

MOKSLO IR STUDIJŲ TIKSLAMS SKIRTOS PROGRAMINĖS ĮRANGOS KŪRIMO IR PLATINIMO VILNIAUS UNIVERSITETE METODINĖS REKOMENDACIJOS

I SKYRIUS BENDROSIOS NUOSTATOS

1. Mokslo ir studijų tikslams skirtos programinės įrangos kūrimo ir platinimo Vilniaus universitete metodinės rekomendacijos (toliau – Rekomendacijos) nustato principus ir gerąsias praktikas, kuriomis rekomenduojama vadovautis mokslinio tyrimo ar studijų tikslais kuriantiems ir (ar) platinantiems programinę įrangą Vilniaus universiteto (toliau – Universitetas) darbuotojams ir Universitete studijuojantiems.

2. Rekomendacijų kontekste mokslinio tyrimo tikslais sukurta programine įranga laikomi programinės įrangos komponentai (pvz., pirminio programos teksto failai, skriptai, skaičiavimo darbo eigos ir t. t.), sukurti arba naudojami mokslinio tyrimo proceso metu ar mokslinio tyrimo tikslams įgyvendinti.

3. Rekomendacijose vartojamos sąvokos:

3.1. **Atsarginė kopija** – periodiškai atnaujinama duomenų arba programinės įrangos failų kopija, padėta saugoti, kad įvykus incidentui būtų galima atkurti prarastus duomenis.

3.2. **Centrinė repozitorija** – pagrindinė repozitorija, laikanti visus visų naudotojų pakeitimus (centralizuoto versijų kontrolės įrankio, pvz., *Subversion* naudojimo atveju) arba viena sutartinė repozitorija, turinti viršenybę prieš kitas jos kopijas (decentralizuoto įrankio, pvz., *Git* atveju).

3.3. **FAIR principai** – principai, kurių laikantis užtikrinama, kad mokslinių tyrimų duomenys būtų randami, prieinami, sąveikūs ir pakartotinai panaudojami.

3.4. **Laisva ir atvirojo kodo programinė įranga (angl. „Free/Libre Open Source Software“, F/LOSS)** – programinė įranga, kurią galima priskirti tiek laisvai, tiek atvirojo kodo programinei įrangai. Programinei įrangai suteikiama laisva licencija bet koku būdu ją naudoti, kopijuoti, tirti ir keisti, o šaltinio kodu yra atvirai dalijamasi, kad žmonės būtų skatinami savanoriškai tobulinti programinės įrangos funkcijas bei dizainą.

3.5. **Licencija** – autorių teisių, gretutinių teisių ar *sui generis* teisių subjekto (licenciaro) leidimas, suteikiantis kūrinio, gretutinių teisių ar *sui generis* teisių objekto naudotojui (licenciatui) teisę naudoti kūrinio, gretutinių teisių ar *sui generis* teisių objekto originalą arba jo kopijas (licencijos dalyką) nurodytoje teritorijoje tokiu būdu ir tokiomis sąlygomis, kaip numatyta licencinėje sutartyje.

3.6. **Pirminis programos tekstas** – instrukcijos, kokius veiksmus turėtų vykdyti kompiuteris, surašytos asmeniui įskaitomu tekstiniu formatu.

3.7. **Programinė įranga** – kompiuterio vykdomų instrukcijų seka, susidedanti iš šaltinio kodo, vykdomųjų failų, naudotojo dokumentacijos, testavimo duomenų, architektūrinio modelio, kuri skirta tam tikriems veiksams atlikti (pvz., duomenų apdorojimo algoritmai, duomenų analizės skriptai, skaičiavimo darbo eigos).

3.8. **Vykdomasis failas** – failas, kuriame laikoma vykdomoji programa.

3.9. **Vykdomoji programa** – programa, kurią kompiuteris gali vykdyti. Paprastai tai programa, sukompiliuota į kompiuterinę kalbą. Gali būti parašyta ir programavimo kalba, jeigu kompiuteris turi tą kalbą suprantantį interpretatorių.

4. Kitos Rekomendacijose vartojamos sąvokos suprantamos taip, kaip jos yra apibrėžtos Universiteto ir Lietuvos Respublikos teisės aktuose.

II SKYRIUS PROGRAMINĖS ĮRANGOS KŪRIMO IR PLATINIMO REKOMENDACIJOS

PIRMASIS SKIRSNIS PROGRAMINĖS ĮRANGOS VERSIJAVIMAS

5. Kuriamą programinę įrangą rekomenduojama versijuoti – tai yra sekti pakeitimus jos pirminiame tekste bei susijusiuose ištekliuose. Versijavimui rekomenduojama naudoti versijų valdymo sistemas, tokias kaip *Git* arba *Subversion*. Rekomenduojama kiekvienai programinei įrangai naudoti vieną centrinę repozitoriją.

6. Versijavimui, kaip ir pradinio kodo platinimui, yra svarbi prieiga prie pirminių redaguojamų failų (pvz., jei redagavimui naudojamas vektorinis paveikslėlis, kuris programinės įrangos pakete dėl vieno ar kitų priežasčių pateikiamas rastriniame formate, rekomenduojama pateikti pradinį vektorinį paveikslėlį, nes rastrinį formatą redaguoti sudėtingiau, be to, dėl vektorinės-rastrinės konversijos prarandamas tikslumas). Ši rekomendacija taikoma visam generuojamam programiniam kodui bei susijusiems ištekliams (pvz., pagal gramatiką generuojamiems sintaksės analizatoriams, minifikuotam *Javascript* kodui, objektiniais failais ir t. t.).

