

ALATI ZA INTERAKTIVNU NASTAVU MATEMATIKE I E-UČENJE

Muzafer Saračević

MUZAFER SARAČEVIĆ

**ALATI ZA INTERAKTIVNU
NASTAVU MATEMATIKE
I E-UČENJE**

PRAKTIKUM

*za nastavnike i profesore matematike, informatike, tehničkog i informatičkog
obrazovanja, fizike, hemije i profesore razredne nastave*

Novi Pazar, 2012

Autor

Muzafer Saračević

Naziv udžbenika

Alati za interaktivnu nastavu matematike i e-učenje, e-praktikum

Izdavač

Univerzitet u Novom Pazaru

Za izdavača

Prof.dr Mevlud Dudić, rektor Univerziteta u Novom Pazaru

Recezent

**Prof.dr Ćamil Sukić
Prof.dr Danijela Milošević
Prof.dr Emruš Azizović**

ISBN 978-86-84389-28-4

Odlukom Senata Univerziteta u Novom Pazaru (14 juna 2012 godine)

CIP - Каталогизација у публикацији
Народна библиотека Србије, Београд

371.39:004.4'2(075.8)(076)

САРАЧЕВИЋ, Музафер, 1984-

Alati za interaktivnu nastavu matematike
i e-učenje : praktikum za nastavnike i
profesore matematike, informatike, tehničkog
i informatičkog obrazovanja, fizike, hemije i
profesore razredne nastave / Muzafer
Saračević. - Novi Pazar : Univerzitet, 2012
(Novi Pazar : Univerzitet). - 103 str. :
ilustr. ; 30 cm

Tiraž 500. - Napomene uz tekst. -
Bibliografija: str. 83-85.

ISBN 978-86-84389-28-4

а) Електронско учење - Програмски алати
COBISS.SR-ID 191904780

U ovom praktikumu se navode konkretni predlozi unapređenja nastave, prvenstveno matematike a i drugih prirodnih i društvenih nauka. Predstavljani su Java apleti kao alati za kreiranje interaktivnog sadržaja za elektronsko učenje i kao savremene metode u nastavi koje direktno utiču na kvalitetno izvođenje nastavnog procesa. Pored kreiranja kvalitetnog interaktivnog sadržaja, apleti se u ovom praktikumu predstavljaju i kao alati za evaluaciju u nastavi matematike, a takođe oni imaju primenu i u okviru korisničke podrške u obliku virtuelnih učionica i laboratorija. Navedene su i mogućnosti java apleta i njihova implementacija u sistemima za e-učenje.

This practicum presents concrete suggestions on improving teaching, especially mathematics and other natural and social sciences. Java applets are presented as tools for creating interactive content for e-learning as modern methods of teaching that directly affect the quality of the teaching process. In addition to creating quality interactive content, applets are presented in this paper as tools for evaluating the teaching of mathematics, and they also have application in the customer support students in the form of virtual classrooms and laboratories. This paper presents the possibilities of Java applets and their in-implementing systems for e-learning.

SADRŽAJ

1. UVOD	5
2. INTERAKTIVNI SADRŽAJI U E-UČENJU I ALATI	
2.1 Simulacije, animacije i interaktivni sadržaji u nastavi	7
2.2 Alati za kreiranje interaktivnog sadržaja u e-učenju	8
2.2.1 Mogućnosti primene alata za e-učenje u nastavi	10
2.2.2 Unapređenje nastave pomoću alata za e-učenje	11
3. ALATI ZA KREIRANJE INTERAKTIVNOG SADRŽAJA	
3.1 Glavne karakteristike Jave i prednosti	13
3.2 Java programski jezik i primena u nastavi	14
3.3 Java apleti - osnovni koncepti	14
3.3.1 Kreiranje i metodi apleta	16
3.4 Primena apleta u nastavi matematike	21
3.4.1 GeoGebra – alat za interaktivne sadržaje u matematici	25
3.4.2 JavaSketchPad i Geometer's SketchPad	32
3.4.3 WIRIS - alat za jednostavno kreiranje apleta	35
3.4.4 Easy Java Simulations - alat za kreiranje interaktivnih simulacija	38
3.4.5 Mathematica - alat za matematičko programiranje	41
3.4.6 Integracija JavaView alata u Notebook (Wolfram Mathematica)	43
3.5 Kolaborativno kreiranje elektronskih sadržaja pomoću apleta	47
3.5.1 WIRIS Conference - math chat	47
3.6 Primena Java obrazovnih igara u nastavi	48
3.7 Mape uma i <i>brejnstorming</i> alati kao Java apleti	50
4. ALATI ZA OCENJIVANJE U NASTAVI MATEMATIKE	
4.1 Metode i postupci provere i ocenjivanja u matematici	51
4.2 Alati za ocenjivanje bazirani na Javi	52
4.3 Primeri upotrebe alata za ocenjivanje u nastavi matematike	54
4.3.1 Primena apleta u izradi testova sa višestrukim odgovorom	60
4.3.2 Provera znanja učenika pomoću SketchPad alata	61
4.3.3 Primeri kreiranja testova u obliku apleta - QuizMaster	62
4.3.4 Primeri kreiranja upitnika, anketa u obliku apleta- ABPoll Master	63

5. JAVA APLETI KAO KORISNIČKA PODRŠKA U SISTEMIMA ZA E-UČENJE

5.1 Praktični razlozi za korisničku podršku	66
5.2 Primena apleta u kreiranju virtuelnih učionica u nastavi matematike	67
5.2.1 Primer java apleta kao virtuelne učionice	70

6. IMPLEMENTACIJA INTERAKTIVNIH SADRŽAJA U SISTEMIMA ZA E-UČENJE

6.1 Pojam i podela sistema za e-učenje	73
6.2 Moodle sistem za e-učenje	73
6.3 Mogućnosti uvoza interaktivnog sadržaja u Moodle sistem	74
6.4 <i>Moodle dodaci</i> za rad sa apletima i animacijama	76
6.4.1 <i>WIRIS integracija</i>	77
6.4.2 <i>GeoGebra filter</i>	79

7. ZAKLJUČAK

81

8. LITERATURA

83

9. PRILOZI

86

10. LISTA SLIKA

101

11. LISTA TABELA

103

1. UVOD

U savremenoj nastavi se susrećemo sa mnoštvom različitih multimedijalnih sadržaja koji doprinose kvalitetu nastave, povećanju motivacije, boljoj realizaciji predmeta i boljem individualnom napredovanju pojedinaca u skladu sa njihovim intelektualnim sposobnostima. Uz upotrebu informaciono-komunikacionih tehnologija, posebno alata i tehnologija u elektronskom učenju, učesnici mogu svoja već stečena znanja proširiti i konkretno primeniti, jer savremeni pristupi u nastavi imaju za cilj da se stečeno teorijsko znanje primeni u praksi.

Interaktivni sadržaji u nastavi su poseban vid nastave pomoću računara koji su primenjivi u svim disciplinama a posebno u prirodnim i tehničkim naučnim oblastima u kojima postoji realna potreba za vizualizacijom procesa. U ovom praktikumu akcenat je stavljen na prirodne nauke, što može izuzetno poslužiti u nastavi informatike, matematike, u fizici, tehnici itd.

Ovaj rad je predlog kako unaprediti nastavu matematike pa su u okviru istog razmatrane mogućnosti Java apleta, koji se mogu lako implementirati u odgovarajuće sisteme za elektronsko učenje. Apleti se realizuju u vidu simulacija, koje pomoću računara obezbeđuju dvosmerni prenos informacija, jer učenik ima mogućnost da samostalno upravlja promenljivima modela.

U prvom poglavlju su navedene opšte karakteristike i prednosti interaktivnih sadržaja u nastavi, uopšteno. Date su definicije za pojam simulacije i animacije, a navedeni su i alati za kreiranje interaktivnog sadržaja u e-učenju, zatim mogućnosti njihove primene u nastavi a samim tim i mogućnost unapređenja nastave pomoću alata za e-učenje.

U drugom poglavlju su navedeni osnovni koncepti Java apleta kao i generalna primena Java programskog jezika u nastavi. Zatim je navedena konkretna primena apleta u nastavi matematike kroz alate za interaktivne sadržaje u matematici kao što su *GeoGebra*, *JavaSketchPad*, *Easy Java Simulations*, *Java View*, *Wiris* i alata *Mathematica* za matematičko programiranje. Dati su primeri i kolaborativnog kreiranja elektronskih sadržaja pomoću apleta. Jedan deo ovog poglavlja se bavi i primenom Java obrazovnih igara u vidu apleta u nastavi i rad sa talentovanim decom, a pomenute su i mape uma i brejnstorming alati kao Java apleti.

Treće poglavlje se odnosi na primenu Java apleta u ocenjivanju u e-učenju. Navedene su metode i postupci provere i ocenjivanje u matematici kao i dostupni alati za ocenjivanje bazirani na Javi. Dati su i konkretni primeri upotrebe alata za ocenjivanje u nastavi matematike kao što su npr. izrada testova sa višestrukim odgovorom a i sa drugim tipovima pitanja. Takođe su navedene i mogućnosti Java apleta kao alata za izradu upitnika i anketa.

Četvrti deo se odnosi na korisničku podršku u sistemima za e-učenje kao i primenu Java apleta u kreiranju virtuelnih učionica i laboratorija. Dati su i praktični razlozi za korisničku podršku u nastavi matematike i predstavljen je konkretan aplet *Virtual Classroom*. Kao primer navedena je laboratorija za matematiku koja je razvijena od strane autora ovog rada za potrebe nastave. Ukoliko obrazovna institucija želi da svoje učenike dodatno podstakne na rad sa eksperimentima to vrlo lako može postići bez preteranih ulaganja u laboratorijski potrošni materijal jer je za mnoge instrumente potrebno daleko više opreme i sredstava nego za korišćenje računara. Naravno treba naglasiti da ni u kom slučaju upotreba računara ne sme da zameni praktičan laboratorijski rad.

U petom poglavlju su navedene mogućnosti implementacije Java apleta i simulacija u sistemima za e-učenje. Naveden je pojam i podela sistema za e-učenje, a predstavljen je Moodle sistem koji je korišćen za potrebe istraživanja i ispitivanja uticaja na postignuće učenika u osnovnoj školi. Dati su i predlozi unapređivanja Moodle sistema kroz dodatke kao što su *Wiris* i *GeoGebra plugin* koji nude mogućnost uvoza apleta u sistem za e-učenje i znatno doprinose kvalitetu interaktivnog sadržaja za e-učenje i boljoj evaluaciji.

2. INTERAKTIVNI SADRŽAJI U E-UČENJU I ALATI

Inovativnost u nastavi je ključni zadatak i potreba obrazovnog sistema jer je cilj učiniti nastavu zanimljivom i kreativnom u cilju približavanja učenicima i tako pratiti njihove potrebe i sposobnosti. Potrebno je obezbediti učenicima prostor u kojem će moći izraziti svoje kreativne ideje i maksimalno iskoristiti svoje potencijale.

2.1 Simulacije, animacije i interaktivni sadržaji u nastavi

Interaktivni sadržaji obuhvataju pristup raznim informacijama bilo da su to video i audio materijali, prezentacije ili multimedijalni katalozi, jednom rečju svi sadržaji prikazani na moderan i atraktivan način.

Gotovi programi sa interaktivnim sadržajem pružaju mogućnost učenja kod kuće bez prisustva nastavnika. Ovakvi programi moraju biti sa što manje grešaka, znači učenik ne sme da nauči pogrešno interpretirane sadržaje i kada mu je potrebna pomoć za razumevanje određene materije računar „mora biti spreman“ da pruži traženu pomoć. Ovakav vid nastave gde računar ima ulogu nastavnika zastupljen je u svetu. Prednosti korišćenja računara u nastavi su višestruke, a kao glavne mogu se istaći interaktivnost i individualna pažnja (Vilotijević,1999).

Prva prednost se odnosi na to da računar omogućava svakom učeniku da ima aktivnu ulogu u procesu učenja, za razliku od pasivne uloge knjige. Učenik više nije posmatrač već aktivni učesnik u procesu učenja.

Druga prednost se odnosi na činjenicu da nastavnici treba da znaju da su učenici različiti, tj. nemaju svi isto predznanje i ne uče na isti način. Međutim, mnogi od naših konvencionalnih prilaza obrazovanju koriste rigidne procedure iste za sve učenike i ne dozvoljavaju uzimanje u obzir ovih razlika. Prednost računara je u tome da se dobrim softverom može individualizovati podučavanje (Đukić,1995). Pošto svi učenici ne uče istom brzinom, tj. potrebno im je različito vreme da prođu kroz gradivo, računar im to takođe omogućava.

Animacija je brzo prikazivanje sekvenci 2D ili 3D slika ili pozicija modela postavljenih tako da stvaraju iluziju pokreta. To je optička iluzija pokreta zbog fenomena istrajnosti vida i može biti napravljena i prikazana na različite načine (Weiss i ostali, 2002). Najčešća metoda prikazivanja animacija je slika u pokretu ili video program.

Računarska simulacija je eksperiment uzrokovanja koji se izvršava pomoću računara (Fishman, 1996). Ključni deo svake računarske simulacije je model koji obuhvata veličine čije su vrednosti nepredvidljive i zato moraju da se uzrokuju iz odgovarajuće populacije. Model se najčešće predstavlja upotrebom računarskog programa, a program u stvari uzrokuje slučajne

promenljive, izvršava proračune modela i izveštava o ishodu, obično u obliku jedne ili više brojčanih vrednosti.

Primena simulacija u nastavi je savremen način izvođenja nastave pomoću računara pogodniji u prirodnim, posebno tehničkim disciplinama, u kojima postoji mogućnost vizualizacije procesa (Namestovski, 2008). Sa aspekta učenika, simulacija pomoću računara obezbeđuje dvosmerni prenos podataka jer učenik može samostalno da upravlja varijablama modela. Iako je ključna prednost simulacije istraživačka funkcija pomoću računara ona ne sme da zameni eksperimentalni, laboratorijski i praktičan rad u nastavi. U uslovima siromašnog školskog sistema, ipak dobija opravdanje da se koristi kao zamena eksperimenta jer je za mnoge instrumente potrebno daleko više opreme i sredstava nego za korišćenje računara.

Još jedna prednost korišćenja računarskih simulacija u nastavi danas jeste i to što je većina obrazovnih računarskih simulacija napisana u programskom jeziku **Java** koji ima besplatnu podršku, veoma je jednostavna za upotrebu i postoji mnoštvo gotovih programskih alata koji izvoze simulacije koje se mogu implementirati u web strane .

Simulacije se mogu smatrati i kao poseban tip tzv. kognitivnih alata, odnosno one daju mogućnost učenicima da proveravaju različite pretpostavke menjajući uslove određenih sistema (npr. fizičkih, hemijskih ili bioloških) pružajući tako dublji uvid u određeni problem. Samo svojstvo simulacija je da pružaju mogućnost menjanja ulaznih i izlaznih podataka modela, što vodi i dubljem razumevanju pretpostavke odnosno modela koji se posmatra jer se dobija svojevršno kognitivno razumevanje kako neki sistem zaista funkcioniše (Cekuš i Namestovski, 2005).

