašina autentifikuojasi. Nuoroda
4.3	Integracija su SSO	Vartotojo sąsajos prieigai turi būti palaikoma integracija su įmonės SSO tiekėjais per OIDC arba SAML protokolus.	Sprendimas suteikia *enterprise* lygio SSO integraciją savo grafinei naudotojo sąsajai („Web UI“). Naudojant OIDC arba SAML autentifikavimo metodus, organizacijos gali įgyvendinti „Prisijungti per SSO“ patirtį, deleguodamos

Eil. Nr.	Charakteristikos pavadinimas	Reikalaujama charakteristika	**Siūloma charakteristika** *(Nurodyti tikslų siūlomos charakteristikos parametrą ir pateikti jį pagrindžiančią informaciją (pvz., gamintojo dokumentacija, internetinės nuorodos, aprašai ar kiti įrodymai))* ***(Pildo tiekėjas)***
			autentifikavimą patikimam tapatybės teikėjui (IdP). Pirma nuoroda Antra nuoroda
5.	**Reikalavimai valdymui, atitikčiai ir operacijoms**		
5.1	Vardų erdvės (Namespaces)	Turi palaikyti vardų erdves, kurios leidžia sukurti izoliuotas, savarankiškas aplinkas viename klasteryje, siekiant atskirti komandų ar projektų paslaptis.	Sprendimas turi vardų erdves („Namespaces“) funkciją. Ji leidžia vieną „Vault“ klasterį suskaidyti į kelis „virtualius Vault“, kurių kiekvienas turi atskiras administravimo ribas, atskirus „Secrets Engine“ ir autentifikavimo metodus. Nuoroda
5.2	Politikos kaip kodas (Sentinel/OPA)	Sprendimas turi integruotą politikos kaip kodo karkasą (pvz., Sentinel, OPA arba lygiavertį), leidžiantį apibrėžti smulkias, sąlygines taisykles paslapčių prieigai ir sistemos konfigūracijai.	Sprendimas turi integraciją su „Sentinel“ – „policy-as-code“ varikliu. „Sentinel“ leidžia peržengti paprastą „leisti / drausti“ prieigos modelį ir taikyti sudėtingas, logika bei sąlygomis pagrįstas taisykles kiekvienai sąveikai su „Vault“. Nuoroda
5.3	Kontrolės grupės (Control Groups)	Turi palaikyti kelių asmenų patvirtinimo (angl. *multi-party authorization*) mechanizmą ypač jautrioms operacijoms, reikalaujant kelių administratorių sutikimo.	Sprendimas įgyvendina kelių šalių autorizaciją naudodamas „Control Groups“. Ši funkcija leidžia organizacijoms apibrėžti aukšto saugumo darbo eigas, kuriose konkretūs API užklausimai yra sustabdomi tol, kol reikiamas patvirtintojų kvorumas (pvz., 2 iš 5 saugumo vadovų) suteikia savo skaitmeninį patvirtinimą. Nuoroda
5.4	Saugi integracija su CI/CD įrankiais	CI/CD procesai turi gebėti saugiai, be slaptažodžių, autentifikuotis paslapčių saugojimo sistemoje naudojant OIDC/JWT standartus.	Sprendimas leidžia CI/CD procesams naudoti autentifikavimą be slaptažodžių, pasitelkiant OIDC/JWT standartus. Nuoroda
5.5	Programinis paslapčių gavimas	Sprendimas turi suteikti CLI įrankį ir išsamią REST API, leidžiančią scenarijams ir aplikacijoms automatizuotai gauti, atnaujinti ir atšaukti paslaptis.	Sprendimas suteikia išsamų REST API ir daugiaplatformį CLI įrankį, leidžiančius įgyvendinti „zero-touch“ paslapčių valdymą. Kadangi „Vault“ kiekvieną operaciją traktuoja kaip API iškvietimą, kūrėjai ir operatoriai gali lengvai automatizuoti paslapčių gavimą, atnaujinimą ir atšaukimą naudodami standartinius įrankius, tokius kaip „curl“, „Python requests“ ar „Vault“ komandų eilutės įrankį. Pirma nuoroda Antra nuoroda