ANTRASIS SKIRSNIS KURIAMOS PROGRAMINĖS ĮRANGOS PRIEGLOBA

7. Mokslinio tyrimo tikslais Universitete kuriamos programinės įrangos versijų kontrolės repozitorijų prieiglobai rekomenduojama naudoti Universiteto institucinę *GitLab* (<https://gitlab.vu.lt>) serverį, tačiau galima naudoti ir kitas programinio kodo prieiglobai skirtas platformas, tokias kaip *GitHub*, *GitLab*, *Codeberg* ir atitinkamo Universiteto padalinio arba asmeninius serverius ar svetaines. Rekomenduojama rinktis platformas, įgalinančias ne tik programinio kodo prieiglobą, bet ir pavišimą. Universitetui nuosavybės teise priklausanti programinė įranga turi turėti kopiją Universiteto programinio kodo saugyklose (pvz., Universiteto instituciniame *GitLab* serveryje), tačiau, jei ateityje būtų ketinama komercinti kuriamą programinę įrangą, rekomenduojama pasirinkti atitinkamus prieinamumo ir matomumo (angl. *access and visibility*) nustatymus, kurie ribotų tokios programinės įrangos prieinamumą ir matomumą.

TREČIASIS SKIRSNIS APSISAUGOJIMAS NUO PROGRAMINĖS ĮRANGOS PRARADIMO

8. Siekiant išvengti programinės įrangos praradimo dėl žmogiškosios klaidos, techninių gedimų ar kitų nenumatytų aplinkybių, rekomenduojama daryti kuriamos programinės įrangos atsargines kopijas. Rekomenduojama vienu metu turėti bent dvi vystomos programinės įrangos repozitorijos atsargines kopijas be centrinės repozitorijos. Talpinant Universiteto instituciniame *GitLab* serveryje apsisaugoma nuo programinės įrangos praradimo, nes šios saugyklos atsarginės kopijos daromos centralizuotai.

KETVIRTASIS SKIRSNIS PROGRAMINĖS ĮRANGOS DOKUMENTAVIMAS

9. Tiek išsaugant programinę įrangą vidiniam naudojimui, tiek atveriant ją plačiajai visuomenei, rekomenduojama laikytis programinės įrangos aprašymo gerųjų praktikų (pvz., aiškiai nurodyti programinės įrangos paskirtį, versiją, diegimo ir naudojimo instrukcijas, priklausomybes, licencijavimo sąlygas ir kontaktinę informaciją). Tvarkinga ir išsami dokumentacija padės būsimiems programinės įrangos naudotojams ją teisingai suprasti, įdiegti, naudoti ir cituoti, taip pat supaprastins darbą prie vystymo prisidėti norintiems Universiteto bendruomenės nariams (darbuotojams ir studijuojantiems).

10. Archyvuojant arba viešinant programinę įrangą, rekomenduojama nuosekliai numeruoti jos versijas (laidas). Laidos yra skirtos unikalčiai identifikuoti kuriamos programinės įrangos išbaigtumo būsenas. Su laida susieta programinės įrangos visuma (paketas) privalo išlikti pastovi ir

nekisti. Laidoms identifikuoti rekomenduojama naudoti griežtai didėjančius skaitinius versijų numerius, turinčius nuo vieno iki keturių komponentų, nes tokius numerius nesunku lyginti tiek asmeniui, tiek kompiuteriui. Rekomenduojama naudoti *SemVer* versijų numerių schemą, pagrįstą skirtingų programinės įrangos laidų tarpusavio suderinamumu arba *CalVer* versijų numerių schemą, pagrįstą laidų datomis. Kartą pasirinkus versijavimo schemą, jos keisti nereikėtų.

11. Rekomenduojama į centrinės repozitorijos arba viešos jos kopijos šakninį lygį įtraukti specialius dokumentacijos failus (šių failų pavyzdžiai pateikiami Rekomendacijų 1–3 prieduose):

11.1. *README* – tekstinio formato informacijos failą, kuriame pateikiama:

11.1.1. trumpas programinės įrangos aprašymas;

11.1.2. laidų (versijų) sąrašas su identifikatoriais, jei jie suteikti;

11.1.3. kūrėjų ir naudotojų gidai (angl. *Developer guide* ir *User guide*);

11.1.4. informacija apie prisidėjimo prie programinės įrangos vystymo galimybes;

11.1.5. kontaktinė / techninio palaikymo informacija;

11.1.6. jeigu nenurodyta specialiai tam skirtuose failuose, informacija apie licencijavimą, autorius ir bendradarbius bei nurodymai, kaip cituoti programinę įrangą.

11.2. *LICENCE* – tekstinį failą, kuriame pateikiamas pilnas pasirinktos programinės įrangos licencijos tekstas.

11.3. Pakeitimų registrą (angl. *Changelog*) – tekstinį failą, kuriame fiksuojami esminiai programos pakeitimai.

11.4. *CITATION.CFF* – *YAML* formato failą, kuriame žmogui ir kompiuteriu nuskaitymu būdu pateikiama visa reikalinga citavimo informacija.

11.5. *Contributing* („bendradarbiavimo gairės“) – tekstinio formato failą, kuriame pateikiama informacija, kaip galima prisidėti prie programinės įrangos. Šiame faile taip pat pateikiamos instrukcijos, kaip pranešti apie problemas, siūlyti pakeitimus ir laikytis kodo rašymo standartų.