Zbog svoje obrazovne vrednosti u skoroj budućnosti računarske simulacije bi trebale biti standardan alat u nastavi. Pogotovo zbog činjenice da simulacije ne zahtevaju nekakve posebne tehničke veštine ili tehnologije. Ovakvim programima moguće je vrlo efektivno doprineti dubljem i na taj način boljem razumijevanju koncepata i problema u različitim modelima, pogotovo u prirodnim i tehničkim naukama.

2.2 Alati za kreiranje interaktivnog sadržaja u e-učenju

Alati za e-učenje su namenjeni za održavanje nastave putem elektronskih uređaja. Važno je napomenuti da održavanje e-nastave nije isto što i održavanje klasične nastave u učionicama. Isto tako predavači kvalifikovani za održavanje nastave u učionicama ne moraju nužno biti kvalifikovani i za održavanje e-nastave (Materijal : E-Lab TFC, L1).

Na osnovu istraživanja (Horton, 2003) alati za e-učenje su podeljeni u tri osnovne grupe:

- Alati za pristupanje e-učenju,
- Alati za isporuku e-učenja,
- Alati za kreiranje sadržaja e-učenja.

U ovom praktikumu akcenat je na treću grupu alata, odnosno osvrnućemo se na alate za kreiranje interaktivnog sadržaja za elektronsko učenje¹ (u daljem tekstu e-učenje). E-učenje je proces razmene i prenosa znanja i veština preko mreže uz korišćenje računarskih aplikacija i okruženja u procesu učenja. Te aplikacije i procesi obuhvataju učenje preko Web-a, preko računara, u digitalnim učionicama, kao i digitalnu saradnju odnosno kolaboraciju. Sadržaji se prenose preko Interneta, audio i video traka, satelitske televizije i CD-ROM-a.

Važno napraviti razliku između pojmova e-učenje i obrazovanje na daljinu. Obrazovanje na daljinu je širi pojam od pojma e-učenje. E-učenje može biti vid obrazovanja na daljinu (u slučaju kada se ono organizuje u online okruženju) ali obrazovanje na daljinu se ne može smatrati elektronskim učenjem jer ono podrazumeva upotrebu i drugih medija (štampe, radio, televizije).

Najprihvatljivijim modelom primene elektronskog učenja (obrazovanja na daljinu) smatra se mešovita (hibridna) nastava koja predstavlja adekvatno balansiranu primenu rada u učionici sa radom uz pomoć medija/tehnologije. Na taj način se omogućava da se u isto vreme iskoriste prednosti oba načina.

Neki od najčešće korišćenih alata za kreiranje multimedijalnih sadržaja za e-učenje su (Ardito i ostali, 2004):

- Camtasia Studio,
- Adobe Captivate ,
- Articulate ,
- Flash ,
- Udutu itd.

Sledeća tabela prikazuje još neki alate za e-učenje koji se koriste za kreiranje interaktivnog sadržaja (animacije, simulacije, prezentacije, kvizovi, testovi itd.).

¹ Najčešće se elektronsko učenje pojednostavljeno shvata kao digitalizacija nastavnih sadržaja i smeštanje ovih materijala u elektronski prostor. Ovo svakako ima svoje prednosti kada je cilj povećanje dostupnosti materijala za učenje, ali to ne vodi postizanju obrazovnih ciljeva i željenih obrazovnih postignuća kod učenika/studenata. Kvalitet primene elektronskog učenja, posmatrano na nivou organizacije nastave, najpre zavisi od profesionalne osposobljenosti nastavnika/predavača. Neki od preduslova su posedovanje određenog nivoa medijske i informatičke pismenosti i onih znanja koja se sve češće smatraju presudnim za uspešan pedagoški rad - poznavanje teorija učenja i osposobljenost za primenu adekvatnih pedagoških modela, principa i nastavnih metoda (<http://www.eucenje.rs>).

Tabela 1: Lista alata za kreiranje interaktivnog sadržaja za e-učenje

Animacije	Prezentacije	Kvizovi i simulacije
<ul style="list-style-type: none"> • <i>Alpoy</i> • <i>Avatar Photo</i> • <i>Avatars</i> • <i>Blabberize</i> • <i>DoInk</i> • <i>Fodey.com</i> • <i>Gickr</i> • <i>GIFup</i> • <i>Meez</i> • <i>PhotoFace</i> • <i>Voki</i> • <i>Xtranormal</i> 	<ul style="list-style-type: none"> • <i>BubbleShare</i> • <i>Capzles</i> • <i>Freepath</i> • <i>Moblyng</i> • <i>MyLiveGallery</i> • <i>PhotoSnack</i> • <i>PresentationsETC</i> • <i>SlideRocket</i> • <i>Slidesix</i> • <i>TimeRime</i> • <i>Timetoast</i> • <i>VoxPix</i> 	<ul style="list-style-type: none"> • <i>Armored Penguin</i> • <i>ClassMarker</i> • <i>ClassTools</i> • <i>Edventure Maze</i> • <i>GeoGebra</i> • <i>Hot Potatoes</i> • <i>Mobile Study</i> • <i>MyStudiyo</i> • <i>Photo Soup</i> • <i>Quibblo</i> • <i>SMILE</i> • <i>TexToys</i>

Alati za kreiranje interaktivnih sadržaja su značajni faktori efikasnosti nastave i učenja, posebno u nastavi prirodnih i tehničkih nauka. Zahvaljujući razvoju digitalne tehnologije i novim mogućnostima računara, omogućen je razvoj računarskog modelovanja i simulacije (Mašović, Saračević i ostali, 2010, *InfoFest*). Primena softverskih modela u nastavi ima za cilj, ne samo stvaranje tehničke pismenosti, već i dodir sa računarskim tehnologijama i mogućnost njihove upotrebe. Veoma su rasprostranjeni alati za procesni pristup razvoja softvera prvenstveno za e-učenje primenom UML alata (Mašović, Saračević i ostali, 2011, *YuInfo*) a takođe postoje i Data Mining² tehnike u e-učenju i evaluaciji (Mašović i Saračević, 2010).

2.2.1 Mogućnosti primene alata za e-učenje u nastavi

Mogućnosti primene informacionih tehnologija u nastavi su višestruke. Ciljevi uvođenja novih metoda u nastavi primenom odgovarajućih alata odražavaju zainteresovanost društva za naučnu i tehničku pismenost, za pripremanje ljudi za široku primenu naučnih dostignuća i za obučavanje onih koji će ta dostignuća dalje razvijati i usavršavati (Cekuš i Namestovski, 2005).

Savremena nastava teži kao individualizaciji³. Samostalni rad učenika uz pomoć nastavnika, jedan je od osnovnih ciljeva savremene nastave. Učenika treba korak po korak voditi

² Inteligentna obrada podataka

³ Individualizacija nastave, kako to definiše M.Đukić, nije samo didaktički princip, već je, među savremenim didaktičarima shvaćena kao najznačajnija inovativna snaga u razvoju i modernizaciji

do samostalnog rada, sistematski i kontinuirano ga osposobljavati i učiniti da postane samostalan u procesu učenja. Samostalan rad se može definisati kao organizovana angažovanost učenika u kontrolisanim uslovima na rešavanju postavljenih zadataka i ispunjavanju određenih ciljeva. U ovom radu se prvenstveno stavlja akcenat na uvođenje e-učenja u obrazovnom sistemu kao prateći način uz tradicionalne metode obrazovanja (Saračević, Mašović i ostali, 2011, *YuInfo*).

Individualna nastava predstavlja najbolje rešenje za izvođenje nastave. U tome mogu znatno doprineti alati za e-učenje koji imaju mogućnost kreiranja interaktivnog sadržaja (multimedija, simulacije, animacije). Sve ovo doprinosi tome da se može vršiti izbor zadataka (sadržaja) koji najviše odgovaraju određenom učeniku. Individualizovati nastavu, znači, orijentisati se na realne tipove učenika, uzeti u obzir razlike među njima, uskladiti ih, i menjati metode i postupke pedagoškog delovanja prema tim razlikama, omogućiti učenicima da napreduju prema svom tempu i mogućnostima (Kurnik, 2000, *Savremena metodika*).

U petom poglavlju ovog rada se navodi i primena alata za e-učenje u praktimumu sa talentovanim decom kao i pružanje podrške u funkciji motivacije talentovanih učenika.

Softver koji se koristi u oblasti obrazovanja predstavlja savremen način učenja, pa se još i naziva i obrazovni računarski softver. On obuhvata programske jezike i alate, određenu organizaciju nastave i učenja, koja se bazira na logici i pedagogiji. Pod obrazovnim računarskim softverom podrazumevaju se računarski programi koji se mogu koristiti u okviru nastave koji pomažu i usmeravaju u fazama individualne nastave (Radosav, 2005).

2.2.2 Unapređenja nastave pomoću alata za e-učenje

Unapređenje kvaliteta nastave podrazumeva između ostalog modernizaciju izvođenja nastave pomoću savremenih nastavnih sredstava (alata, programa, tehnika) odnosno korišćenjem savremene informacione tehnologije. Ono što je bitno navesti prilikom procesa unapređenja nastave pomoću alata za e-učenje, potrebno je da postoje opremljeni informatički kabineti sa mogućnošću korišćenja interneta. Da bi proces unapređenja nastave išao u pravom smeru i da bi imao pozitivne rezultate bitno je da nastavnici pohađaju seminare koji imaju za cilj da unaprede rad nastavnika u primeni savremenih nastavnih metoda i oblika rada, zatim ciljevi su i

nastave, odnosno kao imperativ vremena u kome živimo, permanentna inovacija, pretpostavka racionalizacije nastave. (Đukić, Mara, 2003:68). Suština individualizovane nastave je „u različitim didaktičko-metodičkim postupcima usmerenim ka zadovoljenju individualnih potreba svakog učenika tako da se do maksimuma utiče na njegovo učenje i razvoj.“ (Đukić, Mara, 1995:63).
Link: <http://www.inkluzija.org>

ovladavanje znanjima iz savremenih tehnologija učenja, osnažiti rad nastavnika u individualizaciji nastave, pružanje pomoći nastavnicima u planiranju nastavnog procesa, pružanje pomoći u ocenjivanju učenika itd.

Seminar tj. interni kurs u okviru škole koji je bio organizovan (od strane autora ovog rada) za potrebe istraživanja, koji je trajao 4 školska časa, je imao za cilj da predstavi nastavnicima matematike, informatike, fizike i tehničkog obrazovanja kao i profesorima razredne nastave, unapređenje nastave kroz primenu modela i simulacija (struktura i plan ovog kursa su navedeni u prilogu). Neki od predloga kako se može unaprediti nastava primenom alata su sledeći (Namestovski, 2008).:

- Unapređenje nastave kroz simulacije, samostalne fleš i druge animacije, multimedijalni obrazovni softver, elektronski testovi znanja kao i programi za samostalan rad. Sve pomenute tehnike zahtevaju individualni oblik rada ili u krajnjem slučaju rad u parovima, u zavisnosti od raspoložive opreme.
- Primena edukativnog video materijala koji se bavi tematikom iz određenog predmeta. Prikaz je moguć i preko video plejera povezanog na televizor ili video bim. Bitno je odabrati video materijale koji su namenjeni prvenstveno učenicima koji bi hteli nešto više da znaju te se može prikazati i na časovima redovne i dodatne nastave, kao i na raznim sekcijama uz veliku mogućnost izbora najboljih i najprikladnijih sadržaja.
- Kao najprostiji oblik primene interaktivnog sadržaja su PowerPoint prezentacije koje su pravljene za frontalni oblik rada i to za one nastavne teme u kojima preovlađuju teorijski sadržaji. U prezentacije je potrebno ugraditi kvizove i *.swf* i *.gif* animacije koje mnogo bolje prikazuju dinamičke procese gde god je to neophodno.

3. Java apleti kao alati za kreiranje interaktivnog sadržaja

Programski jezik JAVA nastao kao rezultat rada na projektu razvoja softvera za kontrolu različitih udaljenih elektronskih uređaja za domaćinstvo. Kako raspoloživi programski jezici nisu dovoljno odgovarali potrebama projekta, James Gosling i tim razvijaju potpuno novi jezik⁴.

⁴ Java je objektno orijentisani jezik razvijen od strane kompanije Sun Microsystems 1991. godine. Upravo zbog toga, Java je morala da bude jednostavna, brza, efikasna, pouzdana i lako prenosiva sa jednog uređaja na drugi. Ove osobine su je učinile idealnom za distribuciju izvršnih programa preko WWW, kao i za razvoj softvera koji je nezavisan od platforme na kojoj se izvršava. Java je doživela posebnu ekspanziju kada je 1994. godine napravljen prvi web browser, HotJava, koji je

Nakon nastanka Java programskog jezika, ubrzo se shvatilo da on daleko prevazilazi svoje polazne namene i da je možda baš to programski jezik koji bi mogao postati standard u budućnosti. To bi značilo da karakteristike takvog jezika ispunjavaju visoke zahteve koji se postavljaju pred savremeni alat za razvoj softvera najšire namene u vreme ekspanzije informaciono-komunikacionih tehnologija (Lemay i Cadenhead, 1998).

3.1 Glavne karakteristike Jave

Kao glavne karakteristike i ključne prednosti Java programskog jezika se mogu navesti (Arnold i Gosling, 2001):

- *Jednostavnost* koja se odnosi na sintaksu Jave, koja se oslanja na programske jezike C i C++, ali ona je daleko jednostavnija od njih. U Javi nema pokazivača, nizovi su realni objekti, a upravljanje memorijom je automatsko.
- *Objektno orijentisani koncepti* koji pružaju mogućnost za kreiranje fleksibilnih, modularnih programa. Kao i većina objektno-orijentisanih jezika, Java uključuje biblioteke klasa koje obezbeđuju rad sa osnovnim tipovima podataka, ulazom i izlazom, osnovnim Internet protokolima, kontrolama za kreiranje korisničkog interfejsa i dr.
- *Nezavisnost od platforme*, to znači da programi koji su pisani u Javi se lako prenose sa jednog kompjutera ili uređaja na drugi, bez obzira na različito radno okruženje tih uređaja. Jedini preduslov je da je na uređaju na kome se program izvršava instaliran interpreter za Javu, nazvan JVM (Java Virtual Machine).
- *Pogodnost za rad u mrežnom okruženju*, prednost da se ovaj programski jezik koristi u većini elektronskih uređaja (TV, VCR, mobilni telefoni, računari). Ovo je veoma bitno za unapređenja e-učenja.
- *Podržava konkurentno programiranje*⁵ pomoću niti (*threads*), što znači da Java programi mogu izvršavati više zahteva istovremeno.

mogao da izvršava Java aplete, posebne programe namenjene za izvršavanje u okviru čitača (*kurs: Internet Programerski Alati - Java programiranje, Autor: Vesna Šatev*).

