

Rukovanje multimedijalnim dokumentom upotrebom GeRuDok alata

Jelena Cupać, Aleksa Goljović, Nemanja Jevtić
Softversko inženjerstvo i informacione tehnologije
Fakultet tehničkih nauka, Univerzitet u Novom Sadu
Novi Sad, Srbija

Apstrakt — Rukovanje dokumentima predstavlja težak posao ukoliko podrazumeva odsustvo upotrebe različitih programskih alata. Razvoj programskih alata, doprineo je bržem i lakšem radu sa dokumentima. Dodatno, upotreba multimedijalnih tipova sadržaja u okviru dokumenata omogućila je prikaz teksta u različitim oblicima, čime je čitaocu povećana pažnja i omogućeno bolje razumevanje napisanog. U ovom radu opisan je jedan od alata namenjen rukovanju multimedijalnim dokumentima u elektronskoj formi – GeRuDok. Za implementaciju alata korišćen je programski jezik Python, uz to i biblioteka PySide i određeni Python moduli. GeRuDok je omogućio rad sa dokumentima širokom krugu korisnika, od onih koji žele samo da se opробaju u radu sa multimedijom, do profesionalaca u različitim oblastima poslovanja. Pored objašnjenja ključnih reči, u prvom delu rada daćemo osnovne karakteristike alata i opisati način rada sa alatom. U drugom delu rada opisaćemo implementaciju i navesti ograničenja. Osim toga, u radu smo uporedili GeRuDok sa nekim od postojećih alata sa sličnom namenom. Poređenjem, zaključeno je da je prednost našeg alata u tome što može da rukuje i grafičkim i tekstualnim podatkom, što nije slučaj kod alata sa kojima je poređen. Međutim, ti drugi alati (pogotovo oni napredniji) podržavaju značajno veći broj funkcionalnosti u radu sa tekstem ili grafikom.

Ključne reči — rukovanje dokumentom; multimedijalni sadržaj; grafički korisnički interfejs

I. UVOD

Usled velike ekspanzije interneta i tehnologije, veliki procenat ljudskih delatnosti podrazumeva upotrebu računara u cilju olakšanog rešavanja zadataka različite kompleksnosti. U oblasti poslovanja jedan od glavnih aspekata je protok informacija. Kako bi se smanjile ljudske greške, ali i ubrzao i olakšao pristup i uređivanje dokumenata, pojavila se elektronska forma dokumenata. Postojanje više tipova podataka u dokumentima, kao što su slike, audio podaci, tekstualni podaci itd. rezultovalo je razvojem softverskih proizvoda koji značajno doprinose olakšavanju rada sa ovakvim podacima.

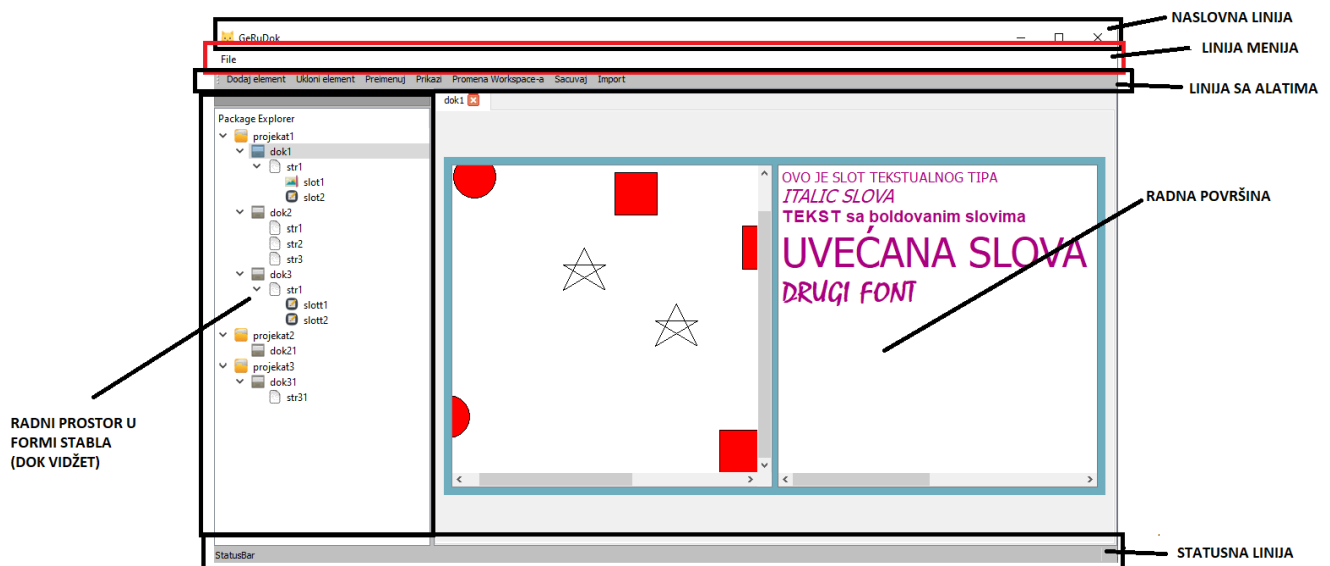
Rad sa dokumentima podrazumeva kreiranje novih dokumenata, kao i pregled i izmenu postojećih. Dokumenti mogu biti jednotipski, odnosno mogu sadržati samo jedan tip

podataka. Međutim, mnogo češće se javljaju dokumenti koji imaju podatke različitog tipa – multimedijalni dokumenti.

