

# KORIŠĆENJE OTVORENOG SOFTVERA U NASTAVI INFORMATIKE

Nemanja Milošević<sup>1</sup>

1: Univerzitet u Novom Sadu, Prirodno-matematički fakultet, Departman za matematiku i informatiku, 21000, Novi Sad, Srbija  
elektronska pošta: [nmilosev@dmi.uns.ac.rs](mailto:nmilosev@dmi.uns.ac.rs)

## REZIME

Odabir softvera koji studenti informatike kod visokoškolskih ustanova svakodnevno koriste u nastavi je izuzetno važan. Pored ispunjenja funkcionalnih i tehničkih zahteva, softver koji se koristi u nastavi mora ispuniti i veoma važan treći zahtev: mogućnost slobodnog korišćenja. Softver koji se koristi u nastavi, ukoliko slobodan, studenti mogu koristiti i nakon studija u profesionalne svrhe. Korišćenje slobodnog i otvorenog softvera takođe upoznaje studente sa kulturom deljenja koda i prednostima koje ovaj vid razvoja donosi. U ovom radu biće opisane prednosti ovakvog pristupa izvođenju nastave na konkretnom primeru iz prakse – izvođenju nastave informatike na Departmanu za matematiku i informatiku Prirodno-matematičkog fakulteta u Novom Sadu.

**Ključne reči:** otvoren softver, slobodan softver, linux, nastava, edukacija.

## 1 Uvod - prednosti otvorenog softvera u nastavi

Korišćenje slobodnog softvera u nastavi ima nekoliko očiglednih prednosti, pogotovo za studente informatike. Mogućnost uvida u izvorni kod softvera koji student koristi je izuzetno dobro oruđe u edukaciji [3]. Studenti koje interesuju detalji implementacije mogu istraživati otvorene izvorne kodove i na taj način steći dodatno znanje o softveru koji koriste. Otvoren softver je takođe više prilagodljiv u odnosu na zatvoren (proprietary) softver i moguće su razne modifikacije po potrebi predavača ili studenata.

Još jedna prednost otvorenog softvera je problem upotrebnih dozvola odnosno licenci. Kod korišćenja zatvorenog softvera često se nailazi na razne probleme licenciranja. Za neke softvere licence su preskupe za visokoškolske ustanove a postoje i ekstremni slučajevi gde nije moguće dobiti odgovarajuću licencu kako bi se softver koristio u učionici. Kod otvorenog softvera ovakvi problemi ne postoje, studenti i nastavno osoblje mogu koristiti otvorene razvojne alate i u nastavi i za privatne potrebe. Ovakav pristup ima i još jednu veliku prednost za studente a to je mogućnost upotrebe istih razvojnih alata (npr. Eclipse) i po završetku studija. Na taj način studenti su po završetku studija spremni za industrijski razvoj softvera bez potrebe za upoznavanjem sa novim alatima.

Jedna od najvećih prednosti korišćenja otvorenog softvera je upoznavanje studenata sa kulturom otvorenog softvera [2]. Svakodnevnom upotrebom otvorenog softvera studenti se upoznaju sa osnovnim konceptima kao što su licence, razlike među licencama i prava i obaveze koje dolaze sa njima, korišćenje softvera za upravljanje verzijama koda (npr. Git), način komunikacije u velikim timovima za razvoj softvera, korišćenje mejling listi, issue tracker softvera i slično. Ukratko, studenti se, već u toku studija, upoznaju sa načinom na koji se softver razvija u velikim timovima, i na taj način stižu

veoma vredno iskustvo koje bi inače teško stekli prilikom nastave. Slobodan softver svojim otvorenim pristupom razvoju i transparentnošću pri razvoju omogućava pojedincima (studentima) da steknu nezanemarljivo iskustvo u razvoju softvera na visokom nivou već u toku studija. [1] [5]

Pored upotrebe otvorenih razvojnih alata, nezamenljivo iskustvo je i upotreba otvorenih projekata drugih studenata, pojedinaca ili organizacija. Kroz razne kurseve/predmete na fakultetu, studenti se upoznaju sa načinima na koje mogu pronaći otvorene projekte, koristiti ih i isto tako doprineti tim projektima ili započeti svoje otvorene projekte. Studentima su baš ti otvoreni projekti (koliko god mali) izuzetno važni za karijeru, kao prvo iskustvo u samostalnom razvoju softvera i kao portfolio na osnovu kojeg mogu dobiti bolju priliku za zaposlenje.