⁵ Konkurentno programiranje predstavlja pisanje takvih programa koji se sastoje od više kooperativnih procesa i niti koje se izvršavaju simultano ili paralelno, pri tom koristeći zajedničke resurse računarskog sistema. Za pisanje konkurentnih programa mogu poslužiti mnogobrojna razvojna okruženja, među kojima su i programski jezik Java i C#, na koje se odnosi ovaj rad. Smatramo da je reč Java ušla u svakodnevni govor informatičkih stručnjaka i da se ona može

3.2 Java programski jezik i primena u nastavi

Analizirajući nastavu iz oblasti računarskih nauka uočava se potreba da se studenti na svojim studijama posebno edukuju za sve širu primenu aplikacija elektronskog učenja i poslovanja, čiji će korisnici u najvećem broju slučajeva oni biti.

U tu svrhu, kao jezik izbora u nastavi programskih jezika predlaže se programski jezik JAVA, smatrajući da je to jezik koji odlično odgovara prirodi i funkcijama aplikacija e-učenja. Kao predstavnik moderne generacije objektno-orijentisanih jezika, on studentima u nastavi omogućava brzo i efikasno savladavanje gradiva. Sa bogatom bibliotekom klasa, pogodan za Internet programiranje, portabilan, jednostavan za savladavanje i samostalno proširivanje znanja, sa komfornim i slobodno dostupnim alatima za brz razvoj aplikacija.

Ključne prednosti java programskog jezika u nastavi:

- većina obrazovnih računarskih simulacija napisana u programskom jeziku Java,
- kvalitetno izvođenje nastave je omogućeno upotrebom besplatnog softvera i apleta,
- nudi mogućnost da se kroz simulacije i modele olakša priprema nastave,
- da se učenicima približe procesi u prirodnim naukama i tehnici,
- java simulacije imaju mogućnost da se pojednostave nastavni sadržaji,
- mnogo mogućnosti koje nude nastavnicima kreiranje časova pomoću apleta,
- metodičke i didaktičke osobine teorijske i laboratorijske nastave su podržana java-om,
- mogućnost korišćenja nekih gotovih besplatnih programa koji su namenjeni nastavi predmeta iz prirodnih nauka.

3.3 Pojam Java apleta i osnovni koncepti

Apleti predstavljaju jednostavne programe koji u sebi sadrže jednostavne grafike. Apleti imaju široku primenu, mogu se koristiti uz bilo koji stil učenja. Aplet kao program je dizajniran na način da objašnjava svaku fizičku pojavu individualno. Apleti su pisani za web, to im nudi mogućnost pokretanja na svakoj platformi, takođe mogu se umetati u html dokumenta. Otvorenog su koda i kao takvi su besplatni, što daje mogućnost stručnim nastavnicima da dalje unapređuju svoju nastavu.

pisati na srpskom jeziku, uključujući ćirilčno pismo, zbog istog značenja kao u engleskom jeziku.
Link: http://www.cet.rs/cetcitaliste/CitalisteTekstovi/Visenitno_1.pdf

JAVA apleti su mali programi koji su uključeni unutar drugih aplikacija, obično neke web stranice prikazane unutar nekog browser-a. Informacija o apletu se nalazi unutar specijalnog taga na nekoj HTML strani. Kada *browser* učita neku stranicu koja koristi aplet, on učitava i sam aplet sa servera i izvršava ga lokalno, na korisnikovom računaru (Milosavljević i Vidaković, 2001). Kada pokrenemo animaciju direktno na ekranu možemo videti kako promena tih parametara utiče na celokupan sistem koji posmatramo. Najfleksibilniji softver za prikaz ovih animacija su ipak JAVA⁶ apleti.

Oni ne sadrže suviše fizičkih podataka (brojeva, jednačina i sl.) koji mogu biti zbunjujući za učenika već korisni za onoga koji dobro poznaje gradivo, prvenstveno misleći na posebno talentovanu decu. Preporučuje se kao dobar i koristan propratni materijal pri učenju. Mogu se koristiti i kod tradicionalnih načina u učenju, npr. kod evaluacije ili kao domaći zadaci.

Sa jedne strane, u ovom praktikumu se navode metode kako da se olakša nastavnicima priprema nastave, a sa druge strane da se učenicima približe procesi u prirodnim naukama i tehnici. Navedeni su i konkretni primeri primene apleta u nastavi matematike, koji mogu biti podrška u funkciji motivacije učenika, pre svega da se pojednostave dodatni složeni nastavni sadržaji talentovanim učenicima i da se predstave mogućnosti i načini kreiranja časa pomoću apleta. Interaktivne simulacije su naročito interesantne i zanimljive u prirodnim i tehničkim naukama jer nam daju mogućnost promene određenih fizičkih parametara.

Iako jedan isti program može biti kreiran i kao aplet i kao aplikacija⁷, postoji veoma značajna razlika između ove dve vrste izvršavanja programa. Ona se ogleda u skupu zabrana koje imaju apleti (Arnold, Gosling, Holmes, 2001):

- apleti ne mogu da čitaju niti pišu u korisnikov sistem fajlova,
- apleti ne mogu da komuniciraju sa bilo kojim serverom, sem onog sa kog su učítani,

⁶ JAVA ima ugrađenu podršku Web-u, jedinstvenu među programskim jezicima. Preko klase Applet iz paketa *java.applet*, omogućeno je pisanje interaktivnih programa koji se izvršavaju u okruženju Web pretraživača. Takvi programi, takozvani apleti, oživljavaju svojom pojavom stranice Interneta. Ova mogućnost je idealna jednostavna alternativa za pisanje inače složenih klijent delova aplikacije. Što se tiče server strane tipičnih Web aplikacija, JAVA takođe ima rešenje u vidu tzv. servleta API (Venkataraman,2001).

Programski jezik Java napravio je velik prodor u oblast Interneta, te je i pored drugih tehnologija (mobilna telefonija, IPod i sl.) njegovo korišćenje postalo gotovo nezaobilazno na web sajtovima (Street i Goodman, 1998).

⁷ Program koji se izvršava nezavisno od internet (web strane)

- apleti ne mogu da pokreću druge programe na korisnikovom računaru,
- apleti ne mogu da učitavaju programe lokalne platforme itd.

3.3.1 Kreiranje i metodi apleta

Da bismo kreirali aplet, potrebno je da kreiramo podklasu klase *Applet*. Ova klasa je deo paketa *java.applet* i ona obezbeđuje metode koji su neophodni za rad unutar *browser-a*.

Definicija glavne klase apleta bi izgledala ovako:

```
public class MojApplet extends java.applet.Applet  
  
    { // telo apleta }
```

Osim ove glavne klase, aplet može da sadrži i više pomoćnih klasa, ali je glavna klasa apleta ta koja se izvršava prilikom pokretanja apleta. Takođe, neophodno je da glavna klasa bude deklarisan kao *public*⁸ (Arnold, Gosling, Holmes, 2001).

Kod aplikacija, Java prilikom pokretanja programa poziva *main()* metod aplikacije. Kod izvršavanja apleta, Java kreira instancu glavne klase apleta i zatim se poziva serija metoda te instance⁹. Različiti apleti mogu da koriste istu klasu, ali različite instance.

Apleti imaju različite aktivnosti koje odgovaraju različitim etapama u životu apleta, kao što su inicijalizacija, iscrtavanje ili događaji miša. Svaka aktivnost ima odgovarajući metod, pa kad se neki događaj desi, Java poziva metod specifičan za taj događaj.

Pet najvažnijih etapa u izvršavanju apleta su inicijalizacija, pokretanje, zaustavljanje, uništavanje i iscrtavanje. U nastavku, kroz primere biće objašnjena inicijalizacija, pokretanje i zaustavljanje kao i rad sa događajima i metod za iscrtavanje (Milosavljević i Vidaković, 2001).

- A) **Postupak Inicijalizacije** (ovoj fazi odgovara metod *init()*) se dešava kada se aplet prvi put učita, slično kao metod *main()* kod aplikacija. Inicijalizacija apleta obično uključuje čitanje i prosleđivanje parametara apletu, kreiranje pomoćnih objekata, podešavanje inicijalnog stanja apleta, učitavanje slika i fontova i dr.
- B) **Pokretanje i zaustavljanje** (pomoću metoda *stop()* i *start()*) - nakon inicijalizacije, aplet se pokreće. Pokretanje se razlikuje od inicijalizacije u tome što aplet može biti pokrenut

⁸ Javna, dostupna svim klasama

⁹ Primerak klase, objekat klase

nekoliko puta, dok se inicijalizacija dešava samo jednom. Na primer, ukoliko korisnik preteći neki link do neke druge strane napusti stranu koja sadrži aplet, podrazumeva se da će sve niti pokrenute u apletu nastaviti sa izvršavanjem.

- C) **Iscrtavanjem** (ovoj fazi odgovara metod *paint()*) aplet praktično crta bilo šta na ekranu (tekst, sliku, grafičke objekte, pozadinu i sl.). Iscrtavanje se dešava veoma često, nakon inicijalizacije, nakon pomeranja prozora *browser*-a, u animacijama i sl. Definicija metoda *paint()* je sledeća:

```
public void paint(Graphics g) { ..... }
```

Ovaj metod koristi argument, objekat klase *Graphics*. Ovaj objekat se kreira i prosleđuje metodu *paint()* od strane *browser*-a. Obzirom da klasa *Graphics* pripada paketu *java.awt*, neophodno je da uvezete ovaj paket u vaš kod.

- D) **Rad sa događajima** (event listeners) - u Javi, događaji se obrađuju pomoću tzv. osluškivača događaja. Obzirom da se jedan događaj (npr. pritisak na dugme) sastoji iz nekoliko akcija (1- pozicioniranje kursora miša na dugme, 2- pritisak levog dugmeta miša, 3- otpuštanje levog dugmeta miša), komponenta na osnovu svih tih preduzetih akcija generiše događaj neke od podklase klase *AWTEvent*, koji se onda prosleđuje odgovarajućem osluškivaču. Osluškivač tada preduzima određene akcije koje su potrebne da se događaj pritiskanja dugmeta obradi, bez ulaženja u detalje o pojedinačnim akcijama od kojih se taj događaj sastoji.

```
class Reakcija implements ActionListener { ..... }
```

Događaji ovog tipa se prosleđuju svim osluškivačima tipa *ActionListener*. Unutar klase *ActionEvent* postoje sledeći indikatori događaja:

- *action_performed* – akcija je izvršena,
- *alt_mask* – bilo je pritisnuto dugme alt za vreme izvršavanja akcije,
- *ctrl_mask* - bilo je pritisnuto dugme control za vreme izvršavanja akcije,
- *shift_mask* – bilo je pritisnuto dugme shift za vreme izvršavanja akcije.

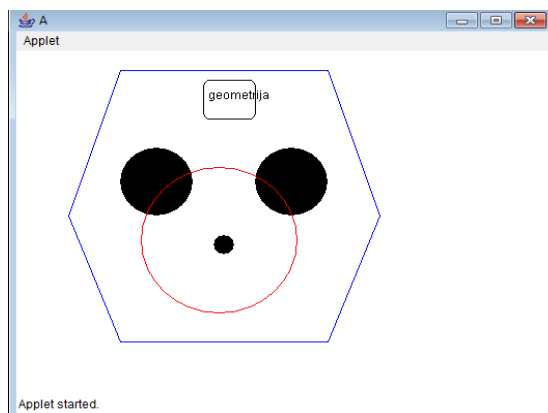
Postoje događaji i na klik miša i taster:

MouseEvent generiše se od strane komponente nad kojom se desila neka akcija mišem. Ova klasa se koristi za sve događaje miša (pritisak na dugme miša, pokretanje kursora, povlačenje i sl.). Svi ovi događaji se prosleđuju osluškivačima tipa *MouseListener* i *MouseMotionListener*.

KeyEvent generiše se od strane komponente nad kojom je pritisnut, otpušten ili otkucan neki taster sa tastature. Ovi događaji se prosleđuju osluškivaču tipa `KeyListener`.

Sledi primer veoma jednostavnog apleta sa metodama *init()* i *paint()* (Saračević, 2011).

```
import java.awt.*;
import java.applet.*;
import java.awt.Polygon;
public class MojApplet extends Applet {
public void init() { }
public void paint(Graphics g) {
    g.drawString("GEOMETRIJA",185, 50 );
    g.drawRoundRect(180,30,50,40,12,12);
    g.fillOval(100,100,70,70);
    g.fillOval(230,100,70,70);
    g.fillOval(190,190,20,20);
    g.setColor(Color.red);
    g.drawOval(120,120,150,150);
    int x[]={ 100,300,350,300,100,50,100};
    int y[]={ 20,20,170,300,300,170,20};
    Polygon poly=new Polygon(x,y,7);
    g.setColor(Color.blue);
    g.drawPolygon(poly);
}
}
```



Slika 1. Aplet sa nizom Geometrijskih oblika

Analiza koda

- Ovaj Aplet iscrtava string - *Geometrija* od koordinate (175,50) i postavlja oko njega zaobljen pravougaonik,
- iscrtava tri ispunjena kruga iz gornjeg levog ugla sa kordinatama i poluprečnikom:
 - (100,100), r=70
 - (230,100), r=70
 - (190,190), r=20
- postavlja boju na crvenu,
- iscrtava ne ispunjen krug od koordinate (120,120) poluprečnika r=150,
- postavlja boju na plavu za naredne grafičke elemente,
- iscrtava poligon kroz temena:
 - $A_1=(100, 20)$ $A_2=(300, 20)$ $A_3=(350, 170)$
 - $A_4=(300, 300)$ $A_5=(100, 300)$ $A_6=(50, 170)$

Ovaj aplet implementira samo metod *paint()*, koji ispisuje poruku na ekranu. Obzirom da je sam aplet veoma jednostavan, metodi *init()* nije ponovo implementiran, tj. preuzeta je njegova implementacija iz klase *Applet*. Metod *paint()* preuzima objekat *g* klase *Graphics* od samog *browser*-a. Ovaj objekat u stvari predstavlja prostor u okviru web strane u kome će se aplet pojaviti.

E) Postavljanje apleta na web stranu (Milosavljević i Vidaković, 2001) gde je najpre neophodno kreirati *HTML* stranu i formirati tag `<APPLET>` taga unutar tela *HTML* stranice. Ovaj tag ima nekoliko atributa pomoću kojih se prosleđuje informacija *browser*-u kako da prikazuje sam aplet.

- *CODE* atribut se odnosi na ime klasnog fajla koji sadrži aplet. Ovaj fajl mora biti u istom direktorijumu kao i sama strana. Takođe, ime fajla mora da uključi i ekstenziju *.class*.
- *CODEBASE* atribut sadrži *URL* ili relativnu putanju do klasnog fajla, ukoliko se on ne nalazi u istom direktorijumu kao i sama *HTML* strana.
- *WIDTH* i *HEIGHT* atributi određuju koliki prostor je dodeljen apletu unutar web strane
- *ALIGN* atribut određuje kako će aplet biti poravnati unutar strane.
- *HSPACE* i *VSPACE* određuju broj piksela između apleta i teksta koji ga okružuje.