Multimedija predstavlja kombinaciju slikovnog, tekstualnog, audio, video i animiranog sadržaja [1]. Multimedija treba da pomogne snalaženju u sadržajima, jer omogućava bolji i pregledniji prikaz istih. Nad svakim elementom multimedije možemo primeniti različite akcije u dokumentu, kao što su dodavanje, brisanje, i različite izmene (npr., u okviru slikovnog sadržaja možemo menjati veličinu, osvetljenost itd., dok u tekstu možemo menjati font, veličinu i boju slova). Sve ove akcije bi bile daleko složenije ukoliko bi zahtevale izvršavanje bez upotrebe različitih aplikativnih softvera.

U ovom radu predstavimo alat *GeRuDok* koji omogućava rukovanje multimedijalnim dokumentima. Danas su prisutne mnoge aplikativne alatke koje se koriste za rukovanje dokumentima. Od najjednostavnijih, kao što su *Microsoft Paint* [3], *GNU Paint*, *Notepad*, pa do složenijih, kao što su *GIMP* [4] i *Visual Studio Code* [5]. Bez obzira na to što oba paint-a i *GIMP* mogu da vrše jednostavnu manipulaciju teksta, oni svoje podatke čuvaju kao slike, te se iz tog razloga ne može reći da vrše obradu teksta, kao što se može reći za *GeRuDok*. Zajednička mana navedenim programima jeste da vrše obradu ili nad tekstem ili nad slikom. Iako neki od navedenih programa imaju značajno više funkcionalnosti od *GeRuDok-a*, njihov nedostatak editovanja istovremeno i grafičkih i tekstualnih podataka se prevazilazi upotrebom *GeRuDok* alata.

Osnovni cilj ovog rada je upoznavanje publike sa karakteristikama *GeRuDok-a* i načinom rada alata. U poglavlju II date su glavne osobine alata i pojašnjen je način rada sa alatom. Istaknute su i dobre i loše strane *GeRuDok-a*. U poglavlju III dato je objašnjenje o personalizaciji, odnosno koje sve vrste podešavanja su dostupne. Poglavlje IV obrađuje programski kod aplikacije, odnosno govori se o arhitekturi, programskom jeziku, kao i o bibliotekama koje su korišćene za implementaciju alata. U poglavlju V analizirane su performanse *GeRuDok-a* i alat je detaljnije upoređen sa drugim, već pomenutim alatima.



Slika 1 Izgled prozora alata

II. KARAKTERISTIKE GERUDOK-A

GeRuDok je interaktivni softverski alat¹ namenjen radu sa dokumentima koji su struktuirani, netipizirani i potencijalno multimedijalni. Svrha ovog alata ogleda se u produkciji i interpretaciji dokumenata.

A. O *GeRuDok-u*

Alat odgovara na problem velikog broja različitih softverskih rešenja namenjenih kreiranju i održavanju sadržaja dokumenata različitog tipa. Zahvaljujući lako razumljivom grafičkom korisničkom interfejsu (eng. *Graphical User Interface*, GUI)², alat mogu koristiti kako profesionalci, tako i svi oni koji žele da rade sa multimedijom. Dakle, namenjen je svima onima koji imaju potrebu za uređivanjem dokumenata, između ostalog izdavačkim kućama, novinarima, fakultetima, arhitektama, dizajnerima, kao i osobama sa potrebom interpretacije istog. Fakultetima bi ovaj alat kroz razne ilustracije, slike i video snimke omogućio lakše i zabavnije učenje za studente, i veći stepen interakcije sa edukativnim materijalom, što znatno olakšava i predavačima posao. Novinarima je korišćenje multimedije, za razliku od tradicionalnog načina reportaže, učinilo posao jednostavnijim, jer osim većeg snalaženja sa rečima, mnogo je praktičnije zainteresovati publiku nekim video snimkom, slikom i sl. Na ovaj način amateri, pa čak i obični ljudi imaju bolji pristup reportaži.

Dokument nad kojim radi *GeRuDok* može da bude bilo kog tipa. Svaki dokument sastoji se iz jedne ili više stranica.

¹ Alat koji trenutno odgovara na akcije korisnika, tj., odgovori korisnika menjaju način funkcionisanja sistema

² Okruženje koje omogućava interaktivnu komunikaciju između korisnika i uređaja na kojem se aplikacija izvršava

Unutar jedne stranice može da postoji više delova koji mogu biti različitog tipa i mogu biti multimedijalni. Da li je stranica, multimedijalna ili ne, zavisi od tipa sadržaja koji su upakovani unutar nje. Isto važi i za dokument. Tip elementa stranice koji u sebi može sadržati samo konkretan sadržaj naziva se slot. Dokumenti se mogu grupisati u projekte, dok se projekti mogu organizovati u radne prostore.

B. Rad sa alatom

Kako bi rad sa alatom bio omogućen, potrebno je prvo instalirati ga na svom računaru. Proces instalacije zahteva svega par koraka. Nakon preuzimanja *wizard-a*, potrebno ga je pokrenuti, pa redom birati opcije *Next*, *Next*, *Install*. Po završetku instalacije, odabrati opciju *Finish*.

