

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Denis Oblak

**Čiščenje operacijskega sistema
GNU/Linux**

DIPLOMSKO DELO

UNIVERZITETNI ŠTUDIJSKI PROGRAM
PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Jurij Mihelič

Ljubljana, 2018

COPYRIGHT. Rezultati diplomskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavo in koriščenje rezultatov diplomskega dela je potrebno pisno privoljenje avtorja, Fakultete za računalništvo in informatiko in mentorja.

Besedilo je oblikovano z urejevalnikom besedil L^AT_EX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Sistemska programska oprema za vzdrževanje operacijskega sistema je že nekaj časa močno prisotna v ponudbi programske opreme. Sem sodijo tudi programi za tako imenovano čiščenje operacijskega sistema, ki skušajo s pregledovanjem, analizo, arhiviranjem in odstranjevanjem datotek izboljšati delovanje sistema. V ponudbi prednjačijo predvsem programi za čiščenje operacijskega sistema Microsoft Windows in tudi MacOS, malo manj pa jih najdemo za GNU/Linux.

V okviru diplomske naloge preglejte področje programske opreme za čiščenje omenjenih treh sistemov. Opišite splošne tehnike čiščenja glede na posamezen operacijski sistem. Nato implementirajte svojo rešitev za sistem GNU/Linux, pri čemer naj bo poudarek na tehnikah, ki sistemskemu administratorju olajšujejo redno vzdrževanje sistema.

ZAHVALA

Zahvaljujem se predvsem družini za podporo in spodbudo v času študija. Zahvalil bi se tudi sodelavki Vesni in šefu Gašotu za pomoč, spodbudo in potrpežljivost. Navsezadnje se zahvaljujem mentorju, doc. dr. Juriju Miheliču, za strokovno pomoč in usmerjanje pri izdelavi diplomskega dela.

Denis Oblak, 2018

Kazalo

Povzetek

Abstract

1	Uvod	1
2	Tehnike čiščenja in pregled obstoječih orodij	3
2.1	Operacijski sistem Microsoft Windows	4
2.1.1	Tehnike čiščenja	4
2.1.2	Orodja za čiščenje registra	7
2.1.3	Orodja za čiščenje ostankov odstranjenih programov	9
2.1.4	Orodja za čiščenje prostora na disku	10
2.2	Operacijski sistem MacOS	11
2.2.1	Tehnike čiščenja	11
2.2.2	Orodja za čiščenje	12
2.3	Operacijski sistem Linux	15
2.3.1	Tehnike čiščenja	15
2.3.2	Upravitelji paketov	17
2.3.3	Orodja	18
3	Predstavitev naše rešitve	21
3.1	Podprte funkcionalnosti	21
3.2	Zasnova aplikacije	24
3.3	Podprto okolje	26

3.4	Uporabniški vmesnik	26
3.4.1	Meniji	27
3.4.2	Navigacija med meniji	27
3.4.3	Interakcija z uporabnikom	28
3.5	Dnevniški zapisi in poročila	28
3.5.1	Dnevniški zapisi	28
3.5.2	Poročila	29
3.6	Podrobnosti izvedbe	29
4	Sklepne ugotovitve	35
	Literatura	38

Seznam uporabljenih kratic

Kratica	Angleško	Slovensko
APT	Advanced Packaging Tool	napredno orodje za delo s paketi
DPKG	Debian Package Management System	sistem vodenja paketov Debian
RHEL	Red Hat Enterprise Linux	Red Hat Enterprise Linux
RPM	Red Hat Package Manager	upravitelj paketov Red Hat
YUM	Yellowdog Updater, Modified	upravitelj paketov Yellowdog

Povzetek

Naslov: Čiščenje operacijskega sistema GNU/Linux

Avtor: Denis Oblak

Cilj diplomskega dela je izdelava aplikacije, ki bo pomagala očistiti operacijski sistem Linux in bo delala v večini distribucij. V teoretičnem delu je obravnavano čiščenje operacijskega sistema Linux, ki sprosti prostor na disku in omogoči boljše delovanje sistema. Sistematično so pregledani in teoretično predstavljeni tehnike čiščenja in obstoječa orodja za operacijski sistem Linux.

V nadaljevanju je predstavljeno čiščenje operacijskih sistemov Windows in MacOS. Hkrati so predstavljeni rezultati primerjanj vseh treh operacijskih sistemov.

Na osnovi ugotovitev primerjanj se osredotočimo na našo rešitev. Skoraj vse funkcionalnosti delujejo v več distribucijah Linuxa. Poleg tega je rešitev preprosta za uporabo. Uporabna je tudi na strežniških izvedbah, kjer ni grafičnega vmesnika, saj je izvedljiva v terminalu. Rešitev je varna, saj pred začetkom vsakega brisanja uporabniku ponudi izbiro, kaj vse se bo pobrisalo in kaj ne. Poleg tega ima možnost razveljavitve.

Ključne besede: Linux, čiščenje, prostor na disku, hitrost.

Abstract

Title: Cleaning up a GNU/Linux operating system

Author: Denis Oblak

The aim of the thesis is to develop an application for cleaning up the Linux operating system that would be able to function on most distributions. The theoretical part discusses the cleaning of the Linux operating system that frees up disk space and allows a better functioning. The cleaning techniques and the existing tools for Linux are systematically reviewed and presented.

The following part examines the cleaning of the Windows and MacOS operating systems. The thesis also compares all three operation systems and describes the results.

The findings form the basis for our solution. Almost all functionalities work on multiple Linux distributions and the solution is very simple to use. It is also useful on server environments with no graphical interface, as it runs from a terminal. The solution is safe to use because the user can always choose which files to delete before the start of the process, which can also be reversible.

Keywords: Linux, cleaning, disk space, speed.

Poglavje 1

Uvod

Operacijski sistem Linux praznuje 25-letnico uporabe in se je v vseh teh letih precej spremenil [10]. Včasih je veljalo, da nima visokih strojnih zahtev, da je brez nepotrebnih datotek in da ne potrebuje veliko vzdrževanja.

Zaradi razvoja se je to precej spremenilo in tudi operacijski sistem Linux potrebuje redno vzdrževanje, da lahko deluje optimalno. Izmed operacijskih sistemov je predvsem Windows znan po tem, da sčasoma postane počasnejši kot ob novi namestitvi. Zato je nujno, da po nekem času počistimo odvečne podatke (to so podatki v registru ali na disku, ki jih ne potrebujemo in so glavni razlog za čiščenje) in na tak način sprostimo prostor na disku ter do neke mere optimiziramo delovanje sistema. Odpor do uporabe orodij za čiščenje je povezan predvsem s tem, da se pri tem pogosto pridružijo neželene reklame, dodatni programi in strah pred brisanjem stvari, ki so potrebne za delovanje operacijskega sistema, in posledično sesutja sistema. Tudi za Linux in MacOS nastaja čedalje več orodij za čiščenje.

Vse operacije, ki jih znajo izvesti ta orodja, lahko izvede vsak posameznik s pomočjo že vgrajenih funkcij operacijskih sistemov. Vendar večina običajnih uporabnikov vseh teh funkcij niti ne pozna, hkrati je uporaba orodij hitrejša kot ročno izvajanje vgrajenih funkcij. Strokovnjaki teh orodij večinoma ne uporabljajo, običajnim uporabnikom pa lahko naredijo več koristi kot škode.

Cilj diplomske naloge je napisati aplikacijo za čiščenje operacijskega sistema Linux, ki bo preprosta za uporabo. Večina funkcij aplikacije bo delovala v več različnih distribucijah, tako za običajne uporabnike [56, 55] kot za podjetja [53, 54].

Najprej bodo predstavljene tehnike čiščenja operacijskih sistemov Linux, Windows in MacOS. V navezavi s tehnikami čiščenja bo sledil pregled funkcionalnosti orodij, ki že obstajajo na trgu. Pregled bo zajemal orodja za vsak operacijski sistem posebej in primerjavo med posameznimi orodji. V opisu rešitve bodo predstavljene funkcionalnosti, ki jih aplikacija podpira. Sledila bo podrobna predstavitev zasnove in uporabniškega vmesnika. Predstavljen bo sistem dnevniških zapisov in poročila. Na koncu bo opis podrobnosti izvedbe posameznih funkcionalnosti.

Poglavje 2

Tehnike čiščenja in pregled obstoječih orodij

V vsakem operacijskem sistemu se sčasoma nakopiči precej datotek, ki jih ne potrebujemo več oziroma na katere smo pozabili. Običajni uporabniki tega niti ne vedo in jih ne znajo poiskati, dokler se to ne opazi na zasedenosti diska ali celo zmogljivosti sistema. Osnovni cilj čiščenja sistema je sprostitev prostora na disku, v nekaterih primerih to privede tudi do boljšega delovanja sistema, predvsem pri operacijskem sistemu Windows. Pomembno je redno čiščenje diska. Osnovno čiščenje diska je mogoče izvesti z vgrajenimi orodji v operacijskem sistemu. To je skupno vsem operacijskim sistemom, načini čiščenja pa se med operacijskimi sistemi precej razlikujejo in bodo predstavljeni v nadaljevanju.

Za vse operacijske sisteme velja, da je smiselno redno odstranjevati programe, ki se ne uporabljajo več. Odsvetovano je ročno brisanje fizičnih datotek programov, saj ostane precej sistemskih in konfiguracijskih datotek. V operacijskem sistemu Windowsu se na ta način tudi ne počisti registra. Bolje je uporabiti vgrajeni pripomoček za odstranjevanje, če ga aplikacija ima, saj jih samodejno odstrani.

Vsem operacijskim sistemom z grafičnim vmesnikom je skupno to, da brskanje po spletu za sabo pušča veliko količino informacij. Sčasoma je lahko

teh veliko in zasedajo precej prostora na disku. Se jih pa da preprosto pobrisati kar v samem brskalniku.

Tehnike čiščenja so v manjši meri skupne, predvsem so specifične za posamezne operacijske sisteme, oziroma se razlikujejo med sistemi [27]. Zato smo jih razdelili glede na operacijski sistem.

Vse navedeno in še več znajo narediti namesto nas posebni programi, namenjeni čiščenju operacijskih sistemov. Tudi ti so v večini pisani za posamezni operacijski sistem, le redki so namenjeni več kot enemu. Zato smo tudi ta orodja razdelili glede na operacijski sistem in jih podrobno predstavili. Zavedati se je treba, da pri programih za čiščenje sistema niso le pozitivne plati; glavni slabosti sta zlonamerna programska oprema in napačno pobrisane sistemske datoteke, kar lahko povzroči nedelovanje sistema.