Između tagova `<APPLET>` i `</APPLET>` možete da ubacite bilo koji tekst. Ovaj tekst će biti prikazan umesto apleta, ukoliko iz nekog razloga aplet ne može da bude prikazan. Znači aplet

nema metodu main (izvršnu metodu), tako da ga ne možemo pokrenuti na do sada poznat način – iz komandne linije. Potrebno je ovakav aplet ugraditi u HTML stranicu u okviru koje će biti prikazan. Sledi primer HTML stranice:

```
<HTML>
<HEAD>
<TITLE>Test strana</TITLE>
</HEAD>
<BODY>
Ukoliko vaš browser to dozvaljava, ispod možete da vidite Mojaplet<br>
<APPLET CODE = "MojAplet.class" WIDTH = "400" HEIGHT = "300"
HSPACE = "0" VSPACE = "0" ALIGN = "middle">
Na ovom mestu bi trebalo da bude prikazan aplet
</APPLET>
</BODY>
</HTML>
```

Da bi se obezbedio veći nivo interaktivnosti može se izvršiti prosleđivanje parametara apletu (Beene, 2004). Ukoliko je potrebno, parametre možete da prosledite apletu pomoću taga `<PARAM>`. Ovaj tag ima dva atributa, `NAME` i `VALUE`. Atribut `NAME` predstavlja bilo koje ime koje odaberete za određeni parametar. Atribut `VALUE` predstavlja konkretnu vrednost koja se prosleđuje. `<PARAM>` tag se ubacuje između tagova `<APPLET>` i `</APPLET>`.

Na primer:

```
<APPLET CODE="MojAplet.class" WIDTH=100 HEIGHT=100>
    <PARAM NAME=font VALUE="TimesRoman">
    <PARAM NAME=velicina VALUE="36">
</APPLET>
```

Prethodni primer prosleđuje dva parametra apletu. Neophodno je ove parametre preuzeti i obraditi u samom kodu apleta. To se postiže unutar metoda `init()` pozivanjem metoda `getParameter()`.

Na primer:

```
String imeFonta = getParameter("font");
```

Metod `getParameter()` vraća vrednost definisanu atributom `VALUE` odgovarajućeg `PARAM` taga. Ime parametra unutar `PARAM` taga i ime prosleđeno metodom `getParameter()` moraju da budu identični.

3.4 Primena apleta u nastavi matematike

Jedan od najvažnijih ciljeva poučavanja matematike je naučiti učenike misliti, odnosno osposobiti ih za rešavanje problema u budućem životu (Knežević, 2004). Konkretno se ovde misli na primenu stečenog teorijskog znanja u praksi. Uloga nastavnika matematike kao izvora informacija znatno je manja u odnosu na ulogu nastavnika koji će voditi i usmeravati učenike na putu sticanja novih znanja. U svom radu (Polya, 1957) mađarski matematičar George Polya opisuje model rešavanja problema u matematici koji se sastoji od četiri koraka:

1. razumevanje problema (SEE),
2. donošenje plana (PLAN),
3. izvođenje plana (DO),
4. pogled unazad (CHECK).

Možemo zaključiti da se istraživački usmerena nastava potpuno uklapa u navedni model rešavanja problema. Odnosno da primena apleta i interaktivnih sadržaja itekako nalazi primenu u ovom modelu. Karadag i McDougall napravili su još jedan korak dalje, Polya-in koncept ugradili su u *GeoGebra* dinamično okruženje za učenje (*dynamic learning environment*).

Posebno mesto zauzimaju alati koji imaju visoke mogućnosti kada je u pitanju vizuelizacija, pa svoju primenu nalaze u geometriji. Ovi alati su nam neophodni u okruženju e-učenja za otkrivanje i ispitivanje forme nastale konceptima geometrije, pa ih možemo svrstati u tri grupe (Mitrović, Devetaković i ostali, 2007):

1. Alati za vizualizaciju u okviru okruženja za e-učenje,
2. Alati izvan okruženja za e-učenje, a čiji je proizvod moguće ubaciti u okruženju,
3. Alati koji su u potpunosti integrisani u okruženju za e-učenje.

Eksplorativni alati¹⁰ za vizuelizaciju mogu biti svrstani u 2D, 3D i 4D okruženja.

2D alati su uglavnom već integrisani u postojeće sisteme za e-učenje i predstavljeni su kroz razne načine postavljanja slika ili konvertovanje tekstualnog ulaza u 2D sliku, 3D prikaz iako već poseduje svoje standarde na internet gotovo kao i formati za prikaz slika, na žalost nije

¹⁰ Konkretnan primer gde je izvršena **integracija apleta u e-learning sistem** je Arhitektonski fakultet, Univerziteta u Beogradu (<http://www.arh.bg.ac.rs>), gde se primenjuje „JavaView lite” alat u arhitekturi (o ovom alatu će biti reči u narednom delu ovog rada) i AMRES eLearning (Akademska mreža Srbije – <http://elearning.amres.ac.rs>, <http://amres.ac.rs>).

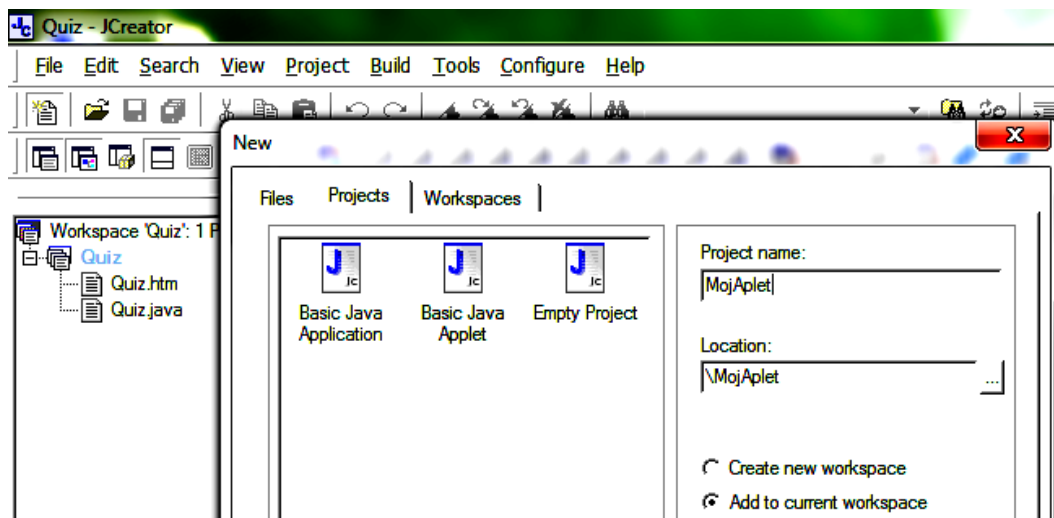
zastupljen u većini sistema za e-učenje ili bar nije deo standardnih paketa, kako komercijalnih tako i besplatnih (Mitrović, Devetaković i ostali, 2007).

4D prikaz je prikaz 3D objekata koji imaju različita stanja u zavisnosti od parametara kojih ih definišu. Ovakav prikaz se najčešće i koristi u arhitektonskoj geometriji, ali ga je i najteže integrisati u samo okruženje za e-učenje. Za to se obično koriste java apleti i specijalizovani programi kao što je *Wolframova Mathematica*, ali ne retko i Adobe Flash aplikacije. Najpopularniji alati koji se koriste za kreiranje apleta za potrebe nastave matematike su:

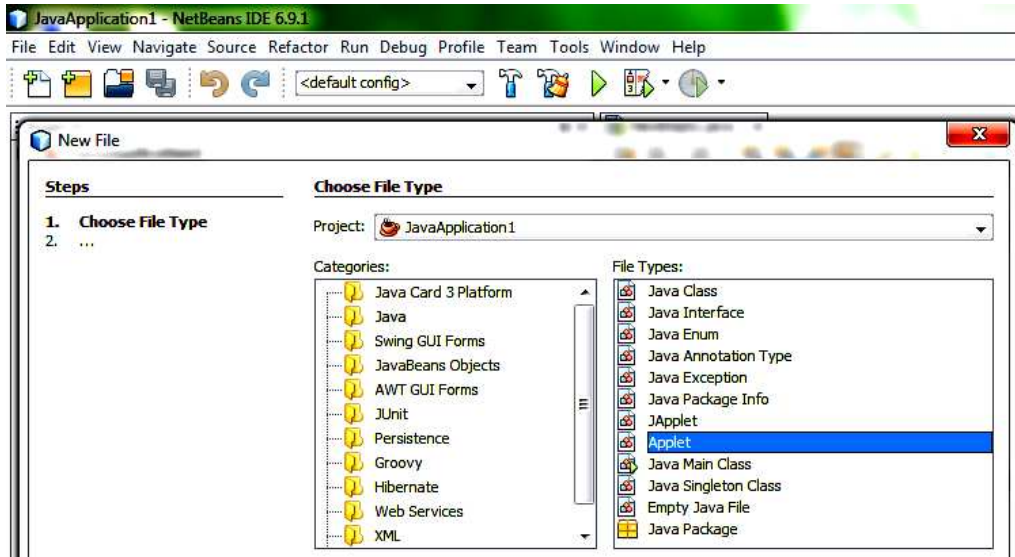
- | | | |
|-------------------|---------------------|--------------------------------|
| ▪ <i>GeoGebra</i> | ○ <i>iGeom</i> | • <i>Tabulae</i> |
| ▪ <i>Jeometry</i> | ○ <i>C.a.R.</i> | • <i>Wolfram Mathematica</i> |
| ▪ <i>Geonext</i> | ○ <i>CaRMetal</i> | • <i>Easy Java Simulations</i> |
| ▪ <i>GRACE</i> | ○ <i>Cinderella</i> | • <i>Java View Lite</i> |

Geogebra je alat koji se se zbog mogućnosti dinamičnog prikaza i interaktivnosti matematičkih objekata u nastavi koristi za objašnjavanje, istraživanje i modeliranje matematičkih koncepta i njihovih međusobnih odnosa.

Alati koji se takođe mogu koristiti za kreiranje apleta koji imaju opštu namenu su i **Jcreator** i **NetBeans**, i ujedno se koriste i za java programiranje.



Slika 2. Okruženje JCreator-a



Slika 3. Izbor kreiranja apleta u NetBeans-u

Sledi primer apleta koji je rađen za potrebe kursa i upoznavanja nastavnika sa strukturom i kreiranjem apleta. Aplet je kreiran u *JCreator* softveru.

Primeri apleta koje je programirao u JCreator-u autor ovog rada za potrebe nastave :

Primer 1. Aplet koji iscrtava string sa tekstom “GEOMETRIJSKA TELA” sa koordinatama (325, 50) i postavlja oko njega zaobljen pravougaonik, sa koordinatama (320,30,140,40,12,12).

- iscrtava neispunjen krug - (60,60,150,150), i ispunjen krug - (60,60,150,150).
- iscrtava neispunjen kvadrat sa koordinatama - (360,130,140,140)
- iscrtava poligon kroz temena:

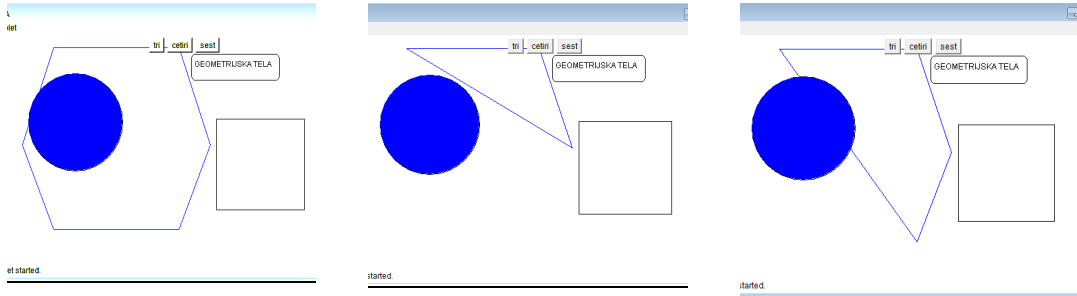
$$A_1=(100, 20) \quad A_2=(300, 20) \quad A_3=(350, 170)$$

$$A_4=(300, 300) \quad A_5=(100, 300) \quad A_6=(50, 170)$$

- Boju ispunjenog kruga definisati kao nova_boja (*početna boja je plava*) i dodeliti funkcionalnost na tastere: B- “plav krug”, G- “zeleni krug”, Y- “zuti krug”.
- Ako se mišem klikne u domen ispunjenog kruga, on se transformiše u manji ispunjen krug (100,100,70,70), a zatim ako se ponovo klikne unutar iscrtava se ispunjen krug - (60,60,150,150). *postaviti uslov za domen \rightarrow if ($x > 60 \ \&\& \ x < 150 \ \&\& \ y > 100 \ \&\& \ y < 220$).

- Formirati tri dugmeta (tri_temena, cetiri_temena, osam_temena), klikom na jedan od tri dugmeta iscrtava se poligon sa određenim brojem temena.

U prilogu **S1** je naveden izvorni kod navedenog apleta sa događajima.



Slika 4. Aplet sa događajima na taster i na klik miša

Primer 2. Napisati applet koji iscrtava neispunjen pravougaonik sa koordinatama (360,130,140,180). U prilogu **S2** je naveden izvorni kod apleta.

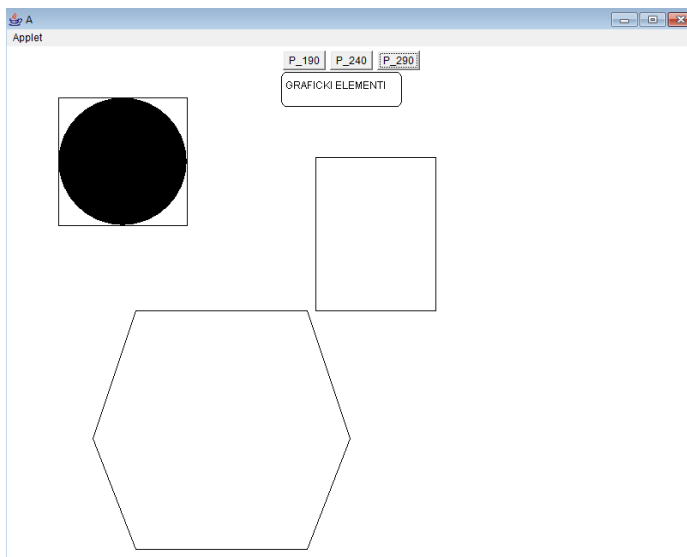
- iscrtava string sa tekstom “GRAFICKI ELEMENTI” sa koordinatama (325, 50) i postavlja oko njega zaobljen pravougaonik (320,30,140,40,12,12).
- iscrtava neispunjen kvadrat - (60,60,150,150), i ispunjen krug K1 - (60,60,150,150).
- iscrtava poligon kroz temena (*postaviti plavu boju za poligon*):

$$P_1=(100, 20) P_2=(300, 20) P_3=(350, 170)$$

$$P_4=(300, 300) P_5=(100, 300) P_6=(50, 170)$$

Uvesti novu promenljivu “Dimenzija_poligona”, i dodeliti funkcionalnost na tastere i klikom na njih iscrtava poligon sa: 4, 5 i 6-temena a 0-briše krug K1 (“clearRect”).

Ako se mišem klikne u domen kruga K1, on se transformiše u String “PLAY”(160,160), crvene boje, *postaviti uslov za domen \rightarrow $if(x>60 \ \&\& \ x<150 \ \&\& \ y>100 \ \&\& \ y<220)$. Formirati tri dugmeta i klikom na jedan od tri dugmeta pomera se poligon za odgovarajuću poziciju: P_{190} -g.translate(50,190), P_{240} -g.translate(50,240), P_{290} - g.translate(50,290).