Nakon instalacije, potrebno je odabrati direktorijum u kojem će se čuvati kreirani projekti. Ovaj direktorijum se kasnije može i promeniti. Alat podržava dva režima rada, režim za produkciju dokumenata i režim za prikaz već postojećih dokumenata. Prvi režim, odnosno režim za produkciju dokumenata, podrazumeva operacije kao što su kreiranje, brisanje, izmena i deljenje dokumenata. Drugi režim, odnosno režim za prikaz postojećih dokumenata, treba da bude dostupan i u toku samog kreiranja dokumenta. Ovaj režim može biti i jedini režim za određenu grupu korisnika čiji cilj nije razvoj novih dokumenata, već samo pregledanje postojećih.

Osnovni elementi prozora alata (Slika 1) su naslovna linija, linija menija, linija sa alatima, statusna linija, organizacija radnog prostora u formi stabla i radna površina. Naslovna linija sadrži opcije za baratanje čitavim glavnim prozorom, kao što su minimiziranje, uvećavanje preko celog ekrana, izlazak iz programa, odnosno zatvaranje prozora. Linija menija sadrži padajuće podmenije u okviru kojih se nalaze različite naredbe za manipulaciju dokumentima. Linija sa alatima se sastoji iz

alata koji omogućavaju uređivanje različitih tipova slotova. U okviru statusne linije prikazane su trenutne informacije o statusu programa. Organizacija radnog prostora u formi stabla podrazumeva prikaz svih projekata i njihovih elemenata koji se nalaze u trenutnom radnom prostoru. Projekti su prikazani na takav način da je moguće shvatiti hijerarhijski odnos među elementima u radnom prostoru. Radna površina služi za prikaz dokumenata koji se obrađuju.

C. Rad sa radnim prostorom

Radni prostor je imenovana kolekcija projekata. U jednom trenutku za vreme rada, sa alatom može biti povezan samo jedan radni prostor, odnosno, nije moguće otvoriti više radnih prostora istovremeno. Prilikom prvog pokretanja aplikacije na računaru se automatski kreira direktorijum koji predstavlja podrazumevani radni prostor. Radni prostor je, ukoliko postoji potreba za tim, moguće promeniti. Podržane operacije nad radnim prostorom su kreiranje, preimenovanje, brisanje i zatvaranje.

Kreiranje radnog prostora prodrazumeva kreiranje novog direktorijuma u kojem će biti smešteni svi projekti napravljeni u tom radnom prostoru. Nakon odabira radnog prostora, u njemu se skladište svi novokreirani projekti sve dok se on ne promeni. Novi radni prostor se kreira tako što se u okviru *File* podmenija izabere opcija *Promena radnog prostora*. U svakom trenutku moguće je preimenovati radni prostor. Isto važi i za brisanje. Brisanjem radnog prostora brišu se i svi projekti u okviru njega.

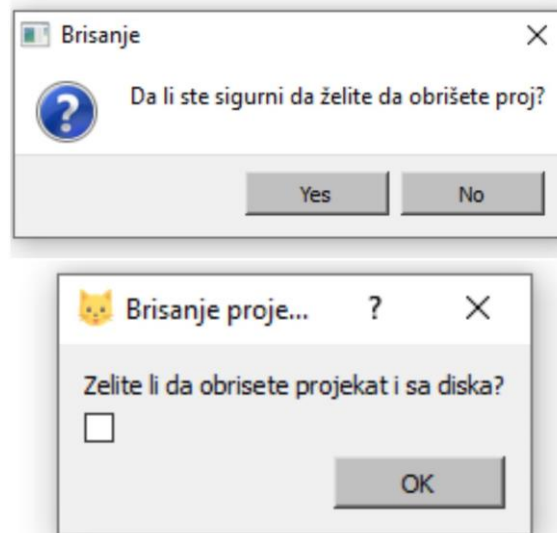
D. Rad sa projektom i dokumentom

Projekat predstavlja kolekciju proizvoljnog broja dokumenata, dok je dokument kolekcija stranica. U trenutku rada sa alatom, može biti otvoreno više projekata, odnosno, otvoreni su svi oni projekti koji se nalaze u trenutnom radnom prostoru i prikazani su u formi stabla radi lakšeg snalaženja i kretanja kroz projekte. Podržane operacije nad projektom su kreiranje, brisanje, zatvaranje, preimenovanje. Sve navedene operacije podržava i dokument, i uz njih i operaciju deljenja. Za projekte postoji još i operacija *import*, koja omogućava referenciranje projekta u odabrani radni prostor.

Projekat se kreira odabirom opcije *Dodaj element* iz *File* podmenija ili odabirom iste opcije u okviru linije sa alatima. Kako ova opcija služi i za kreiranje dokumenta, stranice, i slota, da bi se kreirao projekat, potrebno je da nijedan element iz stabla radnog prostora ne bude selektovan. Kreiranje projekta podrazumeva pojavljivanje dijaloga u okviru kojeg je potrebno uneti ime, a zatim potvrditi akciju. U okviru istog radnog prostora ne mogu postojati dva projekta sa istim imenom. Posledica kreiranja projekta je kreiranje direktorijuma na disku, kao i njegov prikaz u stablu. U svakom trenutku moguće je preimenovati projekat, tada će se preimenovati i odgovarajući direktorijum na disku.

Pod brisanjem projekta podrazumeva se logičko i fizičko brisanje. Logičko brisanje projekta je brisanje iz strukture stabla radnog prostora, dok je fizičko, brisanje i sa diska. Oba tipa brisanja, podrazumevaju gotovo iste korake kako bi se primenili tj. potrebno je prvo selektovati projekat u stablu, a

zatim odabrati opciju *Ukloni element*. Tada se otvara dijalog gde je potrebno potvrditi brisanje, a nakon potvrde, novi dijalog gde je omogućeno čekiranje opcije za brisanje sa diska (fizičko brisanje). Logičko i fizičko brisanje prikazano je na slici 2, tj., prikazani su dijalozi koji su prethodno navedeni.