2.1 Operacijski sistem Microsoft Windows

Pri operacijskem sistemu Windows lahko polnijo disk ali upočasnjujejo delovanje sistema programi, ki se samodejno zaženejo ob zagonu sistema, začasne datoteke, podvojene datoteke in podvojene bližnjice na namizju. Zato je treba operacijski sistem očistiti. Če želimo sistem počistiti, lahko to storimo brezplačno brez uporabe posebnih programov. Windows ima vgrajena orodja za čiščenje, s katerimi lahko naredimo skoraj vse, kar bi naredil povprečni program za čiščenje.

Velikokrat je pri čiščenju operacijskega sistema Windows govora tudi o čiščenju registra. Mnenja o čiščenju registra so različna. Čiščenje ne škodi, a vendar moramo biti pri tem zelo previdni.

2.1.1 Tehnike čiščenja

Microsoft zagovarja uporabo vgrajenih orodij operacijskega sistema Windows in odsvetuje uporabo orodij drugih proizvajalcev. Priporoča [40]:

- odpravljanje težav z zmogljivostjo (angl. *performance troubleshooter*):

to je vgrajeno orodje, ki poišče in odpravi težave z zmogljivostjo, najde datoteke in bližnjice, ki jih ne uporabljamo več;

- odstranjevanje neuporabljenih programov: programe, ki jih nikoli ne uporabljamo, je smiselno pobrisati, saj zasedajo prostor na disku in po nepotrebnem obremenjujejo sistem. Odstranimo jih lahko s pomočjo vgrajenega orodja ali uporabimo katerega od specializiranih programov;
- omejevanje števila samodejno zagnanih programov: zagon operacijskega sistema je lahko zelo upočasnjen, če se ob zagonu odpre veliko programov. Manjša hitrost je posledica obremenitve diska in pomnilnika. To lahko rešimo tako, da ob zagonu računalnika nastavimo samo odpiranje nujno potrebnih programov. V Windows 10 to dosežemo z uporabo upravitelja opravil [51]. V starejših različicah operacijskega sistema Windows uporabimo aplikacijo msconfig. Spremembe bodo veljale šele po ponovnem zagonu, ko se bosta sprostila (počistila) disk in pomnilnik [39];
- uporaba vgrajenega orodja za čiščenje diska (angl. *disc cleaner*): redno čiščenje prostora na disku pripomore k večji zmogljivosti sistema. Eden izmed vgrajenih načinov je orodje čiščenje diska. Izberemo lahko, kateri disk želimo pregledati, nato se pokaže seznam datotek, primernih za brisanje, razvrščen po tipu datotek. Vstavimo kljukice pri tistih, ki jih želimo pobrisati, in izberemo brisanje. Brisanje je varno, saj orodje pokaže samo datoteke, ki niso ključne za delovanje sistema. Postopek lahko večkrat ponovimo;
- sočasno izvajanje čim manjšega števila programov: veliko število uporabnikov ima sočasno odprto veliko zavihkov brskalnika, prav tako je pogost pojav veliko sočasno odprtih elektronskih sporočil. Vse to upočasnjuje delovanje operacijskega sistema, zato je treba zapirati programe, ki jih ne potrebujemo več, in s tem počistiti sistem;
- ponovni zagon sistema: to za sistem pomeni nov začetek. Na tak način

se znebimo programov, ki delujejo v ozadju in postopoma obremenjujejo glavni pomnilnik;

- izklop indeksiranja datotek za hitrejše iskanje: operacijski sistem ves čas skrbi, da ima ažuren seznam lokacij datotek, da je morebitno iskanje čim hitrejše. Ažuriranje seznama lahko izklopimo in na tak način nekoliko razbremenimo sistem (počistimo), vendar je zaradi tega iskanje datotek počasnejše;
- pregled sistema za viruse in vohunske programe: virusi ali vohunska programska oprema lahko upočasnijo delovanje računalnika. Zato je treba imeti dobro protivirusno programsko opremo in jo redno uporabljati. Pomembno je, da izberemo protivirusni program, ki dobro zaščiti računalnik in hkrati ne zmanjša učinkovitosti delovanja računalnika, saj slab protivirusni program že sam upočasni delovanje računalnika. Dober protivirusni program odstrani (počisti) viruse in programe, ki prikazujejo nadležne reklame;

Prostor lahko dodatno sprostimo s čiščenjem zgodovine brskanja v brskalniku, orodja za to so vgrajena v brskalniku.

Tudi koš za smeti pomaga pri čiščenju diska. Možno ga je omejiti na odstotek velikosti celotnega diska in na ta način se stvari samodejno brišejo in uporabniku ni treba skrbeti za to.

Fragmentacija je zapisovanje datotek na več mest na disku. Pri vsakdanjem delu redno spreminjamo in brišemo datoteke. Ko brišemo datoteke z diska, ostane na mestu, na katerem je bila datoteka, mesto za zapis novih podatkov. Z večkratnim brisanjem se prosta mesta povečujejo. Kadar na take diske shranjujemo večje datoteke, jih sistem shrani na prvo prosto lokacijo na disku. Če je datoteka večja od razpoložljivega prostora, jo sistem zapisuje na novo prosto mesto, torej v novi fragment, dokler celotna datoteka ni shranjena na disku. S spreminjanjem datoteke gre sistem po vrsti do prvega fragmenta, ga prebere, nato prebere, kje se nahaja ostanek datoteke, ker je lokacija drugačna, premesti bralno glavo na nov naslov in tako

naprej. Sistemi z močno fragmentiranimi diski so zato občutno počasnejši. Defragmentacija združi podatke skupaj na en prostor, preden razdelimo eno particijo na dve ali celo več. Defragmentacija poveča življenjsko dobo in hitrost delovanja trdega diska z zmanjšanjem premikov bralne glave in z lažjim dostopom datotek do operacij [22].

Če zgoraj navedeni postopki ne pomagajo, zagotovo pomaga ponovna namestitvev operacijskega sistema Windows. V nasprotnem primeru je težava v strojni opremi.

2.1.2 Orodja za čiščenje registra

Nekatera orodja delujejo v vseh operacijskih sistemih, nekatera so specifična za posamezni operacijski sistem. Prva večja razlika se pojavi pri programih za čiščenje registra. Linux nima koncepta registra, zato tudi programov za čiščenje tega ni. Tudi operacijski sistem MacOS nima registra, a je vseeno kar nekaj orodij za ta sistem, ki naj bi znala čistiti register.

Vse, kar počnemo v operacijskem sistemu Windows, se zabeleži v register. To so podatki o odprtih slikah in programih (nameščenih in odstranjenih). Teh podatkov je veliko in tudi niso dobro urejeni, saj jih Windows ne počisti učinkovito, ko opravlja dnevna opravila. Ves čas dodaja nove zapise, vendar le redko pobriše tiste, ki niso več potrebni. Zato postane register poln nesnage, ki upočasni sistem. Zaradi velikosti in kompleksnosti je ročno čiščenje nepraktično in skoraj nemogoče. Veliko orodij ne loči med tipi napak registra in posledično lahko kdaj pobrišejo zelo pomemben ključ, kar povzroči nedejlovanje sistema. Prav tako jih veliko s sabo prinese nadležno programsko opremo. Vseeno se je izkazalo, da previdno in občasno poganjanje teh orodij ne škodi. Mnenja o tem, koliko zares pomaga, so zelo različna [48].

Pri uporabi programov za čiščenje registra veljajo naslednja pravila:

- narediti je treba varnostno kopijo podatkov,
- narediti je treba varnostno kopijo registra,
- pozoren je treba biti in ne klikati naprej,

- pred čiščenjem registrov je treba zapreti vse programe,
- treba je biti previden in upoštevati zgornja pravila.

Vsakodnevno poganjanje programov za čiščenje registra ni priporočljivo. Bolje je redno uporabljati napredna orodja za odstranitev programov, ki iz registra počistijo ostanke odstranjenih programov.

CCleaner Najbolj popularen med uporabniki je CCleaner [8]. Dober je za optimizacijo in čiščenje sistema. Ena redkih slabosti je, da ga je večinoma treba pognati večkrat, da popolnoma počisti register. Vendar je po drugi strani zelo varen, saj ne briše registra preveč globoko in je skoraj nemogoče, da bi sistem po njegovi uporabi postal neuporaben. Je hiter in naj ne bi vseboval zlonamerne programske opreme. Vendar so pred kratkim odkrili, da je spletnim napadalcem uspelo vanj skriti zlonamerno programsko kodo, a so to hitro odkrili in rešili [7].

Comodo System Utilities Še eno orodje, ki zna več kot samo čistiti register, je Comodo System Utilities [19]. To orodje zna iti globlje kot CCleaner, vendar je še vedno med najbolj varnimi. Orodje je brezplačno. Sestavljeno je iz orodja za čiščenje registra, ki temeljito pregleda in popravi register, hkrati ima tudi sistem za preprečevanje napak, ki skrbi, da sistem po zagonu orodja ni pokvarjen.

Jv16 powertools Jv16 powertools [47] je dobro orodje za čiščenje registra. Ima dva načina delovanja. Pri prvem naredimo varnostno kopijo registra in potem pustimo, da orodje vse naredi samo od sebe. Ta način je primeren za začetnike. Bolj izkušeni uporabniki bodo raje uporabljali način, pri katerem lahko izbiramo, kaj bomo popravljali. Vendar je ta način precej zamuden, saj je treba pregledati vsak problem posebej in ni možno videti vseh problemov naenkrat ter izbrati tistih, ki jih želimo. Poleg tega bi bilo dobrodošlo, če bi program pokazal resnost težave.

Wise Registry Cleaner [58, 59] je brezplačno orodje in je namenjeno izključno čiščenju registra. Ima samodejno arhiviranje (angl. *backup*) pred vsakim poganjanjem, zato je možno obnoviti vse pobrisano. Prav tako ima tri načine izvajanja glede na globino iskanja: normalen, varen in globok. Po pregledu sistema ponudi možnost izbire, kaj se bo brisalo in kaj ne. Vključeno ima tudi defragmentacijo registra. Edina resna slabost je ta, da poskuša namestiti dodatne programe.

JetClean Osnovna različica [29] je brezplačna in je namenjena predvsem čiščenju registra. Program je preprost za uporabo, saj omogoča samodejno izdelavo varnostne kopije, je hiter in ima prenosno različico. Žal je nekaj funkcionalnosti na voljo samo v profesionalni različici in pri namestitvi poskuša namestiti še nekaj nepotrebnih dodatnih programov.