Slika 4.1. Aplet sa mogućnošću menjanja pozicije oblika

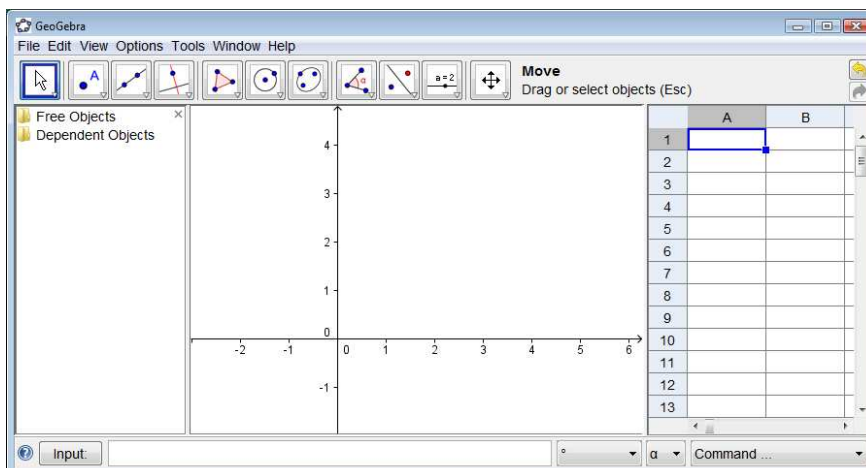
3.4.1 GEOGEBRA, alat za dinamičku matematiku

GeoGebra je program za dinamičku matematiku koji povezuje geometriju, algebru i matematičku analizu. Razvili su ga Markus Hohenwarter i međunarodni tim programera, za nastavu i učenje matematike u školama.

GeoGebra ima tri različita prikaza matematičkih objekata:

1. grafički prikaz,
2. algebarski (brojčani) prikaz i
3. tabelarni prikaz.

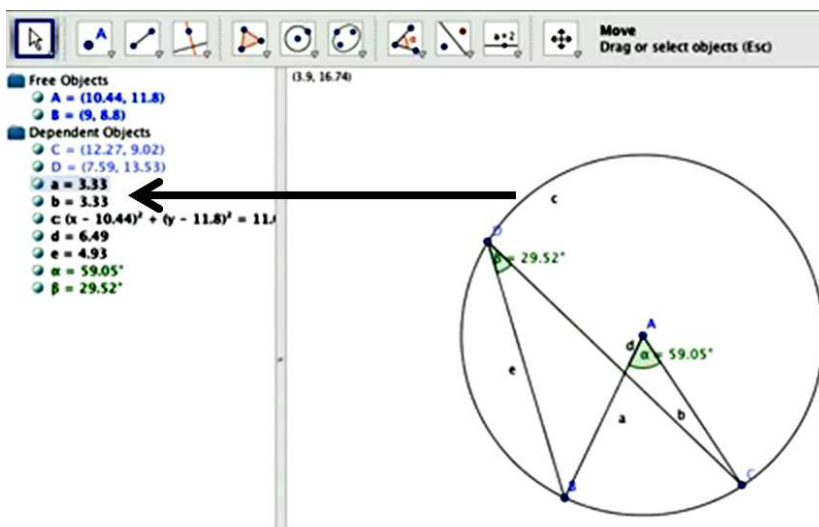
Pomoću njih je moguće prikazati matematičke objekte u tri različita oblika: grafički (na primer, tačke, grafici funkcija), algebarski (na primer, koordinate tačaka, jednačine) i u ćelijama tabele. Pri tome su svi načini prikaza istog objekta dinamički povezani i automatski se prilagođavaju svakoj promeni koja se izvrši u bilo kojem prikazu, nezavisno od načina na koji su objekti nastali.

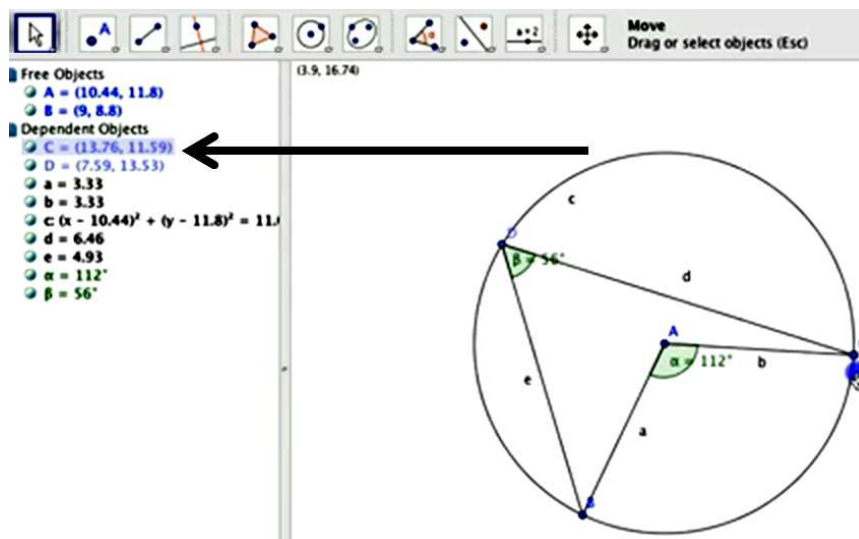


Slika 5 – okruženje programa GeoGebra

Geometrijske konstrukcije se prave u grafičkom prikazu, pomoću miša i alata za konstrukcije koji se nalaze na traci sa alatima. Svi objekti koje napravite u grafičkom prikazu imaju i algebarsku reprezentaciju u algebarskom prikazu.

Bitno je napomenuti da postoji mogućnost da se pomeraju objekte u grafičkom prikazu tako što ćemo ih prevlačiti pomoću miša. U isto vreme, njihova algebarska reprezentacija u algebarskom prikazu se dinamički ažurira.





Slika 5.1. Primer dinamičkog ažuriranja podataka u GeoGebri

Svaka ikona u traci sa alatima predstavlja jednu kutiju sa alatima koja sadrži slične alate za konstrukciju. Kutiju sa alatima se otvara klikom na malu strelicu u donjem desnom uglu njene ikone. Glavna prednost GeoGebra alata je što postoji mogućnost kreiranja interaktivnih Web stranica, tzv. dinamičkih crteža (**Apleta**).

Prilikom snimanja dinamičkog crteža biće napravljeno nekoliko datoteka:

- HTML datoteka (na primer, krug.html) – ova datoteka sadrži Web stranicu
- GGB datoteka (na primer, krug.ggb) – ova datoteka sadrži konstrukciju u GeoGebri
- geogebra.jar (nekoliko datoteka) – ove datoteke sadrže GeoGebru i čine stranicu interaktivnom

Primer *GGB datoteka* unutar HTML datoteke:

```
<applet code="geogebra.GeoGebraApplet"
  archive="geogebra.jar"
  width=200 height=40>
<param name="filename" value="krug.ggb"/>
<param name="type" value="button"/>
<param name="bgcolor" value="#FFFFFF"/>
Please <a href="http://java.sun.com/getjava">instalirajte Javu</a>
da bi videli aplet.
</applet>
```

Prednosti alata GeoGebre su (Markus i Hohenwarter, 2003):

- GeoGebra omogućava animiranje jednog ili više brojeva i/ili uglova istovremeno.
- Pored toga što je objekte moguće prikazati ili sakriti, moguće je podesiti da njihova vidljivost zavisi od određenog uslova. Na primer, može se podesiti objekat tako da se on pojavi na ekranu ukoliko se uključi polje za potvrdu ili se klizač podesi na neku određenu vrednost.
- GeoGebra dozvoljava kreiranje sopstvenih alata zasnovanih na postojećim konstrukcijama. Kada se naprave, alati mogu da se koriste pomoću miša i kao naredbe u *polju za unos*. Svi alati se automatski snimaju u datoteku zajedno sa konstrukcijom.
- **JavaScript interfejs** u GeoGebri može biti zanimljiv korisnicima koji imaju iskustva u radu sa HTML-om.
- GeoGebra apleti imaju JavaScript interfejs koji može da proširi i poboljša njihovu interaktivnost. Na primer, može se napraviti dugme koje će na slučajan način raspoređivati elemente dinamičke konstrukcije.
- Slojevi u GeoGebri služe tome da se odredi koji objekat se bira ili pomera kada korisnik klikne na više objekata.
- Redefinisanje objekata je veoma moćan alat za izmenu konstrukcije. Veoma je bitno obratiti pažnju na to da se može izmeniti i redosled koraka u opisu konstrukcije.
- Objekti mogu da ostave trag u grafičkom prikazu kada se pomeraju.

PRIMERI NASTAVNIH JEDINICA U TOKU KURSA I PRATEĆI APLETI

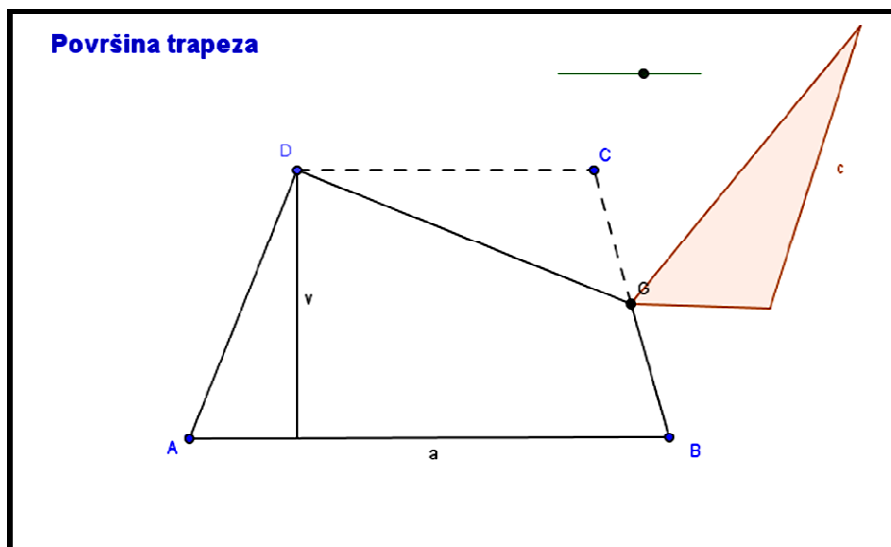
6 razred

Oblast – POVRŠINA TROUGLA I ČETVOROUGLA

Nastavna jedinica – Površina trapeza

Redni broj časa – 124 , mesec maj 2011.

Zadatak – Izračunaj površinu trapeza...



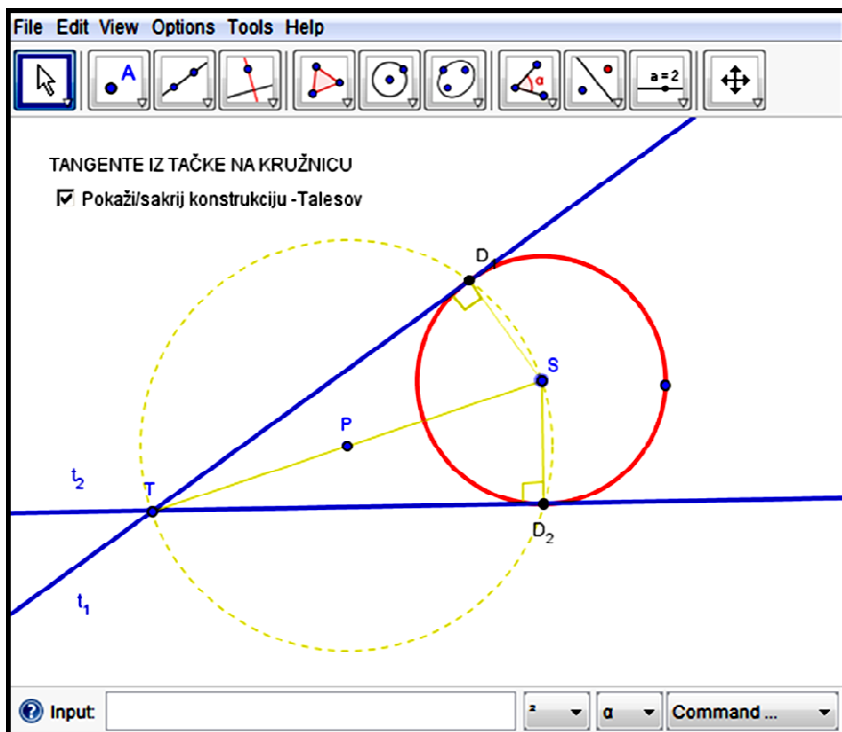
7 razred

Oblast - SLIČNOST

Nastavna jedinica – *Talesova teorema*

Redni broj časa – 125 , mesec maj 2011.

Zadatak – dokaz Talesove teoreme



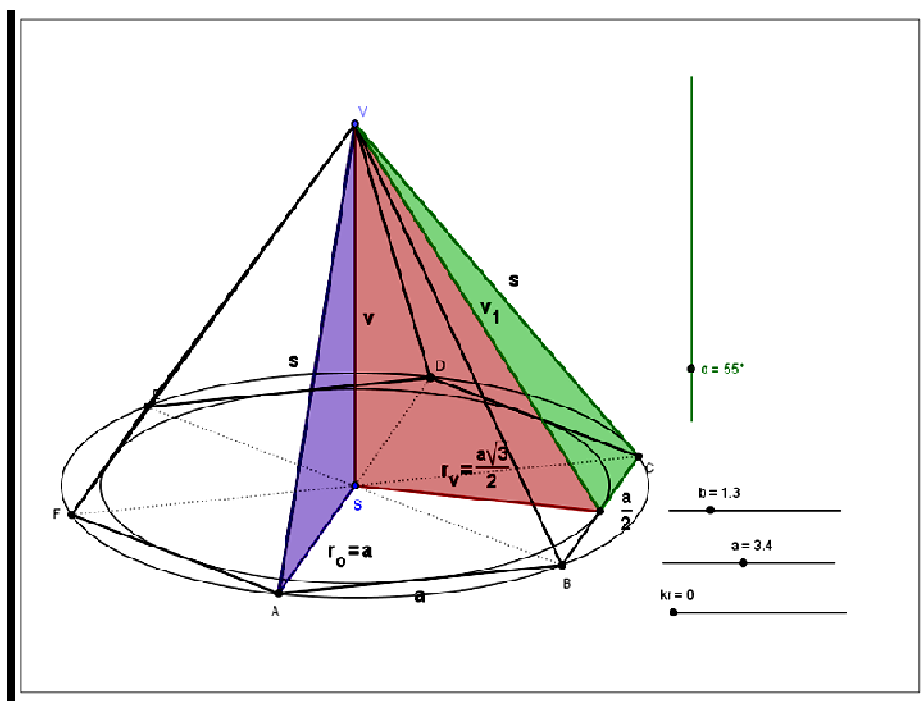
8 razred

Oblast – GEOMETRIJSKA TELA

Nastavna jedinica – **Primena Pitagorine teoreme na prizmu i piramidu**

Redni broj časa – 120 , mesec april 2011.

