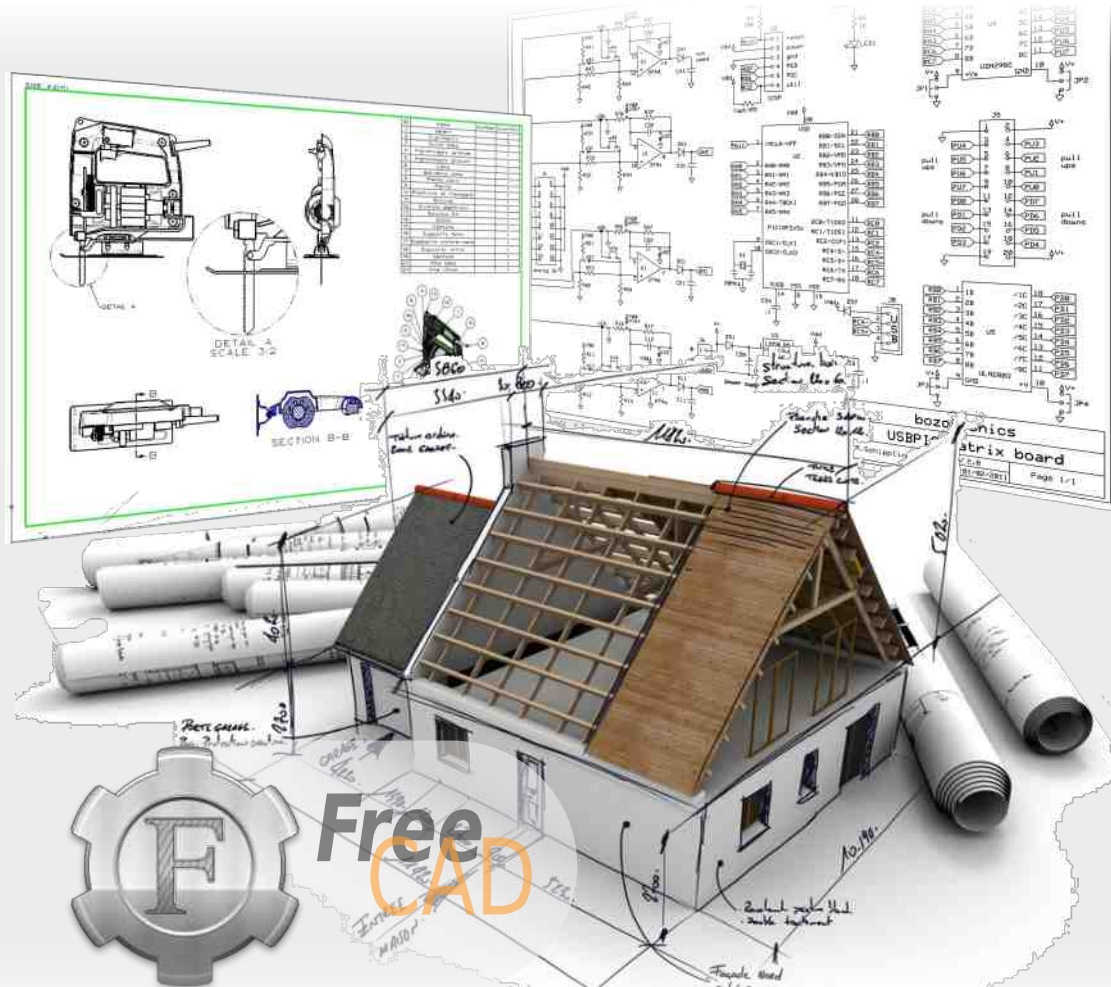


April 2015. Broj 35

LIBRE!

Časopis o slobodnom softveru



JOŠ IZDVAJAMO

Ansibl

Gugl - Savršena reklamna agencija



Creative Commons Ауторство-Некомерцијално-Делити под истим условима

Gde je danas *FLOSS* u svetu?

Sve više istraživanja u svetu pokazuju da softver otvorenog koda preuzima dominaciju u svim oblastima. Nedavno su kompanije Nort Bridž i Blek Dak sprovele istraživanje o tome koliko se slobodni softver koristi u poslovnom okruženju (firmama i kompanijama). Donedavno je poslovno okruženje bilo neprikosnovo carstvo vlasničkog softvera. Najnovije istraživanje sada pokazuje da čak 78% firmi i kompanija u svetu koristi delimično ili u potpunosti slobodan softver u svom poslovanju. Istraživanje je pokazalo da je još uvek u ovom segmentu nizak nivo „open-sors pismenosti“, ali da se stanje znatno popravlja u odnosu na ranije godine.

Činjenica je da je vlasnički softver stariji i da prati razvoj stonih (desktop) personalnih računara već više od trideset pet godina. Slobodan softver je znatno kasnio u osvajanju personalnih računara. Tek u poslednjih deset godina sa razvojem interneta i razvoj slobodnog softvera je primetan i snažan. Zahvaljujući bržem i snažnijem razvoju, približava se kvalitetu vlasničkog softvera i na personalnim računarima. Oblasti u kojima je slobodni softver danas superioran u odnosu na vlasnički su tehnologije koje su se razvijale uporedo sa slobodnim softverom: internet, mobilna telefonija, mobilni računari i tako dalje.

U ovom broju časopisa vam predstavljamo Fri-ked (eng. *FreeCAD*). Ovaj softver je najbolji primer brzog i приметnog napretka slobodnog softvera u svim oblastima korišćenja računara. Pre tri godine u časopisu smo prvi put pisali o slobodnim *CAD* programima. Tada smo konstatovali da je stanje u toj oblasti jako loše i da ti programi mogu da služe samo za skice. Danas predstavljamo već ozbiljan projekat koji još ima dosta svojih mana, ali u odnosu na takve programe od pre samo tri godine ovo je ogroman



napredak.

Istraživanje koje smo pomenuli kaže da je još u mnogo firmi obaveštenost o slobodnom softveru na niskom nivou. Istovremeno, neke firme su izgradile već imperiju na slobodnom softveru. Kad to kažemo, prvenstveno mislimo na najjaču - Gugl.

Prosto je neverovatno kako je čovek u stanju od svakog izuma da napravi oružje. Iako je novac jak motivator za rad, u zapadnoj kulturi, pre ili kasnije, onaj koji zaradi mnogo novca poželi da osvoji svet. O tome kako Gugl tumači reč „Free” iz akronima FLOSS i šta se desilo sa „L” iz istog akronima, možete takođe da pročitate u ovom broju.

Gde su Srbija i region u ovoj oblasti? Kao i u svemu, kasnimo deset godina. Sve najnovije tehnologije koristimo kao i svi u svetu. Problem najviše leži u tome što je većina korisnika potpuno neobaveštena i „open-sors nepismena”. Zbog tog neznanja sve plaćamo više nego što je neophodno i ugrožena nam je privatnost, a da toga nismo svesni. LiBRE! zato postoji da bi se promenila ova statistika i kod nas. Krajnje vreme je da se opismenimo i u slobodnom softveru kako ne bismo plaćali danak neznanju, i to ne samo među „kompjuterskim stručnjacima”, nego i među običnim korisnicima. Slobodni softver već odavno nije „igračka” za štrebere, već tehnologija koja nas okružuje. Uključite se i vi u ovo „open-sors opismenjavanje” i pišite nam na našu već poznatu adresu elektronske pošte [libre \[et\] lugons \[dot\] org](mailto:libre@lugons.org).

Do čitanja,

LiBRE! tim

Sadržaj

Vesti

str. 6

Predstavljamo

FreeCAD

str. 9

Kako da...?

Uvod u programski jezik C (11. deo)

Vagrant (2. deo)

str. 17

str. 20

Oslobađanje

Rezvoj slobodnog softvera

str. 24

Slobodni profesionalac

Vaš posao, *open-source* posao -

Gugl - Savtšena reklamna agencija

Ansible (1. deo)

str. 29

str. 34

Internet mreže i komunikacije

Šifrovani čet (2. deo) - *Niltalk*

str. 38

Sam svoj majstor

Bootstrap - Prvi koraci

str. 41

Hardver

BagleBone Black Rev C: Vodič od prvog dana (6. deo)

Biglbon Blek kao Tor egzit

str. 45

Moć slobodnog
softvera





LIBRE! prijatelji



REGIONALNI
LINUX PORTAL

linuxzasve.com



Grupa korisnika GNU/Linux operativnih sistema u Lovčencu

info i tutorijali na srpskom
lubunturs.wordpress.com



Broj: 35

Periodika izlaženja: mesečnik

Izvršni urednik: Stefan Nožinić

Glavni lektor:

Admir Halilkanović

Lektura:

Jelena Muncan

Saška Spišjak

Milena Beran

Milana Vojnović

Aleksandar Božinović

Grafička obrada:

Dejan Maglov

Ivan Radeljić

Dizajn: White Circle Creative Team

Autori u ovom broju:

Nenad Marjanović

Zlatan Vasović

Nikola Hardi

Kriptopank

Ivan Radeljić

Nikola Nenadić

Ostali saradnici u ovom broju:

Marko Novaković

Mihajlo Bogdanović

Počasni članovi redakcije:

Željko Popivoda

Željko Šarić

Vladimir Popadić

Aleksandar Stanisavljević

Kontakt:

IRC: #floss-magazin na irc.freenode.net

E-pošta: libre@lugons.org

Vesti

19. april 2015.

Pedeset godina Murovog zakona

Pre tačno pedeset godina je publikovan Murov zakon, koji predviđa da se broj komponenti na čipu duplira tokom vremena dok cena i površina ostaju isti. Ovo čini da broj komponenti u čipu raste eksponencijalno.

Koristan link: <http://j.mp/1EjJTHg>



28. mart 2015.

Evropska komisija će povećati upotrebu slobodnog softvera

Evropska komisija je odlučila da ažurira svoju strategiju o internoj upotrebi softvera, koja predviđa upotrebu slobodnih softverskih sistema kao i dodatni razvoj softverskih rešenja otvorenog koda.

Koristan link: <http://j.mp/1EwagHt>



1. april 2015.

Objavljen Fajerfoks 37

Mozilla je objavila novu verziju svog veb-pretraživača otvorenog koda.

Koristan link: <http://j.mp/1Pq3Jj5>





3. april 2015.

Github je bio izložen DDoS napadu

Sumnja se da je za ovaj napad odgovorna kineska vlada, što se naslućuje iz činjenice da su napadi stizali iz Kine sa zvaničnih servera kineske vlade.

Koristan link: <http://j.mp/1DolKH4>



3. april 2015.

Rajesh De o Snoudu

Generalni savetnik američke Državne bezbednosne agencije (NSA) objašnjava da je jako teško za ovu američku agenciju da ostane bez kritika u medijima i da ne može da veruje da Snouden, koji je sedeo sa njima i brinuo se o sigurnosti građana SAD-a, mogao da uradi tako nešto kao što je objavljivanje tajnih podataka ove agencije. On takođe kaže da treba naći balans između sigurnosti i privatnosti građana.

Koristan link: <http://j.mp/1OLzMI0>



6. april 2015.

DORS i KHKL od 18. do 20. maja

Dani otvorenih računarskih sustava (DORS) i Konferencija Hrvatskih korisnika Linuksa (KHKL) održaće se u Zagrebu na Fakultetu elektrotehnike i računarstva od 18. do 20. maja.

Koristan link: <http://j.mp/1QpaWl0>



Vesti

9. april 2015.