Slika 2 Logičko i fizičko brisanje (dijalozi)

Poslednja operacija podržana nad projektom je *import*. Postoje dve vrste importa: sa kopiranjem i bez kopiranja. Obe vrste podrazumevaju referenciranje projekta u strukturu stabla, ali je razlika u tome što import sa kopiranjem podrazumeva i kopiranje projekta u direktorijum trenutnog radnog prostora pri čemu se svaka nastala promena čuva u kopiranom projektu.

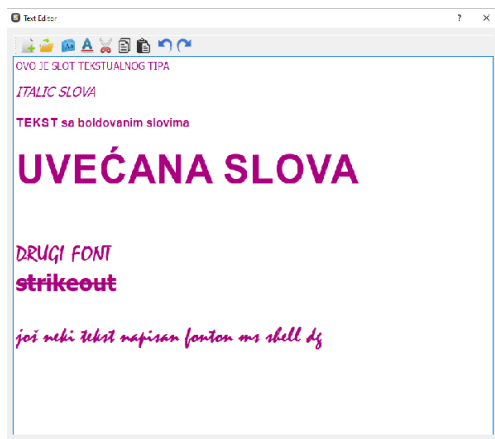
Dokument se kreira tako što se u okviru strukture stabla odabere projekat u kojem je potrebno kreirati dokument, a zatim se odabere opcija *Dodaj element*. Sledeće, potrebno je uneti naziv za novi dokument. U okviru istog projekta, nije dozvoljeno postajanje dokumenata istog naziva. Kao i za projekte, i dokumentu je moguće u svakom trenutku promeniti naziv.

Operacija brisanja dokumenta se razlikuje od brisanja projekta jer se vrši isključivo fizičko brisanje. Dokument se briše tako što se selektuje u stablu, a zatim izabere opcija *Ukloni element*. Pogodno je što se može otvoriti više dokumenata (svaki otvoreni dokument predstavlja jednu karticu), ali samo jedan može biti u fokusu. Nad dokumentom, podržana je operacija deljenja, koja podrazumeva da se isti dokument nalazi u više projekata. Ovakav dokument naziva se hiperdokument. Svaka promena koja se izvrši nad dokumentom u bilo kom projektu, vidljiva je u svim projektima u kojima se dokument nalazi. Da bi se dokument podelio, potrebno je selektovati ga, pa u okviru *File* podmenija odabrati opciju *Deli*, a zatim odabrati projekat u koji želimo da dodamo dokument.

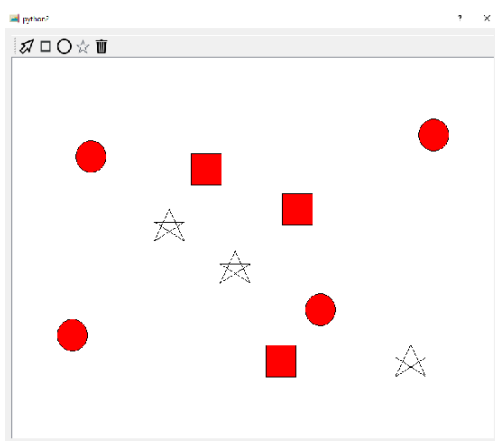
U okviru dokumenta, moguće je raditi sa stranicama, a u okviru stranica sa slotovima. Stranica se kreira na sličan način kao i dokument, samo je umesto projekta, potrebno

selektovati odgovarajući dokument. Sve ostale operacije nad stranicom su iste kao i nad dokumentom. Isto važi i za slotove, ali za njih postoje dodatne operacije koje zavise od tipa slot. Da bi se slotovi uređivali, potrebno je dvokliknuti na željeni slot, a zatim se pojavljuje novi prozor u okviru kojeg je postavljen toolbar za editovanje odgovarajućeg tipa slot.

Slotovima tekstualnog tipa moguće je menjati font, veličinu, boju, uz to se mogu i dodatno uređivati – podebljavanje, podvlačenje itd. (Slika 3). Slotovi grafičkog tipa podrazumevaju importovanje različitih fajlova ekstenzija .png, .img, kao i samostalno crtanje jednostavnih crteža pomoću oblika kao što su krug, zvezda, pravougaonik (Slika 4).



Slika 3 Prozor za uređivanje tekstualnog slot. U okviru toolbar-a nalaze se različite opcije, čijim odabirom se otvara odgovarajući dijalog. Prve dve opcije su za kreiranje novog slot i otvaranje postojećeg, zatim slede opcije za menjanje fonta i boje, pa opcije katovanja, kopiranja. Poslednje dve opcije su Undo i Redo.



Slika 4 Prozor za uređivanje grafičkog slot postojećim oblicima. U okviru toolbar-a dostupna su tri oblika, kao i opcije za selekciju i brisanje oblika.

E. Dobre i loše strane GeRuDok-a

Jednostavan interfejs i besplatnost samo su neke od dobrih strana *GeRuDok-a*. Podržanost više vrsta medija – grafika, tekst – daju *GeRuDok-u* potencijalno široku primenu u oblasti manipulisanja dokumentima. Više nije neophodno koristiti alate koji su specijalizovani za samo neku od ovih vrsta multimedije, *GeRuDok* rukuje obema.