PENKTASIS SKIRSNIS

PROGRAMINĖS ĮRANGOS ATVĖRIMO IR NEATVĖRIMO ATVEJAI

12. Universitete yra skatinami atvirojo mokslo principai ir siekiama, kad moksliniai tyrimai, tyrimų procesai bei rezultatai būtų kiek įmanoma atviri. Universitete siekiama, kad mokslinio tyrimo tikslais sukurta programinė įranga būtų platinama kaip laisva ir atvirojo kodo programinė įranga, t. y. būtų atvira prieinama su licencijomis, leidžiančiomis teisėtu būdu ją vykdyti, kopijuoti, tirti, keisti ir platinti.

13. Mokslinio tyrimo tikslais sukurtos programinės įrangos, o ypač jos pirminio teksto, atvėrimas didina jos matomumą, gali paskatinti naujų ryšių užmezgimą ir bendradarbiavimą. Viešai paskelbus programinės įrangos pirminį tekstą, jį išanalizuoja platesnė auditorija, o tai gali padėti nustatyti klaidas, gauti pasiūlymų dėl tobulinimo, taip užtikrinant aukštesnę programinės įrangos kokybę. Be to, tam tikri su atvertos programinės įrangos panauda susiję rodikliai (pvz., atsisiuntimų skaičius, priklausomybės, programinės įrangos citavimai) gali pasitarnauti kaip poveikio ir mokslinio indėlio įrodymai.

14. Mokslinio tyrimo tikslais sukurtos programinės įrangos atvėrimas taip pat naudingas ir pačiam mokslui, nes tai prisideda prie mokslinių tyrimų skaidrumo užtikrinimo. Programinio kodo, naudoto mokslinių tyrimų rezultatams gauti, paviešinimas leidžia kitiems tyrėjams patikrinti rezultatus, išsamiai suprasti metodologiją ir užtikrintai remtis darbu vystant tolesnius tyrimus. Programinės įrangos atvėrimas užtikrina darbo tęstinumą, kai ją sukūręs darbuotojas ir (ar) studijuojantysis palieka Universitetą.

15. Vis tik, kai kuriais atvejais atverti programinę įrangą gali būti neišmintinga ir netgi pavojinga. Ypatingas atsargumas reikalingas, kai dėl atliekamų funkcijų pobūdžio programinė įranga patenka į Europos Parlamento ir Tarybos reglamente (ES) 2021/821 pateikiamą dvejojo naudojimo prekės apibrėžimą, t. y. tais atvejais, kai programinė įranga galėtų būti naudojama ir civiliniams, ir gynybos tikslams. Jeigu iškyla abejonių dėl to, ar kuriama programinė įranga patenka į šią kategoriją

ir ar reikėtų imtis kokių nors atsargumo priemonių, siūloma prieš imantis bet kokių su platinimu susijusių veiksmų kreiptis į savo tiesioginį vadovą, kuris nukreips į atsakingą specialistą.

16. Atverti programinę įrangą nerekomenduojama ir tada, kai ji neatsiejamai susijusi su neskelbtiniais ar konfidencialiais duomenimis (pvz., išlaptinta valstybės informacija, konfidenciali verslo įmonės informacija ar fizinio asmens duomenimis). Jeigu programinę įrangą susijusi su valstybės informacija, patariama visais atvejais vadovautis nuostatomis, numatytais prieigą prie šios informacijos suteikiančiuose dokumentuose. Jei programinę įrangą kuriama, pasitelkiant verslo duomenis, kurie yra arba gali būti konfidencialūs, būtina atkreipti dėmesį, ar Universitetas su įmone yra sudaręs susitarimą dėl įmonei priklausančių duomenų naudojimo atvėrimo mokslinio tyrimo ir (ar) studijų tikslais. Jeigu programinę įrangą susijusi su asmenis identifikuoti leidžiančia informacija, būtina atkreipti dėmesį į Bendrojo duomenų apsaugos reglamento (BDAR) reikalavimus. Nepakanka tik formaliai siekti duomenų anonimiškumo ar asmeninių duomenų panaikinimo, tačiau reikia ir užtikrinti, kad ir iš likusių duomenų nebūtų galimybės pažeisti asmenų ar grupių privatumo ir informacijos konfidencialumo. Jei programinės įrangos kūrimo ar platinimo metu tvarkomi asmens duomenys, rekomenduojama kreiptis į Universiteto duomenų apsaugos pareigūną (el. pašto adresu: dap@vu.lt), kuris įvertins, ar būtinas poveikio duomenų apsaugai vertinimas ir (ar) kiti pagal BDAR numatyti veiksmai.

17. Kai kuriant programinę įrangą buvo panaudoti komponentai, kurių licencijos draudžia atvirą dalijimąsi išvestiniais kūrinių, programinės įrangos negalima atverti dėl intelektinės nuosavybės apribojimų.

ŠEŠTASIS SKIRSNIS

INTELEKTINĖ NUOSAVYBĖ IR PROGRAMINĖS ĮRANGOS LICENCIJAVIMAS

18. Platinti ar kopijuoti programinę įrangą galima tik tada ir tik tokiomis sąlygomis, kokiomis tai daryti leidžia autorių turtinių teisių savininko suteikta licencija. Jeigu prie vykdomosios programos ar jos pirminio teksto nėra nurodyta jokia jiems suteikta licencija, iš esmės, bet koks jų panaudojimas be išreikšto autorių turtinių teisių savininko leidimo yra draudžiamas. Todėl labai svarbu, kad kiekviena viešai publikuojama programinė įrangą turėtų priskirtą licenciją, kuri nurodytų, kaip ją galima teisėtai naudoti.