Linuks fondacija hostuje projekat *Open Encryption*

Ova neprofitna organizacija će hostovati projekat koji će omogućiti veb-stranicama da dobiju sigurnosne sertifikate u roku od nekoliko minuta, čime će veb učiniti globalno bezbednijim.



Koristan link: <http://j.mp/1Ewaw9l>

11. april 2015.

Podrška za enkripciju za *EXT4* sistem datoteka

U Guglu je razvijena podrška za enkripciju u sklopu *EXT4* sistema podataka. Biće ugrađena u novu verziju Androida M i biće poslata glavnom razvojnom timu da se doda u novi Linuks kernel.



Koristan link: <http://j.mp/1FhPJ8j>

22. april 2015.

GCC 5.1 objavljen

Objavljena je nova verzija Gnuovog kompajlera GCC.



Koristan link: <http://j.mp/1Domr38>



FreeCAD



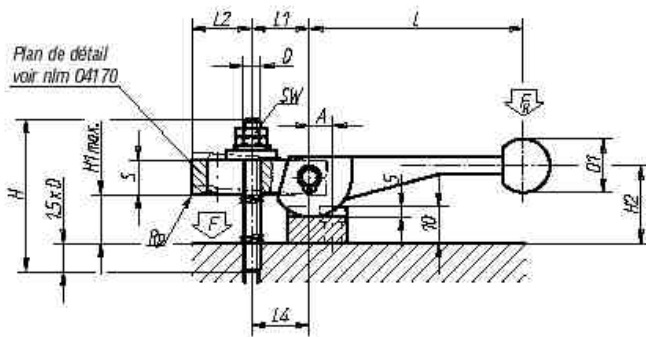
Autor: Nikola Hardi

Odricanje od odgovornosti

Autor se ne bavi profesionalno tehnikom ili kompjuterskim konstruisanjem. Ovaj tekst je nastao iz želje autora da predstavi program koji mu je pomogao da reši problem pred kojim se našao. Ovo neće biti profesionalni opis programa, ali se nadamo da će biti dovoljno dobar opis programa koji može mnogima da bude od koristi.

CAD/CAM softver

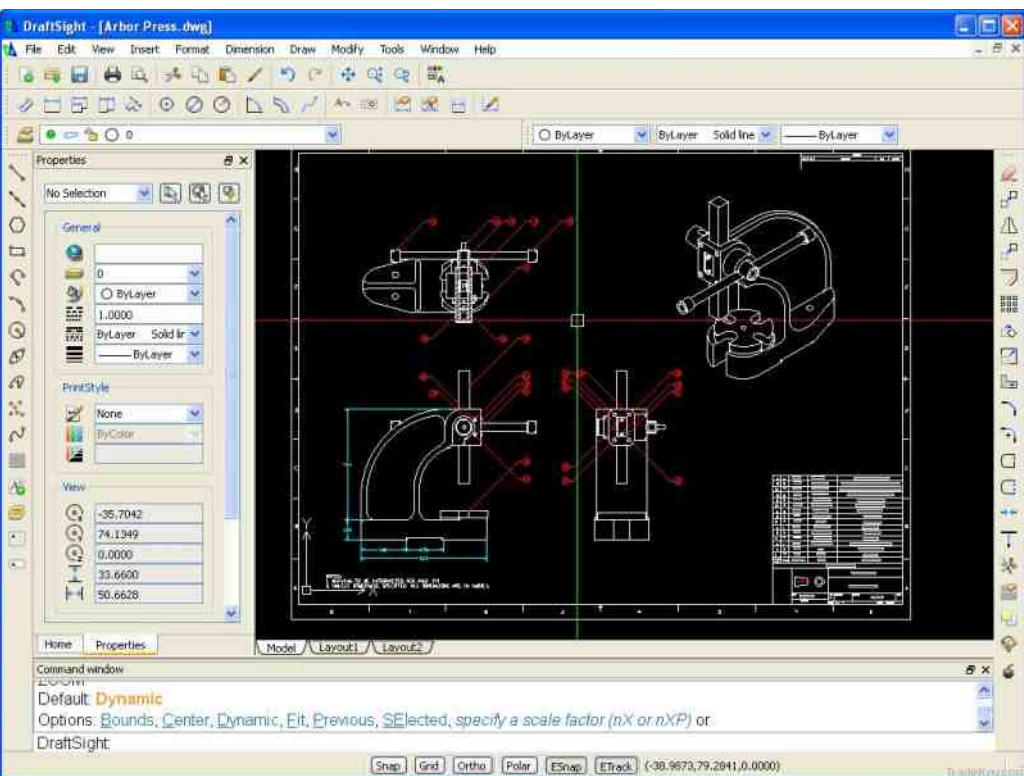
CAD/CAM (eng. *computer-aided design / computer-aided manufacturing*) sistemi - u prevodu: kompjuterski sistemi koji pomažu pri kreiranju, modifikovanju, analizi i optimizaciji dizajna sa jedne strane, i upravljanju numerički upravljanih mašina alatki u proizvodnji, sa druge strane. Laičkim rečnikom rečeno, u ovom tekstu je predstavljen softver za tehničko crtanje. Da podsetimo, tehničko crtanje je precizno crtanje sa jasnim pravilima koja svima u lancu proizvodnje omogućavaju čitljivost crteža.



Predstavljamo

CAD/CAM softver je, uprošteno, softver za 2D i 3D vektorsku grafiku, sa naglaskom na precizno crtanje alatima za brzo i lako dodavanje tehničkih specifikacija.

Ovakav tip programa je bolna tačka u svetu slobodnog softvera jer, kako kažu ljudi iz struke, ne postoji alat odgovarajućeg kvaliteta za profesionalnu upotrebu. Postoje programska rešenja koja nisu otvorenog koda, ali mogu biti pokrenuta na slobodnim platformama kao što su Briksked (*BricsCAD*) i Draft-Sajt (*DraftSight*). Neka profesionalna rešenja su postala otvorena i slobodna, kao BRL-CAD. O jednom potpuno slobodnom CAD softveru smo već pisali u ovom časopisu (*LibreCAD, LIBRE!* broj 1).





Možda je najveći problem u tome što je Autodeskov Autoked (*Autodesk AutoCAD*) postao standard za profesionalnu upotrebu, a on je zaštitio svoj **DWG** format. Bez tog formata, nema komunikacije sa nekim drugim CAD programima, pa prema tome ni saradnje sa kolegama koji koriste samo Autoked.

U ovom tekstu predstavimo još jedan slobodni CAD softver - Friked (*FreeCAD*).

2D CAD

Jedan pristup je crtanje krajnjih pogleda na element koji se dizajnira. Ovo podrazumeva crtanje 2D projekcija, uključujući finalno označavanje dimenzija, šrafiranje, popunjavanje zaglavlja itd. Treba napraviti razliku između CAD programa i drugih programa za 2D vektorsku grafiku, kao što su Libreofis drou (*Libreoffice Draw*) i Inkskej (p) (*Inkscape*). Iako je pomoću oba ova programa moguće napraviti crtež nalik na tehničku dokumentaciju, oni imaju svoje nedostatke u odnosu na CAD programe. Pre svega, nemaju potrebnu preciznost, unošenje izmena, postavljanje dimenzija i ostalih tehničkih specifikacija zahteva mnogo više truda, što smanjuje produktivnost.

3D modelovanje

Oznaka „3D“ se često sreće u svetu softvera. Postoje 3D igre, 3D filmovi, alati za 3D modelovanje i 3D CAD/CAM alati. Primer programa za 3D modelovanje je Blender. Problem je što je i u ovom programu akcenat na izgledu, a ne na dimenzijama. Moguće je nacrtati objekat koji u prostoru izgleda kako treba, međutim problem nastaje kod zadavanja dimenzija, označavanja dimenzija i pravilnog iscrtavanja projekcija i preseka.

Objekti nacrtani ovakvim alatom nemaju zapreminu, već imaju samo omotač. Programi kao što je Friked, ili neki komercijalni, kao što su Autoked, Inventor, Katia (*Catia*) ili Pro inženjer (*PRO/Engineer*) - omogućuju korisniku da nacrti 3D model elementa koji će zaista imati i zapreminu, odgovarajuće dimenzije, parametre i moći će propisno da se predstave projekcijama u vidu tehničkih crteža. Sve u svemu, pravi CAD softveri inženjerima pružaju više informacija o nacrtanom telu od običnih 3D dizajnerskih softvera i takođe pomažu u daljoj obradi tih informacija.

Predstavljamo

Primer upotrebe

Friked ima donekle standardni način projektovanja. Potrebno je nacrtati skicu elementa. Zatim se od skice pravi 3D objekat izvlačenjem ili rotacijom. Postupak se ponavlja uz primenjivanje dodatnih alata kao što je obrada ivica, bušenje itd. Naposljetku, element se projektuje u crtež i vrši se njegovo kotiranje.

Okruženje programa

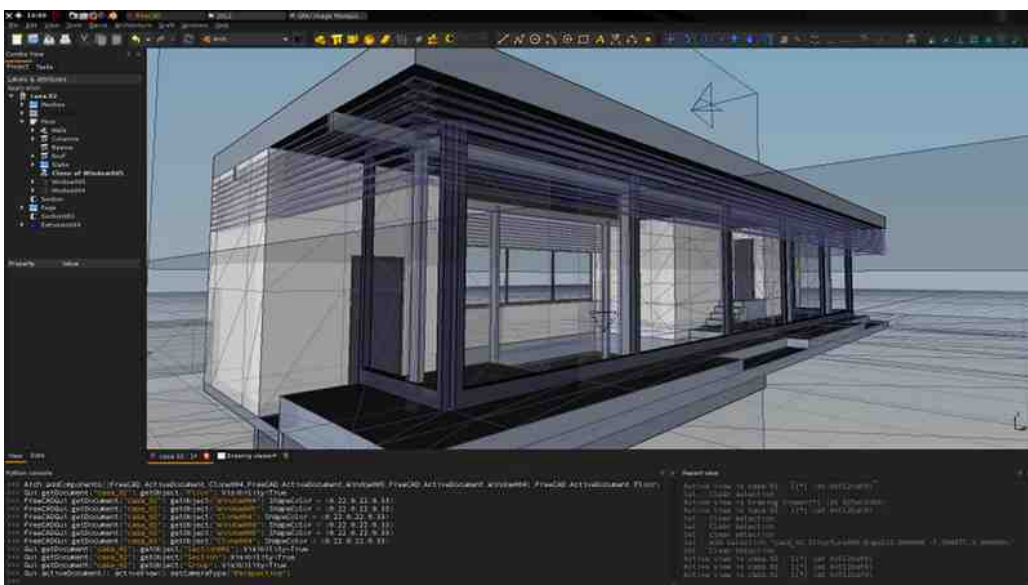
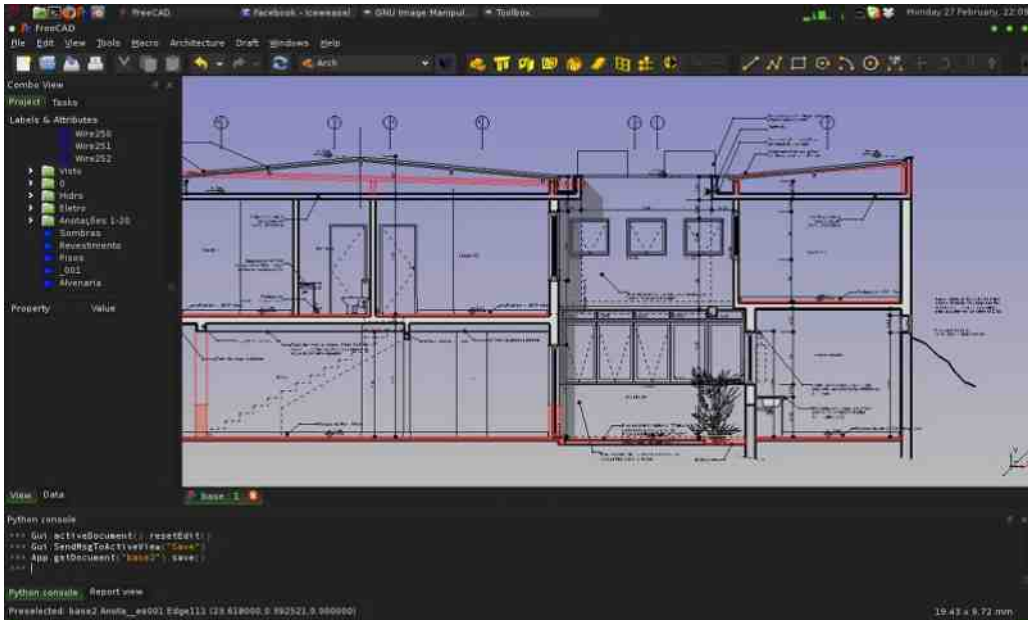
Friked se sastoji od više radnih okruženja (eng. *workbench*). Neka od tih okruženja su: početno okruženje, okruženje za rad sa delovima (eng. *part*), okruženje za dizajn delova (eng. *part design*), rad sa crtežima (eng. *drawing*) itd. Svako radno okruženje ima drugačiju namenu i različit set alata. Prvi korak je naravno kreiranje novog projekta. U levom delu prozora postoji deo za prikaz strukture trenutnog projekta (eng. *model*) i deo za prikaz zadataka dostupnih u trenutnom okruženju (eng. *tasks*).