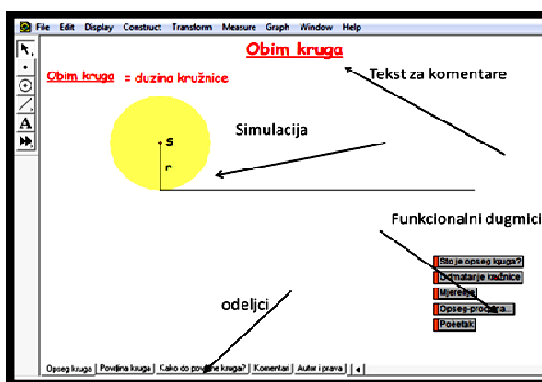
Zadatak – kako se izračunava površina i zapremina piramide



3.4.2 Javasketch i Geometers Sketchpad

JavaSketchpad je program koji omogućava jednostavno kreiranje i korišćenje apleta kao i objavljivanje istih na Internetu. Ovaj alat se može koristiti da bi se distribuirali interaktivni sadržaji *Dinamične Geometrije*. Pošto postoje mogućnosti integracije, napredni korisnici, programeri i matematički istraživači mogu koristiti JavaSketchpad da bi dodali vizuelizacije Dinamične Geometrije drugim programima ili problemima. Alat JavaSketch se temelji na aktivnostima u Sketchpad modulu¹¹. Kada se pogleda stranica koja sadrži JavaSketchPad, mnoge komponente su u interakciji “iza scene” da bi prikazali Dinamičnu Geometriju na implementiranom apletu.

Alat Geometer’s SketchPad omogućava kreiranje apleta koristeći sve alate za crtanje, konstruiranje i oblikovanje. Ovaj način je uobičajeniji za kreiranje apleta u matematici. Sketchpad je otvoreni matematički alat koji je veoma jednostavan za korišćenje i daje učenicima idealne uslove za razvijanje njihovih matematičkih kreativnosti, izražavanja i dostignuća.



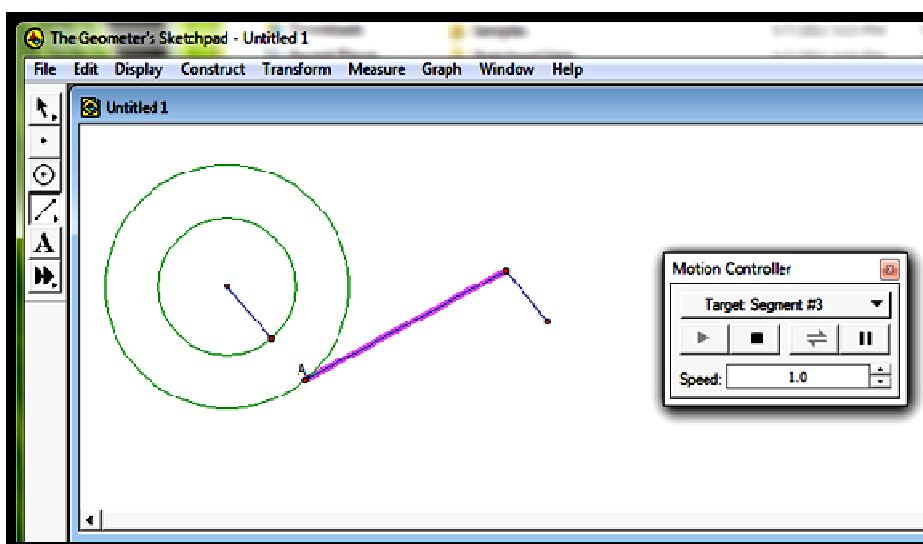
Slika 6. Radno okruženje alata Geometer’s Sketchpad

¹¹ Članak - **Sketchpad pomaže pri učenju matematike** – “Sketchpad je alat namenjen za istraživanje Dinamične Geometrije. Key Curriculum Press (KCP) je, godine 1991. osmislio i zaštitio termin Dynamic Geometry. Naziv opisuje Sketchpadovu sposobnost zadržavanja matematičkih odnosa u geometrijskoj konstrukciji kad korisnici pomiču komponente konstrukcije koristeći miš. Java komponenta Sketchpada omogućuje programu stvaranje HTML Dynamic Geometry stranica koje se mogu pregledavati i menjati putem interneta”. Portal o SketchPadu i matematici - <http://www.proven.hr/>

Okruženje ovog alata je veoma prilagodljivo za korisnike. Nudi mogućnost da se jednostavno urade dinamičke simulacije sa mogućnošću lakog dodavanja komentara (teksta), kao i dodatnih funkcionalnih dugmića koji mogu predstavljati linkove na neke naredne strane odnosno druge aplete. Takođe još jedna prednost je što se može aplet organizovati u vidu slajdova (odeljaka) tako da u ovom slučaju imamo mogućnost da odradimo aplet za više nastavnih jedinica u okviru jedne oblasti, što je veoma korisno za učenike. Ovo omogućava učenicima da imaju više apleta na jednom mestu.

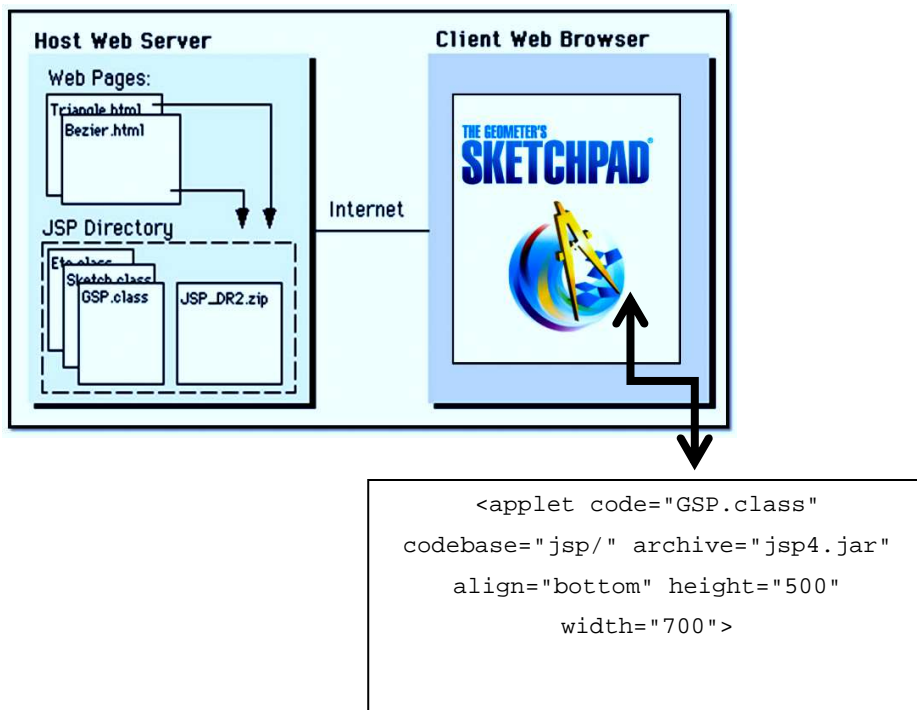
Ovaj alat omogućava učenicima i studentima da vide i razumeju matematičke pojmove i da primenjuju gradivo iz različitih oblasti. Nastavnici mogu koristiti Sketchpad kao dinamičnu ploču da bi prezentovali i ilustrovali moć vizuelizacije matematike.

Ono što može biti korisno učenicima i nastavnicima je da ne veoma jednostavan način mogu kreirati animacije pomoću tzv. *Motion Controller-a*, gde postoje parametri za unos brzine ili za odabir odgovarajuće komponente za koju želimo odraditi pokrete.



Slika 6.1 Jednostavan primer u SketchPad-u i Motion Controller

Postoji veoma jednostavno snimanje u obliku *html* strane, a sledeći dijagram prikazuje tipičnu vezu interakcija potrebnih za stvaranje JavaSketchpad (JSP) apleta u web pretraživaču, kao i način pozivanja apleta u *html* editor.



Slika 6.2 Način realizacije i implementacija apleta u html editoru

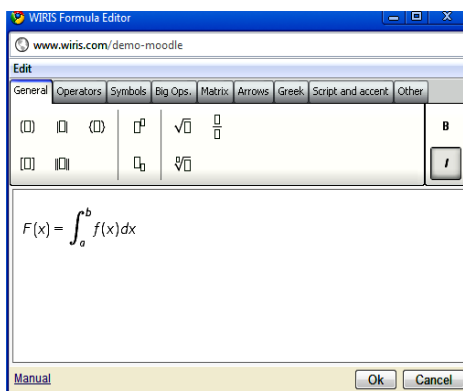
3.4.3 WIRIS, alat za jednostavno kreiranje apleta i testova u matematici

Wiris je alat koji može biti u potpunosti integrisan u *e-learning* okruženje, odnosno može postati deo samog sistema za e-učenje. Ovaj alat se pokazao veoma dobro u okruženjima za e-učenje i ima nekoliko prednosti nad drugim moćnijim alatima:

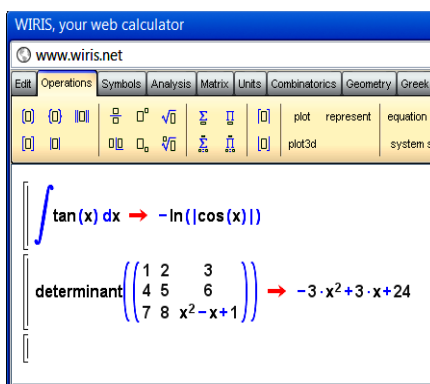
- za njegovu upotrebu dovoljan je web browser,
- nezavistan je od operativnog sistema,
- korisnik ga može koristiti u e-learning okruženju pa ih i doživljava kao deo okruženja,
- veća produktivnost,
- sama aplikacija ne mora obavezno da bude na istom serveru sa e-learning sistemom,
- jednu aplikaciju može da koristi više e-learning sistema.

U poglavlju 6, u kome je naveden primer integracije sa sistemima za e-učenje biće predstavljene i još neke mogućnosti ovog alata u Moodle sistemu. *Wiris* alat poseduje sledeće module:

- Editor ,
- Onlajn okruženje - Web kalkulator ,
- *Wiris* Kviz (mogućnosti ovog alata i primena u ocenjivanju, su detaljnije objašnjene u poglavlju br.4),
- *Wiris* - Bela tabla.

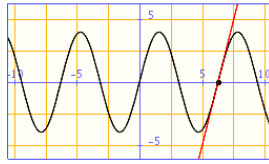


Slika 7. editor *Wiris* alata



Slika 8. *Wiris* u obliku onlajn okruženja

1 Find the expression of the line tangent to $f(x) = 4 \cdot \sin(x)$ at the point of x-coordinate $2 \cdot \pi$.



Answer

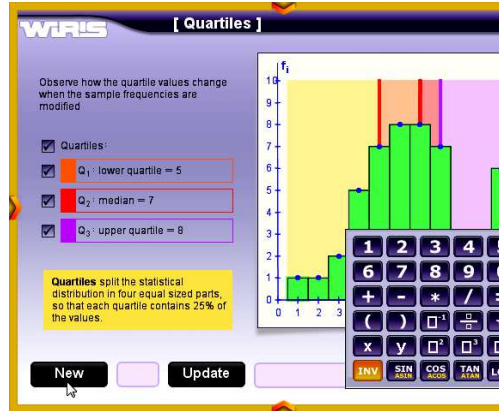
General Operators Symbols Big Ops. Matrix Arrows Greek Script and accent.

(\square) (\square) (\square) \square^{\square} $\sqrt{\square}$ $\frac{\square}{\square}$

(\square) (\square) \square_{\square} $\sqrt[\square]{\square}$ \sum \prod

$2 \cdot \pi + 4 \cdot \cos(0) \cdot x$

Slika 9. Wiris alat za izradu kvizova

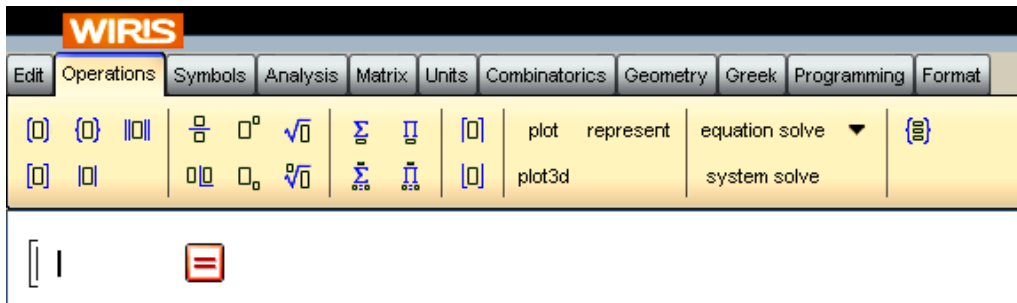


Slika 10. Wiris Bela tabla

Veoma je bitno naglasiti da se Wiris alat može veoma jednostavno integrisati u različite LMS, CMS I HTML editore. Alat obuhvata rad sa sledećim oblastima :

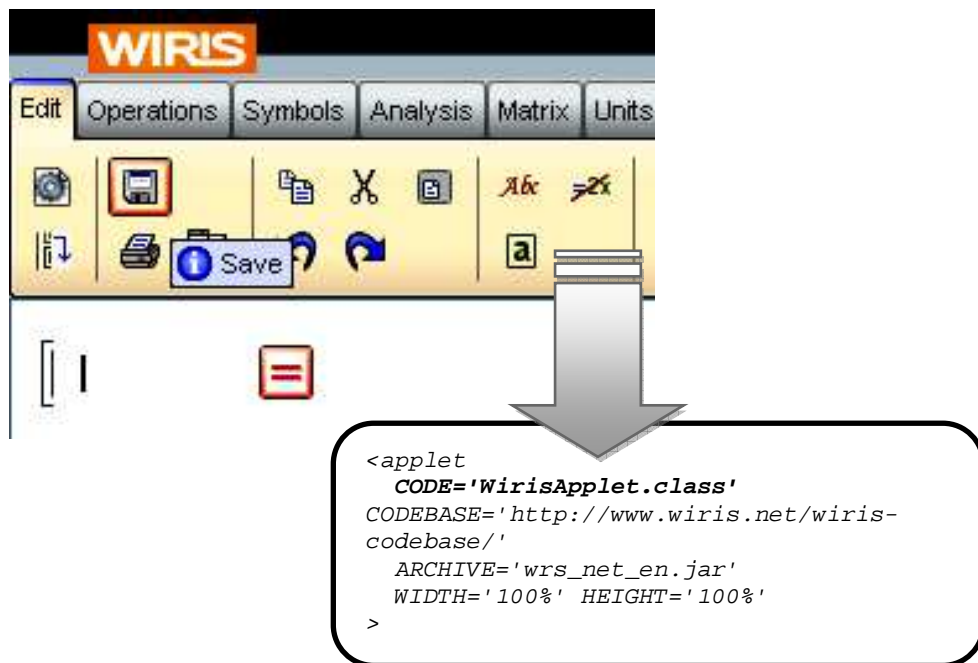
- | | |
|-----------------------------|-----------------------------|
| A. Operacijama i simbolima, | D. Geometrija, |
| B. Matricama, | E. Programiranje , |
| C. Kombinatorika, | F. Matematička Analiza itd. |

Na slici 11 je prikazano okruženje Wiris alata u onlajn verziji. Ovo je velika prednost ovog alata zato što ne zahteva da se kompletan softver preuzme sa interneta već se jednostavno može koristiti kao *Onlajn editor*.