## 2 Otvoreni softver u nastavi

U ovom delu diskutovaćemo upotrebu otvorenog softvera (razvojnih alata) u nastavi sa strane predavača. Neke od prednosti su već spomenute, a pored lakšeg upravljanja licencama i slobodne upotrebe bitno je napomenuti da je otvorenim softverom lakše i upravljati. Priroda otvorenog softvera je da ima jasno definisane otvorene interfejsne kojima je moguće pristupiti na neki način. Ovo može biti slučaj i sa zatvorenim (proprietary) softverom, ali iz našeg iskustva češće je lakše upravljati otvorenim softverom.

U nastavi na Departmanu za matematiku i informatiku Prirodno-matematičkog fakulteta koristi se veliki broj otvorenih razvojnih okruženja kao što su:

- Eclipse okruženje za Java/Java EE i Web razvoj <sup>1</sup>
- Eclipse okruženje za C/C++ razvoj i paralelno programiranje
- Papyrus okruženje za UML modelovanje (bazirano na Eclipse projektu) <sup>2</sup>
- Spyder3 okruženje za Python razvoj <sup>3</sup>
- Jupyter okruženje za Python razvoj <sup>4</sup>
- Visual Studio Code za Web razvoj <sup>5</sup>
- Android Studio za Android razvoj <sup>6</sup>
- pgModeler za upravljanje i modelovanje PostgreSQL bazama podataka <sup>7</sup>
- WireShark za nadzor računarskih mreža <sup>8</sup>

Lista ovde pomenutih alata i razvojnih okruženja je naravno delimična, naveden je samo podskup korišćenih alata. Konkretno, razvojni alati predstavljeni u listi koriste se na Katedri za informacione tehnologije i sisteme, gde autor ovog teksta predaje kao asistent. Predmeti koji su u nadležnosti pomenute katedre su: Baze podataka 1 i 2, Računarske mreže, Razvoj poslovnih sistema, Proces razvoja

---

<sup>1</sup><https://www.eclipse.org>

<sup>2</sup><https://www.eclipse.org/papyrus/>

<sup>3</sup><https://www.spyder-ide.org/>

<sup>4</sup><https://jupyter.org/>

<sup>5</sup><https://code.visualstudio.com/>

<sup>6</sup><https://developer.android.com/studio>

<sup>7</sup><https://pgmodeler.io/>

<sup>8</sup><https://www.wireshark.org/>

informativnih sistema, Veštačka inteligencija 1, Razvoj mobilnih aplikacija, Prostorne baze podataka i drugi.

Većina ovih razvojnih alata ima zatvorene alternative, poneke su čak nekada ranije bile korišćene na Departmanu ali nakon višegodišnje upotrebe otvorenih razvojnih alata možemo potvrditi da je ovde navedeni softver više nego adekvatan za upotrebu u nastavi (a i profesionalno). Pored ovih alata, gotovo svi kursevi koriste iz našeg iskustva nezamenljivu Moodle platformu za deljenje materijala i organizaciju kurseva.

Još jedna prednost korišćenja otvorenih razvojnih alata je što su često kompatibilni sa Linux operativnim sistemom. Korišćenje Linux otvorenog operativnog sistema u učionicama je od izuzetnog značaja i ima puno prednosti koje smo već pomenili kao što su licenciranje, bolja podrška za stariji i slabiji hardver, i lakše upravljanje (između ostalog).