19. Universiteto darbuotojai gali platinti ir skelbti savo sukurtą programinę įrangą vienos iš *Free Software Foundation* (FSF) arba *Open Source Initiative* (OSI) patvirtintų atvirų licencijų pagrindu vadovaujantis Vilniaus universiteto atvirojo mokslo politikos gairėse įtvirtintais principais.

20. Universitete kuriamą programinę įrangą platinti galima ir uždaros licencijos pagrindu (pvz., kai norima apsaugoti potencialiai patentuotinus išradimus, verslo įmonės konfidencialius duomenis ir pan.). Tokiu atveju sprendimas dėl jos licencijavimo ir platinimo priimamas vadovaujantis Universiteto intelektinės nuosavybės valdymo nuostatų ir kitų teisės aktų nustatyta tvarka.

21. Platinant programinę įrangą, kurią sukūrė Universiteto darbuotojai drauge su Universitete studijuojančiais, rekomenduojama aptarti, kurios programinės įrangos dalys priklauso kuriems autoriams, ir susitarti dėl programinės įrangos platinimo licencijų. Atviros licencijos leidžia bendrai naudoti ir platinti programinę įrangą, net jei ji sukurta kelių autorių ir turi keletą turtinių teisių savininkų.

22. Parenkant licencijas programinei įrangai, reikia atsižvelgti į licencijų suderinamumą, konkrečių licencijų apribojimus ir programinės įrangos taikymo sritis. Jei programinę įrangą naudoja kitų autorių sukurtus komponentus, tų komponentų licencijos gali apriboti ir naujo paketo licencijos pasirinkimą (pvz., viena populiariausių atvirų programinės įrangos licencijų *GPL* reikalauja, kad išvestiniai kūriniai (angl. *derivative works*), paremti ja licencijuota programine įrangą, būtų platinami su ta pačia licencija. Jei kuriant programinę įrangą naudojama paprogramių biblioteka, platinama su *GPL* licencija, arba panaudojami ženklūs kodo fragmentai iš programos, platinamos pagal šią licenciją, ji bus laikoma „išvestiniu kūriniu“ ir taip pat turės būti platinama pagal *GPL* licenciją). Jei manoma, kad kuriama programinė įrangą galės būti panaudota komerciniams, uždaro kodo ar

naudojantiems dvigubą licenciją (angl. *dual-license*) projektams, patartina taikyti „liberalesnes“ licencijas, tokias kaip *LGPL*, *LGPL* su išimtimis, *BSD* ar *MIT*, kurios nedraudžia derinti atviro kodo su sukurtu kodu.

23. Licenciniai apribojimai priklauso ir nuo tam tikrų techninių detalių (pvz., *GPL* atveju jie priklausys nuo to, ar programa yra surenkama (angl. *linked*) su *GPL* licencijuotais fragmentais į vieną vykdomąjį failą, ar *GPL* licencijuotas kodas kompiliuojamas į atskirą vykdomąjį failą, kurį galima paleisti ir (ar) naudoti nepriklausomai nuo visos sistemos. Jei *GPL* licencijuotas kodas yra atskirame vykdomajame faile kaip iškviečiama išorinė programa, ji bus leidžiama pakviesti ir iš nuosavybinio ar *MIT* / *BSD* licencijuoto kodo, nes kviečiančiam kodui *GPL* apribojimai negalioja. Taip pat *GPL* neriboja galimybių panaudoti *GPL* licencijuotą kodą saityno paslaugoms (angl. *web services*) teikti (serveriams)). Jei kuriama programa sukurta pagal kliento serverio modelį bei gali būti naudojama kaip saityno paslauga internete, ir norima reglamentuoti, kaip tokia programa gali būti naudojama galimų paslaugos teikėjų, galima taikyti *GNU Affero* licenciją.

24. Laisvos ir atviro kodo licencijos, įskaitant (*L*)*GPL*, *Affero*, *MIT* ir *BSD*, neuždraudžia komercinio programinės įrangos panaudojimo. Licencijų apribojimai nukreipti ne į komercinės naudos gavimą, o į pamatinių programinės įrangos naudotojų teisių apsaugą. Tokiu būdu, net ir paskelbus programinę įrangą atviros licencijos pagrindu, išlieka galimybės jos komerciniam panaudojimui. Atviros licencijos neperduoda turtinių teisių potencialiems programinės įrangos naudotojams. Jos suteikia neišimtinę teisę naudotis programine įranga, laikantis tam tikrų apribojimų. Turtinių teisių savininkai gali licencijuoti tą pačią programinę įrangą ir naudodami kitas licencijas, tarp jų ir nuosavybines, jei to reikėtų.

25. Rekomenduojama mokslinio tyrimo ir studijų tikslais Universitete kuriamos programinės įrangos dokumentaciją taip pat licencijuoti atvirosiomis licencijomis, prieš tai įvertinus Rekomendacijų 15, 16 ir 20 punktų nuostatas. Dokumentacija gali būti platinama pagal tą pačią licenciją, kaip ir programinės įrangos pirminis tekstas, tačiau programinei įrangai skirtos licencijos nėra pritaikytos tekstiniais dokumentais. Dėl šios priežasties, rekomenduojama dokumentaciją platinti pagal kitas atviras, su programinės įrangos paketo licencija suderinamas licencijas, tokias kaip *CC BY* ar *CC BY-SA*, arba su *CC0* žyma.