Slika 11 . Glavni odeljci Wiris alata

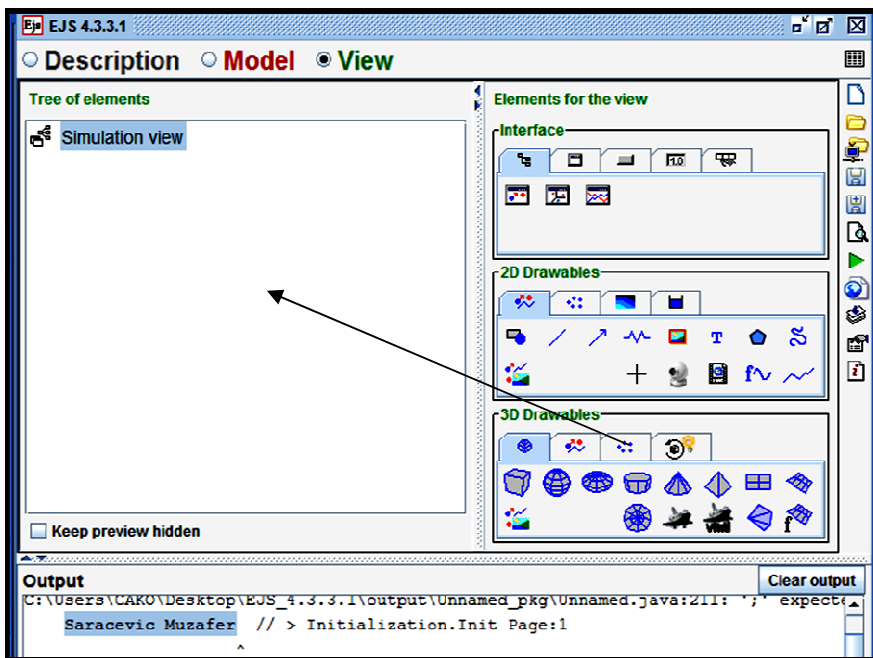
Još jedna prednost ovog alata je jednostavnost snimanja u obliku apleta i kasnija mogućnost da se taj aplet modifikuje. Sledi primer implementacije Wiris apleta u html kodu:



Slika 12. Mogućnost snimanja Wiris sadržaja

3.4.4 Easy Java Simulations, alat za razvoj interaktivnih simulacija

Easy Java Simulations (dalje u tekstu *EJS*¹²) je softverski alat koji omogućava kreiranje interaktivnih simulacija (apleta). Ovaj alat je koristan naročito za osobe koje imaju osnovno znanje iz programiranja kao i kreiranje simulacija fizičkih i tehničkih fenomena i sistema na brz i lak način, pa je pogodan za korišćenje u inženjerskom obrazovanju (Francisco, 2005).



Slika 13. Korisnički interfejs Easy Java Simulations

Aplikacije kreirane pomoću *EJS* mogu biti samostalne Java aplikacije ili apleti. Izvorni fajlovi *EJS* aplikacija se čuvaju u *.xml* formatu. Grafički korisnički interfejs koji nudi *EJS* je strukturiran tako da se sastoji od tri dela: opisa, modela i vizuelizacije (Popović i Naumović,

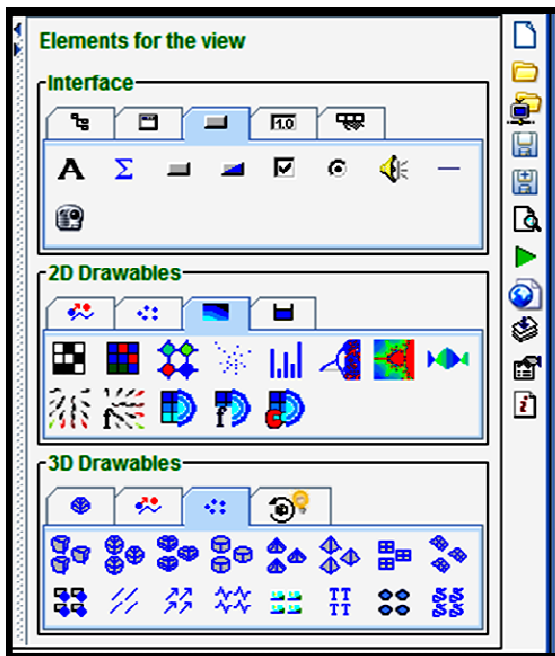
¹² *EJS* je namenjen je nastavnicima, istraživačima i studentima koji imaju osnovno znanje iz programiranja, a ne mogu sebi priuštiti veliki utrošak vremena za kreiranje kompletnog sofisticiranog interaktivnog grafičkog korisničkog interfejsa.

Kako je *EJS* baziran na Java-i, to je moguće simulacije kreirane u njemu koristiti kao nezavisne programe koji rade pod različitim operativnim sistemima, distribuisati ih preko Interneta i pokretati ih kao html stranice iz većine popularnih web čitača (Francisco, 2005).

2010). Opis je deo u kojem se definiše fenomen ili sistem koji se simulira, odnosno u njemu se nalazi teorijski deo neophodan za razumevanje ponašanja procesa ili sistema.

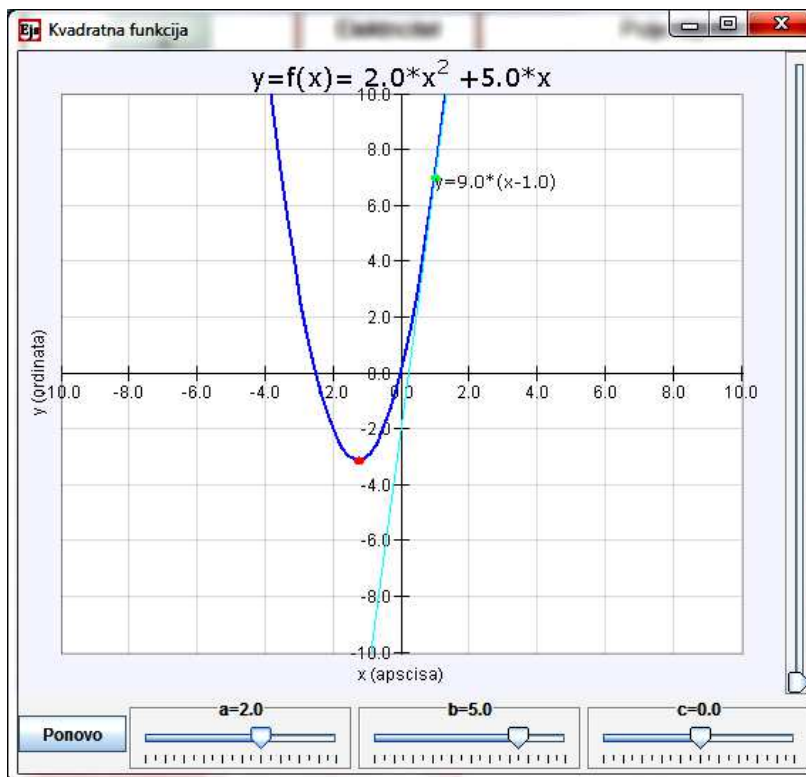
Model je deo koji se odnosi na stranice sa Java kodom, diferencijalnim jednačinama ili na povezivanje sa eksternim aplikacijama (npr. MATLAB, SYSQUAKE).

Vizuelizacija se odnosi na grafički prikaz aplikacije kao i na elemente korisničkog interfejsa potrebnog za interakciju. Model i vizuelizacija su povezani tako da se bilo kakva promena nastala u modelu automatski odražava na vizuelni prikaz i obrnuto.



Slika 13.1 Deo korisničkog interfejsa Easy Java Simulations

MATLAB je softverski paket koji mnoštvom svojih alata omogućava jednostavnu analizu, sintezu, modelovanje i simuliranje sistema (Francisco i Jose, 2005). Za kreiranje simulacija u Matlab-u koristi se *Simulink*, alat namenjen za modelovanje dinamičkih sistema u grafičkom okruženju. Simulacioni modeli koji su kreirani u ovom alatu su prikazani u obliku *blok-dijagrama* koji se sastoji od pojedinačnih modula (integrator, diferencijator, operator sabiranja, operator množenja, pojačanje itd.). Parametri sistema se obično definišu u samim blokovima pre početka simulacije.



Slika 14. Primer apleta u Easy Java Simulations

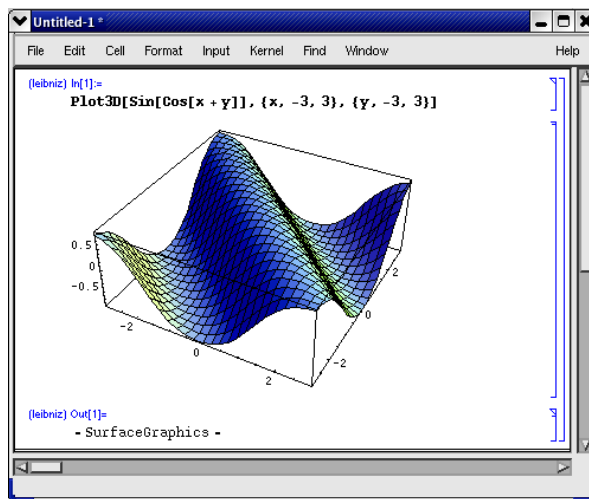
3.4.5 MATHEMATICA, alat za matematičko programiranje

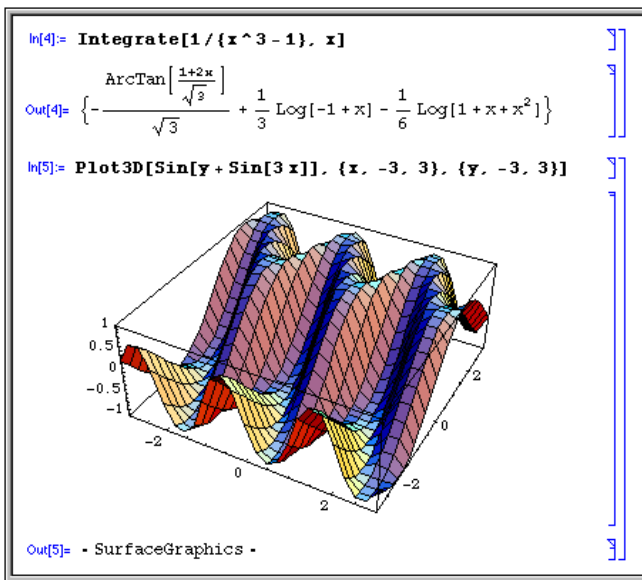
MATHEMATICA je programski paket za matematičke i druge primene. Do sada je ovaj alat upotrebljivan u praktične svrhe, a pomogao je u rešavanju mnogih teorijskih problema. Takođe, *Mathematica* je izučavana i od strane studenata, posebno na Matematičkim fakultetima, primenjuje se na osnovnim studijama i na Tehničkim i Arhitektonskim fakultetima.

Primena ovog alata se rasprostire u svim poljima nauke. Ovaj programski paket ima velike mogućnosti. Alat je posebno pogodan za sledeće primene (Blachman,1992):

- A. obrada numeričkih podataka,
- B. sposobnost simboličkog procesiranja,
- C. sistem za grafičko prikazivanje podataka i funkcija.

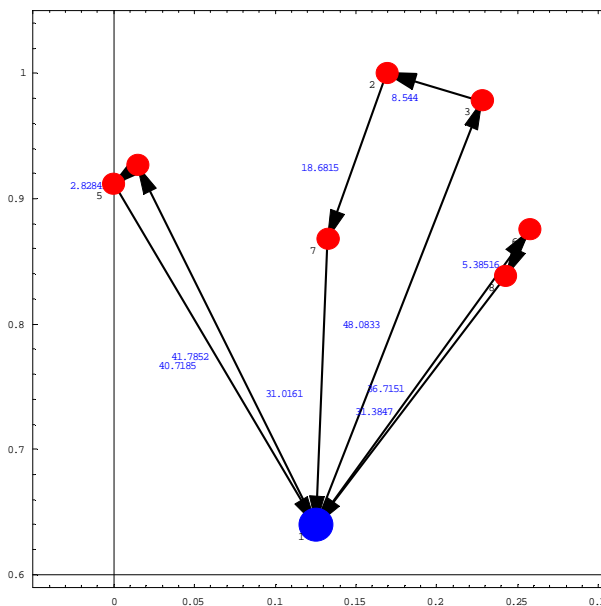
Ključna ideja je bilo otkriće simboličkog programskog jezika koji može da manipuliše vrlo velikim opsegom objekata koji se pojavljuju u tehničkim izračunavanjima. Ona se aktivno koristi i u obrazovanju. Više stotina kurseva, počev od visokoškolskih institucija pa do srednjih škola bazirano je na paketu *Mathematica*.





Slika 15 - rad sa 3D grafikom u alatu Mathematica

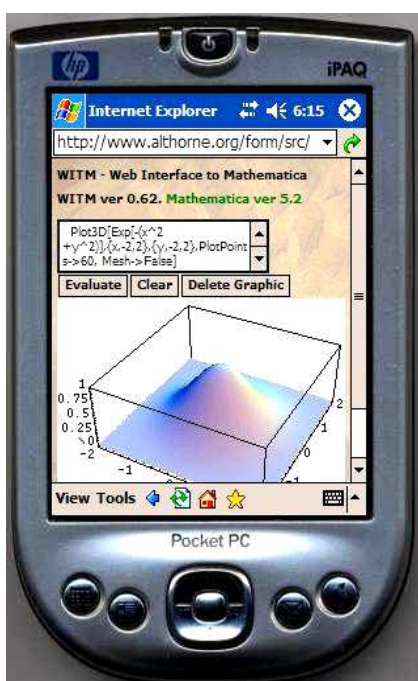
Mathematica se široko koristi kao najvažniji podvig softverskog inženjerstva. Predstavlja jednu od najvećih aplikacija koja je ikada razvijena, i sadrži ogroman niz novih algoritama i važnih tehničkih inovacija. Među mnogim inovacijama je koncept interaktivnog dokumenta, poznat kao *notebook* (Maeder, 1996).



Slika 16. Demonstracija izračunavanja rastojanja tačaka u koordinatnom početku

Mathematica je softver (jezik i paket) za simboličko i numeričko rešavanje problema iz svih poznatih oblasti matematike, fizike i drugih oblasti nauke, tehnologije, finansija, medicine, istraživanja, obrazovanja, itd. Namenjena je kako korisnicima (đacima, studentima, inženjerima) za rešavanje već poznatih, proučenih problema, tako i istraživačima koji je mogu upotrebiti za najkomplicovanije proračune i analize (Gray i Glynn, 1991).