Pažljivo osmišljeni dizajn omogućava lako razumevanje i snalaženje u programu. Sa programerske strane, prednost alata je što su funkcije za obradu multimedijalnih elemenata u aplikaciju dodate kao plagini, što dozvoljava dodatno proširivanje novim funkcijama za obradu različitih tipova podataka. Isto tako, grafički i tekstualni editor mogu se ponovno iskoristiti, tj. kako predstavljaju plagine, mogu se instalirati u okviru programa koji imaju adekvatne interfejse za dodavanje ovih plagina. Za programere, pogodno je i postojanje dokumentacije, gde se može videti detaljna implementacija alata, kao i opisa ponašanja alata pomoću domen-specifičnog jezika *Gherkin* [6], koji daje laganu strukturu za opis primera ponašanja alata [7]. *Gherkin* definiše scenarije ponašanja na osnovu kojih je moguće lakše sagledati način rada alata.

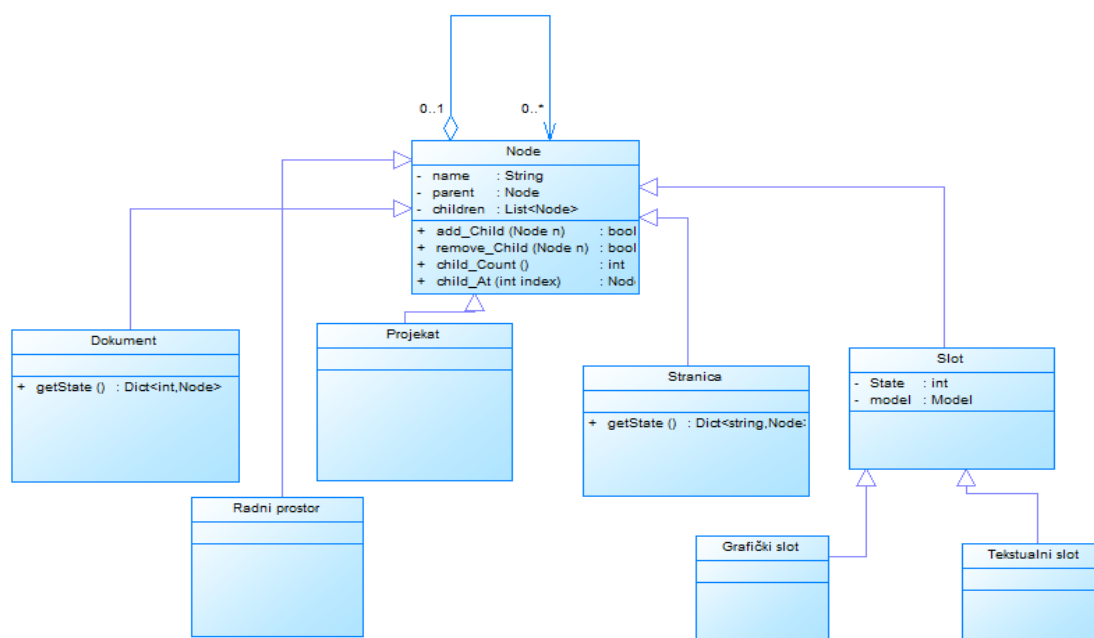
Loša strana alata je što je dostupan samo određenim računarima (više o tome u petom poglavlju) i nedostatak podrške za audio i video sadržaj. Uz to, za crtanje različitih grafičkih slotova dostupno je svega nekoliko geometrijskih oblika.

III. PODEŠAVANJA

U okviru rada sa alatom, omogućena je i personalizacija samog alata, odnosno dostupna su podešavanja alata prema volji korisnika. Podešavanja alata su *look and feel* i to bez prekidanja rada u programu i bez uticaja na trenutni rad korisnika. Podešavanja koja su dostupna jesu promene izgleda alata, zatim, promena jezika, veličine i sl. U nastavku su ukratko opisana moguća podešavanja.

Među prvim podešavanjima, dozvoljena je promena izgleda alata. Ona, između ostalog, podrazumeva izmenu pozicija osnovnih elemenata prozora alata. Neki od elemenata prozora čija pozicija može da se menja su linija sa alatima i dok vidžet (Slika 1). Dok vidžet je deo prozora na kojem je predstavljen radni prostor u formi stabla. On se može postaviti ili sa leve ili sa desne strane prozora alata. Linija sa alatima može da bude u gornjem delu prozora ili u donjem. Omogućena je i promena teme celokupnog alata (dostupno je nekoliko standardnih tema koje se mogu izabrati), promena boje dok vidžeta i radne površine, fonta i veličine teksta.

Kako bi alat bio dostupan velikom broju ljudi, postoji mogućnost odabira velikog broja jezika na kom će alat biti prikazan. U inicijalnoj verziji, dostupni su samo engleski i srpski. U narednim verzijama, očekuje se značajno proširivanje broja dostupnih jezika.



Slika 5 Dijagram klasa

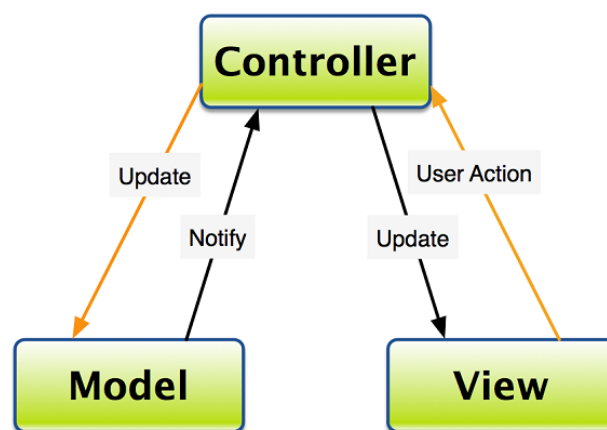
IV. IMPLEMENTACIJA

U ovom delu, detaljnije je opisana implementacija alata. Opisan je programski jezik koji je korišćen, kao i sve uključene biblioteke. Uz to, dodatno je objašnjen MVC arhitektura na kojoj je ovaj alat baziran i ukratko su opisane najvažnije klase.