Crtanje skice

U okruženju za dizajn delova, u delu za zadatke, moguće je dodati novu skicu u željenoj ravni. Crtanje skice je prvi korak u dizajnu novog dela. Skica može biti presek dela ili neki od tri osnovna pogleda na deo. Na raspolaganju su alati za crtanje geometrijskih primitiva: linija, krugova, lukova, pravougaonika itd. Skice treba crtati slobodno, jer je kasnije svakako neophodno dodati ograničenja (eng. *constraints*). Ograničenja postavljaju odnose između elemenata skice. To može da bude ugao između dve linije, rastojanje između tačaka ili dužina linije. Linije mogu da budu međusobno paralelne. Tačke mogu da budu preklapljene. Sa leve strane postoji kontrola da li je skica dobro definisana ili upozorenje da neka ograničenja ne mogu da budu ispunjena (eng. *solver messages*). Praktičan način kojim može da se proveri stanje skice je nasumično povlačenje delova skice. Cilj je da skica nikada ne izgubi svoj oblik. Po završetku crtanja skice može se napustiti crtanje skice (eng. *exit the editing of the sketch*).

Izvlačenje elementa

Crtanje skice je samo početni korak u dizajnu jednog dela. Od crteža u ravni



Predstavljamo

potrebno je napraviti trodimenzionalna tela. Za prelazak iz ravni u prostor mogu da posluže alati za istiskivanje ili izvlačenje skice (eng. *extrude*) ili za kreiranje tela rotiranjem skice oko neke ose. Osim ovih alata moguće je iskoristiti alat za zaobljavanje ivica ili napraviti udubljenje.

Dodavanje crteža

Kada je deo poprimio željeni oblik, može se preći na pravljenje crteža. Potrebno je preći u okruženje za rad sa crtežima i u traci sa alatima dodati novi list željenog formata. Sledeći korak je dodavanje dela na crtež. U levom delu ekrana, u prikazu strukture projekta, treba označiti željeni deo, a u traci sa alatima izabrati akciju dodavanja pogleda ili ortografske projekcije na crtež. Ukoliko u projektu postoji više listova, pri dodavanju novog pogleda je potrebno u strukturi projekta označiti i deo i list u koji pogled treba da bude dodat.

Dodatak za dimenzionisanje crteža

Jedan veliki nedostatak programa je kotiranje crteža. Dobra stvar je što postoji sjajan dodatak za ovu namenu. Reč je o dodatku koji dodaje okruženje za kotiranje crteža (eng. *drawing dimensioning*). Uputstvo za instalaciju i kod dodatka je moguće preuzeti sa sledeće adrese: https://github.com/hamish2014/FreeCAD_drawing_dimensioning. Kada je pogled dodat na crtež, prelaskom u okruženje za kotiranje pojavljuju se alati za kotiranje. Najavljeno je da ovaj dodatak biti uključen u naredne verzije programa. Iako je osnova projekta napisana u programskom jeziku C++ uz pomoć Qt biblioteka, dodaci mogu da budu napisani u Pajtonu (*Python*). Pajton konzola je uvek dostupna u jednom delu prozora i omogućava automatizaciju mnogih zadataka i istraživanje koda programa „naživo“.

Prevod programa

Dostupan je prevod na hrvatskom jeziku, što je uglavnom prihvatljivo za naše područje. Prevod na srpski je takođe započet - prevedena je približno polovina sadržaja. Međutim, prevod iz nekog razloga nije dostupan za izbor u programu. Ukoliko postoji dovoljan broj zainteresovanih da se ovaj prevod dovrši, naš časopis će rado podržati takvu akciju.



Nekoliko komentara za kraj

Friked je daleko od savršenog ili profesionalnog rešenja, ali ipak je upotrebljiv. Za kraj ostavljamo nekoliko komentara, dobrih i loših. Budite slobodni da podelite sa nama i ostalim čitaocima vaše crteže ili iskustva u radu.

Rad sa crtežima

Rad sa crtežima je pomalo neintuitivan. Najveća zamerka odlazi na pomeranje elemenata na crtežu. Nije moguće pomeranje elemenata na crtežu kursorom miša, već se pozicija elemenata mora postaviti u prozoru za osobine tog elementa. Ovo može da bude nezgodno i zamorno. Nadamo se da će ovaj problem uskoro biti rešen.

Drugi problem u okruženju za rad sa crtežima je popunjavanje zaglavlja. Slično kao i sa pozicijama, sadržaj zaglavlja crteža može biti izmenjen u osobinama crteža, ne i klikom na tekst u prikazu samog crteža. Nije strašan problem ali zasigurno bi klik na tekst u zaglavlju bio praktičniji.

Stabilnost

Iako na sajtu programa postoji izričito upozorenje da je projekat u ranoj fazi razvoja i da su mogući izvesni problemi u redu, takvi problemi nisu primećeni. I pored našeg dobrog iskustva, i mi ćemo ostaviti upozorenje da budete pažljivi i redovno čuvate izmene u vašim projektima.

Brzo se razvija

Projekat postoji više od 10 godina, a u proteklih nekoliko godina primetan je veliki napredak u razvoju. Na primer, u Ubuntuu 14.04 je dostupna verzija 0.13 koja vrlo oskudeva sa mogućnostima u odnosu na trenutnu najnoviju verziju 0.15. Dobra vest je da postoje paketi sa novijim verzijama, a za Ubuntu je dostupan i PPA repozitorijum sa novim stabilnim verzijama i dnevnim (eng. *daily*) verzijama.

Predstavljamo

Stanje dokumentacije

Zvanična dokumentacija nije baš sjajna. Međutim, na internetu su dostupna mnoga uputstva i saveti korisnika. Posoji mnogo video primera na Jutjubu (*YouTube*). Ipak, pozivamo vas da doprinesete kvalitetu dokumentacije prevodom ili pisanjem novih uputstava.

Neistražene mogućnosti

Frieked nudi mnogo mogućnosti koje nisu ni pomenute u ovom tekstu. Naprimera, dostupna su okruženja za arhitekturu, brodogradnju, rad sa sklopovima, simulacije, projektovanje robota itd. Dostupni su mnogi dodaci koji još uvek nisu dospeli u standardno izdanje. Sve u svemu, postoji pregršt mogućnosti za istraživanje.



Pregled popularnosti GNU/Linuks i BSD distribucija za mesec april

Distrowatch

1	Mint	3196>
2	Ubuntu	1944>
3	Debian	1533>
4	openSUSE	1241<
5	CentOS	1210<
6	elementary	1174<
7	Fedora	1173>
8	Mageia	921<
9	LXLE	879>
10	Arch	856>
11	Manjaro	855<
12	Android x86	821<
13	Lite	711=
14	MakuluLinux	621>
15	PCLinuxOS	618>
16	Puppy	607>
17	Lubuntu	591>
18	Kali	539<
19	Antergos	539>
20	KaOS	514>
21	Simplicity	512>
22	Zorin	495>
23	Bella	495>
24	SparkyLinux	492>
25	Xubuntu	469>

Pad <
Porast >
Isti rejting =
(Korišćeni podatci sa Distrovoča)

Kako da...?

Rad sa bazom podataka



U prethodnom delu serijala je spomenuto da je datoteke pogodno zameniti bazama podataka. Rad sa bazama podrazumeva mnogo stvari, poput uspostavljanja veze sa serverom. Zanimljiv pristup je i rad sa *SQLite* (Es-kju-lajt) bazom koja je sačuvana u jednoj datoteci. Ceo sistem za upravljanje bazom je enkapsuliran u jednu datoteku i jednu biblioteku - *SQLite* upravljački program (drajver).

Komunikacija sa udaljenim računarima

Međuračunarske mreže mogu biti vrlo složene. Komunikacija između dva računara može da se vrši na raznim nivoima i raznim protokolima. Međutim, korisnicima (i programerima) uglavnom je dovoljno da prenesu neki niz bajtova iz jednog programa u drugi. Ovo se može obaviti pomoću transportnog sloja, mehanizmom „mrežnih utičnica“ (eng. *network sockets*) i protokola *TCP* ili *UDP*. Potrebno je svega desetak linija *C* koda da bi se uspostavila veza između dva programa putem mreže, a potom se prenos podataka sastoji od pripreme podataka u formi poruke (string i memorijski blok), slanja tih podataka i eventualnog čekanja na odgovor. Ključne reči za pretragu na internetu su „*BSD sockets*“ ili „*Berkleu sockets*“.

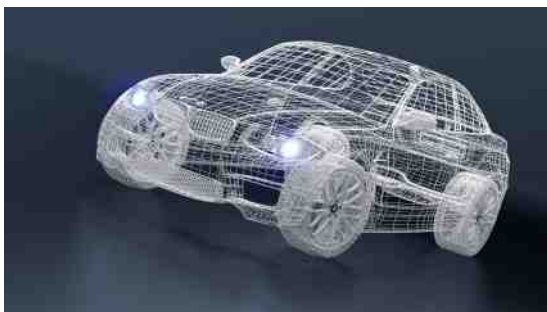
Napredna konzolna aplikacija



Konzolne aplikacije mogu da imaju složeniji interfejs - moglo bi se reći čak i grafički interfejs. Reč je o biblioteci *Ncurses*, koja ima veoma dugu istoriju. Pomoću ove biblioteke moguće je izgraditi zaista zanimljive konzolne aplikacije.