2.1.3 Orodja za čiščenje ostankov odstranjenih programov

Revo Uninstaller [49] najprej poizkusi uporabiti vgrajeno orodje za odstranitev programov in če to ne uspe, počisti ostanke. Odstrani tudi datoteke v mapi /tmp, zgodovino brskanja in seznam zadnje odprtih datotek v programskem paketu Office. Ne zna pa počistiti orodnih vrstic in vtičnikov. Prav tako nima nadzora (angl. *monitoring*), ki bi samodejno pomagal pravilno odstraniti program oziroma počistiti ostanke takoj po odstranitvi. Orodje je brezplačno.

Geek Uninstaller [24] je za uporabo boljši od Revo Uninstallerja v tem, da zna pokazati, kje v registru se nahajajo ostanki programa in kje na disku se nahajajo datoteke. Poleg tega je popolnoma prenosen, ne potrebuje namestitve na sistem in ga je možno poganjati iz USB-ja ali kar iz Dropboxa.

IObit Uninstaller Glavna prednost programa IObit Uninstaller [28] je, da

zna počistiti orodne vrstice in vtičnike. Prav tako zna upravljati s programi iz zagona. Hkrati ga ni treba nameščati, ampak deluje s preprostim zagonom.

2.1.4 Orodja za čiščenje prostora na disku

CCleaner Tako kot za čiščenje registra je tudi pri čiščenju prostora na disku največkrat uporabljen CCleaner [8]. Sprostiti zna prostor na disku in omogočiti operacijskemu sistemu Windows hitrejše delovanje in več prostora na disku. Prav tako zna počistiti zgodovino brskanja v brskalniku. Je hiter in ne vsebuje zlonamerne programske opreme [4].

Comodo System Utilities To je še eno orodje, ki zna več kot samo čistiti register [19]. Poleg čiščenja registra je namenjen čiščenju diska, kjer zna med drugim poiskati podvojene datoteke, tudi če nimajo istega imena. Prav tako je namenjen čiščenju zasebnosti in zna poleg zgodovine brskanja počistiti tudi priljubljene strani in zaznamke.

BleachBit [5] Je hiter program za čiščenje diska in varovanje zasebnosti, ki dela v okolju Windows, Linux in MacOS. Sprostiti zna predpomnilnik, piškotke, zgodovino brskanja. Pobrisati zna začasne datoteke, brisati dnevnik in zavreči smeti, za katere niti vemo ne, da jih imamo. Poleg brisanja odvečnih datotek zna tudi pobrisati datoteke tako, da jih ni možno obnoviti. Je tudi odprtokodni program.

Fast Duplicate File Finder [23] Je orodje, specializirano za iskanje podvojenih datotek. Pregledati zna tudi video, glasbo in slike. Primerjave ne dela samo po imenih datotek, ampak tudi na binarni ravni, kar je še posebej pomembno pri multimedijjskih datotekah. Ponudi nam seznam podvojenih datotek in predogled razlik ter možnost ročnega izbora datotek, ki se bodo brisale. Je preprosto, hitro in brezplačno orodje.

2.2 Operacijski sistem MacOS

Vsem operacijskim sistemom je skupno to, da priporočajo izdelavo varnostne kopije in ne neposrednega brisanja. Če se pri operacijskem sistemu MacOS odločimo za brisanje, najprej prestavimo datoteke na zunanji disk, da se preveri delovanje, potem se lahko datoteka pobriše z zunanjega diska. Pri ročnem čiščenju je marsikdo moral ponovno namestiti operacijski sistem, ker je pobrisal nekaj, za kar je menil, da ni potrebno, a se je izkazalo, da je to povzročilo nedelovanje sistema.

Programe v MacOS zapremo na drugačen način kot v operacijskem sistemu Windows. Če kliknemo na rdeči križec v levem zgornjem kotu za zapiranje aplikacije, se ta skrije, ne pa tudi zapre. Aplikacija teče v ozadju. Če imamo naenkrat v ozadju več aplikacij, lahko to povzroči počasnejše delovanje sistema. Odprto aplikacijo zapremo z desnim klikom na ikono aplikacije, ki jo želimo zapreti v sidrišču. Tam izberemo ukaz Quit. Tako počistimo pomnilnik, da ni zaseden z aplikacijami, ki jih ne potrebujemo [33].

Pri vzdrževanju sistema je pomembno tudi to, da so programi posodobljeni.

2.2.1 Tehnike čiščenja

Obstaja precej načinov čiščenja operacijskega sistema MacOS. Ročno čiščenje sistema MacOS je težje in manj uspešno kot čiščenje operacijskega sistema Windows. Priporočilo glede prostega prostora na disku je naslednje: treba je imeti vsaj 30 odstotkov prostora na disku, to je še posebej pomembno za sistemski disk [46].

Začnemo lahko z Activity Monitor, kjer najdemo problematične ali odvečne aplikacije. Podobno kot pri operacijskem sistemu Windows je tudi tukaj treba preveriti, katere aplikacije se samodejno zaženejo ob zagonu sistema in po nepotrebnem zasedajo prostor v pomnilniku. Te je treba zapreti, da se sprost prostor v pomnilniku. MacOS nima orodja za odstranjevanje aplikacij. Na voljo imamo dve možnosti. Če ima aplikacija vgrajene pripomočke za od-

stranjevanje, je najboljša izbira uporaba teh pripomočkov. Druga možnost je, da ikono damo v koš. Potem koš izpraznimo [35]. Sami lahko pobrišemo datoteke, za katere vemo, da jih ne potrebujemo več.

Za kaj več je potrebno znanje. Zelo znana problematična lokacija je mapa prenos, v kateri lahko pobrišemo posnetke diska. Prav tako je dobro redno čistiti lokacijo, kamor se odlagajo odprte priloge iz elektronske pošte.

Priporočljivo je tudi čiščenje predpomnilnika (angl. cache) [16]. MacOS ima tri vrste predpomnilnika: uporabnikov, sistemski in predpomnilnik brskalnika. Vse lahko pobrišemo tudi ročno. Uporabnikov in sistemski predpomnilnik se čistita na enak način, in sicer tako, da se pobriše vsebino map `~/Library/Caches` in `/Library/Caches`. Po tem brisanju je treba počistiti koš, da se dejansko opazi sproščeni prostor na disku. Čiščenje predpomnilnika brskalnika je vezano na brskalnik, saj ima vsak brskalnik svojo lokacijo, kamor shranjuje predpomnilnik.

Za vse navedeno je pomembno redno praznjenje koša.

Fragmentacija diska ni tako velika težava, a prisotna je tudi pri MacOS. Najboljši način za defragmentacijo diska v sistemu MacOS je, da skopiramo vse datoteke na zunanji disk, jih na originalnem pobrišemo, pobrišemo koš in jih skopiramo nazaj na originalni disk.

2.2.2 Orodja za čiščenje

Operacijski sistem MacOS slovi po tem, da ne potrebuje orodij za čiščenje in optimizacijo. Vseeno jih najdemo kar precej. Z njimi lahko poiščemo in odstranimo podvojene datoteke. MacOS poskuša sam odstraniti trenutne datoteke, vseeno pa znajo določena orodja najti in odstraniti še več trenutnih datotek. Odstraniti znajo tudi odvečne jezikovne datoteke, počistiti datoteke, ki so ostale po odstranitvi programov. MacOS nima samo enega koša. Nekatera orodja znajo počistiti vse koše naenkrat. Če uporabljamo vgrajeno aplikacijo za pošiljanje elektronske pošte dlje časa, je velika verjetnost, da se na disku nakopiči veliko prilog iz elektronske pošte, ki zasedajo veliko prostora na disku. Nekatera orodja jih uspešno najdejo in počistijo.

Od Applovega operacijskega sistema Sierra dalje ima operacijski sistem vgrajeno tudi orodje za čiščenje diska Optimized Storage. Za delovanje potrebujemo iCloud, Applovo spletno storitev. Načini čiščenja prostora na disku so [44]:

- shranjevanje v iCloudu: orodje zna delati z datotekami na namizju, z mapo dokumenti in s slikami ter videoposnetki. Najnovejše datoteke in pred kratkim odprte se vedno hranijo na lokalnem disku. Vse ostalo se hrani v iCloudu. S tem se sprostí prostor na lokalnem disku. Tako je možno kadarkoli prenesti potrebne datoteke iz iClouda na lokalni disk;
- optimiziranje shranjevanja: orodje zna samo pobrisati videoposnetke, ki smo jih pogledali. Prav tako obdrži samo najnovejše priloge iz elektronske pošte in pobriše vse ostale, saj se nahajajo na poštnem strežniku. Obe nastavitvi je možno izklopiti;
- avtomatsko čiščenje koša: če je ta možnost vklopljena, se vse, kar je v košu dlje kot 30 dni, samodejno pobriše;
- zmanjšanje šuma: to lahko naredimo tudi ročno z orodjem Finder, vendar je s pomočjo optimiziranega shranjevanja bolj pregledno in hitreje. Orodje nam pokaže seznam velikih datotek, ki jih morda ne potrebujemo več. Nato sami izberemo, kaj bomo dejansko brisali;

Dodatna orodja, ki se najbolj pogosto uporabljajo, so:

- Onyx [42]: brezplačno multifunkcijsko orodje, ki zna preveriti strukturo datotečnega sistema, popraviti pravice za dostop in skrite nastavitve, ki niso prosto dostopne. Pri čiščenju je to orodje učinkovito v čiščenju prepomnilnika in praznjenju koša[43];
- AppZapper [1]: orodje, ki pomaga pri odstranjevanju programov iz sistema. Operacijski sistem MacOS namreč nima metode za odstranjevanje programov. Programe naj bi kar pobrisali iz datotečnega sistema. AppZapper poišče datoteke, povezane s programom, ki ga želimo

odstraniti tudi v mapah, kjer se aplikacija ne nahaja, tako da najde konfiguracijske datoteke, ostanke paketov in pokaže, kaj vse lahko odstranimo. Brezplačno lahko odstranimo samo pet programov, potem je treba orodje plačati [2];