9. PASLAUGŲ UŽSAKYMŲ TEIKIMO TVARKA

9.1. **Užsakymo pateikimas:** Paslaugos teikiamos pagal Perkančiosios organizacijos Tiekėjui pateiktus paslaugų užsakymus (toliau – Užsakymas). Užsakyme nurodoma:

9.1.1. užsakomos paslaugos;

9.1.2. kuriamas funkcionalumas ir /ar jo elementas (-ai);

9.1.3. siektinas rezultatas;

9.1.4. numatomi paslaugų suteikimo terminai;

9.1.5. preliminarus Tiekėjo specialistų ir jų darbo valandų skaičius;

9.1.6. Užsakymo kaina;

9.1.7. Kita svarbi informacija.

9.2. **Procedūra:** Konkreti paslaugų užsakymo teikimo ir vykdymo procedūra turi būti aprašyta Reglamente.

9.3. **Užsakymo patvirtinimas:** Užsakymas, pateiktas raštu Sutarties Specialiųjų sąlygų priede Nr. 3 „Atsakingi asmenys“ Tiekėjo nurodyto įgalioto asmens elektroninio pašto adresu, laikomas gautu jo išsiuntimo dieną. Tiekėjas ne vėliau kaip per 5 (penkias) darbo dienas patvirtina Užsakymą ir pateikia: Užsakymo įgyvendinimo aprašymą, įvykdymo terminą, specialistus, vykdysiančius Užsakymą, jų darbo valandų skaičių bei viso Užsakymo įgyvendinimo kainą ir kitą svarbią informaciją. Tiekėjas, prieš patvirtindamas Užsakymą, įvertina numatytų teikti paslaugų apimtį, techninius, funkcinius, saugumo ir kokybės reikalavimus, paruošia pasiūlymą ir pateikia jį Perkančiajai organizacijai, kuriame nurodo:

9.3.1. Užsakymo įgyvendinimo detalų aprašymą;

9.3.2. sąlygas, prielaidas, žmogiškuosius ir finansinius išteklius, būtinus paslaugų teikimui;

9.3.3. Paslaugų suteikimo trukmę darbo valandomis ir galutinį Užsakymo įvykdymo terminą.

9.4. **Paslaugų teikimo pradžia ir vykdymas:** Užsakyme nurodytų paslaugų teikimo pradžia laikoma kita darbo diena po galutinio abiejų Šalių Užsakymo patvirtinimo.

9.5. **Užsakymo įgyvendinimo sprendimai:** Perkančioji organizacija, gavusi iš Tiekėjo Užsakymą ir jame numatytų paslaugų aprašymą bei apimčių įvertinimą, priima sprendimą dėl Užsakymo įgyvendinimo:

9.5.1. jei Perkančioji organizacija nustato, kad paslaugos, nurodytos gautame Užsakyme, yra nereikalingos dėl netinkamo kaštų ir naudos santykio – Užsakymas atšaukiamas, apie tai informuojant Tiekėją;

9.5.2. jei Užsakyme numatytų suteikti paslaugų aprašymas yra neaiškus, Perkančioji organizacija gali paprašyti Tiekėjo detalizuoti Užsakyme aprašytas paslaugas bei jų teikimo laiko sąnaudų įvertinimą;

9.5.3. jeigu Perkančioji organizacija nustato, kad Užsakyme nurodytos paslaugos yra reikalingos, paslaugų detali analizė, suteikimo terminai, apimtys ir paslaugų teikimo biudžetas yra raštu patvirtinami Užsakyme. Užsakymas laikomas suderintu, kai jį pasirašo abi Šalys;

9.5.4. Perkančioji organizacija ne vėliau kaip per 5 (penkias) darbo dienas turi patvirtinti Tiekėjo Užsakymą, o Tiekėjas per protingą laiką inicijuoja darbų vykdymą.

9.6. **Dokumentacija:** Visa paslaugų teikimo dokumentacija turi būti pateikta elektroninėmis priemonėmis (pvz., el. paštu ar kitomis su Tiekėju ir Perkančiąja organizacija suderintomis priemonėmis) lietuvių ir/ar anglų kalba bei patalpinta su Perkančiąja organizacija suderintoje aplinkoje (pvz., SharePoint).