Rad sa grafikom

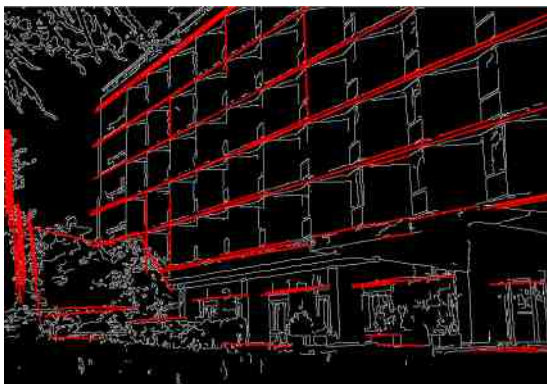


(recimo, 3D igrom) postoje potpuni pokretači (eng. *game engine*) kao što je *OGRE 3D (Object-Oriented Graphics Rendering Engine)*.

Rad sa grafikom takođe može da podrazumeva mnogo toga. Pre svega, priča se deli na 2D i 3D grafiku. Za rad sa 2D grafikom može poslužiti *SDL* (Es-di-el), a za 3D *OpenGL* (Open-dži-el). Postoje i druge biblioteke sa sličnim funkcionalnostima. Za

napredniji rad sa 3D grafikom

Kompjuterska vizija



automatska korekcija fotografija. Za ovakve zadatke od neprocenjive važnosti je biblioteka *OpenCV (Open-si-vi)*.

Iako i ovi projekti na neki način spadaju u rad sa grafikom, zaslužuju poseban odeljak. Reč je naučno-istraživačkim projektima koji nastaju zarad istraživanja u oblasti mašinskog učenja i veštačke inteligencije u domenu kompjuterske vizije. Recimo, primeri ovakvih projekata bili bi prepoznavanje lica, prepoznavanje objekata i

Learn C Programming

Kako da...?



VAGRANT

(2. deo)

Autori: Ivan Radeljić i Stefan Nožinić

U prošlom broju pisali smo o prednostima korišćenja Vagranta pri podešavanju razvojnog okruženja. Istakli smo zašto svaki od članova tima ima koristi. Sada ćemo vam pokazati kako da instalirate Vagrant i kako da koristite Vagrant komande.

Instalacija

Najpre je potrebno da posetite stranicu <http://www.vagrantup.com/downloads> i da preuzmete odgovarajući paket za vaš operativni sistem i arhitekturu. Instalater će automatski dodati *vagrant* komandu u vaš sistemski *PATH*, tako da će komanda biti dostupna globalno u terminalu. Po završetku instalacije u terminalu napišite komandu:

```
vagrant -v
```

i program će vam ispisati verziju koja je instalirana.

Vagrant se takođe nalazi u riznicama paketa mnogih distribucija Linuksa, pa time možete Vagrant instalirati kao i svaki drugi paket koji instalirate za vašu distribuciju. Naprimer, ako koristite Ubuntu, procedura je sledeća - otvorite terminal i Vagrant instalirate upravnikom paketa na sledeći način:

```
sudo apt-get install vagrant
```



Ako imate bilo kakvih problema tokom instalacije, možete nas kontaktirati na našu adresu elektronske pošte *libre [AT] lugons.org*.

Inicijalizacija

Da biste uspešno integrisali Vagrant u vaš projekat, potrebno je prethodno instalirati Virtual boks (*VirtualBox*) ili neki drugi provajder. Potom treba pronaći i instalirati Vagrant kutije (*Vagrant boxes*). U zavisnosti od projekta, biramo odgovarajuću kutiju. Na internet adresi <https://atlas.hashicorp.com/boxes/search> možemo vršiti pretragu kutija. Za dodavanje nove kutije koristimo:

```
vagrant box add USER/BOX
```

Sada je sve spremno za inicijalizaciju projekta.

```
vagrant init USER/BOX
```

Ova komanda će kreirati Vagrant konfiguracionu datoteku (*vagrantfile*) u početnom direktorijumu projekta. Ova datoteka je sve što nam je potrebno da uspešno koristimo Vagrant dalje u projektu. Naravno, *vagrantfile* datoteka sadrži sve potrebne konfiguracije i, kada želimo da priključimo novog člana postojećem timu, dovoljno je da mu prosledimo *vagrantfile* datoteku. Ovo možemo uraditi tako što ćemo tu datoteku podeliti na Git repozitorijumu ili nekom sistemu za timski razvoj softvera.

Up and running

U ovom stadijumu Vagrant okruženje je spremno za pokretanje. Koristimo komandu:

```
vagrant up
```

i naše virtuelno razvojno okruženje radi. Od ovog momenta sve promene u direktorijumu Vagrant projekta na host mašini će se automatski odraziti na naše virtuelno okruženje. Sinhronizacija na virtuelnoj mašini se izvršava u direktorijumu */vagrant*. Ovo možemo proveriti tako što ćemo pristupiti virtuelnoj mašini preko *SSH* protokola. Komanda za to je:

Kako da...?

```
vagrant ssh
```

Nakon ove komande i pristupanja virtuelnom okruženju, možemo koristiti sve nama dobro poznate Linuks terminalne komande. Po završetku rada korišćićemo komandu:

```
vagrant halt
```

da bismo isključili virtuelnu mašinu i sačuvali sve promene. Postoji i komanda:

```
vagrant destroy
```

koja briše ceo projekat i sve zavisnosti koje smo instalirali, tako da ovu komandu koristite tek onda kada ste sigurni da je vaš rad na projektu završen.

Konfiguracija

Kao što je već napomenuto, konfiguracija Vagrant mašine za konkretan projekat se obavlja u *vagrantfile* konfiguracionoj datoteci. Ova datoteka je Rubi (*Ruby*) skripta koja pruža konkretne informacije Vagrantu o tome koja se kutija koristi, koji se sistem automatske konfiguracije koristi (eng. *provisioning*) i slična podešavanja.

Ova datoteka treba da se postavi na sistem kontrola verzija (eng. *version control system*) i time će svaki član tima dobiti istu konfiguraciju i, jednostavno, komandom *vagrant up* dobiti okruženje kao i Vi.

Svaki projekat koji koristi Vagrant mašine treba da ima samo jedan *vagrantfile*. Ova konfiguraciona datoteka ne zavisi striktno od platforme koju koristite, osim ako se u njoj ne pozivate na funkcije koje su specifične za neku platformu, što je moguće, ali se retko radi i nije preporučljivo.

Kada pokrenete Vagrant, on traži *vagrantfile* u trenutnom direktorijumu. Ako tu ne pronađe konfiguracionu datoteku, on je traži u roditeljskom direktorijumu i tako dalje. Ovo je veoma korisno jer se možete kretati kroz direktorijume unutar vašeg projekta i uvek ćete moći da pozivate Vagrant bez potrebe da se vraćate u glavni direktorijum vašeg projekta.

Podrazumevani *vagrantfile* dobijate kada u trenutnom direktorijumu izvršite



vagrant init. Ova konfiguraciona datoteka koristi Rubi sintaksu, ali znanje ovog programskog jezika nije potrebno da biste razumeli konfiguraciju i podesili istu u skladu sa vašim potrebama i potrebama vašeg tima i projekta.

Automatska konfiguracija

Vagrant podržava takozvani *provisioning*, odnosno automatsku konfiguraciju kutije prilikom prvog pokretanja. Ovo je veoma korisno jer ne morate praviti kutiju za svaki projekat, već, recimo, možete koristiti početnu Debian konfiguraciju, a podesiti specifičnu konfiguraciju u zavisnosti od projekta. Postoji mnogo sistema za automatsku konfiguraciju. Najjednostavniji sistem je *shell*, odnosno da napravite skriptu koja će se pokrenuti prilikom prvog pokretanja mašine i podesiti automatski sve što ste naveli u njoj. Ovde možete instalirati svoje pakete koji su vam potrebni, podesiti veb-server i slično. Naravno, datoteke za sistem automatske konfiguracije morate takođe podeliti sa vašim timom kroz sistem kontrola verzija.

Pored *shell* pristupa moguće je koristiti i druge sisteme kao što je Ansibl, o kojem možete čitati u našem časopisu.

Sistem automatske konfiguracije, kao i specifični parametri za konkretan sistem, podešavaju se u *vagrantfile*.

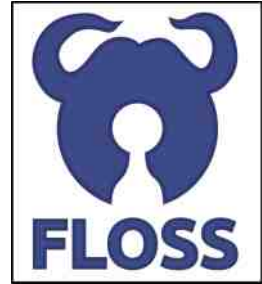
U narednim brojevima ćemo detaljnije pričati o ovoj tematici, a preporučujemo vam i da konsultujete zvaničnu dokumentaciju i da se sami oprobate u konfigurisanju i eksperimentisanju.



VAGRANT

Oslobađanje

Razvoj slobodnog softvera



Autor: Nikola Hardi

Slobodan softver otvorenog koda postoji, to je činjenica. Često se postavlja pitanje ko i zašto razvija ovakav softver. Drugo pitanje koje se često postavlja je: „Kako mogu da se uključim u neki projekat koji mi se dopada?” Ovim tekstom ćemo pokušati da damo odgovor na oba ova pitanja.

Razlozi i načini za uključenje u projekat slobodnog softvera

Osećaj pripadnosti

Razloga ima mnogo i uglavnom su lične prirode. Najvažniji, a verovatno i najbolje prikriiven, je osećaj pripadnosti. Ljudi su društvena bića iako programeri možda ne izgledaju tako. Svi projekti su pre svega zajednice: imaju svoj karakter, pravila ponašanja i istoriju. Neki su prestižni, neki su veliki, neki su nevidljivi. Želimo da budemo deo nekih od njih.

Deljenje koda

Sledeći razlog koji je vrlo čest je deljenje koda. Ljudi vole da se pohvale svojim delom. Programi jesu autorska dela, imaju različitu namenu, ali i različitu vrednost. Programeri vole da se pohvale i uglavnom to rade ćutke. Jedan od predloga za osnivanje uspešne zajednice je da se započne rad samostalno, objavi javno što pre i nastavi sa radom. Tek onda će se drugi polako priključivati i nastajće zajednica koja će se dalje brinuti o životu tog projekta.

Lokalizacija

Programi koji su prevedeni ili potpuno lokalizovani za naše područje mogu da budu



Razvoj slobodnog softvera

u ozbiljnoj prednosti u odnosu na druge, možda čak i naprednije programe. Postoji veliki interes za rad na slobodnom softveru upravo zbog ovog razloga.

Podrška

Zajednice formirane oko slobodnog softvera su zajednice u kojima najčešće vladaju meritokratija i znanje. Svi ti članovi su u nekom trenutku dobili podršku od drugih članova zajednice i često osećaju potrebu da tu uslugu uzvrate. Pomagati drugima znači napredovanje u hijerarhiji zajednice i sticanje jednog oblika ugleda i poštovanja.

Testiranje

Razvoj bilo kakvog softvera podrazumeva mnogo aktivnosti pored samog pisanja koda. Tu je i pisanje dokumentacije, pisanje predloga za nove mogućnosti, testiranje itd. Korisnici su ti koji mogu da daju korisnu informaciju o tome kako bi projekat trebalo da napreduje. Pored toga, korisnici mogu da pruže informacije o tome koji delovi programa se ne ponašaju očekivano. I ovo je način na koji neko doprinosi razvoju projekta. Sledi više konkretnih detalja kako se uključiti u razvoj jednog projekta.