Na slici 5 prikazan je dijagram klasa. Najvažnije klase koje su prikazane na dijagramu modeluju radni prostor, projekat, dokument, stranicu, grafički slot i tekstualni slot. Glavna klasa je klasa *Node*, koja modeluje čvor u strukturi stabla, i iskorišćena je kao roditeljska klasa za klase *Projekat*, *Dokument*, *Radni prostor*, *Slot* itd. Njeni glavni atributi su naziv, roditelj (čvor na višem nivou hijerarhije) i deca (čvorovi na nižem nivou hijerarhije). Klasa *Radni prostor*, kao što joj ime kaže, modeluje radni prostor i glavna metoda ove klase je dodavanje projekata. Klase *Projekat*, *Dokument*, *Stranica*, *Slot* su analogne klasi *Radni Prostor*. Klase *Grafički slot* i *Tekstualni slot* modeluju konkretne tipove slotova. *Grafički slot* je klasa koja nasleđuje klasu *Slot* (bazna klasa za sve vrste slotova), ali je povezana sa pluginom za obradu grafičkog sadržaja. Dva bitna atributa ove klase su *stanje* – daje podatke o trenutnom razmeštaju oblika, ukoliko su korišćeni dostupni oblici za crtanje – kao i *grafikModel* – atribut koji predstavlja model ove vrste slota, sadrži metode za serijalizaciju i deserijalizaciju slota. Klasa *Tekstualni slot* je analogna klasi *Grafički slot*.

Arhitektura koja se često koristi za interaktivne softverske sisteme i koja je u razvoju ovog alata primenjena je Model-Pregled-Kontroler (eng. *Model-View-Controller*, MVC) [8]. Ova arhitektura se zasniva na sinhronom radu tri komponente: *Model*, *Pregled* i *Kontroler*. Ključna osobina je nezavisnost podataka od načina njihove prezentacije. Komponenta *Model* modeluje podatke i procese, ona komunicira sa bazom podataka i vrši sva potrebna izračunavanja. Komponenta

Pregled predstavlja prezentacioni sloj podataka, dok *Kontroler* omogućava vezu između korisnika i podataka. *Kontroler* je zadužen za upravljanje tokom aplikacije, modelom i pregledom. Ovom arhitekturom obezbeđeno je lakše upravljanje tokom aplikacije. Upravo zbog nezavisnosti, lakog upravljanja tokom, mogućnosti prikaza podataka na više načina, ali i drugih prednosti MVC-a [9], ova arhitektura je izabrana u implementaciji. Autori u [9] takođe navode da se primenom MVC arhitekture dobija skalabilniji i fleksibilniji program. Na slici 6 prikazana je saradnja ove tri komponente.



Slika 6 Saradnja komponenta u okviru MVC arhitekture. Kontroler prima zahteve i inicira obradu podataka u Modelu, a zatim preuzima odgovarajuće podatke i aktivira prezentacioni sloj kako bi se podaci prikazali.

U okviru alata *GeRuDok*, komponentu *Model* čine prethodno opisane klase. Komponentu *Pregled* čine klase koje renderuju izgled dokumenta, stranice i sl. Dok je *Kontroler*

predstavljen klasom *ActionManager* koja upravlja klasama koje su zadužene za akcije kao što su dodavanje novog čvora, prikaz dijaloga za odabir radnog prostora, promena imena čvorovima itd., kao i klasama koje spadaju u prethodne dve komponente.

Programski jezik u kojem je implementiran *GeRuDok* je Python [10]. Python spada u grupu interpretiranih programskih jezika³ i podržava veliki broj različitih paketa i modula. Uz to, jednostavna sintaksa, širok spektar mogućnosti samo su jedni od razloga upotrebe ovog programskog jezika u implementaciji.

GUI je grafički prikaz aplikacije na ekranu. Njegova namena je lakša interakcija korisnika sa programom. Radi implementacije GUI-a, korišćena je Python biblioteka *PySide* [12], koja je omotač biblioteke *QT* napisane u C++ jeziku [14]. *PySide* biblioteka sadrži veliki broj klasa, od kojih su za implementaciju bile najvažnije *QApplication*, *QMainWindow*, *QAction* i sl. *QApplication* je osnovna klasa za upravljanje GUI-em aplikacije. *QMainWindow* je iskorišćena za glavni prozor aplikacije i kako je ona kontejnerska, sadrži različite komponente, kao što su barovi, vidžeti itd. Klasa *QAction* namenjena je pozivu akcija iz menija, barova i korišćenjem prečica. Ova klasa poslužila je za različite akcije, kao što su dodavanje oblika u okviru rukovanja grafičkim slotom, akcija čuvanja, importovanja itd. Bitni atributi ove klase su signali koji označavaju da je akcija pokrenuta, pa se onda izvršava metoda koja je „zakačena“ na tu akciju.