- CleanGenius [12]: ponaša se s hitrostjo. Zna odstraniti uporabnikove osebne trenutne datoteke, datoteke dnevniškega zapisa obdelav, smeti in spletne datoteke. Prav tako zna najti podvojene datoteke glede na vsebino, tudi če je ime datoteke drugačno. Uporabna je tudi funkcionalnost prikaza datotek po velikosti. Zna sprostiti tudi pomnilnik in opozoriti, ko zasedenost pomnilnika doseže nastavljeno mejo [13];
- Monolingual [37]: program, ki lahko sprosti več sto MB prostora na disku. Odstraniti zna vse nepotrebne jezikovne datoteke za jezike, ki jih ne potrebujemo [38];
- CleanMyMac [14]: ena izmed najbolj preprostih in učinkovitih aplikacij za čiščenje sistemov MacOS. Zna prepoznati in izbrisati nepotrebne datoteke. Med te sodijo systemske datoteke, log datoteke, trenutne datoteke, jezikovne datoteke, stare posodobitve programske opreme. Program ima možnost nastavitve urnika. Odkrije tudi pokvarjene inštalacije ali nepopolno pobrisane. Brisanje se lahko izvede neposredno ali na bolj varen način prek koša. Lahko nastavimo tudi urnik. Programska oprema ponuja možnost, da se izognemo polnjenju koša za smeti pri brisanju datotek in varno brisanje občutljivih materialov [15];
- CCleaner: deluje na več operacijskih sistemih, smo ga opisali že pri operacijskem sistemu Windows. Program izvede analizo stanja in pokaže, kaj vse bo počistil in koliko prostora bo prihranil. Lahko se izvede čiščenje vsega ali samo določenih datotek. Ponuja tudi pomoč pri odstranitvi programov in popolno brisanje datotek, teh potem ni več mogoče obnoviti [9];
- DaisyDisk [20]: namenjen iskanju in brisanju velikih datotek. Pro-

gram omogoča prikaz analize celotnega diska, posamezne particije ali celo posamezne mape. Za izbrano možnost pokaže urejene datoteke od največje do najmanjše. Potem lahko samo primemo in odložimo zelene datoteke v koš, iz katerega jih pozneje pobrišemo. Program je zelo hiter in preprost za uporabo [21];

- MacCleanse Mac Application [31]: še eden izmed podobnih programov. Zna vse, kar zna CCleaner, poleg tega ponuja zelo natančen opis, kaj vse se bo brisalo. Prav tako omogoča samodejno brisanje v mapah, ki jih izberemo. Vendar je vprašanje, če te stvari odtehtajo ceno nakupa, saj so drugi podobni programi brezplačni [32].

Zelo dober primer, zakaj je treba biti zelo previden pri takšnih programih, je MacKeeper [34], ki je bil spoznan kot zelo nevaren zlonamerni program. Sporočal naj bi neresnične pomanjkljivosti sistema z namenom plačila njihove odprave.

2.3 Operacijski sistem Linux

Za razliko od Windowsa Linux samodejno počisti vsečasne datoteke. Nismo registrirani v Linuxu, kar še zmanjša potencialne težave. V najslabšem primeru je treba pobrisati nekaj pokvarjenih paketov, ki niso več potrebni, zgodovino brskanja in predpomnilnik.

2.3.1 Tehnike čiščenja

Vendar to ne pomeni, da Linux ne potrebuje občasnega vzdrževanja v obliki čiščenja sistema. Eden izmed glavnih razlogov je sprostitev prostora na disku. Linux med izvajanjem ustvarja podatke v obliki datotek dnevniških zapisov, predpomnilnika in trenutnih datotek. Če imamo disk SSD, je ta običajno manjši, zato je prostor na disku dragocen.

Dodatno lahko prostor na disku sprostimo z brisanjem starih jeder. Postopek brisanja starih jeder je drugačen za vsako distribucijo in ni možno

popolnoma ročno brisanje. V vsakem primeru je potrebno neko orodje. Vendar ni nujno, da je to orodje za čiščenje, lahko uporabimo posebna orodja, ki pridejo skupaj z distribucijo in so namenjena izključno upravljanju z jedrom.

Še en način sprostitve prostora na disku je praznjenje koša za smeti. Včasih kar pozabimo na to, čeprav je dobro po nekaj časa pognati čiščenje koša.

Obstaja tudi način, kako preprečiti pretirano polnjenje diska. Včasih se lahko zgodi, da zaradi napake na sistemu narastejo sistemske datoteke dnevniških zapisov preko vseh obsegov. Zato je pametno omejiti maksimalno velikost teh datotek.

Z naslednjim ukazom poiščemo, kaj zavzema prostor na disku: `sudo du -s -h -x /*`. S tem dobimo skupno količino zavzetega prostora (-s) za vsako datoteko in mapo na najvišji ravni korenskega uporabnika (/*), tako da ne gledamo na drugi datotečni sistem (-x) in izpišemo v obliki, prijazni za ljudi (-h).

Treba je preveriti vsako datoteko, preden jo pobrišemo, a načeloma se sistema zagotovo ne dotaknemo, če brišemo iz naslednjih map:

- /tmp – začasni podatki uporabnika, ki se običajno brišejo samo ob ponovnem zagonu sistema,
- /var/tmp – tiskalnik v ozadju (angl. *print spool*) in drugi sistemski začasni podatki,
- /root – domača mapa korenskega uporabnika,
- /opt – tukaj najdemo večino programske opreme tretjih strank, ki se ne počistijo za sabo,
- /var/log – datoteke dnevniških zapisov.

Ravno zadnje datoteke dnevniških zapisov lahko ob brisanju povzročijo nede-lovanje programa, na katerega se nanašajo, včasih so zaklenjene in brisanje niti ni možno. Zato je rešitev ta, da se namesto brisanja datoteke izvede

praznjenje vsebine datoteke. Prav tako je mnogokrat uporabno, če poiščemo datoteke, ki jih zelo dolgo že nismo spreminjali in jih morda niti ne potrebujemo več. Te poiščemo z naslednjim ukazom: `find /var/log -atime +90 -ls`

Linux navadno uporablja datotečni sistem ext4. Ta načeloma ne potrebuje defragmentacije, ker ext4 dobro skrbi, da ne pride do fragmentacije. Vseeno se lahko v nekaterih primerih zgodi, da do fragmentacije pride. Takšna primera sta, če je disk več kot 80 odstotkov poln in če je na njem veliko velikih datotek.

2.3.2 Upravitelji paketov

Obstaja več upraviteljev paketov, s katerimi upravljamo s programskimi paketi. Tudi naš program bo za delo s paketi uporabil upravitelje paketov.

S pomočjo posodobljanja paketov je upravljanje s programi, ki jih želimo namestiti ali pobrisati, preprosto. Tudi nameščanje vseh programov je mogoče z enim korakom in ni treba večkrat ponovno zagnati računalnika kot v operacijskem sistemu Windows. Prav tako je vse na enem mestu in ni treba iskati programov po spletu. Vsak upravitelj paketov zna upravljati s paketi, poiškati vse dodatne datoteke, ki so potrebne pri posameznem paketu, ponudi pakete v formatu, ki omogoča preprosto namestitev in preverjanje kakovosti paketov.

Upravitelje paketov ločimo glede na vrsto distribucije [45]. Vse distribucije Linux, ki izhajajo iz Debiana, uporabljajo uporabno in močno orodje DPKG. Upravitelj, ki je najbolj popularen, je APT. Znotraj tega upravitelja poznamo naslednje ukaze:

- `apt-get` (install, remove, autoremove, clean, update, upgrade),
- `apt-cache` (search, install, show, depends, rdepends, pkgnames),
- `aptitude` (update, install, remove, clean, purge).

Fedora in CentOS sta si zelo podobni distribuciji, tesno povezani z RHEL. Glavna razlika je v tem, kako so paketi povezani z njihovimi skladišči. CentOS uporablja YUM za interakcijo s sistemskimi skladišči in RPM za interakcijo s posameznimi paketi. Fedora od različice 22 dalje uporablja dnf namesto YUM. Oba v ozadju uporabljata orodje na nižji ravni imenovano RPM. Večinoma ni potrebna uporaba ukazov RPM, nekateri so včasih vseeno uporabni.

Med znanimi upravitelji so še Pacman za Arch Linux, Zypper za Open SUSE in Portage za Gentoo.

2.3.3 Orodja

Odstranjevanje nameščenih programov v Linuxu je precej bolj preprosto kot odstranjevanje pri operacijskem sistemu Windows. Zato včasih orodij za čiščenje sistema Linux ni bilo veliko. Tudi sedaj jih je veliko manj kot za operacijski sistem Windows, vendar se sedaj za Linux najde precej orodij, katerih namen je, uporabniku olajšati delo. Vsa orodja so podobna in uporabljajo že obstoječe sistemske ukaze.

Synaptic [52] je grafični program za pomoč pri upravljanju s paketi APT in RPM. Pomaga pri nameščanju, odstranjevanju in posodobljanju aplikacij.

BleachBit [6] za Linux zna brisati vse tipe datotek. Ponudi nam izbiro, katere tipe datotek bi radi brisali. Pred brisanjem nam ponudi predogled, kaj se bo brisalo, da preprečimo nezaželene posledice. Prav tako omogoča trajno brisanje datotek; tako smo prepričani, da jih ni mogoče obnoviti, če recimo prodamo disk. Dodatno zna prepisati prazen prostor z nepomembnimi podatki, ki jih potem pobriše, da ne ostanejo sledi starih datotek. Tudi brisanje sistemskih datotek lahko preprečimo, če program poženemo kot navaden uporabnik. Ko si tega res želimo, je možno v programu to nastaviti. Dodatni prihranek prostora na disku

je mogoč z brisanjem jezikovnih datotek jezikov, ki jih ne potrebujemo. Poleg tega ga je možno poganjati iz terminala in ne potrebuje grafičnega vmesnika. Ta način uporabe omogoča tudi samodejno izvajanje programa s pomočjo skript v ozadju [57].

GtkOrphan [25] je grafično orodje za brisanje paketov v Ubuntu Linuxu, ki jih ne potrebujemo več. V terminalu sta za to dve orodji. Eno je `Sudo apt-get autoremove`, ki ne omogoča izbire paketov, ki se bodo brisali. Deluje po načelu: vsi ali noben. Podobno orodje je `DebOrphan`, ki omogoča izbiro paketov, ki se bodo brisali. `GtkOrphan` je grafično orodje, ki pokaže pakete, jih razvrsti glede na kategorijo (Ruby, Python ...), prav tako pokaže osnovne informacije o paketu [26].

Cruft [17] je program, ki poišče datoteke, ki niso del nameščenih paketov. Program preveri za vse nameščene pakete, katere datoteke so možne in jih primerja z datotekami na disku. Pokaže, kje se datoteka nahaja, da vemo, ali je del sistemskih map ali je morebiti na primerni mapi `/home` in ni za brisanje. Težava programa je, da težko sledi vsem spremembam na paketih, zato njegovemu poročilu ne smemo slepo zaupati [18].