Oslobađanje

Uključivanje u projekat: korak po korak

Komunikacija

Prvi korak bi trebalo da bude upoznavanje sa zajednicom. Veb-sajt jednog projekta ili zajednice može da oda pogrešnu sliku o njoj. Kao primer možemo slobodno da navedemo početne stranice LUGoNS ili LIBRE! projekata, koje nisu preterano zanimljive, iako su obe zajednice vrlo aktivne. Ključ je u tome što zajednice mogu da budu najaktivnije na nekim drugim kanalima za komunikaciju. Recimo, slobodno se može reći da je LUGoNS zajednica najaktivnija na svom *IRC* kanalu, a to nije toliko primetno nekome ko poseti sajt ove zajednice. Slično je i sa mnogim drugim zajednicama. Treba zaviriti i na *IRC* kanale, forume, dopisne liste (eng. *mailing list*) itd. Ovi kanali su pravo mesto za upoznavanje zajednice, učenje, traženje i pružanje podrške i informisanje. Pomagati drugima da reše problem u vezi sa jednim projektom ujedno znači i pružiti podršku tom projektu.

Dokumentacija

Problemi se često ponavljaju i postaje zamorno da se znanje prenosi „usmenim” putem u formi razgovora. Zbog toga postoji dokumentacija u raznim oblicima koja zahteva kako pisanje, tako i održavanje. Rad na dokumentaciji može značiti i uređivanje hijerarhije u dokumentaciji, uklanjanje zastarelih informacija i usmeravanje korisnika na čitanje dokumentacije. Dokumentacija u *OSS* (eng. *open source software*) projektima je često dostupna u formi vikija (eng. *wiki*). U poslednjih nekoliko godina, popularna je i forma elektronskih knjiga (*epub*, *PDF*, *web*...). Još jedan oblik dokumentacije koji treba pomenuti je sajt <https://readthedocs.org> koji prikazuje dokumentaciju zapisanu u kodu u jednom prijatnijem obliku. Svi ovi vidovi dokumentacije zahtevaju, pisanje i održavanje, te predstavljaju sjajan način za početak rada na nekom projektu.

Lokalizacija

Lokalizacija uglavnom podrazumeva prevod programa ili dokumentacije. Deo teksta koji se koristi u programima i može da se prevede smešten je u posebnim fajlovima. Često je reč o tzv. **.po** fajlovima i *GNU gettext* standardu. Tada se prevod programa vrši tako što se preuzmu **.po** fajlovi za odgovarajući jezik, učitaju se u uređivač za **.po** datoteke. Tada može biti dodat nov sadržaj ili izmenjen postojeći. Drugi, moderniji način koji se takođe oslanja na *gettext* su veb-servisi za uređivanje **.po** fajlova. Korisnici mogu pomoću internetskog pregledača da pristupe



Razvoj slobodnog softvera

takvom servisu, dodaju predloge prevoda i ocenjuju druge prevode. Ovakav način znatno olakšava spajanje svih izmena u jednu celinu, kontrolu prevoda i komunikaciju među korisnicima koji se bave prevodom. Neki od takvih sajtova su <https://transifex.org> i <https://launchpad.net>.

Prijavljivanje problema

Oni koji su spremni da još više „zaprljaju ruke“ mogu da pomognu testiranjem najnovijih verzija svog omiljenog programa. Mnogi projekti imaju takozvene najtli bildove (eng. *nightly build*). U takvoj verziji programa se nalaze najsvežije izmene. Ideja je da se noću sve izmene u kodu sakupe i distribuiraju korisnicima koji žele da se bave testiranjem. Od korisnika se očekuje da primećene probleme prijave na sajtu za praćenje problema (eng. *issue/bug tracker*). Problemi se ne moraju odnositi na najtli bild već može biti u pitanju bilo koja verzija. Problemi mogu da budu najrazličitijeg karaktera - od pogrešnog prevoda, preterane upotrebe resursa, do neočekivanog ponašanja programa. Različiti projekti očekuju prijave o greškama na različitim mestima. Neki projekti za to koriste svoje privatne „bagzile“ (eng. *bugzilla*) ili koriste usluge drugih sajtova, kao što su lončped (eng. *launchpad*), github (eng. *github*) ili sors fordž (eng. *source forge*). Najbolje je zaviriti u dokumentaciju ili se konsultovati sa ostalima o tome gde se nalazi taj sistem. Važno je upamtiti da prvo treba da se proverí da li je problem već prijavljen, a potom detaljno opisati karakter problema, verziju programa i sistema na kom se problem dogodio, i način na koji se može reprodukovati takav problem.

Održavanje paketa

U prethodnom broju bilo je reči o načinima kako se programi mogu distribuirati i objašnjeni su pojmovi repozitorijuma, paketa i njihov značaj. Slobodan softver se distribuira u paketima različitog tipa. Čest je slučaj da mnogo distribucija koristi isti tip paketa. Međutim, distribucije imaju svoje repozitorijume. Ljudi koji se bave održavanjem neke distribucije se, između ostalog, brinu i o tome da se softver lepo uklopi, da ne postoje konflikti i da sve međuzavisnosti budu ispoštovane. Ti ljudi se u žargonu zovu paketari, a njihov posao je pravljenje (pakovanje) i održavanje paketa. Paketari su često organizovani u timove koji se brinu o određenom skupu paketa slične namene. Pakovanje softvera zahteva veliku odgovornost i iskustvo. Neko bi mogao da kaže da je čak jednostavnije napraviti izmenu u kodu nego pravilno spakovati taj kod. Zbog toga paketi pored imena autora programa sadrže i ime osobe koja je napravila paket. Pravljenjem paketa se podržavaju i distribucije i

Oslobađanje

programi i ovo je sjajan način da se pomogne zajednicama, a u pitanju je i velik izazov.

Pisanje koda

Pod razvojem slobodnog softvera uglavnom se podrazumeva programiranje. Rad na kodu zahteva pronalaženje ažurnog repozitorijuma koji sadrži kod, preuzimanje koda odgovarajućim VCS-om (eng. *versioning control system* - naprimer, git). Zatim kod može da bude izmenjen, a izmena poslata na reviziju. Izmene se tradicionalno razmenjuju zakrpama ili pečevima (eng. *patch*), a danas se često sreće i razmena izmena u obliku pul rikvesta (eng. *pull request* - zahtev za prihvatanje izmena). Pri pisanju izmena u kodu treba voditi računa o dogovorima i standardima za uređivanje koda. Često postoje zapisane smernice za pisanje izmena (eng. *contributing guide*).

Osnivanje projekata i zajednica

Sasvim drugi nivo podrške nekom projektu bi bio osnivanje lokalne zajednice. Lokalne zajednice se uglavnom bave pružanjem podrške. Međutim, mogu se baviti i lokalizacijom, i drugim stvarima. Kod osnivanja zajednica oko već postojećih projekata treba imati na umu da zajednice često nadžive projekat; kvalitet zajednice je od ključne važnosti za kvalitet ostalih delova projekta. Preduslov za dolazak novih ljudi u zajednicu je infrastruktura pogodna za rad. To podrazumeva infrastrukturu koja je standardna, lako dostupna i redovno održavana. Nadamo se da smo vas ohrabрили i da ćete pomoći nekom projektu čije proizvode često koristite. Ukoliko ovaj tekst nije bio dovoljan, slobodno nas kontaktirajte i pomoći ćemo vam da se na pravi način uključite u željeni projekat.





Vaš posao, open-sors posao (3. deo)

Gugl - Savršena reklamna agencija

Autor: Dejan Maglov

Prošlo je skoro godinu dana otkako je ovaj važan serijal prekinut zbog sprečenosti izvornog autora serijala Marka Kažića da ga nastavi. Međutim, posle dugog čekanja nastavljamo sa novim autorom. Novi autor će doneti malo drugačiji koncept i redosled u analizu ove oblasti i nadamo se da vam to neće smetati.

U dvadeset i četvrtom broju LIBRE! časopisa bio je objavljen uvod u ovaj serijal i najavljen je članak o kompaniji Red Het (objavljen već u sledećem, dvadeset i petom broju). Takođe je bio najavljen i članak o Kanonikalu (*Canonical*) za neki od narednih u serijalu, koji je, nažalost, naglo prekinut pa taj članak nikada nije objavljen.

Novi autor je odlučio da malo promeni redosled i da priču o Kanonikalu odloži za neki naredni put, dok će sada biti reči o najjačoj kompaniji koja je svoje poslovanje zasnovala na slobodnom softveru - Guglu (*Google*)

Novi početak

Na novom početku serijala moramo da se izvinimo čitaocima što neće biti korišćena zvanična terminologija iz sveta ekonomije. Ovi modeli finansiranja imaju svoja zvaničnu terminologiju u ekonomiji, ali s obzirom na to da autor nije ekonomista kao ni većina čitalaca, ona, radi jednostavnosti teksta, neće biti korišćena u ovom serijalu.

Primarna ideja je da se u ovom serijalu identifikuju modeli i da se pokaže da je moguće razviti biznis u svetu slobodnog softvera i bez direktne prodaje softvera

Slobodni profesionalac

kao intelektualne svojine.

Slobodan softver nije produkt intelektualnog rada niti pojedinca niti jedne kompanije. On je produkt velike zajednice tako da je nemoguće zaštititi intelektualna prava na taj softver i tako stvoriti preduslov za prodaju. Pa ipak, programeri, nosioci projekata, samo su ljudi koji imaju svoje potrebe i ne mogu živeti od ideje i filozofije. Za dalji, brži razvoj projekata slobodnog softvera potreban je novac. Nemogućnost direktne prodaje softvera zahteva više maštovitosti u kreiranju biznisa na osnovi slobodnog softvera.

U tekstu o kompaniji Red Hat, predstavili smo savršen model biznisa koji se ne zasniva na prodaji GNU/Linuksa, nego na uslugama održavanja, implementacije, edukacije, prvenstveno u poslovnom okruženju. Dokaz da ovaj model, barem u Americi, funkcioniše, je činjenica da je još pre 2 godine kompanija Red Hat preskočila magičnu cifru od milijardu dolara profita.

Gugl



Još uspješniji model biznisa je razvila kompanija Gugl. Bespotrebno je dokazivati uspešnost ove kompanije. Njeni proizvodi kao što su internet-pretraživač, internet-aplikacije i servisi, Gugl Krom (*Google Chrome*) internet-pregledač, Gugl Krom OS, Android OS za mobilne uređaje (ima izmenjeno Linuks jezgro) i drugi - slobodni su softveri ili servisi i pritom potpuno besplatni. Gledajući samo softvere i servise koje Gugl razvija i podržava moglo bi se zaključiti

da se radi o pravom FLOSS projektu. Nažalost, komercijalizacija po Guglovom modelu je potpuno iskrivila suštinu filozofije slobodnog softvera.

Kada je Gugl u pitanju, sasvim je jasno odakle potiču njegovi prihodi. Gugl je postao savršena globalna reklamna agencija. Popularnost svojih proizvoda je



Open-sors posao

iskoristio da naplaćuje i plasira reklamni materijal.

Svi znamo da Gugl nudi reklame svakom korisniku ali se mora priznati da to rade i sa dosta diskrecije. Još se nikada nije desilo da korisnik za pretragu koristi Gugl pretraživač, a da kao rezultat dobije samo plaćene reklame koje nemaju veze sa predmetom pretraživanja. Umetnost Guglovog reklamiranja je plasiranje reklame na takav način da korisnik nije svestan da mu je reklama podmetnuta, već da ima utisak da je sam pronašao rešenje svog problema.