Naša računarska učionica koristi Fedora Linux <sup>9</sup> operativni sistem koji je odabran iz više razloga. Potpuna otvorenost samog operativnog sistema i dostupnost najnovijih stabilnih verzija razvojnog softvera u repozitorijumima ovog operativnog sistema je glavni razlog zbog kojeg koristimo baš ovu distribuciju Linux operativnog sistema. Neki od primera biblioteka/softvera koje se najčešće koriste i za koje želimo da koristimo najnovije stabilne verzije su OpenJDK i Python. Takođe, dobra podrška za hardver, aktivnost zajednice i aktivna upotreba baš ove distribucije u edukaciji <sup>10</sup> su još neki razlozi zbog kojih smatramo da je Fedora dobar izbor. Fedora Linux je u našem računarskom centru (sa 25 radnih stanica) u upotrebi od 2016. godine i do danas nismo imali velikih poteškoća sa upotrebom. Koristimo osnovnu Workstation verziju sa GNOME Desktop okruženjem a verzija operativnog sistema se ažurira jednom godišnje (pred početak semestra). Za upotrebu “kancelarijskog” softvera su takođe zaslužni otvoreni projekti kao što su Chromium, Firefox, Geany i naravno LibreOffice.

Još jedna sporedna posledica upotrebe Linux-a u nastavi je što se studenti sve češće odlučuju na upotrebu Linux operativnog sistema (Fedora ili drugi) i na kućnim računarima što je važan korak unapred ka većoj upotrebi otvorenog i slobodnog softvera.

Na kraju, i dalje postoje načini za unapređenje našeg postojećeg sistema. U izvođenju nastave i dalje postoji mali broj komercijalnih alata koji se koriste u nedostatku dobre alternative. Na primer za modeliranje baza podataka uz pomoć ER dijagrama koristi se Oracle SQL Developer kao besplatan i Linux-kompatibilan ali softver zatvorenog koda. Takođe, ponekad postoji potreba i za korišćenjem veoma specifičnog softvera, na primer za konfiguraciju mrežne opreme kao što je MikroTik Winbox, kompatibilan samo sa Microsoft Windows operativnim sistemom, koji se u nastavi koristi uz pomoć softvera Wine. U ovakvim situacijama, gde je korišćenje komercijalnog softvera neizbežno, jedno od najboljih rešenja je razvoj novih alata otvorenog koda koji imaju traženu namenu.

### 3 Upravljanje otvorenim računarskim centrima

Upravljanje računarskim centrima/učionicama može biti naporan posao ukoliko ne postoji neka vrsta automatizacije. Neki od problema su: inicijalna instalacija softvera na sve radne stanice, konfiguracija svih računara, problemi ažuriranja verzija softvera, distribucija fajlova, nadzor studenata i slično.

Postoje razna oruđa koja se mogu koristiti za rešavanje ovih problema. Na primer, Ansible <sup>11</sup> ili Kickstart <sup>12</sup> se mogu koristiti za inicijalno podešavanje i instalaciju računara. Ansible se čak može

---

<sup>9</sup><https://getfedora.org/>

<sup>10</sup><https://labs.fedoraproject.org/en/python-classroom/>

<sup>11</sup><https://www.ansible.com/>

<sup>12</sup>[https://docs.fedoraproject.org/en-US/fedora/rawhide/install-guide/advanced/Kickstart\\_Installations/](https://docs.fedoraproject.org/en-US/fedora/rawhide/install-guide/advanced/Kickstart_Installations/)

koristiti i za kasnije upravljanje računarima. Naravno, ne možemo da ne spomenemo i iTALC [4]<sup>13</sup> otvoreni softver specifično napravljen za upravljanje učionicama.

Naše rešenje za upravljanje učionicama je jednostavnije ali i dalje efektivno. Koristimo internu Python biblioteku koja je wrapper oko SSH protokola. Python biblioteka rc3-scripts<sup>14</sup> je otvorena i može se koristiti u bilo kojoj učionici sa Linux radnim stanicima. Jedini zahtev je mogućnost korišćenja SSH protokola.