Pristup koji je primenjen pri izradi alata je Interfejs za višestruke dokumente (eng. *MultipleDocumentInterface*, MDI) koji podrazumeva da u okviru jedne instance aplikacije može postojati više otvorenih dokumenata. U *PySide* postoje dve komponente namenjene MDI radu – *QmdiArea* i *QTabWidget*, koje su i korišćene.

Pored biblioteke *PySide*, u razvoju alata korišćeni su Python moduli, *OS* [15], *Shutil* [16] i *Pickle* [17]. Modul *OS* služi za rad sa operativnim sistemom, u konkretnom slučaju, za *GeRuDok* je korišćen radi rada sa fajlovima na masovnoj memoriji. Modul *Shutil* je korišćen radi podrške kopiranja i brisanja fajlova, dok je modul *Pickle* korišćen radi serijalizacije i deserijalizacije podataka.

V. OGRANIČENJA

Osnovna ideja *GeRuDok-a*, pored uspešnog rešavanja zadataka vezanih za dokument, bila je dostupnost svima. Međutim, postoje određena ograničenja za korišćenje alata koja se tiču kako operativnog sistema, tako i računara na kojem se izvršava program. U nastavku su navedena ograničenja, ali i kratak osvrt i na druge alate koji su upoređeni sa *GeRuDok-om*.

A. Operativni sistem i performanse

GeRuDok alat je namenjen isključivo desktop i laptop računarima. Može da se izvršava na računarima sa *Windows* operativnim sistemom i to 7, 8, 8.1 i 10 verzijom. Performanse programa zavise od hardverske konfiguracije računara, stoga je preporučljivo imati minimum 500 MB slobodnog prostora na hard disku, 1 GB RAM memorije i procesor sa dva jezgra i 2 GHz radnog takta.

B. *GeRuDok* i drugi alati

GeRuDok nije prvi alat koji omogućava editovanje teksta i grafike. Neki od poznatijih alata jesu *Visual Studio Code* i *GIMP*, koji su ranije spomenuti. Ono što je zajedničko za *GeRuDok* i *Visual Studio Code* jeste da oba mogu da manipulišu tekstualnim podacima. Bez obzira na to što je *Visual Studio Code* pre svega editor [5], on je namenjen prvenstveno programerima kako bi u značajnoj meri olakšao pisanje koda. Iz tog razloga, ne poseduje mogućnost manipulisanja grafičkim podacima, ali uključuje mnoge funkcionalnosti za obradu teksta, kao što su isticanje sintakse, popunjavanje koda i mnoge druge koje ne postoje u *GeRuDok-u*. Razlog tome je različita namena ova dva alata. *GeRuDok* je alat opšte namene, dok je *Visual Studio Code* alat namenjen, pre svega, programerima. *GeRuDok* podržava standardne akcije izmene teksta, što ga čini boljim izborom za one koji traže jednostavnu obradu teksta.

Najjednostavniji program za obradu teksta je *Notepad* koji je namenjen korisnicima koji zahtevaju jednostavnu upotrebu koja podrazumeva pisanje, brisanje, izmenu fonta, veličine i slično. Sve navedene funkcionalnosti podržava i *GeRuDok*. Za razliku od *Notepad-a*, *GeRuDok* može da rukuje i grafikom, ali ne podržava štampanje niti grafičkih, niti tekstualnih podataka. Napredniji programi za manipulaciju teksta su *Gedit* i *Notepad++* [18] koji pružaju značajno više funkcionalnosti od *GeRuDok-a* i *Notepad-a*. Jedna od značajnih funkcionalnosti ova dva programa je prikaz sadržaja više otvorenih dokumenata istovremeno na ekranu (eng. *side by side*), čime je omogućen rad nad više dokumenata istovremeno. Podsetimo, *GeRuDok* podržava otvaranje više dokumenata istovremeno, ali samo jedan može biti prikazan na ekranu (dok su ostali prikazani kao kartice), ali se u okviru jednog dokumenta istovremeno može videti više slotova. *Notepad++* podržava i *clone document* [18] što omogućava da jedan isti dokument bude otvoren u dve kartice i da se obe kartice istovremeno prikazuju na ekranu jedna do druge. Promene koje nad tim dokumentom budu izvršene u okviru jedne kartice, istog trenutka su vidljive i u drugoj kartici. *GeRuDok* ne podržava ovu funkcionalnost.

Jedan od poznatih programa za obradu slika je *GIMP* (eng. *GNU Image Manipulation Program*). Usko je specijalizovan za obradu grafičkih podataka, te iz tog razloga nudi veliki broj funkcionalnosti od običnog crtanja do izmena formata slika [4], dok je, sa druge strane, *GeRuDok-ova* sposobnost manipulacije grafičkim podacima dosta ograničena s obzirom da se još uvek nalazi u početnoj fazi razvoja.

Najjednostavniji programi za obradu grafike su *Microsoft paint* i *GNU paint*. U odnosu na *GIMP*, pružaju značajno manje funkcionalnosti. Mogu da rukuju sa više boja i oblika

³ Programi napisani na interpretiranom programskom jeziku samo se prilikom prvog pokretanja prevode, nakon toga ih je moguće pokretati bez prevođenja

nego *GeRuDok*, ali nemaju mogućnost obrade više grafičkih podataka istovremeno. Za razliku od *GeRuDok-a*, oba paint-a imaju mogućnost štampanja slika, ali ne i mogućnost obrade teksta.