SEPTINTASIS SKIRSNIS LICENCINĖ SUTARTIS

26. Programinė įranga retai būna sukuriamą vieno asmens. Jei išoriniai autoriai (ne Universiteto darbuotojai ir (ar) studijuojantieji) reikšmingai prisideda prie programinės įrangos kūrimo, jų pateiktam programiniam kodui turi būti numatyta licencinė sutartis (angl. *Contributors License Agreement, CLA*), naudojama siekiant užtikrinti, kad tiek bendraautoriai, tiek pradinės programinės įrangos kūrėjai turėtų visas būtinas teises į savo įnašus ir į galutinį kūrinį (informaciją, kaip galima įgyvendinti įeinančios programinės įrangos licencijas pateikiama Maracke, C. (2013). Copyright management for open collaborative projects – inbound licensing models for open innovation. *SCRIPTed* 10(2), 140-148. Edinburgh University Library, prieiga per internetą: <https://doi.org/10.2966/scrip.100213.140>). Praktiniam įeinančios programinės įrangos licencijos įgyvendinimui galima naudoti *CLA* vedlį *GitHub* platformoje (<https://cla-assistant.io/>, <https://github.com/cla-assistant/cla-assistant>) arba analogiškus įrankius, pasirūpinus, kad jie būtų suderinami su Lietuvos Respublikos ir Universiteto teisės aktais, įskaitant, bet neapsiribojant, Lietuvos Respublikos autorių teisių ir gretutinių teisių įstatymą ir Vilniaus universiteto intelektualinės nuosavybės valdymo nuostatus.

AŠTUNTASIS SKIRSNIS PROGRAMINĖS ĮRANGOS REGISTRAVIMAS IR IDENTIFIKATORIŲ SUTEIKIMAS

27. Platinant laisvą ir atviros kodo programinę įrangą, sukurta mokslinio tyrimo tikslais, svarbu užtikrinti jos surandamumą ir prieinamumą. Programinės įrangos prieinamumą užtikrinti

padės atviros platformos naudojimas programinės įrangos prieigai. Programinės įrangos surandamumą užtikrinti galima užregistruojant ją specialios paskirties talpykloje (pvz., *Software Heritage* ar *Code Ocean Open Science Library*) arba tarpdisciplininėje duomenų talpykloje, leidžiančioje įgyvendinti FAIR principus (pvz., *Zenodo*, Nacionalinis atviros prieigos mokslinių tyrimų duomenų archyvas (MIDAS), *Figshare*, *Open Science Framework*, *Dryad* ir pan.). *Software Heritage* talpykla automatiškai registruoja įvairiose programinio kodo prieigos platformose, tokiose kaip *GitHub*, *GitLab* ir daug kitų, atvirai prieinamą programinę įrangą, todėl jeigu kuriama programinė įranga buvo atverta per vieną iš šių platformų, verta pasitikrinti, ar jos archyvinė kopija nėra jau prieinama *Software Heritage* talpykloje.

28. Norint užtikrinti programinės įrangos surandamumą ir cituojamumą, svarbu jai suteikti unikalų nekintantį identifikatorių (pvz., DOI, Handle, Archival Resource Key (ARK) ar SWHID). Kiekvienai dokumentuotai programinės įrangos versijai turėtų būti suteikiamas atskiras identifikatorius. Tai leis tiksliai nurodyti, kuri versija buvo naudojama moksliniame tyrime ar publikacijoje, ir palengvins programinės įrangos citavimą.

DEVINTASIS SKIRSNIS

FAIR PRINCIPAI

29. FAIR principai yra palaikomi tiek Universiteto, tiek Lietuvos mokslo tarybos, Europos Sąjungos koordinuojamų programų ir daugelio kitų aukštųjų mokyklų bei institucijų. FAIR principai sukurti mokslinių tyrimų duomenims, tačiau pagrindiniai jų išreiškiami siekiniai – surandamumas, pasiekiamumas, sąveikumas ir tinkamumas pakartotinai naudoti – gali būti pritaikomi ir programinei įrangai. Šios Rekomendacijos atitinka FAIR principus ir Rekomendacijų laikymasis padeda šiuos principus veiksmingiau įgyvendinti.

III SKYRIUS

REKOMENDACIJOS PROGRAMINĖS ĮRANGOS CITAVIMUI

30. Tvarkingas ir išsamus mokslinio tyrimo metu naudotos programinės įrangos citavimas prisideda prie mokslo rezultatų atkuriamumo didinimo, skatina pakartotinį programinės įrangos naudojimą ir jos tobulinimą, padeda susidaryti išsamesnį vaizdą apie tai, kokie mokslinės veiklos produktai sukuriama vykdant vienos ar kitos mokslo srities tyrimus ir užtikrina pelnytą programinę įrangą sukūrusių tyrėjų indėlio pripažinimą.

31. Jeigu generuojant mokslinėje publikacijoje aprašytus rezultatus buvo panaudota tam tikra programinė įranga, ypač jeigu jos naudojimas turėjo reikšmės tyrimo naujumui ir aktualumui, ši programinė įranga paminima publikacijos literatūros sąrašė. Cituojant dirbtinio intelekto generatyviais modeliais paremtus rezultatus, be įprastų programinės įrangos citavimo gairių, būtina atsižvelgti į Dirbtinio intelekto naudojimo Vilniaus universitete gairėse pateikiamas rekomendacijas. Į literatūros sąrašą įtraukti nebūtina bendros paskirties įrankių (pvz., teksto redaktorių), taip pat nebūtina cituoti naudotą operacinę sistemą. Jei bendrosios paskirties programinės įrangos ar operacinės sistemos pasirinkimas gali turėti įtakos tyrimo rezultatų atkuriamumui, verta jas paminėti aprašant tyrimo metodus. Kaip ir skelbiant publikaciją, publikuojant duomenų rinkinį, prie jo pridedamoje informacijoje (metaduomenyse arba README faile) rekomenduojama cituoti duomenims gauti ir (arba) apdoroti naudotą programinę įrangą. Dalinantis naujai sukurta programine įranga, jos README faile taip pat reikėtų paminėti programinę įrangą, kuri buvo panaudota kaip komponentas ar turėjo reikšmingos įtakos pasidalintos programinės įrangos idėjai arba kūrimo procesui.