Biblioteka rc3-scripts ima sledeće mogućnosti:

- Provera koji računari su uključeni
- Gašenje svih računara u učionici
- Korišćenje x11vnc<sup>15</sup> za monitoring svih računara u učionici
- Mogućnost izvršenja bilo koje bash komande na svim računarima (najčešće korišćena mogućnost)

Korišćenjem ove biblioteke rešen je velik broj problema upravljanja računarskim centrom. Moguće je automatski ažurirati sav softver, instalirati novi softver, ažurirati operativni sistem, pratiti ekrane svih studenata tokom testova i slično. Takođe, pošto je moguće izvršiti bilo koju komandu na svim računarima istovremeno (paralelno), postoje i naprednije mogućnosti:

- Korišćenje lokalnih torenata (uz pomoć aria2<sup>16</sup> biblioteke) za efikasnu distribuciju velikih fajlova. Na primer: distribucija virtuelnih mašina za konferencije/radionice koje se održavaju u računarskoj učionici
- Distribucija fajlova sa katedre – glavnog računara na sve studentske računare (uglavnom sa HTTP Web serverom)
- Preuzimanje fajlova sa studentskih računara (SCP protokol ili rsync)

Za inicijalnu konfiguraciju i instalaciju softvera koristimo odličnu Linux distribuciju CloneZilla<sup>17</sup> koja nam omogućava da jedan podešeni računar iskoristimo kao obrazac za podešavanje drugih računara (kloniranjem hard diskova).

## 4 Zaključak i budući rad

Korišćenje otvorenog softvera i otvorenih razvojnih alata je od izuzetnog značaja u nastavi informatike. Studenti se upoznaju sa kulturom deljenja koda i razvoja softvera u velikim timovima što je nezamenljivo iskustvo u ranim fazama njihovih karijerama. U ovom radu prikazali smo neke prednosti ovakvog pristupa, kako i koji se otvoreni softver koristi u nastavi na Departmanu za matematiku i informatiku Prirodno-matematičkog fakulteta i kako mi upravljamo našim otvorenim računarskim centrom. Ovo iskustvo i znanje može biti primenjeno i na drugim institucijama i nadamo se da će naš primer podstaći i druge da koriste sve više otvorenog softvera u nastavi.

<sup>13</sup><https://sourceforge.net/projects/italc/>

<sup>14</sup><https://github.com/nmilosev/rc3-scripts>

<sup>15</sup><https://github.com/LibVNC/x11vnc>

<sup>16</sup><https://aria2.github.io/>

<sup>17</sup><https://clonezilla.org/>

Što se tiče budućeg rada, uvek tražimo načine da unapredimo sistem koji smo opisali u ovom radu, kao što je na primer izrada specifične distribucije Fedora Linux operativnog sistema sa instaliranim svim potrebnim softverom za naše studente. Druga, možda važnija mogućnost je dodatna edukacija studenata o konceptima otvorenog i slobodnog softvera kroz radionice kao što su Linux Install dani, Fedora Release događaji i slično.

## Literatura

- [1] Carrington, D., and Kim, S.-K. Teaching software design with open source software. In *33rd Annual Frontiers in Education, 2003. FIE 2003.* (2003), vol. 3, IEEE, pp. S1C–9.
- [2] Dionisio, J. D. N., Dickson, C. L., August, S. E., Dorin, P. M., and Toal, R. An open source software culture in the undergraduate computer science curriculum. *ACM SIGCSE Bulletin* 39, 2 (2007), 70–74.
- [3] Singh, V., and Holt, L. Learning and best practices for learning in open-source software communities. *Computers & Education* 63 (2013), 98–108.
- [4] Ternauciuc, A., and Ivanc, D. Remote desktop solutions used in e-learning scenarios. In *2011 6th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI)* (2011), IEEE, pp. 369–372.
- [5] Xing, G. Teaching software engineering using open source software. In *Proceedings of the 48th Annual Southeast Regional Conference* (2010), pp. 1–3.