VI. ZAKLJUČAK

U ovom radu predstavljen je alat za rukovanje multimedijalnim dokumentima, *GeRuDok*. Motivacija je bila što bi se ovakav alat mogao koristiti za jednostavniji rad sa dokumentima, kako u oblasti poslovanja, tako i u oblasti obrazovanja. Želja je bila da alat bude prisutan kroz segmente koji imaju pozitivan predznak u društvu – kultura, nauka; da se omogući ekspanzija elektronskih udžbenika i druge literature. Alat se sastoji iz dva režima: režim za produkciju dokumenata i režim za prikaz postojećih dokumenata.

Režim za produkciju dokumenata podrazumeva osnovne operacije nad dokumentom – kreiranje, brisanje, deljenje i izmenu. *GeRuDok* podržava dve vrste slotova: grafički i tekstualni. Nad grafičkim slotom omogućeno je crtanje, a nad tekstualnim izmene fonta, boje, veličine slova i sl. Režim za prikaz dokumenata, podrazumeva samo prikazivanje dokumenta bez mogućnosti manipulacije. Moguće je otvoriti više dokumenata, ali se samo jedan može prikazati na ekranu. Implementacija alata bazirana je na MVC arhitekturi, a alat je implementiran u programskom jeziku Python, uz korišćenje *PySide* biblioteke i Python modula, *OS*, *Shutil* i *Pickle*.

Rad sa alatom i sam alat mogli bi se poboljšati dodavanjem plugina za obradu tipova multimedije koji u trenutnoj verziji nisu podržani, kao što su audio, animacija i video sadržaji. Na ovaj način, *GeRuDok* bi mogao u potpunosti zameniti rad sa više editora od kojih je svaki specijalizovan za određeni tip multimedije, jer bi *GeRuDok* predstavljao sve u jednom. Dodavanje novih složenijih funkcionalnosti postojećim pluginovima (za obradu grafičkih i tekstualnih podataka), kao što su photoshop, autokorekcija, pretrage, obeleživači doprinelo bi povećanju broja korisnika ovog alata. Pored toga, unapređenje se takođe može steći prilagođavanjem alata mobilnim uređajima, ali i računarima koji nemaju *Windows* operativni sistem. Skup jezika koji je podržan, mogao bi se proširiti kako bi ljudi sa različitih govornih područja efikasno i lako mogli koristiti alat.

LITERATURA

- [1] Costello, Vic. *Multimedia foundations: Core concepts for digital design*. CRC Press, 2016, pp. 3-39.
- [2] Pavithra, A., M. Aathilingam, and S. M. Prakash. "Multimedia and its applications." *International journal for research & development in technology* 5 (2018), pp. 271-276.
- [3] A Guide to Microsoft Paint. Dostupno na adresi: [https://www.reading.ac.uk/web/files/www/File_Library/A_Guide_to_Microsoft_Paint_\(XP\).pdf](https://www.reading.ac.uk/web/files/www/File_Library/A_Guide_to_Microsoft_Paint_(XP).pdf) [pristupljeno 4.4.2020.]
- [4] Dokumentacija programa GIMP. Javno dostupna na linku: <https://docs.gimp.org/2.10/en/> [pristupljeno 4.4.2020.]
- [5] Dokumentacija i korisničko uputstvo programa Visual Studio Code, javno dostupna na linku: <https://code.visualstudio.com/docs> [pristupljeno 4.4.2020.]
- [6] Dokumentacija domen-specifičnog jezika Gherkin. Dostupna na adresi: <https://cucumber.io/docs/gherkin/reference/> [pristupljeno 11.4.2020.]

- [7] Wynne, Matt, Aslak Hellesoy, and Steve Tooke. *The cucumber book: behaviour-driven development for testers and developers*. Pragmatic Bookshelf, 2017. pp. 25-39
- [8] Sharan, Kishori. "Model-view-controller pattern." *Learn JavaFX* 8. Apress, Berkeley, CA, 2015. 419-422
- [9] Thakur, Ram Naresh, and U. S. Pandey. "The Role of Model-View Controller in Object Oriented Software Development." *Nepal Journal of Multidisciplinary Research* 2.2 (2019): 1-6
- [10] Miller, Bradley N., David L. Ranum, and Julie Anderson. *Python programming in context*. Jones & Bartlett Learning, 2019.
- [11] Srinath, K. R. "Python–The Fastest Growing Programming Language." *International Research Journal of Engineering and Technology (IRJET)* 4.12 (2017): pp. 354-357.
- [12] Harwani, Bintu M. *Qt5 Python GUI Programming Cookbook: Building responsive and powerful cross-platform applications with PyQt*. Packt Publishing Ltd, 2018
- [13] Dokumentacija biblioteke PySide. Javno dostupna na linku: <https://srinikom.github.io/pyside-docs/> [pristupljeno 1.4.2020.]
- [14] Dokumentacija Qt biblioteke. Javno dostupna na linku: <https://doc.qt.io/> [pristupljeno 1.4.2020.]
- [15] Dokumentacija Python modula OS. Javno dostupna na linku: <https://docs.python.org/3/library/os.html> [pristupljeno 2.4.2020.]
- [16] Dokumentacija Python modula shutil. Javno dostupna na linku: <https://docs.python.org/3/library/shutil.html> [pristupljeno 2.4.2020.]
- [17] Dokumentacija Python modula pickle. Javno dostupna na linku: <https://docs.python.org/3/library/pickle.html> [pristupljeno 2.4.2020.]
- [18] Dokumentacija programa Notepad++. Javno dostupna na linku: <https://npp-user-manual.org/> [pristupljeno 5.4.2020.]