Computer Janitor [11] je program, ki deluje samo v distribuciji Ubuntu z grafičnim vmesnikom. Prikazati zna seznam nameščenih aplikacij, na vrhu seznama pokaže aplikacije, ki jih najdlje nismo uporabljali. Potem sami izberemo, katere bi radi odstranili. Uporaba je podobna kot pri Windowsovi vgrajeni funkciji za odstranjevanje programov [30].

Stacer je eden najnovejših orodij, napisan za Ubuntu. Ima zelo lep grafični uporabniški vmesnik z ogromno funkcionalnosti z zavijki [50]. Glavne funkcionalnosti so:

- hiter vpogled v stanje sistemskih resursov,
- čistilec sistema(angl. *system cleaner*) za sprostitev prostora na disku z enim klikom,

- nadzor aplikacij, ki se zaženejo ob zagonu in vplivajo na zmogljivost sistema,
- nadzor nad servisi,
- iskanje in odstranitev aplikacij za sprostitev prostora na disku.

Poglavje 3

Predstavitev naše rešitve

V operacijskem sistemu Linux se tako kot v vseh ostalih operacijskih sistemih sčasoma nakopiči veliko začasnih datotek spletnih brskalnikov, nepravilno zaprtih programov in podobnega. Osnovna ideja aplikacije je bila, olajšati sistemskemu administratorju enega od korakov pri vzdrževanju operacijskega sistema Linux, to je čiščenje diska. To smo dosegli, prav tako se je izkazalo, da je aplikacija uporabna tudi za običajne uporabnike, saj ponudi uporabne priporočljive stvari, za katere običajni uporabniki niti ne vedo ali imajo premalo znanja iz ukazov, ki so uporabljeni za doseglo posameznih funkcionalnosti. Vsekakor je ena izmed prednosti aplikacije ta, da se izvaja v terminalu in deluje v vseh tipih (minimalna namestitev, strežnik, GNOME namizje, itd.) namestitve operacijskega sistema Linux.

3.1 Podprte funkcionalnosti

Funkcionalnosti aplikacije so razdeljene na analizo sistema, čiščenje sistema (brisanje ali prestavitev datotek), razveljavitev in poročilo. Vse funkcionalnosti so povezane s čiščenjem prostora na disku, kar lahko privede do boljšega delovanja sistema. Dodatno lahko na ta način preprečimo pomanjkanje prostora na disku, saj je lahko pomanjkanje prostora vzrok za nedelovanje sistema.

Analiza sistema

Pred čiščenjem moramo analizirati sistem in identificirati možnosti za izboljšanje delovanja sistema. Analizo sestavljajo funkcionalnosti, ki jih navajamo v nadaljevanju.

Najprej poiščemo začasne datoteke. Običajno jih najdemo v mapi /tmp, ki jo lahko nastavimo, da se vsebina briše samodejno. Aplikacija poišče začasne datoteke še v mapi, ki jo izbere uporabnik. Išče datoteke, katerih ime vsebuje .swp ali .bak ali poseben znak ~.

Dnevniške datoteke so problematične za sistem, saj lahko ob težavah v delovanju operacijskega sistema hitro narastejo. Vsekakor je ena od rešitev, da jim omejimo velikost. Naša aplikacija poišče stare dnevniške datoteke v mapi /var/log.

Sledi identifikacija neuporabljenih, osirotelih paketov in starih jeder. Eden izmed razlogov za osirotele pakete je lahko nadgradnja na novejšo različico, uporaba drugega paketa (na primer postfixa namesto sendmaila) ali pa smo paket pri namestitvi izbrali, čeprav ga sploh ne potrebujemo (npr. program za delo z bluetoothom na sistemu brez bluetootha). Paket je lahko nameščen kot odvisnost pri namestitvi drugega programa, ki po njegovi odstranitvi še vedno ostane v sistemu. Moramo pa biti previdni, saj lahko odvisne pakete potrebuje tudi drug program. Naša aplikacija poskrbi, da se odstranijo samo osiroteli, resnično nepotrebni paketi. Pri starih jedrih pokaže vse različice jedra.

Naslednji korak pri analizi sistema je iskanje podvojenih datotek. Večina uporabnikov iz več virov odlaga slike, videoposnetke, dokumente in glasbo na en računalnik. S tem se velikokrat veliko datotek podvojuje, kar lahko zasede veliko prostora. V tem primeru jih je najbolje poiskati in ohraniti le eno kopijo. Naša aplikacij naredi ravno to: poišče in prikaže vse podvojene datoteke v izbrani mapi, nato lahko uporabnik sam izbere, kaj naj se briše in kaj naj se ohrani.

Sledi iskanje neuporabljenih datotek. Če imamo velike količine podatkov, je med njimi zagotovo precej takšnih, ki jih ne potrebujemo več in niti ne

vemo, da so še vedno na disku. Z aplikacijo lahko iščemo po vsem disku ali samo v določenih mapah. Iščemo lahko glede na datum zadnje uporabe ali glede na velikost datotek.

Iskanje neuporabljenih datotek, prenešenih ob nameščanju paketov s programom yum. Yum samodejno briše prenesene datoteke, ko niso več potrebne, če je operacija uspešna. S tem je optimizirana poraba prostora na disku. Če pa smo velikokrat brez spletne povezave, je možno vklopiti predpomnilnik yum in imamo datoteke na voljo v mapi yum. Tudi ta predpomnilnik je treba čistiti, kar naša aplikacija omogoča.

Iskanje neuporabljenih jezikovnih paketov je zadnja funkcionalnost. Ob namestitvi operacijskega sistema Linux ponavadi nismo pozorni na število privzeto nameščenih jezikovnih paketov. Lahko se tudi zgodi, da uporabnik potrebuje en dodaten jezikovni paket, in namesto enega namesti vse možne. Če pobrišemo odvečne in obdržimo samo tiste, ki jih res potrebujemo, lahko prihranimo nekaj prostora na disku. Aplikacija pokaže trenutno izbrani jezikovni paket in seznam vseh, ki so nameščeni. Nato lahko izberemo, katere bomo obdržali, ostali se odstranijo, lahko pa izberemo obratno.

Čiščenje sistema

Pri tem ločimo funkcionalnosti, pri katerih so zadaj ukazi GNU in kjer se uporabijo ukazi, ki so lahko v različnih distribucijah drugačni. Za funkcionalnosti, ko se išče datoteke na datotečnem sistemu, se pri analizi za vsako funkcionalnost ustvari svoja datoteka. V njej je zapisan seznam datotek, ki smo jih za posamezno funkcionalnost našli. Uporabnik lahko vsebino te datoteke pogleda in jo po potrebi uredi. Lahko se namreč odloči, da neke datoteke ne bo brisal, in jo odstrani s seznama. Ko izbere brisanje, aplikacija odstrani datoteke na seznamu. Datoteka pri večini funkcionalnosti v resnici ni dokončno pobrisana, ker se zaradi možnosti razveljavitve prenese v mapo /tmp/trash, ki jo aplikacija kreira, če ne obstaja. Odstranjena pa je iz originalne lokacije. Za funkcionalnost brisanja osirotelih paketov, starih jeder, jezikovnih paketov in datotek predpomnilnika yum je možno samo brisanje

in ne tudi prestavljanje datotek v poljubno mapo in razveljavitev brisanja.

Uporabnik ima pri funkcionalnosti iskanja datotek na datotečnem sistemu namesto brisanja možnost izbire, da bo datoteke raje prestavil v poljubno mapo. V tem primeru razveljavitev ni možna, zato mora uporabnik sam obnoviti datoteke iz izbrane mape, kamor se datoteke prestavijo.

Razveljavitev

Če uporabnik ugotovi, da je katera od funkcionalnosti odstranila datoteke, ki jih uporabnik ali sistem vseeno potrebuje, se lahko odloči za razveljavitev. Na osnovi seznama, ki se je uporabil za brisanje, se datoteke iz mape `/tmp/trash` prestavijo nazaj v originalno mapo. To ne velja za brisanje osirotelih paketov, starih jeder, jezikovnih paketov in predpomnilnika yum, pri katerih se uporabijo posebni ukazi. Tukaj nimamo točnega seznama datotek, ki se bodo brisale, da bi jih lahko potem obnovili. Možno je ročno ponovno namestiti programske ali jezikovne pakete, če bi jih potrebovali.

Poročilo

Omogoča prikaz prihranka prostora za vsako posamezno funkcionalnost posebej in skupni prihranek prostora.

3.2 Zasnova aplikacije

Kot smo že omenili, je bila osnovna ideja aplikacije, olajšati delo sistemskemu administratorju. Običajno se za naloge sistema administratorja uporablja lupina bash. Najpogostejša Linuxova lupina je bash. Bash je splošni skriptni jezik in je odličen za delo z datotečnim sistemom. Dodaten razlog za uporabo lupine bash je, da je del vsake distribucije Linuxa, zato ni treba nameščati kakršnihkoli orodij, kot je to treba pri perlu, pythonu, lupini C (angl. *C shell*), lupini Z (angl. *Z shell*) itd. Z uporabo lupine bash se poveča možnost prenosljivosti skripte med strežniki in celo med operacijskimi sistemi.

Naša aplikacija je skupek skript bash, s katerimi avtomatiziramo naloge ukazne vrstice z uporabo istega jezika kot pri ročnem izvajanju ukazov. Pogajanje istih ukazov vsak dan je lahko zamudno. Bolje je napisati skripto bash z osnovnimi konstrukti programskega jezika. Na ta način lahko neposredno uporabimo vsa možna orodja, ki jih imamo v ukazni vrstici. S katerikoli drugim jezikom bi morali najprej izvršiti ta ukaz in preusmeriti rezultat, kar bi bilo bolj zapleteno.

Prvo različico aplikacije smo naredili brez posebnega načrtovanja in vse skupaj v eni kompleksni skripti bash z velikim številom neurejenih funkcij. Kmalu se je pokazalo, da bo brez načrta oziroma strukture projekta, vse skupaj postalo preveč nepregledno in neobvladljivo. Pri dodajanju vsake nove funkcionalnosti smo izgubili ogromno časa, rešitve so postale neoptimalne, nepreglednost je naraščala.