10. PASLAUGŲ TEIKIMO, PERDAVIMO IR PRIĖMIMO TVARKA

10.1. Techninėje specifikacijoje pateikiami detalūs reikalavimai Debesijos valdymo platformos sukūrimui ir įdiegimui, taip pat Tiekėjui ir jo teikiamoms paslaugoms.

10.2. Tiekėjo paslaugoms teikti yra būtini Debesijos valdymo platformos komponentai.

10.3. Debesijos valdymo platformos sukūrimui ir įdiegimui reikalingi komponentai turi būti perduoti ne vėliau kaip per 10 (dešimt) kalendorinių dienų nuo Sutarties įsigaliojimo dienos. Į komponentų perdavimo terminą įskaičiuojamas ir Prekių perdavimo – priėmimo akto pasirašymas.

10.4. Perkančioji organizacija priima Debesijos valdymo platformos sukūrimui ir įdiegimui reikalingus komponentus (Techninės specifikacijos 2.4.1. papunktis), kai Tiekėjas pateikia Prekių perdavimo–priėmimo aktą.

10.5. Tiekėjo Užsakyme nurodytos ir teikiamos paslaugos (Techninės specifikacijos 2.3 papunktis) priimamos Tiekėjui pateikiant Paslaugų perdavimo–priėmimo aktą, kuriame nurodomi paslaugų teikimui paskirti specialistai, kiekvieno iš šių specialistų faktiškai dirbtų valandų skaičius ir kita reikalinga informacija, kuri buvo nurodyta Užsakyme.

10.6. Perkančioji organizacija Tiekėjo paslaugas, skirtas Debesijos valdymo sprendimo sukūrimui ir įdiegimui, užsakys pateikdama atskirus Užsakymus. Konkrečiame Užsakyme nurodoma užsakomų paslaugų apimtis.

10.7. Perkančioji organizacija Paslaugų užsakymą suformuoja teikiamoms paslaugoms pagal 9 skyriuje numatytą tvarką. Paslaugų perdavimo – priėmimo aktą Perkančioji organizacija pasirašo, kai:

10.7.1. yra sėkmingai atliktas sukurtų Debesijos valdymo platformos funkcionalumų priėmimo testavimas gamybinėje ir/arba testinėje aplinkose;

10.7.2. gamybinėje ir/arba testinėje aplinkose galima įsitikinti, kad Užsakyme sulygti atitinkami atskiri darbai, kuriant Debesijos valdymo platformos funkcionalumą pagal Užsakymą ir pan., atlikti numatyta apimtimi, nėra likusių Perkančiosios organizacijos nurodytų ir neištaisytų klaidų, o naujai į gamybinę aplinką įkeltas funkcionalumas, jeigu tai susijęs su Debesijos valdymo platformos kūrimu bei įdiegimu, veikia kokybiškai.

10.8. Jei suteiktų paslaugų perdavimo-priėmimo metu Perkančioji organizacija negali pilnai patikrinti suteiktų paslaugų atitikimo Sutartyje ir Užsakyme nustatytiems reikalavimams, tai paslaugų perdavimo-priėmimo akto pasirašymas jokiu būdu neapriboja Perkančiosios organizacijos teisės po Paslaugų perdavimo-priėmimo akto pasirašymo reikšti Tiekėjui pretenzijas dėl paslaugų neatitikimo Sutartyje ir Užsakyme nustatytiems reikalavimams ir/ar trūkumams pašalinti.

11. REIKALAVIMAI TIEKĖJO ATASKAITOMS

11.1. Tiekėjas Sutarties vykdymo laikotarpiu turės parengti ir Perkančiajai organizacijai pateikti šių paslaugų teikimo ataskaitas, kaip paslaugų teikimo rezultatus:

11.1.1. testavimo scenarijų ataskaita (-os);

11.1.2. pateikiamas naudotinų komponentų techninės architektūros aprašymas;

11.1.3. parengiamas visą sprendimą apimančios techninės ir technologinės architektūros aprašas;

11.1.4. mokymų medžiaga;

11.1.5. kiti su Paslaugų teikimu ir Sutarties vykdymu susiję dokumentai.