32. Cituojant programinę įrangą, rekomenduojama:

32.1. jeigu egzistuoja kelios skirtingos programinės įrangos pateikimo formos, pvz., pirminis programos tekstas, vykdomasis failas ir programą aprašanti publikacija, rekomenduojama cituojant pirmenybę teikti pirminiam programos tekstui. Autoriams pageidaujant, galima kartu su pirminiu tekstu pacituoti ir programinę įrangą aprašančią (-as) publikaciją (-as);

32.2. jeigu programinė įranga yra prieinama tiek versijų kontrolei skirtoje platformoje, tiek ilgalaikiam archyvavimui skirtoje talpykloje (pvz., *Software Heritage*), rekomenduojama cituojant nurodyti ilgalaikio archyvavimo vietą, nes joje pateikiamas turinys mažiau linkęs kisti;

32.3. jeigu cituojama programinė įranga buvo sukurta kitos programinės įrangos pagrindu, tačiau ta programinė įranga nebuvo tiesiogiai panaudota tyrime, ją cituoti nėra būtina;

32.4. jeigu į cituojamos programinės įrangos paketą buvo įtrauktas *CITATION.CFF* failas, rekomenduojama cituojant naudoti šiame faile pateikiamą informaciją;

32.5. cituojant programinę įrangą rekomenduojama nurodyti jos autorių, pavadinimą, naudotą versiją, nekintantį identifikatorių (jei toks suteiktas), ir kur ar koku būdu galima gauti prieigą prie jos. Taip pat rekomenduojama pridėti žymą, kad cituojamas objektas yra būtent programinė įranga. Papildomai galima nurodyti, kokia programinės įrangos forma cituojama;

32.6. APA formatu paremti programinės įrangos citavimo pavyzdžiai lietuvių ir anglų kalbomis:

32.6.1. Hebbur Dayananda, S., de Vicente, A., del Pino Alemán, T., Shchukina, N. ir Trujillo Bueno, J. (2025). P-CORONA (v1) [Programinė įranga: paketas]. Zenodo. <https://doi.org/10.5281/zenodo.15195461>;

32.6.2. Hebbur Dayananda, S., de Vicente, A., del Pino Alemán, T., Shchukina, N., & Trujillo Bueno, J. (2025). P-CORONA (v1) [Software: Package]. Zenodo. <https://doi.org/10.5281/zenodo.15195461>;

32.7. akademiniai leidėjai gali turėti specifinių reikalavimų programinės įrangos citavimui, todėl cituojant programinę įrangą publikacijoje, reikėtų vadovautis pasirinkto leidėjo gairėse autoriams pateikiamomis rekomendacijomis.

IV SKYRIUS BAIGIAMOSIOS NUOSTATOS

33. Šios Rekomendacijos viešai skelbiamos Universiteto ir jo padalinių, kurių darbuotojai ir (ar) studijuojantieji mokslinio tyrimo ar studijų tikslais kuria programinę įrangą arba administruoja tokios programinės įrangos kūrimą, interneto svetainėse.

34. Universiteto bibliotekos Mokslinės komunikacijos ir technologijų departamento Mokslinės informacijos ir duomenų skyrius užtikrina šių Rekomendacijų reguliarių (ne retesni nei kas 5 metai) atnaujinimą atsižvelgiant į technologijų vystymąsi ir Universiteto bendruomenės poreikius.

35. Universiteto bibliotekos Mokslinės komunikacijos ir technologijų departamento Mokslinės informacijos ir duomenų skyrius konsultuoja Universiteto bendruomenės narius šių Rekomendacijų įgyvendinimo klausimais.

Mokslo ir studijų tikslams skirtos
programinės įrangos kūrimo ir
platinimo Vilniaus universitete
metodinių rekomendacijų
1 priedas

README FAILO PAVYZDYS

Šiame priede pateikiamas *README* failo pavyzdys. Jis pateikiamas kaip paprastas tekstas su *Markdown* kalba užkoduotu formatavimu. Pateikiami du variantai – vienas anglų ir vienas lietuvių kalba.

Vietoje pilka spalva pažymėto pavyzdinio teksto reikia įrašyti konkrečios programinės įrangos informaciją.

README failo pavyzdys anglų kalba

Software Name and Current Version number

Software Name is ...(software type). It does XYZ/ enables X/ is designed for Y.

[Further information] (<https://...>)

Package Content

Item	Comment
Folder 1	Comment
File 1	Comment
File 2	Comment
File 3	Comment
Folder 2	Comment
File 1	Comment
File 2	Comment
File 3	Comment
CITATION.cff	
Contributing	
CHANGELOG	
LICENSE	
README	

Other Published Versions

- Author name & Vilnius University (2025). Software Name. v1.1. PID or URL.

- Author name & Vilnius University (2024). Software Name. v1.0. PID or URL.