Zato smo problem razbili na module. Za vsako funkcionalnost smo naredili svoj modul (svoja skripta bash). Vsaka funkcionalnost v grobem predstavlja eno izmed specifičnih čistilnih tehnik. Prav tako smo združili funkcije glede na namen (branje vhoda, prikaz menija itd.). Z združevanjem funkcij smo ustvarili nekakšne knjižnice. Te so omogočile čelno-zaledno (angl. *front-end*, *back-end*) zasnovi.

Uporabniški vmesnik skrbi za interakcijo z uporabnikom, ki prek menijev izbere zelene operacije in parametre. Za tekstovni uporabniški vmesnik v terminalu smo se odločili zato, da aplikacija deluje v vseh tipih namestitve in v vseh distribucijah Linuxa, brez namestitve kakršnih koli dodatnih paketov. Prav tako se kompleksne naloge izvršijo hitreje kot v grafičnem vmesniku in aplikacija je bolj prilagodljiva. Ti meniji so predvideni predvsem za običajne uporabnike. Izvedbo operacij uporabniški vmesnik doseže s klicem knjižnic z izbranimi parametri. To je dodatna prednost aplikacije, saj takšna zasnova omogoča preprosto menjavo tekstovnega uporabniškega vmesnika z grafičnim uporabniškim vmesnikom. Tukaj namesto prek menijev v grafičnem vmesniku izberemo zelene operacije in parametre ter pokličemo ustrezno knjižnico oziroma modul, ki izvede zelene operacije. Sistemski admi-

nistrator lahko uporabi menije ali namesto njih sam pokliče ustrezne knjižnice oziroma module. Lahko si celo nastavi opravila, ki se samodejno izvedejo enkrat tedensko.

3.3 Podprto okolje

Aplikacijo je mogoče izvesti na katerikoli distribuciji operacijskega sistema Linux, vendar za vsako ne omogoča vseh funkcionalnosti. Vsekakor je ena izmed prednosti ta, da za delovanje ne potrebuje nobenih dodatnih programov ali paketov. Uporablja namreč samo ukaze programske opreme GNU, ki jo uporablja operacijski sistem Linux, in posledično vse njegove distribucije. Za delovanje je dovolj že terminal in s tem so podprte tudi strežniške izvedbe. Aplikacijo smo razvijali najprej za distribucijo CentOS, zato v tej distribuciji delujejo vse funkcionalnosti. Funkcije, ki uporabljajo za izvedbo operacij ukaze programske opreme GNU delujejo v vseh distribucijah. Pri čiščenju predpomnilnika, osirotelih paketov, starih jeder in jezikovnih paketov pa so zadaj za vsako distribucijo drugačni ukazi. Pri izbiri ene izmed teh funkcionalnosti aplikacija javi, ali je ta funkcionalnost podprta, saj zazna in izpiše vrsto distribucije.

3.4 Uporabniški vmesnik

Uporabniški vmesniki v aplikacijah so namenjeni čimbolj preprostemu in preglednemu prenašanju informacij od aplikacije do uporabnika in obratno. Naš tekstovni uporabniški vmesnik zagotovo izpolnjuje kriterij za preprost in pregleden vnos informacij od uporabnika do aplikacije. Prikaz rezultatov operacij pa bi bil v grafičnem vmesniku lahko lepši, vendar je za naše potrebe in želje vsekakor ustrezen tekstovni vmesnik. Uporabnik pokliče aplikacijo kot vsako drugo skripto bash. Odpre se mu tekstovni uporabniški vmesnik, ki izriše osnovni meni.

3.4.1 Meniji

Naš uporabniški vmesnik je sestavljen iz več menijev v treh ravneh. Iz osnovnega menija, v katerem izberemo eno od osnovnih funkcionalnosti in še ne ponuja nobene končne akcije, pridemo na drugo raven, kjer lahko za nekatere operacije že izberemo končno zeleno akcijo. Pri nekaterih podmenijih se moramo pomakniti še raven nižje in tam izvesti zeleno akcijo. V nekaterih menijih, v katerih izvajamo končne akcije, uporabniku ponudimo še izbiro mape, nad katero želi, da se izbrana operacija izvede.

```
Option 1 Picked - Analyze
*****
**  1) Temporary files
**  2) Old log files
**  3) Packages
**  4) Old files
**  5) Empty files
**  6) Duplicate files
**  7) Yum cache
**  8) Large files
**  9) Language packs
** 100) Return to menu
*****
Please enter a menu option and enter or q to exit..
█
```

Slika 3.1: Izgled menija: Analiza sistema

3.4.2 Navigacija med meniji

Vsak meni ponudbi več akcij, ki jih lahko uporabnik izvede. Pred vsako izbiro je zaporedna številka akcije. Uporabnik prek tipkovnice vnese zaporedno številko akcije, ki bi jo rad izvedel, in s pritiskom na tipko za potrditev (angl. *enter*) to akcijo izvede. Premika se lahko med tremi ravnmi menijev. Ko izvede določeno akcijo in se ta zaključi, se ponovno pokaže isti meni. V

menijih druge in tretje ravni lahko uporabnik izvede končno akcijo ali se vrne eno raven višje oziroma z vnosom črke q in s pritiskom na tipko potrditev zapusti aplikacijo.

3.4.3 Interakcija z uporabnikom

Interakcija uporabnika z aplikacijo poteka z uporabo menijev, v obratni smeri pa prek zaslona in vmesnih datotek. Pri vsaki operaciji, ki jo uporabnik izvede, se rezultat operacije pokaže na zaslonu. Pri nas gre za tekstovni uporabniški vmesnik, ki se izvaja v terminalu. Operacija lahko kot rezultat vrne veliko podatkov in v tem primeru je prikaz na zaslonu v terminalu slabo pregleden, posledično je sprehajanje med vrsticami in iskanje informacij oteženo. Zato aplikacija rezultat vrne tudi v obliki vmesne datoteke, kamor se zapiše isti rezultat operacije kot na zaslonu. Uporabnik se lahko v poljubnem urejevalniku besedil sprehaja med zadetki in išče želene informacije. Uporabnik ima na voljo tudi enega izmed menijev, ob izbiri se mu na zaslonu izpiše poročilo o prihranku prostora za posamezne operacije.

3.5 Dnevniški zapisi in poročila

3.5.1 Dnevniški zapisi

V računalništvu je dnevniški zapis datoteka za beleženje dogodkov, ki se pojavijo v operacijskem sistemu ali na drugi programski opremi, ali sporočila med različnimi uporabniki komunikacijske programske opreme. V našem primeru gre za beleženje vseh operacij, ki jih uporabnik izvede v aplikaciji. Vsa sporočila so zapisana v eni datoteki. Ob izvedbi vsake operacije se v datoteko zapišejo čas izvedbe, izbrana operacija in ukaz, ki se je izvedel. Primer izgleda nekaj zapisov v dnevniški datoteki:

```
Sun Jan 21 21:40:51 CET 2018;Analyze - Flash, Macromediaobjects;
find /root/.macromedia/* /root/.adobe/* -type f | xargs du -c
Sun Jan 21 21:40:58 CET 2018;Analyze - temporary files;
find / -type f -name *~
Sun Jan 21 21:41:01 CET 2018;Analyze - old log files;
find /var/log/*.gz -type f -print
Sun Jan 21 21:41:24 CET 2018;Analyze - oprhan packages;
package-cleanup --quiet --leaves --exclude-bin
Sun Jan 21 21:57:51 CET 2018;Analyze - old files 30 days;
find /test2 -mtime +30 -type f
Mon Jan 22 06:59:30 CET 2018;Delete - temporary files;
mv /run/log/journal/1643f1/system@000562.journal~ /tmp/trash/cleanup_rm_tmp_all/
Mon Jan 22 06:59:36 CET 2018;Delete - temporary files;
mv /run/log/journal/1643f1/system@c9fe8e.journal~ /tmp/trash/cleanup_rm_tmp_all/
Mon Jan 22 06:59:41 CET 2018;Delete - temporary files;
mv /run/log/journal/1643f1/system@000562.journal~ /tmp/trash/cleanup_rm_tmp_all/
Mon Jan 22 07:09:13 CET 2018;Move - temporary files;
mv /run/log/journal/bc89e4/system@78a0637.journal~ /move_folder/cleanup_rm_tmp_all/
Mon Jan 22 07:11:22 CET 2018;Move - temporary files;
mv /run/log/journal/bc89e4/system@5608b20.journal~ /move_folder/cleanup_rm_tmp_all/
Mon Jan 22 07:12:42 CET 2018;Move - temporary files;
mv /run/log/journal/bc89e4/system@923714b.journal~ /move_folder/cleanup_rm_tmp_all/
```

Slika 3.2: Primer zapisa: Dnevniški zapis

3.5.2 Poročila

Aplikacija ima med drugim sistem poročanja. Ena izmed izbir na prvem meniju je poročilo. Ob izvajanju vsake izmed analiz sistema, se zapišejo informacije o prihrankih. Poročilo za vse izvedene analize na zaslonu pokaže, koliko datotek bi se izbrisalo in koliko prostora na disku bi se sprostilo ob dejanski izvedbi operacij. Na koncu prikaže še prihranek prostora na disku za vse funkcionalnosti skupaj v primeru brisanja. Poleg izpisa na zaslonu se poročilo zapiše tudi v datoteko. Omogoča lažjo odločitev pri izbiri funkcionalnosti brisanja, ko potrebujemo dodaten prostor, a ne bi radi brisali vsega, ne spomnimo pa se natančno, kakšni so bili posamezni prihranki za vsako izmed analiz sistema. Primer izgleda za en testni sistem:

3.6 Podrobnosti izvedbe

Kot smo že omenili, je osnovna ideja aplikacije ta, da sistemskemu administratorju poenostavimo upravljanje rednih preprostih, a pomembnih opra-

```
Option 4 Picked - Report
Temporary files - Flash, Macromediaobjects:      6      24
Temporary files - Desired folder:                1       4
Old log files:                                   2       8
Older than 30 days:                              4      16
Older than 90 days:                              3      12
Older than 180 days:                             2       8
Older than 365 days:                             1       4
Empty files:                                     1       0
Duplicate files:                                 5      16
Yum cache:                                       19    1296.9
Large files:                                     12    1560
Large folders:                                   5     2000
Large files and folders:                          4     100
```

Slika 3.3: Izgled poročila

vil, združenih v en paket, in da deluje na večini distribucij. Uporabili smo osnovne ukaze Linuxa brez namestitve dodatnih orodij in za izvajanje teh je bila primerna uporaba skript bash.