11.2. Paslaugų teikimo ataskaitos Perkančiajai organizacijai raštu (pateikiant el. paštu) privalo būti pateiktos ne vėliau kaip per 2 (dvi) darbo dienas nuo Paslaugos suteikimo laikotarpio pabaigos. Apie Tiekėjo ataskaitos pateikimą konkrečiu darbų vykdymo metu kaip paslaugų teikimo rezultatą, turi būti numatyta Užsakyme.

12. REIKALAVIMAI BANDOMAJAI EKSPLOATACIJAI

12.1. Bandomosios eksploatacijos tikslas – užtikrinti Debesijos valdymo platformos funkcionalumų kokybę testuojant, identifikuoti ir pašalinti bandomosios eksploatacijos metu pastebėtus defektus, siekiant perkelti funkcionalumą į realios aplinkos konfigūraciją.

12.2. Tiekėjas turi parengti bandomosios eksploatacijos planą, kurio tikslas – aprašyti bandomosios eksploatacijos eigą bei apibrėžti bandomosios eksploatacijos dalyvių atsakomybes.

12.3. Tiekėjas testavimo aplinkoje turi atlikti suteiktų paslaugų rezultatų testavimą ir pateikti testavimo rezultatų ataskaitą. Testavimai turės būti atliekami, derinami atskirai kiekvieno Užsakymo metu. Funkcionalumo diegimai turi būti atliekami testavimo aplinkoje, siekiant patikrinti sukurtą funkcionalumą ir suderinamumą su VIPVIS. Testavimo metu pastebėtos klaidos turi būti registruojamos ir jų šalinimas valdomas Tiekėjo bandomajai eksploatacijai paruoštoje aplinkoje. Bandomosios eksploatacijos aplinka turi būti suteikta Tiekėjo.

12.4. Laikoma, kad išlaidos bandomajai eksploatacijos aplinkai sukurti, užtikrinti palaikymą Sutarties vykdymo metu yra įskaičiuotos į bendrą sutarties kainą.

12.5. Bandomosios eksploatacijos planas turi būti parengtas likus be mažiau kaip 5 (penkioms) darbo dienoms iki bandomosios eksploatacijos pradžios.

12.6. Bandomosios eksploatacijos trukmė – 30 (trisdešimt) kalendorinių dienų, skirta ištestuoti Debesijos valdymo platformą, jos komponentus ir pagal sukurtą funkcionalumą įgyvendintų sprendimų atitiktį Perkančiosios organizacijos Užsakymams.

12.7. Bandomosios eksploatacijos plane minimaliai turi būti aprašyta:

12.7.1. bandomosios eksploatacijos dalyvių atsakomybės ir komunikavimo priemonės;

12.7.2. testavimo metu sutrikimų registravimo ir jų šalinimo tvarka;

12.7.3. bandomosios eksploatacijos priėmimo kriterijai (pvz., funkciniai, nefunkciniai, saugumo, funkcionalumo stabilumo ir pan.).

12.8. Bandomosios eksploatacijos metu elektronine forma turi būti vedamas pastebėtų klaidų (problemų) ir jų būsenų kaupimo registras.

12.9. Tiekėjas privalo pašalinti bandomosios eksploatacijos metu pastebėtus trūkumus, užregistruotus klaidų (problemų) ir jų būsenų registre.

12.10. Perkančioji organizacija pradės priėmimo veiklas tik tada, kai testuojamas kuriamos Debesijos valdymo sistemos funkcionalumas atitiks bandomosios eksploatacijos plane apibrėžtus priėmimo kriterijus.

12.11. Tiekėjas privalo, prieš perduodamas bandomosios eksploatacijos metu gautus rezultatus, Perkančiajai organizacijai pateikti dokumentaciją (testavimo scenarijų ataskaitą) ir išeities kodo versijas, jeigu jos buvo pakeistos nuo paskutinio pateikimo. Perkančiajai organizacijai turi būti sudarytos galimybės atlikti diegimo testą, siekiant įsitikinti pateiktų išeities kodų tinkamumu tolimesniam naudojimui.