User Guide

- These instructions were tested on ...(operating system name and version)

- Installing Software Name requires you to ...(list of dependencies, pre-conditions, and rights required)

- Other useful information (if applicable)


```

## Install Dependencies

###Dependency Name 1
[Download link] (https://....) or file path (if installation file provided
with the software package)

###Dependency Name 2
[Download link] (https://....) or file path (if installation file provided
with the software package)

###Dependency Name 3
[Download link] (https://....) or file path (if installation file provided
with the software package)

## Install Software Name

###Step 1
Instructions (text)
'''
Code
'''

###Step 2
Instructions (text)
'''
Code
'''

###Step 3
Instructions (text)
'''
Code
'''

-----
# Developer guide

- These instructions were tested on ...(operating system name and
version)
- Installing Software Name requires you to ...(list of dependencies, pre-
conditions, and rights required)
- Other useful information (if applicable)

## Install Dependencies

###Dependency Name 1
[Download link] (https://....) or file path (if installation file provided
with the software package)

###Dependency Name 2
[Download link] (https://....) or file path (if installation file provided
with the software package)

###Dependency Name 3
[Download link] (https://....) or file path (if installation file provided
with the software package)

```

Install `Software Name`

###Step 1

Instructions (text)

'''

Code

'''

###Step 2

Instructions (text)

'''

Code

'''

###Step 3

Instructions (text)

'''

Code

'''

Contributing

See `CONTRIBUTING` (=provide file name and path)

Copyright and Licence

Copyright (c) 2025, Vilnius University and other copyright holders, if applicable

Documentation licence: `[Licence type]` (link to licence text)

Source code licence: `Licence type`. For full details, see `/LICENCE.md` (=provide file name and path)

Third-Party Components used

Component 1, (c) Copyright Holder, licenced under `Licence`

Component 2, (c) Copyright Holder, licenced under `Licence`

Component 3, (c) Copyright Holder, licenced under `Licence`

Contacts

If you have any questions or suggestions regarding `Software Name`, feel free to reach out to us. To contact our team, please use the following email address: `contact@faculty.vu.lt`

Acknowledgments

`Software Name` has its origins in: `A Publication | A research project | Another Software | ...`

`Software Name` has evolved in response to | feedback from: `Name, Institution; Name, Institution; ...`

We also acknowledge the funding of Funder | the assistance of Partner Institution | ...

README failo pavyzdys lietuvių kalba

Programinės Įrangos Pavadinimas ir dabartinės versijos numeris

Programinės Įrangos Pavadinimas yra ... Ji skirta XYZ/ įgalina X/ sukurta tikslui Y.

[Daugiau informacijos] (<https://...>)

Paketo turinys

Elementas	Komentaras
-----	:-----
Katalogas 1	Komentaras
Failas 1	Komentaras
Failas 2	Komentaras
Failas 3	Komentaras
Katalogas 2	Komentaras
Failas 1	Komentaras
Failas 2	Komentaras
Failas 3	Komentaras
CITATION.cff	
Contributing	
CHANGELOG	
LICENSE	
README	

Kitos publikuotos versijos

- Autorius & Vilniaus universitetas (2025). Programinės Įrangos Pavadinimas. v1.1. Nekintantis identifikatorius arba internetinė nuoroda.
- Autorius & Vilniaus universitetas (2024). Programinės Įrangos Pavadinimas. v1.0. Nekintantis identifikatorius arba internetinė nuoroda.

Naudotojo vadovas

- Šios instrukcijos buvo išbandytos su ... (operacinės sistemos pavadinimas ir versija)
- Norint įdiegti Programinės Įrangos pavadinimas, bus reikalinga... (priklausomybės, reikalingos išankstinės sąlygos ir teisės)
- Kita naudinga informacija (jei yra)

Priklausomybių diegimas

###Priklausomybė 1

[Atsisiuntimo nuoroda] (<https://...>) arba kelias iki failo (jei pateikiama su programinės įrangos paketu)

###Priklausomybė 2

[Atsisiuntimo nuoroda] (<https://...>) arba kelias iki failo (jei pateikiama su programinės įrangos paketu)

###Priklausomybė 3

[Atsisiuntimo nuoroda] (<https://...>) arba kelias iki failo (jei pateikiama su programinės įrangos paketu)

```
## Programinės Įrangos Pavadinimas diegimas
```

```
###1 žingsnis
```

```
Tekstinė instrukcija
```

```
'''
```

```
Diegimui reikalingas programinis kodas
```

```
'''
```

```
###2 žingsnis
```

```
Tekstinė instrukcija
```

```
'''
```

```
Diegimui reikalingas programinis kodas
```

```
'''
```

```
###3 žingsnis
```

```
Tekstinė instrukcija
```

```
'''
```

```
Diegimui reikalingas programinis kodas
```

```
'''
```

```
-----
```

```
# Kūrėjo vadovas
```