Analiza sistema

Pri analizi sistema se v ozadju izvedejo ukazi programske opreme GNU in posebni ukazi za upravljanje s programskimi paketi, jedri in jezikovnimi paketi. Osnovne funkcionalnosti lahko razdelimo glede na to, kateri ukaz uporabimo.

Funkcionalnosti, ki uporabljajo ukaze programske opreme GNU, so navedene v nadaljevanju.

Uporaba ukaza find Ukaz find je resnično močan za iskanje po datotečnem sistemu. Uporabili smo ga za iskanje začasnih datotek: iskali smo po različnih delih imena in z uporabo metaznakov. Stare dnevniške datoteke poiščemo hkrati po imenu in lokaciji, saj vemo, da se nahajajo v mapi /var/log. Z datumom zadnje uporabe smo si pomagali pri iskanju starih datotek. Za iskanje praznih datotek smo uporabili parameter -size ukaza find, ki smo ga nastavili na 0. V večini primerov smo uporabili parameter ukaza za iskanje v globino. Vsaka funkcionalnost ima svojo funkcijo z največ tremi vhodnimi parametri, od katerih sta dva

obvezna. Prvi je pot in ime za datoteko, kamor se vpiše rezultat iskanja pri analizi in se uporabi za izpis rezultata na zaslonu. Drugi je pot in ime datoteke, v katero se zapiše seznam datotek, ki smo jih našli glede na izbrano operacijo. Zadnji parameter je odvisen od operacije in pove, kje se bo iskalo datoteke, če pot ni vnaprej določena (npr. /var/log za dnevniške datoteke). Kompleksnejši je primer uporabe iskanja podvojenih datotek. Ukaz, ki smo ga pri tem uporabili, lahko vidimo v kodi 3.1:

Koda 3.1: Iskanje podvojenih datotek.

```
$ find path -not -empty -type f -printf "%s\n" |  
  sort -rn | uniq -d | xargs -I{} -n1 find path -  
  type f -size {}c -print0 | xargs -0 md5sum |  
  sort | uniq -w32 --all-repeated=separate | cut -  
  f3 -100 -d ' ' | xargs du -c
```

Najprej poiščemo neprazne datoteke, izpišemo njihove velikosti, jih uredimo in pokažemo samo podvojene. Dve enako veliki datoteki nista nujno enaki po vsebini, vseeno pa najprej omejimo izbor s tem filtrom, saj je iskanje po velikosti veliko hitrejše kot iskanje po vsebini. V drugem delu dobimo kot vhod datoteke z enakimi velikostmi, kjer nato z metodo MD5 poiščemo enake. Zgoščevalna funkcija (hash function) je algoritem, ki dobi kot vhod poljubno dolgo sporočilo, kot izhod pa vrne fiksno dolgo binarno vrednost (hash value). Uporabljamo jo za preoblikovanje poljubno dolgih vhodnih sporočil v izhodne vrednosti dolžine 128 bitov. Pomembnost te funkcije je, da je nepovratna ali enosmerna: to pomeni, da je nemogoče najti vhodno sporočilo, če poznamo izhodno vrednost. Temu, da dvema različnima vhodnima datotekama ustreza enak povzetek, rečemo kolizija. Jasno je, da za vsako zgoščevalno funkcijo obstajajo kolizije, ker imamo neskončno mnogo možnih vhodov, ki jih preslikamo v končno mnogo izhodov. Zgoščevalna funkcija MD5 je v našem primeru dovolj dobra, ker zagotovo nimamo več kot 2^{64} enako

velikih datotek[36] in je hitrejša kot novejšje zgoščevalne funkcije SHA1, SHA2 idr. Na koncu le še uredimo izpis in dodamo velikosti datotek in skupen prihranek.

du S preprostim ukazom 3.2 poiščemo velikosti izbrane mape in vseh njenih podmap, urejeno po velikosti v obratnem vrstnem redu. Potem lahko za velike mape pogledamo in po velikosti uredimo datoteke z ukazom 3.3.

Koda 3.2: Izpis velikosti map.

```
$ du -k path --max-depth=1 | sort -nr
```

Koda 3.3: Izpis velikosti datotek

```
$ du -a /test2 --max-depth=1 | sort -n
```

Na voljo imamo še ukaze, ki niso del programske opreme GNU. Uporabili smo jih, ker zeleno z ukazi GNU ne moremo doseči, a so funkcionalnosti za čiščenje sistema pomembne.

Package-cleanup Ta ukaz z različnimi parametri lahko uporabimo za iskanje in brisanje osirotelih paketov in starih jeder.

yum clean Pri analizi sistema pogledamo velikost mape `/var/cache/yum`, da ugotovimo, koliko prostora lahko prihranimo z brisanjem datotek, ki so ostanki nameščanja paketov.

locale Pri analizi z ukazom `locale` pokažemo izbrani jezik in tudi vse ostale nameščene jezike v sistemu.

Vračanje rezultata analize

Tukaj bomo pojasnili, zakaj smo se odločili za zapis rezultata operacij v datoteke in ne neposredno na zaslone. Bash nam ponuja več možnosti za vračanje vrednosti. Prvi način je vračanje izhodnega statusa (0 je uspeh, vse ostalo neuspeh). Drugi način je izpis rezultata na standardni izhod. Tretji je klic funkcije, ki sprejme ime spremenljivke kot del ukaza in jo nastavi

na vrednost funkcije [3]. Za vračanje rezultata funkcij bi lahko uporabili katerega od teh načinov, vendar smo se odločili za uporabo vmesnih datotek zaradi želje po čim bolj preprosti zamenjavi uporabniškega vmesnika. Vse, kar mora uporabniški vmesnik narediti, je, da pokliče funkcijo, ki jo želi izvesti, nato pa rezultat preprosto izpiše iz datoteke. Drugi razlog je ta, da je lahko rezultat funkcije zelo obsežen (veliko število datotek) in je posledično izpis v terminalu precej nepregleden. Tudi iskati in premikati se med rezultati v terminalu je težko. V našem primeru pa ima uporabnik možnost odpreti datoteko v urejevalniku besedil in preprosto analizirati rezultat. Tretji razlog pa je, da mu je rezultat določene operacije na voljo tudi po izvajanju kakšne druge analize, ko rezultati niso več na zaslону. Rezultat ostane namreč zapisan v datoteki, dokler ponovno ne izvede enake analize. Takrat se prepiše z novim rezultatom. Vsaka funkcionalnost pri analizi sistema ima namreč svojo datoteko, v katero se zapisujejo rezultati analize. V tej datoteki je zapisan seznam datotek s potmi, ki ustrezajo pogojem izbrane operacije. Zraven so zapisane velikosti datotek in na koncu seštevek vseh velikosti, to je skupni prihranek prostora na disku v primeru brisanja.

Brisanje, prestavitev in razveljavitev

Naslednji trije sklopi funkcionalnosti so brisanje, prestavitev in razveljavitev. Brisanje in razveljavitev sta obratno sorazmerna. Datoteke, ki jih pobrišemo, z razveljavitvijo lahko dobimo nazaj. To smo dosegli tako, da se ob izvajanju analize za vsako operacijo naredi svoja datoteka s seznamom datotek, ki ustrezajo kriterijem izbrane operacije. Uporabnik lahko datoteko odpre in jo poljubno ureja. Na ta način lahko dodatno izbere, katere od datotek se bodo brisale, ostale pa odstrani iz datoteke. Pri brisanju se sprehodimo čez vse datoteke in jih zaradi možnosti razveljavitve prestavimo v mapo `/tmp/trash`. Če se uporabnik odloči za prestavitev namesto brisanja, se datoteke iz seznama prestavijo v mapo, ki jo uporabnik izbere prek uporabniškega vmesnika. Lahko izbere tudi čiščenje mape `/tmp/trash`, vendar šele ko je prepričan, da ne potrebuje nobene datoteke, izbrane za brisanje. Namreč po

brisanju vsebine mape `/tmp/trash` razveljavitev ni več mogoča. Na to uporabnika tudi opozorimo. Namestitvene pakete pobrišemo s pomočjo ukaza `yum clean all`, jezikovne pakete pa z ukazom `localedef`.

Poglavje 4

Sklepne ugotovitve

Diplomskega dela smo se lotili z namenom izdelave aplikacije za čiščenje operacijskega sistema Linux. Predstavili smo tehnike čiščenja operacijskih sistemov Linux, Windows in MacOS. V povezavi s tehnikami čiščenja smo naredili pregled funkcionalnosti orodij, ki že obstajajo na trgu. Zajeli smo orodja za vsak operacijski sistem posebej in naredili primerjavo med posameznimi orodji. Na spletu je že kar nekaj orodij za čiščenje operacijskega sistema Linux. Njihova težava je, da so običajno omejeni na določene distribucije, imajo premalo funkcionalnosti ali pa s sabo prinašajo nezaželeno zlonamerno programsko opremo.

Ena izmed prednosti naše rešitve je prav gotovo preprostost. Prek preprostega uporabniškega vmesnika omogoča običajnemu uporabniku izvedbo večine opravil, povezanih z vzdrževanjem operacijskega sistema in s prostorom na disku, ki jih običajno izvaja sistemski administrator. Ker v ozadju uporablja osnovne ukaze Linuxa, je primerna za večino distribucij Linuxa, po vsej verjetnosti bo delovala tudi v prihajajočih distribucijah. Narejena je kot skripta bash, kjer lahko vsakdo vidi, da ne vsebuje zlonamerne programske kode. Zaradi modularne zasnove, preglednosti in preprostosti je aplikacijo z minimalnim trudom mogoče prilagoditi lastnim potrebam. Prav tako je dodajanje novih funkcionalnosti preprosto. Zaradi čelno-zaledne zasnove je preprosta tudi zamenjava uporabniškega vmesnika. Uporabnik lahko name-

sto uporabniškega vmesnika uporabi kar razporejevalnik opravil in si nastavi izvajanje samo določenih opravil, ki jih aplikacija ponuja, ob poljubnem času.

Dodobra smo raziskali področje in s preprostimi ukazi ter s preprosto uporabo pokrili potrebe po čiščenju diska v operacijskem sistemu Linux. Tako smo pripravili osnovo za prilagajanje aplikacije prihajajočim spremembam in željam uporabnikov. Diplomski izdelek si lahko ogledate na naslovu [41].