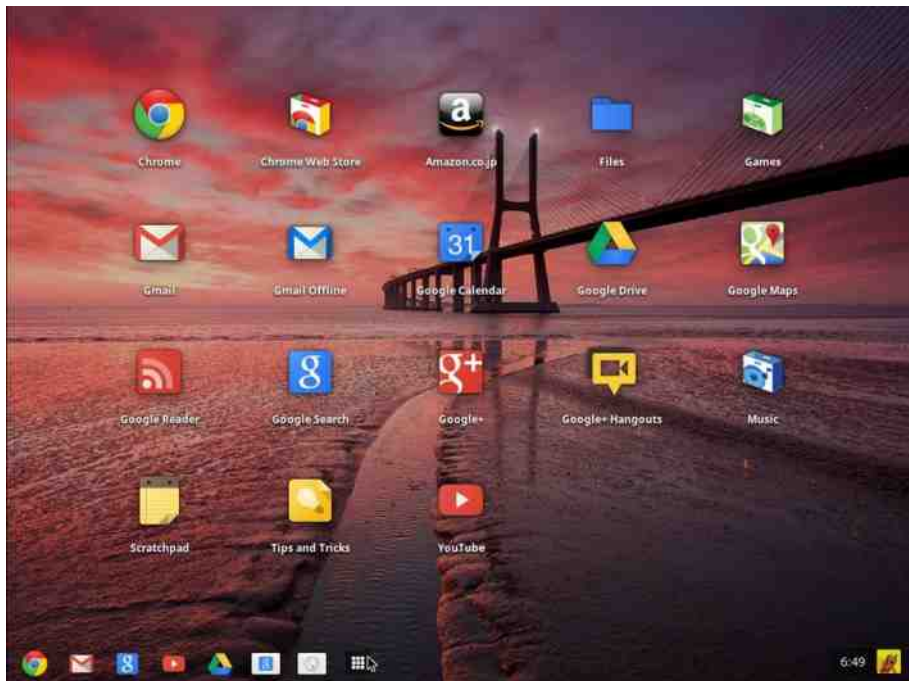
Gugl svoju superiornost u pretraživanju internet sadržaja bazira ne samo na dobrom poznavanju predmeta pretraživanja, nego i na dobrom poznavanju korisnika, praćenjem njegovih navika i interesovanja. Personalizovanje rezultata pretrage daje kvalitetnije rezultate za korisnika jer sa manje reči nalazi ono što ga zanima. Cena za to je zadiranje u privatnost korisnika. Možda nam ne bi smetalo ovo zadiranje u privatnost da se ne bojimo od zloupotrebe tih podataka i to ne samo od strane Gugla, nego i nekih drugih zainteresovanih strana koje bi mogle da preko Gugla dođu do njih.



Gugl je našao interes u FLOSS-u i pretvorio ga u alat koji će preusmeriti korisnike na njegov pretraživač i servise, čime je taj softver postao samo besplatni softver otvorenog koda, a slobodu korisnika je izgubio. Prema Gulu, budućnost desktop računara je u jeftinim mašinama bez tvrdog diska koji pokreće operativni sistem koji ima samo internet pregledač i koji u potpunosti preusmerava korisnika na Guglove već postojeće web aplikacije: Gugl poštu, Gugl drajv, Gugl dokumente, Gugl pikasu, Gugl mape, Gugl kalendar, Gugl plus, Gugl igre, Gugl internet

Slobodni profesionalac

prodavnicu... Na taj način korisnik bi poprilično uštedeo na uređaju i softveru, ali bi izgubio i ono malo prividne privatnosti koju ima sad.



Međutim, Gugl svoje korisnike ne tera ni na šta. Niko nije prisiljen da kupi Android telefon, koristi Krom internet-pregledač i Gugl pretraživač. Korisnici sami pristaju na to jer dobijaju kvalitetne proizvode i imaju utisak da su sami izabrali najbolje. Slobodni softver ima potencijal da bude najbolji, ali ga Gugl zloupotrevlja. Umesto potpune slobode, Gugl je svog korisnika strpao u matriks svet privida da sam odlučije o svemu dok je zapravo samo baterija za Gugl mašineriju.

* * *

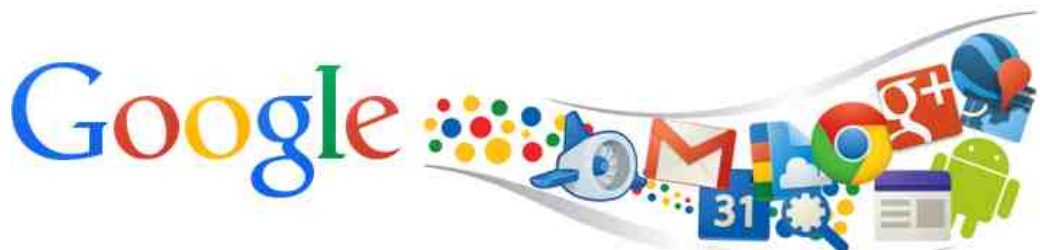
Gugl je troškove razvoja softvera prebacio sa korisnika na strane zainteresovane za



Open-sors posao



reklamiranje, kao i na njihove potrošače. Zadovoljni su projektanti softvera, korisnici softvera, proizvođači robe, jer lako dolaze direktno do kupca, a i kupci, jer lako dolaze do tražene robe. Gugl ne samo da tim svojim prihodima razvija svoj slobodni softver, nego i preko programa *Google Summer of Code* podržava i nezavisne projekte slobodnog koda. Stiče se utisak da je, ipak, i pored zloupotrebe, sam slobodni softver više na dobitku nego na gubitku. Otvoreni kod i antimonopolski zakoni mogu da isprave trenutno stanje u kojem Gugl postaje globalni Veliki brat.



Slobodni profesionalac



ANSIBLE (1. deo)

Autor: Stefan Nožinić

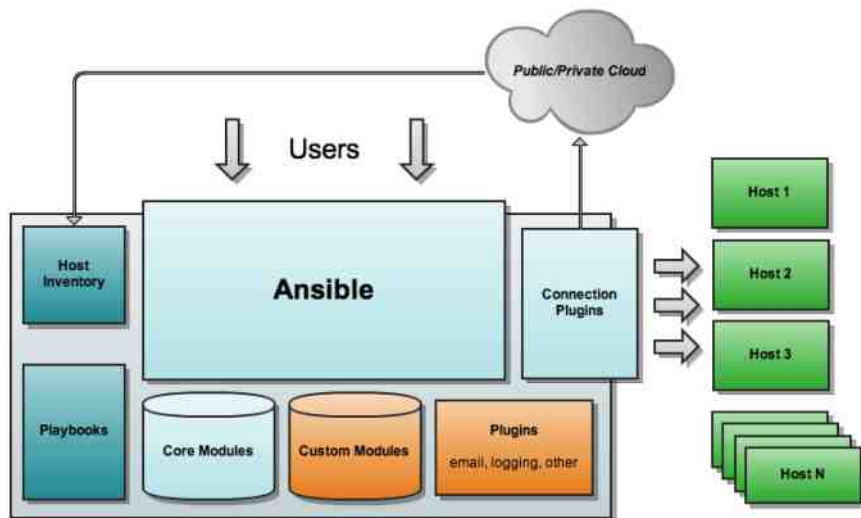
Uvod

Administracija kućnog sistema nikad nije predstavljala problem. Sve što ste ikada trebali da uradite jeste po koje ažuriranje i instalacija potrebnog softvera. Kada treba raditi administraciju ozbiljnijih sistema kao što su to brojni serveri, tu se već prelazi u drugi domen kompleksnosti. Ovde je namerno upotrebljena množina jer se želi naglasiti da je reč o više servera koje treba održavati. Tokom vremena su razvijeni mnogi alati za ovakav menadžment. Dosta takvih alata je zahtevalo instalaciju specifičnog softvera na sve servere kako bi se oni mogli spojiti sa glavnim računarom preko kojeg je administrator upravljao čitavom grupom servera. Ovo je na prvi pogled izgledalo sasvim prihvatljivo ali se vremenom pokazalo kao pogrešno. Softver je postajao sve kompleksniji i imao je sve više grešaka. Ovo je kasnije administratorima napravilo više problema nego što su imali pre.

Ansibl (*Ansible*) je jedan od retkih i kvalitetnih softverskih rešenja koje je često korišćeno i koje omogućava laku automatizaciju održavanja više servera. Pored toga, on ne zahteva znanje nekog skript-jezika za neke osnovne stvari. Takođe, ne zahteva instalaciju na svim serverima jer umesto svojih agenata na svakom serveru, on povezuje glavni server i ostale preko *SSH* (eng. *Secure Shell*) protokola. Ansibl koristi jezik za konfiguraciju koji je jednostavan i može se savladati bez mnogo učenja. Ovo omogućava administratorima da se fokusiraju



na probleme koji se tiču same strategije održavanja sistema umesto na kucanje koda.



Zbog ove jednostavnosti u konfiguraciji, Ansibl mogu koristiti programeri, menadžeri, sistemski administratori i ljudi ostalih vrsta profesija. Konfiguraciju je lako pisati, čitati ali i održavati posle nekog vremena.

Dakle, da sumiramo, Ansibl nam omogućava da kontrolišemo svoje servere sa jednog računara. Ako nam treba instalacija baze podataka na deset servera, nema problema - Ansibl nam to omogućava putem jednostavne konfiguracije i to možemo uraditi sa našeg laptop računara makar bili i kilometrima daleko od tih servera. Važno je samo da imamo *SSH* pristup tim serverima.

Instalacija

Ansibl je podržan u skoro svakoj distribuciji. Tako postoje paketi za Ubuntu, Debian, Arčlinuks, Džentu i ostale velike distribucije. U slučaju da ne postoji paket za vašu distribuciju, Ansibl možete preuzeti i instalirati pomoću alatke *pip*. Preporučujemo, ako imate vremena i volje, da u tom slučaju napravite paket za vašu distribuciju, prosledite to zajednici i time budete deo razvojnog tima slobodnog softvera. Nema bolje satisfakcije za jednog Linuks korisnika, zar ne?

Slobodni profesionalac

Ako, kojim slučajem, naiđete na problem tokom instalacije, ili ne znate uopšte kako da instalirate Ansibl na vaš računar, možete nas kontaktirati putem naše elektronske pošte koja vam je, nadamo se, do sada već poznata - [libre \[et\] lugons \[dot\] org](mailto:libre@lugons.org).

Još malo uvoda

Pre nego što krenemo sa osnovama, treba pojasniti kako Ansibl komunicira sa ostalim računarima. Ansibl koristi *SSH* protokol da komunicira sa ostalim serverima preko kojeg im govori koje komande izvršavaju. *SSH*, kao što verovatno znate, može se koristiti da se spojite na udaljeni server pomoću šifre ili pomoću ključa (*RSA / DSA*). Ansibl omogućava oba načina, ali takođe i, pored upotrebe *SSH* protokola, konekciju na *chroot*, *lxc*, *jail* ili lokalno.

Vaše udaljene servere navodite u posebnom fajlu tako da Ansibl tačno zna kome treba da pristupi.

Prvi primer

Sada, kada imamo malo predznanja o tome šta Ansibl radi i čemu konkretno služi, možemo se baviti tehničkim stvarima i pogledati kako to da konkretno iskoristimo u praksi.

Pretpostavimo da imamo tri servera sa sledećim adresama: 192.168.1.3, a.home.net, b.home.net i da te servere želimo da administriramo odjednom.

U fajlu */etc/ansible/hosts* navodimo adrese tih servera u svakom redu posebno na sledeći način:

```
192.168.1.3
a.home.net
b.home.net
```

Pretpostavlja se da imate pristup tim serverima preko *SSH* protokola, odnosno da imate njihove ključeve (u ovom primeru pretpostavljamo da želimo koristiti ključeve umesto šifre).

Sada možemo „pingovati“ sve naše servere navedene u gore opisanom fajlu.



```
ansible all -m ping
```

Ansibl će se povezati na udaljene servere koristeći korisničko ime sa vašeg sistema odakle pokrećete Ansibl. Ovo je moguće promeniti, ali to nećemo objašnjavati u ovom tekstu. Ako vas zanimaju konkretno neki detalji, možete nas kontaktirati.