- Šios instrukcijos buvo išbandytos su ... (operacinės sistemos pavadinimas ir versija)
- Norint įdiegti Programinės Įrangos pavadinimas, bus reikalinga... (priklausomybės, reikalingos išankstinės sąlygos ir teisės)
- Kita naudinga informacija (jei yra)

```
## Priklausomybių diegimas
```

```
###Priklausomybė 1
```

```
[Atsisiuntimo nuoroda] (https://....) arba kelias iki failo (jei pateikiama su programinės įrangos paketu)
```

```
###Priklausomybė 2
```

```
[Atsisiuntimo nuoroda] (https://....) arba kelias iki failo (jei pateikiama su programinės įrangos paketu)
```

```
###Priklausomybė 3
```

```
[Atsisiuntimo nuoroda] (https://....) arba kelias iki failo (jei pateikiama su programinės įrangos paketu)
```

```
## Programinės Įrangos Pavadinimas diegimas
```

```
###1 žingsnis
```

```
Tekstinė instrukcija
```

```
'''
```

```
Diegimui reikalingas programinis kodas
```

```
'''
```

```
###2 žingsnis
```

```
Tekstinė instrukcija
```

```
'''
```

```
Diegimui reikalingas programinis kodas
```

```
'''
```

```
###3 žingsnis
```

```
Tekstinė instrukcija
```

```
'''
```

```
Diegimui reikalingas programinis kodas
```

```
'''
```

```
-----  
# Bendradarbiavimas
```

```
Žr. CONTRIBUTING (=nurodyti atitinkamo failo pavadinimą ir kelią iki failo).
```

```
-----  
# Autorių teisės ir licencija
```

```
Turtinių teisių turėtojas: (c) 2025, Vilniaus universitetas ir kiti  
turtinių teisių turėtojai, jei tokių yra
```

```
Dokumentacijos licencija: [Licencijos tipas] (nuoroda į licencijos  
tekstą)
```

```
Programinio kodo licencija: Licencijos tipas. Išsamesnė informacija:  
LICENCE (=nurodyti atitinkamo failo pavadinimą ir kelią iki failo).
```

```
# Naudoti išoriniai komponentai
```

```
Komponentas 1, (c) Turtinių Teisių Turėtojas), licencijuotas Licencija  
Komponentas 2, (c) Turtinių Teisių Turėtojas), licencijuotas Licencija  
Komponentas 3, (c) Turtinių Teisių Turėtojas), licencijuotas Licencija
```

```
-----  
# Kontaktinė informacija
```

```
Jei turite klausimų ar pasiūlymų, susijusių su Programinės Įrangos  
Pavadinimas, prašome kreiptis el. pašto adresu:  
kontaktinis.asmuo@fakultetas.vu.lt
```

```
# Padėka
```

```
Programinės Įrangos Pavadinimas sukurta: Publikacija | Projektas | Kitos  
programinės įrangos pagrindu | ...
```

```
Programinės Įrangos Pavadinimas kūrėjai dėkoja Vardas, Institucija;  
Vardas, Institucija; ... už vertingas pastabas.
```

```
Programinės Įrangos Pavadinimas kūrėjai dėkoja (Finansuojanti  
organizacija) už paramą ir (Institucija partnerė) už bendradarbiavimą.
```

Mokslo ir studijų tikslams skirtos
programinės įrangos kūrimo ir
platinimo Vilniaus universitete
metodinių rekomendacijų
2 priedas

CITATION.CFF FAILO PAVYZDYS

Šiame priede pateikiamas *CITATION.CFF* failo pavyzdys. Jis pateikiamas kaip paprastas tekstas su *Markdown* kalba užkoduotu formatavimu.

Vietoje pilka spalva pažymėto pavyzdinio teksto reikia įrašyti konkrečios programinės įrangos informaciją.

```
Cff-version: 1.1.0
message: If you use this software, please cite it as below.
title: Software Name
abstract: Software Name is ...(software type). It does XYZ/ enables X/ is
designed for Y.
authors:
- family-names: Tyrėja
  given-names: Tyrėjauskienė
  orcid: "https://orcid.org/0000-xxxx-xxxx-xxxx"
- family-names: Tyrėjas
  given-names: Tyrėjauskas
  orcid: "https://orcid.org/0000-xxxx-xxxx-xxxx"
- family-names: Studentas
  given-names: Studentauskas
  orcid: "https://orcid.org/0000-xxxx-xxxx-xxxx"
version: 0.11.2
date-released: "2025-06-01"
identifiers:
- description: This is the collection of archived snapshots of all
versions of My Research Software
  type: doi
  value: "10.xxxx/yyyyyyy.xxxxxx"
- description: This is the archived snapshot of version 0.11.2 of My
Research Software
  type: doi
  value: "10.xxxx/yyyyyyy.xxxxxx"
license: Licence type
repository-code: "https://..."
keywords: - research software
          - software maintenance
          - ...
```

Mokslo ir studijų tikslams skirtos
programinės įrangos kūrimo ir
platinimo Vilniaus universitete
metodinių rekomendacijų
3 priedas

CHANGELOG FAILO PAVYZDYS

Šiame priede pateikiamas *CHANGELOG* failo pavyzdys. Jis pateikiamas kaip paprastas tekstas su *Markdown* kalba užkoduotu formatavimu. Pateikiami du variantai – vienas anglų ir vienas lietuvių kalba.

Visi reikšmingi pakeitimai programinėje įrangoje turi būti nuosekliai fiksuojami *CHANGELOG* faile. Pakeitimai turėtų būti grupuojami pagal versijas, nurodant jų išleidimo datą.

CHANGELOG failas gali būti pildomas rankiniu būdu arba generuojamas automatizuotai pagal aiškiai apibrėžtus pakeitimų registro formatus.

Vietoje pilka spalva pažymėto pavyzdinio teksto reikia įrašyti konkrečios programinės įrangos informaciją.

CHANGELOG failo pavyzdys anglų kalba

```
## [1.2.0] - 2025-06-15
### Added
- Added user login screen
- New API endpoint: `/users/list`

### Changed
- Refactored authentication module
- Updated user interface design

### Fixed
- Fixed an issue where password reset did not work on mobile browsers

### Removed
- Removed the old authOld.js file
```

CHANGELOG failo pavyzdys lietuvių kalba

```
## [1.2.0] - 2025-06-15
### Added
- Pridėtas naudotojo prisijungimo langas
- Naujas API endpoint: `/users/list`

### Changed
- Refaktoriizuotas autentifikacijos modulis
- Atnaujintas vartotojo sąsajos dizainas

### Fixed
- Pataisyta klaida, kai slaptažodžio atstatymas neveikdavo
mobiliajame naršyklėje

### Removed
- Pašalintas senas `authOld.js` failas
```