13. REIKALAVIMAI GARANTINIAM APTARNAVIMUI

13.1. Viešosios debesijos paslaugų valdymo platformos garantinis aptarnavimas – tai Tiekėjo teikiamos sukurtos ir įdiegtos Debesijos valdymo platformos valdymo klausimais naudotojams (VSSA darbuotojams) konsultavimo paslaugos, kurios įskaičiuotos į numatytų įsigyti Debesijos valdymo platformos atitinkamų komponentų kainą.

13.2. Debesijos valdymo platformos garantija teikiama 36 (trisdešimt šešis) mėnesius nuo Debesijos valdymo platformos perdavimo–priėmimo akto pasirašymo dienos.

13.3. Garantinės priežiūros paslaugos apima:

13.3.1. Sprendimo neatitikimų funkciniams reikalavimams ir veikimo klaidų ir/ar trikdžių šalinimą;

13.3.2. garantinės priežiūros metu nustatytų klaidų taisymą;

13.3.3. eksploatuojamo Sprendimo darbingumo atstatymą, pavyzdžiui, įvykus atskirų jos komponentų darbo sutrikimams, kai tai įvyksta ne dėl Perkančiosios organizacijos kaltės;

13.3.4. duomenų atkūrimą, kai gedimo priežastis yra Debesijos valdymo platformos komponento (-ų) programinės įrangos netinkamas veikimas;

13.3.5. Sprendimo programinės įrangos naujumo garantiją, pateikiant gamintojo išleidžiamas naujas programinės įrangos versijas ir jų pataisymus.

13.4. Garantinio aptarnavimo laikotarpiu visos Sprendimo veikimo klaidos ir/ar trikdžiai klasifikuojami:

13.4.1. kritinės klaidos – tokios klaidos, dėl kurių negali funkcionuoti Debesijos valdymo platforma ar keliama itin didelė žala viešosios debesijos paslaugų platformai ir jos teikiamoms paslaugoms.

13.4.2. svarbios klaidos – tai tokios klaidos, kurios leidžia nekorektiškai funkcionuoti Debesijos valdymo platformai, bet nekelia didelės žalos viešosios debesijos paslaugų platformai ir jos teikiamoms paslaugoms;

13.4.3. nesvarbios klaidos – tai visos kitos užfiksuotos klaidos. Konkretūs reikalavimai detalizuojami atskirame susitarime (SLA), kuris parengiamas ir suderinamas su Perkančiąja organizacija iki garantinio periodo pradžios.

13.5. Klaida taip pat laikoma situacija, kai Tiekėjo suteiktos paslaugos garantinio aptarnavimo laikotarpiu neatitinka paslaugų perdavimo–priėmimo akte nurodytų rezultatų, Techninės specifikacijos reikalavimų ar testavimo scenarijų, kurie buvo suderinti tarp Šalių.

13.6. Tiekėjas privalo identifikuoti ir ištaisyti Sprendimo veikimo klaidas ir/ar trikdžius bei pateikti Perkančiajai organizacijai jų šalinimo įgyvendinimo būdo aprašymą pagal šiuos grafikus:

13.6.1. kritinės klaidos atveju – ne vėliau kaip per 8 (aštuonias) valandas nuo kritinės klaidos užregistravimo (pvz., Tiekėjo atstovo informavimo el. p., telefonu ar kitomis priemonėmis kaip Teams ir pan.);

13.6.2. svarbios klaidos atveju – ne vėliau kaip per 2 (dvi) darbo dienas nuo klaidos užregistravimo;

13.6.3. kitais atvejais – ne vėliau kaip per 7 (septynias) kalendorines dienas nuo klaidos užregistravimo.

13.7. Garantija apima sukurtos programinės įrangos sutrikimų šalinimą. Tiekėjas sutrikimus turi ištaisyti savo sąskaita ne vėliau kaip per 2 (dvi) darbo dienas nuo Perkančiosios organizacijos raštiško pranešimo (el. paštu, ar kita rašytine abiejų sutarties šalių sutarta forma) apie pastebėtus sutrikimus arba pateikti laikiną sprendimą.

13.8. Tiekėjas turi pateikti detaliai aprašytus garantinio aptarnavimo darbų organizavimo principus ir suderinti juos su Perkančiąja organizacija iki garantinio aptarnavimo pradžios.

_____