Ako smo ustanovili da sve do sada radi kako smo očekivali, možemo pokrenuti komandu na svim serverima na sledeći način:

```
ansible all -a "/bin/echo hello"
```

Jednostavno, zar ne?

Za kraj

Treba naglasiti da je Ansibl mnogo više od pokretanja komandi na više računara. Pravu moć ovog alata tek želimo da vam pokažemo, ali, kao što znate, ne možemo toliko toga da stavimo u jedan članak. Treba ostaviti prostora i za druge teme. Ono što možemo da vam obećamo jeste da ćemo se potruditi da u narednim tekstovima o ovom alatu damo što više primera i zanimljivih informacija. Nadamo se da ćete i vi ubuduće nekomе prenositi sjajne utiske o ovom alatu kao što mi sada prenosimo vama.

Ako imate neko pitanje, predlog, zanimljivost ili, pak, želite da nam ukažete na tehnički ili bilo kakav drugi propust, kontaktirajte sa nama na [libre \[et\] lugons \[dot\] org](#) i mi ćemo dati sve od sebe da budemo još bolji.



ANSIBLE

Šifrovani čet (2. deo)

Niltok

Autor: Kriptopank

Kao što smo u prošlom broju pisali o novoj i fantastičnoj Subrosi, ovde ćemo pomenuti jedno novorođenče u svetu brzog šifrovanog dopisivanja. Da se ne lažemo, nije ni blizu po praktičnosti i pametnim rešenjima kao što je to slučaj kod Subrose, ali uvek je lepo osetiti da se ljudi trude u ovoj oblasti stvarajući nove i inovativne aplikacije - takozvane aplikacije-debitante.

Kad smo već kod Subrose, njihov sajt je već neko vreme nedostupan (<http://goo.gl/Nf8neL>), ali to ne bi trebalo da bude naročiti problem jer je moguće podići Subrosa server iz udobnosti vašeg doma (eng. *selfhost*), što su neki već uradili kako bi sebi priuštili veću sigurnost, privatnost i, što je najvažnije, nezavisnost od glavnog udaljenog servera Subrose.

Iza imena „Niltok“ (*Niltalk*), koje veoma dobro opisuje ovu veb aplikaciju, krije se





Šifrovani čit

veoma prosta ideja za šifrovana dopisivanja, iskovana preko mreže u programskom jeziku Golang.

Stvari funkcionišu ovako: Kada posetite njihovu veb-stranicu (<http://goo.gl/N4wVUO>), videćete polje u koje treba da unesete šifru, nakon čega se pravi vaša nova soba za ćaskanje i jedinstveni link koji vodi njoj. Ta šifra će biti korišćena za šifrovanje teksta svih poruka i biće potrebna kao identifikacija za pristup sobi za ćaskanje koju ste upravo napravili. Sada morate da drugima pošaljete ovaj link i šifru. Ovde se javlja i jedan problem koji se pojavljivao i kod novijih sigurnih i-mejl provajdera poput Protonmejla (*Protonmail*) i Skiptmejla (*SCYPTmail*), a to je da tu šifru sada nekako morate javiti ili poslati drugima sa kojima želite da sigurno ćaskate, ali tako da to niko osim njih na internetu ne zna.

Logično se nameće rešenje da u te svrhe koristite šifrovanu poštu, ali verujemo da bi to delovalo malo odbojno i moglo bi da obeshrabri ljude da koriste ovu aplikaciju, jer ko zna kada i da li drugi pregledaju svoju internetsku poštu. Ovde treba obratiti pažnju i na to da od trenutka kada se nova soba za ćaskanje kreira, ona će se zatvoriti posle sto dvadeset minuta ukoliko joj do tada niko ne pristupi.

 niltalk



Free, instant, private, disposable chat

How does it work?

Niltalk lets you create instant, password protected chat rooms without the need to signup. Simply click the "Create" button, and share the unique chat URL with your peers.

All communication happens over SSL. Niltalk doesn't record or log IP addresses, messages, or peer handles anywhere.

Once a room is created, it has a lifetime of 120 minutes before the first login. Then, the room is automatically deleted after 10 minutes of inactivity (no messages exchanged).

While in a room, you or any of your peers can dispose of the room with the click of a button.

Why can any connected peer dispose of a room?

Niltalk is meant for holding short private conversations between groups of people who have mutually agreed to converse. There is no concept of ownership of a room, and introducing ownership complicates the otherwise simple privacy feature of instant disposal by any participant. This also means that Niltalk isn't really meant for starting conversations by opening up a room to

Internet mreže i komunikacije

Ovo, iako dobra predostrožnost, zapravo predstavlja prebacivanje problema sa sebe (Niltoka) na korisnike, koji u većini slučajeva ne razmišljaju o sigurnosti. Ne treba da pominjemo da bi upotreba nekog drugog vida šifrovanog dopisivanja za razmenu šifre za Niltok bila besmislena iz dva razloga. Prvi, ako je već taj vid šifrovanog ćaskanja siguran, zašto koristiti Niltok - i drugi, ako nije siguran, zašto preko njega razmenjivati tajne?

Naravno, ove sobe nisu u vlasništvu onoga ko ih kreira i on nema nikakvih privilegija u odnosu na druge sagovornike u istim.

Ali, recimo da ste nekako sigurno razmenili ključeve i sada se već lagodno dopisujete sa prijateljima, primetićete dugme „Dispoze” za napuštanje trenutne sobe. Takođe, soba će se automatski sama obrisati i time postati nedostupna nakon deset minuta bez ikakve aktivnosti (poruka) unutar iste. Niltok tvrdi da ne čuva nikakve logove IN adresa, a sva komunikacija sa Niltokom uvek ide preko sigurne SSL konekcije.

Niltok je samo aplikacija za ćaskanje i nema opcije poput privatnih poruka nekom sagovorniku na listi van glavnog kanala koji svi u sobi vide. Nema podrške ni za video, ni za deljenje fajlova, a cela audio podrška je zvučno obaveštenje o novoj poruci unutar sobe u kojoj ste.

Međutim, nije sve tako loše. Kao prvo, Niltok je (naravno) otvorenog koda sa AGPL3 licencom i dostupan je za preuzimanje sa Githuba (<http://goo.gl/ZeeEM0>), a druga pozitivna strana je da nema nikakvog registrovanja i ostavljanja bilo kakvih ličnih podataka. Možete ovu aplikaciju koristiti i preko Tor anonimne mreže; samo treba da smislite dobar način da društvu javite šifru (ovo nikako ne radite preko društvenih mreža).





Bootstrap

Prvi koraci

Autor: Zlatan Vasović

Uvod

U tridesetom broju LIBRE! časopisa upoznali smo vas sa Butstrapom (*Bootstrap*) i rekli smo da je to frejmvork (eng. *framework*) za veb-dizajn ili još jednostavnije rečeno, to je skup predefinisanih alata koji olakšavaju dizajn veb-prezentacija i veb-aplikacija.

Ovog puta idemo korak dalje i pokazujemo vam kako da počnete da koristite Butstrap u svojim projektima.

Prvi koraci sa Butstrapom nisu teški. Potrebno je preuzeti spremljene CSS i Džavaskript (*JavaScript*) Butstrap datoteke, odabrati neki od šablona - ili, ipak, „krenuti od nule” i napraviti sajt bez gotovog šablona.

Preuzimanje

Sve što je potrebno za preuzimanje Butstrapa je odlazak na <https://github.com/twbs/bootstrap/releases/>, zatim nalaženje potrebne verzije (poslednja stabilna verzija trenutno je 3.3.2) i preuzimanje „*dist*” datoteke, koja je među linkovima preuzimanja ispod verzije. Aktuelni *dist* (unapred spremljen) paket je <https://github.com/twbs/bootstrap/releases/download/v3.3.1/bootstrap-3.3.1-dist.zip>.

Sam svoj majstor

Nakon raspakivanja ZIP arhive u rut (eng. *root*) folder vašeg sajta i povezivanja sa HTML stranama sajta naredbama

```
<link href="css/bootstrap.min.css" rel="stylesheet">
<script src="js/bootstrap.js"></script>
```

- Butstrap je spreman za korišćenje.

Butstrap se može učitati i putem takozvanog *CDN*-a (eng. *content delivery network*). Linkove do *CSS* i *JS* datoteka, kao i odgovarajući kod za njih, možemo naći na <http://www.bootstrapcdn.com/>.

Dokumentacija

Dokumentacija za Butstrap (<http://getbootstrap.com/getting-started/>) sadrži sve što je potrebno korisnicima da počnu izradu veb-sajtova.

Bootstrap Getting started CSS Components JavaScript Customize Expo Blog

Getting started

An overview of Bootstrap, how to download and use, basic templates and examples, and more.

Download

Bootstrap (currently v3.3.4) has a few easy ways to quickly get started, each one appealing to a different skill level and use case. Read through to see what suits your particular needs.

- Bootstrap**
Compiled and minified CSS, JavaScript, and fonts. No docs or original source files are included.
[Download Bootstrap](#)
- Source code**
Source Less, JavaScript, and font files, along with our docs. **Requires a Less compiler and some setup.**
[Download source](#)
- Sass**
Bootstrap ported from Less to Sass for easy inclusion in Rails, Compass, or Sass-only projects.
[Download Sass](#)

Bootstrap CDN

The folks over at [MaxCDN](#) graciously provide CDN support for Bootstrap's CSS and JavaScript. Just use these [Bootstrap CDN links](#).

- Download
- What's included
- Compiling CSS and JavaScript
- Basic template
- Examples
- Tools
- Community
- Disabling responsiveness
- Migrating from 2.x to 3.0
- Browser and device support
- Third party support
- Accessibility
- License FAQs
- Translations
- Sign up

Postoji više prevoda celog sajta dokumentacije, ali među njima nije srpski, tako da



će vam biti neophodno određeno znanje engleskog jezika za čitanje dokumentacije.

Podjeljena je na četiri dela - „Getting started” (kako početi), „CSS” (osnovne CSS komponente), „Components” (dodatne CSS komponente) i „JavaScript” (Džavaskript komponente).

U delu „Getting started” se nalaze linkovi za preuzimanje, sadržaj Butstrapa, lista poznatih grešaka, postupak prelaska sa Butstrapa 2 na verziju 3, kao i nekoliko sekcija za napredne korisnike Butstrapa. Ostala tri dela dokumentacije prilično detaljno predstavljaju sve komponente Butstrapa.