Literatura

- [1] AppZapper. Dosegljivo: <https://en.wikipedia.org/wiki/AppZapper>, 2016. [Dostopano: 12. 3. 2017].
- [2] AppZapper. Dosegljivo: <http://macreview.com/2012/10/appzapper-review/>, 2012. [Dostopano: 12. 3. 2017].
- [3] Bash Funkcije. Dosegljivo: <http://www.linuxjournal.com/content/return-values-bash-functions>, 2009. [Dostopano: 2. 1. 2018].
- [4] Andrew Bettany. Windows File System Troubleshooting, 2015.
- [5] BleachBit. Dosegljivo: <https://en.wikipedia.org/wiki/BleachBit>, 2017. [Dostopano: 10. 3. 2017].
- [6] BleachBit for Linux. Dosegljivo: <https://www.howtogeek.com/116971/7-tips-to-get-the-most-out-of-bleachbit-a-c-cleaner-for-linux/>, 2012. [Dostopano: 19. 3. 2017].
- [7] CCleaner vdor. Dosegljivo: <https://www.racunalniskenovice.com/novice/dogodki-in-obvestila/ce-imate-program-c-cleaner-ga-takoj-posodobite-sicer-vam-bo-zal.html>, 2017. [Dostopano: 01. 10. 2017].
- [8] CCleaner. Dosegljivo: <https://en.wikipedia.org/wiki/CCleaner>, 2017. [Dostopano: 10. 3. 2017].
- [9] CCleaner. Dosegljivo: <http://www.pcmag.com/article2/0,2817,2400840,00.asp>, 2012. [Dostopano: 12. 3. 2017].

-
- [10] Linux praznuje četrto stoletja! Kje je pravzaprav danes? Dosegljivo: <http://www.racunalniske-novice.com/novice/dogodki-in-obvestila/linux-praznuje-cetrto-stoletja-kje-je-pravzaprav-danes.html>, 2016. [Dostopano: 6. 3. 2017].
- [11] Computer Janitor. Dosegljivo: <http://www.linuxlinks.com/article/20110101011003402/ComputerJanitor.html>, 2015. [Dostopano: 12. 3. 2017].
- [12] CleanGenius. Dosegljivo: <https://en.wikipedia.org/wiki/CleanGenius>, 2016. [Dostopano: 12. 3. 2017].
- [13] CleanGenius. Dosegljivo: <https://itunes.apple.com/us/app/cleangenius-pro/id486153141?mt=12>, 2012. [Dostopano: 18. 3. 2017].
- [14] Clean My Mac. Dosegljivo: <https://en.wikipedia.org/wiki/CleanMyMac>, 2017. [Dostopano 18.3.2017].
- [15] Clean My Mac. Dosegljivo: <https://www.softwarehow.com/cleanmymac-review/>, 2016. [Dostopano 18.3.2017].
- [16] How to Clear Cache on Mac OS X. Dosegljivo: <https://macpaw.com/how-to/clear-cache-on-mac>, 2015. [Dostopano: 6. 3. 2017].
- [17] Cruft. Dosegljivo: <https://wiki.debian.org/Cruft>, 2014. [Dostopano: 12. 3. 2017].
- [18] Cruft. Dosegljivo: <https://raphaelhertzog.com/2011/02/28/debian-cleanup-tip-5-identify-cruft-that-can-be-removed-from-your-debian-system/>, 2011. [Dostopano: 19. 3. 2017].
- [19] Comodo System Utilities. Dosegljivo: https://en.wikipedia.org/wiki/Comodo_System_Uilities, 2017. [Dostopano: 10. 3. 2017].
- [20] DaisyDisk. Dosegljivo: <https://en.wikipedia.org/wiki/DaisyDisk>, 2017. [Dostopano: 12. 3. 2017].

-
- [21] DaisyDisk. Dosegljivo: <http://www.macworld.com/article/2996212/storage/daisydisk-4-review-now-faster-than-ever-at-cleaning-out-unwanted-files-on-your-mac.html>, 2015. [Dostopano: 18. 3. 2017].
- [22] Defragmentacija. Dosegljivo: <https://sl.wikipedia.org/wiki/Defragmentacija/>, 2016. [Dostopano: 22. 10. 2017].
- [23] Fast Duplicate File Finder. Dosegljivo: https://en.wikipedia.org/wiki/Fast_Duplicate_File_Finder, 2016. [Dostopano: 10. 3. 2017].
- [24] Geek Uninstaller. Dosegljivo: <http://www.makeuseof.com/tag/geekuninstaller-uninstall-programs-thought-windows/>, 2012. [Dostopano: 10. 3. 2017].
- [25] gtkorphan. Dosegljivo: <http://www.hecticgeek.com/2011/12/gui-tool-to-remove-unused-packages-ubuntu-linux/>, 2011. [Dostopano: 12. 3. 2017].
- [26] gtkorphan. Dosegljivo: <http://www.junauza.com/2012/06/best-system-cleaning-software-for-linux.html>, 2012. [Dostopano: 19. 3. 2017].
- [27] How to Clean Your Computer System. Dosegljivo: <http://www.wikihow.com/Clean-Your-Computer-System>, 2016. [Dostopano: 6. 3. 2017].
- [28] IObit Uninstaller. Dosegljivo: https://en.wikipedia.org/wiki/IObit_Uninstaller, 2016. [Dostopano: 10. 3. 2017].
- [29] JetClean. Dosegljivo: <http://jetclean.software.informer.com/>, 2018. [Dostopano: 10. 3. 2017].
- [30] Neal Krawetz. Ubuntu Powerful Hacks and Customizations, 2010.

-
- [31] MacCleanse. Dosegljivo: <https://thetechreviewer.com/software/maccleanse-mac-application-review/>, 2013. [Dostopano: 12. 3. 2017].
- [32] MacCleanse. Dosegljivo: <https://itunes.apple.com/us/app/maccleanse/id415970551?mt=12>, 2016. [Dostopano: 18. 3. 2017].
- [33] Mac close app. Dosegljivo: https://support.apple.com/kb/PH25174?viewlocale=en_US&locale=en_US, 2017. [Dostopano: 22. 8. 2017].
- [34] MacKeeper. Dosegljivo: <https://en.wikipedia.org/wiki/MacKeeper>, 2017. [Dostopano: 12. 3. 2017].
- [35] Mac OS remove app. Dosegljivo: https://support.apple.com/kb/PH25083?locale=sl_SI, 2017. [Dostopano: 22. 8. 2017].
- [36] MD5. Dosegljivo: <http://www.si-ca.si/kripto/kr-zgo.htm>, 2006. [Dostopano: 5. 1. 2018].
- [37] Monolingual. Dosegljivo: <https://www.macupdate.com/app/mac/7758/monolingual>, 2017. [Dostopano: 12. 3. 2017].
- [38] Monolingual. Dosegljivo: <http://www.makeuseof.com/tag/monolingual-remove-languages-you-dont-use-from-your-mac/>, 2013. [Dostopano: 18. 3. 2017].
- [39] msconfig. Dosegljivo: <http://www.online-tech-tips.com/cool-websites/guide-to-msconfig-windows/>, 2014. [Dostopano: 27. 8. 2017].
- [40] Microsoft recommends. Dosegljivo: <https://support.microsoft.com/en-us/help/15055/windows-7-optimize-windows-better-performance>, 2016. [Dostopano: 12. 3. 2017].
- [41] Denis Oblak. CleanLin. Dosegljivo: <https://github.com/nusasara/cleanlin>, 2018.

-
- [42] OnyX. Dosegljivo: <https://en.wikipedia.org/wiki/OnyX>, 2017. [Dostopano: 12. 3. 2017].
- [43] Onyx. Dosegljivo: <http://macreview.com/2012/10/onyx-review/>, 2012. [Dostopano: 12. 3. 2017].
- [44] Optimize Storage. Dosegljivo: <https://support.apple.com/en-us/HT206996>, 2017. [Dostopano: 30. 9. 2017].
- [45] Upravitelji paketov. Dosegljivo: <https://www.linode.com/docs/tools-reference/linux-package-management>, 2017. [Dostopano: 20. 3. 2017].
- [46] Basic Troubleshooting and Maintenance Tips. Dosegljivo: http://www.gballard.net/macrant/osx_troubleshooting.html, 2014. [Dostopano: 23. 8. 2017].
- [47] Jv16 powertools. Dosegljivo: https://en.wikipedia.org/wiki/Jv16_powertools, 2016. [Dostopano: 10. 3. 2017].
- [48] Registry cleaner. Dosegljivo: https://en.wikipedia.org/wiki/Registry_cleaner, 2016. [Dostopano: 10. 3. 2017].
- [49] Revo Uninstaller. Dosegljivo: https://en.wikipedia.org/wiki/Revo_Uninstaller, 2016. [Dostopano: 10. 3. 2017].
- [50] Stacer. Dosegljivo: <http://www.techrepublic.com/article/how-to-install-stacer-for-quick-linux-system-optimization/>, 2017. [Dostopano: 2. 9. 2017].
- [51] Startup manager. Dosegljivo: <https://support.microsoft.com/en-us/help/4026268/windows-change-startup-apps-in-windows-10>, 2017. [Dostopano: 27. 8. 2017].
- [52] Synaptic. Dosegljivo: <https://help.ubuntu.com/community/SynapticHowto>, 2013. [Dostopano: 12. 3. 2017].

- [53] Najbolje Linux distribucije za male tvrtke. Dosegljivo: <http://www.ictbusiness.info/poslovna-rjesenja/najbolje-linux-distribucije-za-male-tvrtke>, 2016. [Dostopano: 6. 3. 2017].
- [54] Top 10 Linux Server Distributions of 2017 . Dosegljivo: <http://www.serverwatch.com/columns/slideshows/top-10-linux-server-distributions.html>, 2017. [Dostopano: 6. 3. 2017].
- [55] Najbolje Linux distribucije u 2016. godini . Dosegljivo: <http://pcchip.hr/softver/najbolje-linux-distribucije-u-2016-godini/>, 2016. [Dostopano: 6. 3. 2017].
- [56] 5 najboljših distribucij Linux . Dosegljivo: <http://www.racunalniskenovice.com/triki/5-trenutno-najboljsih-linux-distribucij.html?RSS81120f575fb05a68ea366bf6a4bd8c94>, 2014. [Dostopano: 6. 3. 2017].
- [57] Emilio Raggi Keri Thomas Trevor Parsons Andy Channelle Sander van Vugt. Beginning Ubuntu Linux.
- [58] Wise Registry Cleaner. Dosegljivo: http://www.pcworld.com/article/256026/wise_registry_cleaner.html, 2012. [Dostopano: 10. 3. 2017].
- [59] Wise Registry Cleaner v9.41. Dosegljivo: <https://www.lifewire.com/wise-registry-cleaner-review-2625142>, 2017. [Dostopano: 10. 3. 2017].