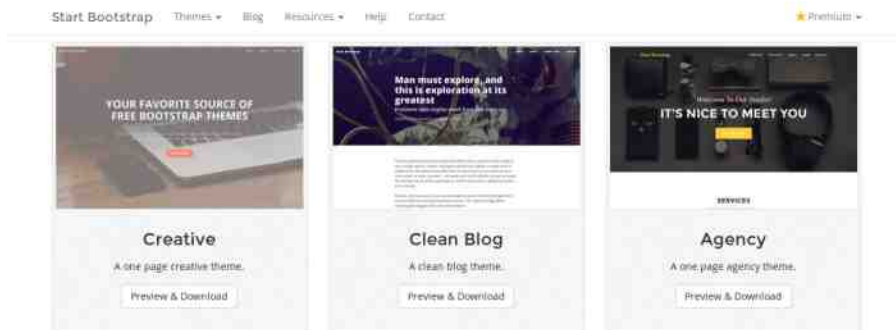
Šabloni

#	Header	Header	Header	Header
1.001	Lorem	ipsum	dolor	elit
1.002	amet	consectetur	adipiscing	elit
1.003	integer	nec	odio	praesent
1.003	libero	Sed	cursum	ante
1.004	dapibus	diam	Sed	nisi

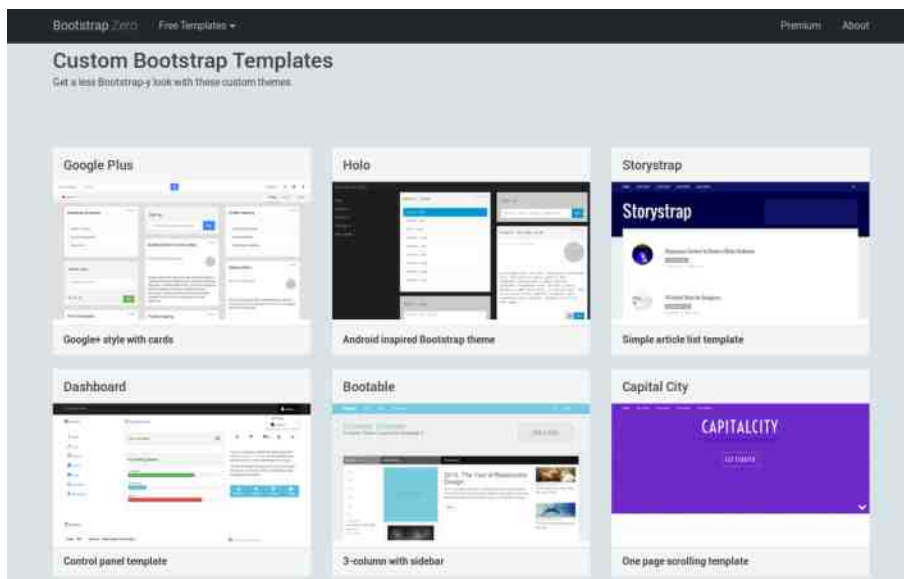
Ipak, nije svima lako snaći se „isprve”. Mnogima će pri izradi sajtova pomoći veliki broj šablona, bilo zvaničnih (<http://getbootstrap.com/getting-started/#examples>), bilo korisničkih, među kojima ima i onih koji se plaćaju.

Pomenućemo nekoliko najpoznatijih sajtova sa besplatnim korisničkim šablonima:

Sam svoj majstor



- <http://startbootstrap.com/>
- <https://wrapbootstrap.com/>
- <http://bootswatch.com/>
- <http://bootstrapzero.com/>



Sada smo spremni za sledeći korak - izradu veb-sajta.



BeagleBone Black Rev C

Vodič od prvog dana (6. deo) - Biglbon Blek kao Tor egzit

Autor: Nenad Marjanović



beaglebone

U nekoliko prethodnih brojeva pisali smo o instalaciji i podešavanjima Biglbona revizije C korišćenjem Debian Linuks distribucije. Danas tačku stavljamo na sigurnost Tor servera ali i samog operativnog sistema. Posle instalacije svakog servera i nakon samo nekoliko sati, a i ranije, proverom logova možemo zaključiti da smo već na meti hakera. Ovo se dešava zbog prisutnosti velikog broja mrežnih skenera i alata za brutalno prijavljivanje na *SSH* servis. Nažalost, ovi napadi traju godinama i verovatno će još dugo biti tu. Analizom prijavnih logova, koji se u ovom slučaju nalaze u **/var/log/auth.log** fajlu, uvek možemo utvrditi da li je neko pokušavao da nas napadne. Komande koje nam mogu olakšati proces provere o pokušajima napada na naš server su sledeće:

```
cat /var/log/auth.log | grep failed  
cat /var/log/auth.log | grep failed | wc -l
```

Kratak opis ovih komandi je sledeći:

- cat** prikazuje zadržaj fajla
- grep** traži željenu reč
- wc -l** prikazuje ukupan broj

Primer na tek instaliranom serveru nakon trideset minuta aktivnosti: SLIKA 1

Hardver

U prethodnom broju smo govorili o promeni porta za SSH konekciju, ali danas

```
root@sd-33602:~# cat /var/log/auth.log | grep failed | wc -l
45
root@sd-33602:~#
```

nastavljamo priču upotrebom geolokalizacijskog softvera. Ovaj softver vrši restrikciju pristupa serveru prema zemljama koje odaberemo. Za ovo rešenje su nam potrebne dve **baš** (eng. *bash*) skripte i dva paketa. Ceo proces započinjemo instalacijom **geoiip** paketa i baze.

```
apt-get install geoiip-bin geoiip-database
```

Da bismo proverili uspešnost instalacije:

```
geoiiplookup pentester.iz.rs
```

```
root@sd-33602:~# geoiiplookup pentester.iz.rs
GeoIP Country Edition: FR, France
root@sd-33602:~#
```

Prvu skriptu kreiramo u **/usr/local/bin** dosijeu i podešavamo dozvole za njeno izvršavanje.

```
nano /usr/local/bin/sshfilter.sh
```

Zatim kopiramo sledeći sadržaj:

```
#!/bin/bash
# UPPERCASE space-separated country codes to ACCEPT
ALLOW_COUNTRIES="RS FR" f [ $# -ne 1 ]; then
  echo "Usage: `basename $0` <ip>" 1>&2
  exit 0 # return true in case of config issue
fi
COUNTRY=`/usr/bin/geoiiplookup $1 | awk -F " " '{ print $2 }' |
awk -F "," '{ print $1 }' | head -n 1`
[[ $COUNTRY = "IP Address not found" || $ALLOW_COUNTRIES =~
```



```
$COUNTRY ]] && RESPONSE="ALLOW" || RESPONSE="DENY" fi [ $RESPONSE
= "ALLOW" ]
then
  exit 0
else
  logger "$RESPONSE sshd connection from $1 ($COUNTRY)" exit 1
fi
```

Potrebno je izmeniti **ALLOW_COUNTRIES** u zavisnosti iz koje se zemlje redovno konektujemo na server. Za Srbiju je to **RS**, za Francusku **FR** itd. Poštujemo velika slova i razmak, a pišemo bez zarezova ukoliko unosimo više zemalja. Podešavamo dozvolu za izvršavanje skripte:

```
chmod +x /usr/local/bin/sshfilter.sh
```

Zaključavanje SSH-a

Da bismo omogućili upotrebu geolokalizacije, moramo izmeniti dva sistemiska konfiguraciona fajla.

```
nano /etc/hosts.deny
```

I na kraju dodajemo:

```
sshd: ALL
```

Zatim menjamo sadržaj u sledećem fajlu:

```
nano /etc/hosts.allow
```

I na kraju fajla dodajemo:

```
sshd: ALL: aclexec /usr/local/bin/sshfilter.sh %a
```

Provera efikasnosti sistema

Ukoliko smo autorizaciju za *SSH* podesili samo na Srbiju, dobićemo sledeće rezultate:

Hardver

```
/usr/local/bin/sshfilter.sh 8.8.8.8  
April 13 16:02:37 pi root: DENY sshd connection from 8.8.8.8 (US)
```

Da bismo proverili efikasnost skripte nakon nekoliko minuta, pokrećemo:

```
cat /var/log/messages | grep DENY
```

Dobićemo sledeći rezultat:

```
April 13 17:59:54 pi logger: DENY sshd connection from 82.221.102.185 (IS)  
April 13 17:47:54 pi logger: DENY sshd connection from 220.227.123.122 (IN)  
April 13 17:43:51 pi logger: DENY sshd connection from 221.229.166.252 (CN)  
April 13 16:49:04 pi logger: DENY sshd connection from 221.208.245.226 (CN)
```

Ažuriranje *GeoIP* baze

Za ovo koristimo još jednu skriptu i **crontab** funkciju:

```
nano /usr/local/bin/updategeoip.sh
```

Kopiramo sledeći sadržaj:

```
#!/bin/bash  
cd /tmp  
wget -q  
http://geolite.maxmind.com/download/geoip/database/GeoLiteCountry/GeoIP.dat.gz  
if [ -f GeoIP.dat.gz ]  
then  
    gzip -d GeoIP.dat.gz  
    rm -f /usr/share/GeoIP/GeoIP.dat  
    mv -f GeoIP.dat /usr/share/GeoIP/GeoIP.dat  
else  
    echo "The GeoIP library could not be downloaded and updated" i
```

Podešavamo dozvolu za izvršavanja fajla:

```
chmod +x /usr/local/bin/updategeoip.sh
```




Ovu bazu možemo ažurirati jednom mesečno i to radimo na sledeći način:

```
crontab -e
```

U fajl upisujemo sledeću liniju:

```
0 6 15 * * /usr/local/bin/updategeoip.sh
```

Ovim podešavanjem imaćemo nadogradnju svakog petnaestog u mesecu u šest sati. I za kraj ćemo rešiti curenje *DNS* informacija pri poseti sajtova od strane korisnika našeg Tor servera. Za te potrebe menjamo Tor konfiguracioni fajl i na kraj fajla dodajemo sledeći sadržaj:

```
AutomapHostsOnResolve 1
AutomapHostsSuffixes .exit, .onion
```

Za kraj ponovo pokrenemo *SSH* i ponovo učitamo Tor:

```
service ssh restart && service tor reload
```

Za migraciju Tor servera obavezno je sačuvati **/etc/tor/torrc** fajl i sadržaj **/var/lib/tor/keys** direktorijuma. U suprotnom gubimo rang u Tor mreži i krećemo ispočetka.

Ovim člankom završavamo pisanje o Tor serveru na Biglbon reviziji C i prelazimo na poglavlje veb servera i instalaciju najsigurnijeg veb servera današnjice - Hijavata (eng. *Hiawatha*).

Do čitanja.



Congratulations. You are using Tor.



GNU 30th

The GNU Manifesto

What's GNU? Gnu's Not Unix!

GNU, which stands for Gnu's Not Unix, is the name for the complete Unix-compatible software system which I am writing so that I can give it away free to everyone who can use it.

(1) Several other volunteers are helping me. Contributions of time, money, programs and equipment are greatly needed.

So far we have an Emacs text editor with Lisp for writing editor commands, a source level debugger, a yacc-compatible parser generator, a linker, and around 35 utilities. A shell (command interpreter) is nearly completed. A new portable optimizing C compiler has compiled itself and may be released this year. An initial kernel exists but many more features are needed to emulate Unix. When the kernel and compiler are finished, it will be possible to distribute a GNU system suitable for program development. We will use TeX as our text formatter, but an nroff is being worked on. We will use the free, portable X Window System as well. After this we will add a portable Common Lisp, an Empire game, a spreadsheet, and hundreds of other things, plus online documentation. We hope to supply, eventually, everything useful that normally comes with a Unix system, and more.

GNU will be able to run Unix programs, but will not be identical to Unix. We will make all improvements that are convenient, based on our experience with other operating systems. In particular, we plan to have longer file names, file version numbers, a crashproof file system, file name completion perhaps, terminal-independent display support, and perhaps eventually a Lisp-based window system through which several Lisp programs and ordinary Unix programs can share a screen. Both C and Lisp will be available as system programming languages. We will try to support UUCP, MIT Chaosnet, and Internet protocols for communication.

GNU is aimed initially at machines in the 68000/16000 class with virtual memory, because they are the easiest machines to make it run on. The extra effort to make it run on smaller machines will be left to someone who wants to use it on them.

To avoid horrible confusion, please pronounce the g in the word "GNU" when it is the name of this project.

Gnu's Not Unix