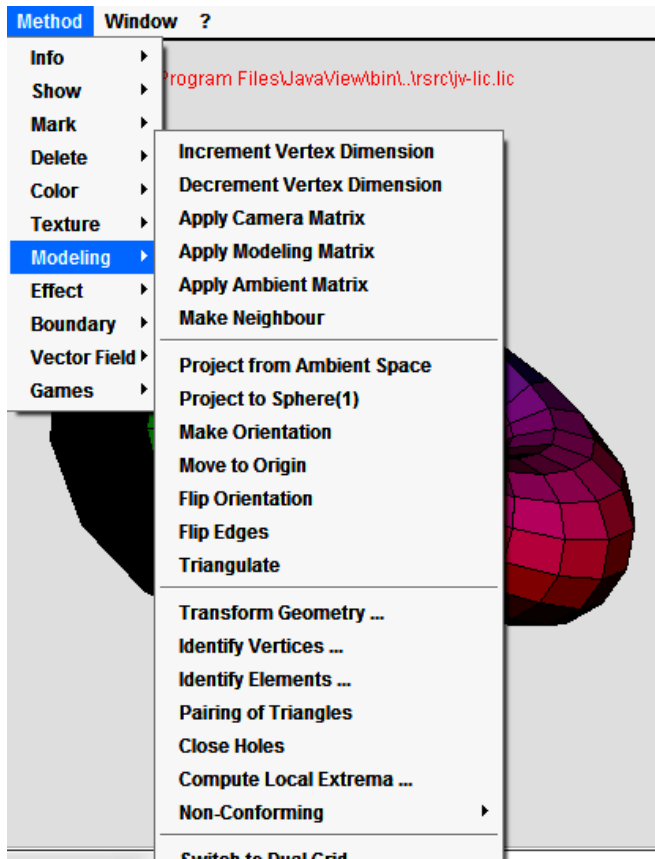
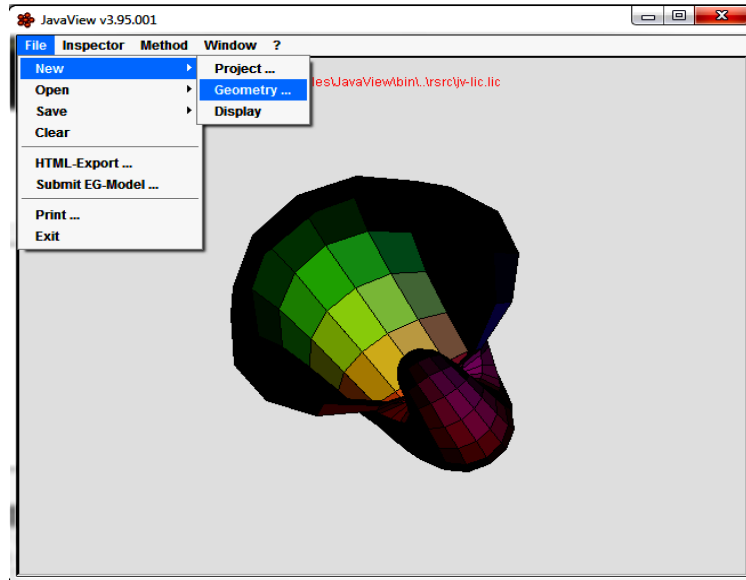
Prednost ovog alata je što se krajnji produkt može izvesti u obliku Java apleta. U radu (Mašović, Saračević i ostali, 2011, *Quality*) se može pročitati da java ima velike mogućnosti kada su u pitanju web bazirani sistemi tako da svaki alat koji ima mogućnost da izveze proizvod u vidu apleta da postoji mogućnost primene istih na različitim platformama.



Slika 17. Implementacija Mathematice na smart telefone

3.4.6 Integracija JAVAVIEW alata sa Notebook (Wolfram Mathematica)

Alat *JavaView* se uglavnom koristi kao integracija sa programskim paketom *Mathematica*. Izuzetne grafičke mogućnosti alata *Mathematica* se mogu iskoristiti za kreiranje zanimljivih apleta koji se mogu koristiti za potrebe nastave prirodnih nauka ili često svoju primenu nalazi u arhitekturi ili računarskoj grafici (Wolfram, 1996).



Slika 18. Primer apleta i njegovih podešavanja u alatu Java View

JavaView se može integrisati u *Mathematica Notebook*, kako bi krajnji proizvod bio Java aplet. Naveden je primer instalacije odnosno pozivanja *JavaView* okruženja u *Mathematica* kodu:

```
<<JavaView`JLink`
InstallJavaView[];
cube = Graphics3D[Cuboid[{0, 0, 0}, {5, 2, 1}]];
JavaView[cube];
```

JavaView pored *Mathematice* se može integrisati i sa *Maple Powertool* i alatom *Polymake*. Sledi primer implementacije apleta koji je kreiran u *JavaView* alatu, u *HTML* editoru:

```
<applet
height="512" archive="jars/javaview.jar"
width="640" code="javaview.class">
<param name=model value="muzafer.jvx">
<param name=displayFile value="muzafer.jvd">
<param name=panel value="material">
</applet>
```




Takođe bitno je napomenuti da se apleti u *JavaView*-u mogu jednostavno implementirati u sisteme za e-učenje (resurse, forume, testove i td.). Detaljnije o ovim mogućnostima u poglavlju 6, u kome su navedene mogućnosti implementacije Java apleta u sistemima za e-učenje.



Slika 19. Primer integrisanja Java View okruženja u funkcionalnost Forum sistema za e-učenje

Naveden je i primer složenog Java apleta koji je rađen u alatu *Mathematica* i njegova implementacija u *HTML kodu*.

Postavke:

Pozadina: #ffffff   

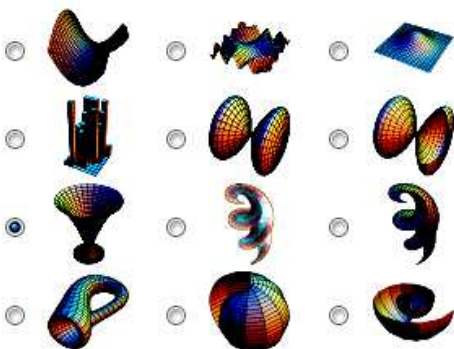
Početni kut rotacije (X):

Početni kut rotacije (Y):

Rotacija (X):

Rotacija (Y):

Povećanje:



Logotip Mathematice:

Oprez: Drugi i treći objekt su procesorski iznimno zahtjevni; preporučuje se pokretanje samo na brzim računalima!



HTML kod:

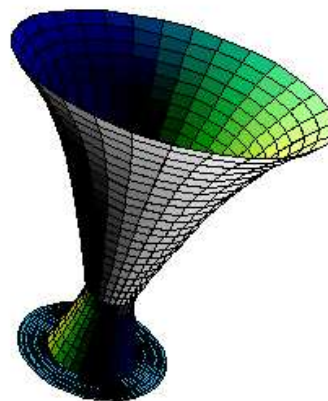
```
<applet archive="live.jar" code="Live.cl"
<param name="BGCOLOR" value="#ffffff" />
<param name="MAGNIFICATION" value="1.2"
<param name="INITIAL_ROTATION" value="0,
<param name="SPIN_X" value="17" />
<param name="SPIN_Y" value="0" />
<param name="INPUT_FILE" value="glass.m"
<param name="INPUT_ARCHIVE" value="glass
```

BB kod (za *math.e forum*):

```
[live=BGCOLOR:#ffffff;MAGNIFICATION:1.2;
```

Datoteka sa slikom: [.m](#) ili [.zip](#).

Applet:



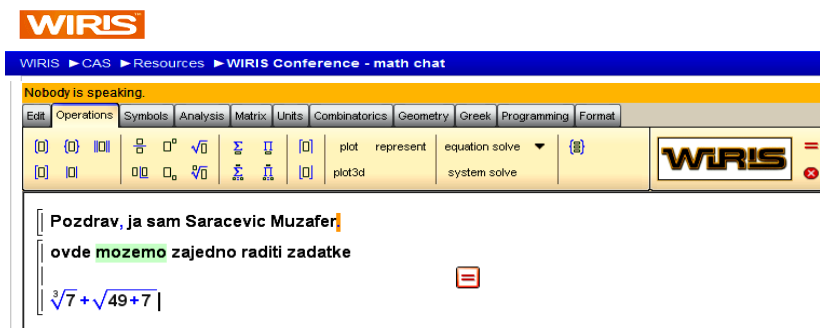
Slika 20. Rad sa 3d grafikom u Mathematici i izvoz u Applet

3.5 Kolaborativno kreiranje elektronskih sadržaja pomoću apleta

Kolaborativnost je jedna od osnovnih pretpostavki nove paradigme u razvoju Web aplikacija, koja se često označava kao Web 2.0¹³. Neka kolaborativna okruženja namenjena su pre svega u radu sa binarnim fajlovima dok su druga više okrenuta tekstualnim sadržajima (Wiki). U svakom slučaju, ova okruženja moraju da omoguće jednostavno dodavanje i promenu fajlova. Sadržaj mora biti tako organizovan i skladišten da ga je moguće jednostavno pronaći. Zato ovi sistemi uglavnom imaju napredne podsisteme za pretragu podataka. Najpoznatiji predstavnici ovih sistema su MediaWiki, Twiki, Plone i Alfresco.

3.5.1 WIRIS Conference - math chat

WIRIS alat je izuzetno koristan za kolaborativan rad u vidu chat-a ili konferencije ali sa nekim dodatnim nespecifičnim mogućnostima. Krajnji proizvod se može sačuvati u vidu Java apleta koji se kasnije može jednostavno implementirati na različitim platformama. *Zamislite da se nalazite u učionici, u toku je čas matematike i učenici rade zadatak.* U toku ove diskusije učenici timski rešavaju zadatak, gde mogu zajedno navoditi svoje ideje u vidu matematičkih formula ili geometrijskih figura i međusobno komunicirati. Zaista ovaj alat pruža dobru korisničku podršku kolaborativnom radu u e-učenju. Postoji i dodatak za implementaciju u sistemima za e-učenje.



Slika 21. Okruženje WIRIS Conference - math chat

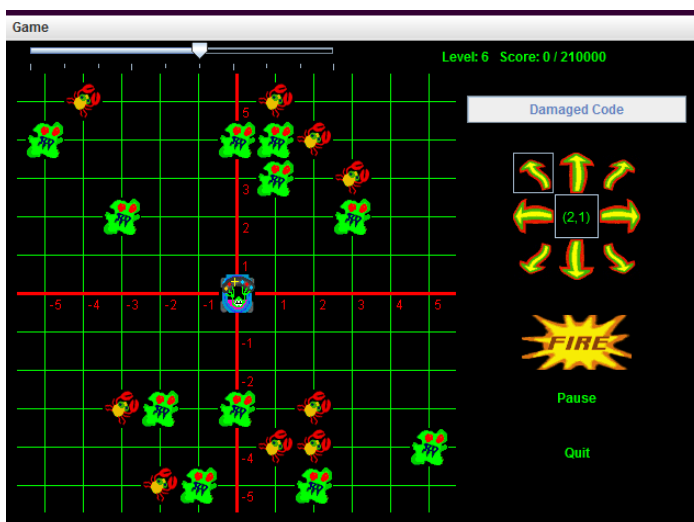
¹³ **Web 2.0** je novi način dizajniranja i projektovanja Web aplikacija, a neki od osnovnih principa su korišćenje Weba kao platforme, korišćenje podataka koje unosi veliki broj ljudi, i implementacija aplikacija pomoću u jednostavnih programskih modela. Opisuje se kao „read-write web“ i nudi sve servise i aplikacije koje omogućuju pojedincima da kreiraju nove sadržaje, dopunjavaju postojeće, međusobno saraduju i razmenjuju sadržaje sa ostalima. Podržava korisnički generisan sadržaj, gde su autori obični korisnici, a ne specijalizovani autori ili eksperti, koristeći raspoložive tehnologije kao što su blogovi, podcast, wiki i drugi. Podstiče društveni aspekt weba korišćenjem socijalnih mreža

(<http://e-lab.tfc.kg.ac.rs/moodle/mod/book/view.php?id=5088>, materijal za predmet – Alati i tehnologije u e-učenju).

3.6 Primena Java obrazovnih igara u nastavi

Pojmove simulacija i igra potrebno je šire shvatati i ne vezivati ih samo za njihovu realizaciju uz pomoć kompjutera. Sadržaj simulacije je simuliranje neke igre, ali igra nije samo vid zabave ili nešto što služi za rekreaciju, za igru, već može biti i oblik nastavne tehnike, metoda kojom se nešto vizuelno prikazuje i uči i na taj način se apstraktno pretvara u konkretno, očigledno, jasnije i lakše razumljivo a time i lakše za učenje.

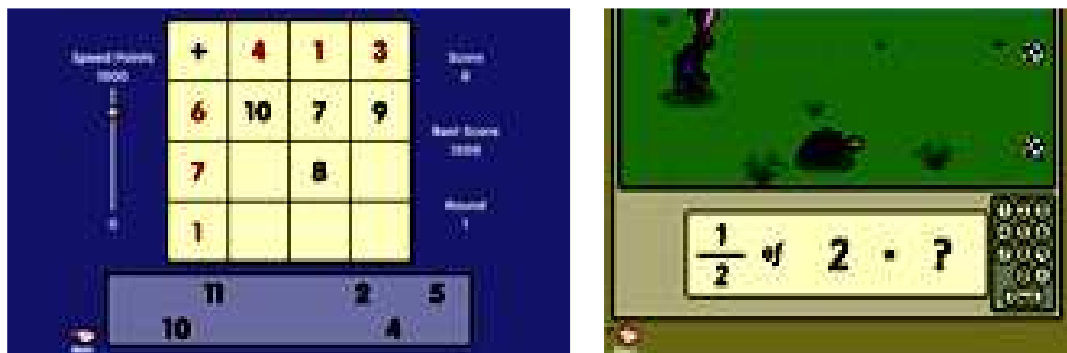
U toku simuliranja učenik se aktivno uključuje pri donošenju odluka, usvajanju stavova, razumevanju pravila, igranju uloga. Učenik tada uči manipulisanjem modelom koji može biti, na primer, predmet, proces, neki složeni sistem.



Slika 22. Primer obrazovne igre za nastavu matematike u obliku apleta

Mnogi pedagozi i psiholozi bave se proučavanjem efekata elektronskih igara na razvoj dece, ali da bi svaka igra našla svoje odgovarajuće mesto, trebalo bi izvršiti njihovu kategorizaciju i klasifikaciju. Shodno tome, može se govoriti o obrazovnim kompjuterskim igrama i kompjuterskim igrama namenjenim zabavi. Danas je sve više pedagoga, psihologa i IKT stručnjaka koji dokazuju da kompjuterske igre mogu podstaći kognitivni razvoj i naći svoju primenu u nastavi (Moreno i Mayer, 2007).

Java je poznata u svetu kompjuterskih igara, pa se ovaj programski jezik može smatrati bitnim faktorom u oblasti obrazovnih igara. Prvenstveno kroz java aplete, jer se oni mogu lako implementirati na web stranice nezavisno od platforme.



Slika 23. Primeri edukativnih java igrica u vidu apleta namenjenih za nastavu matematike

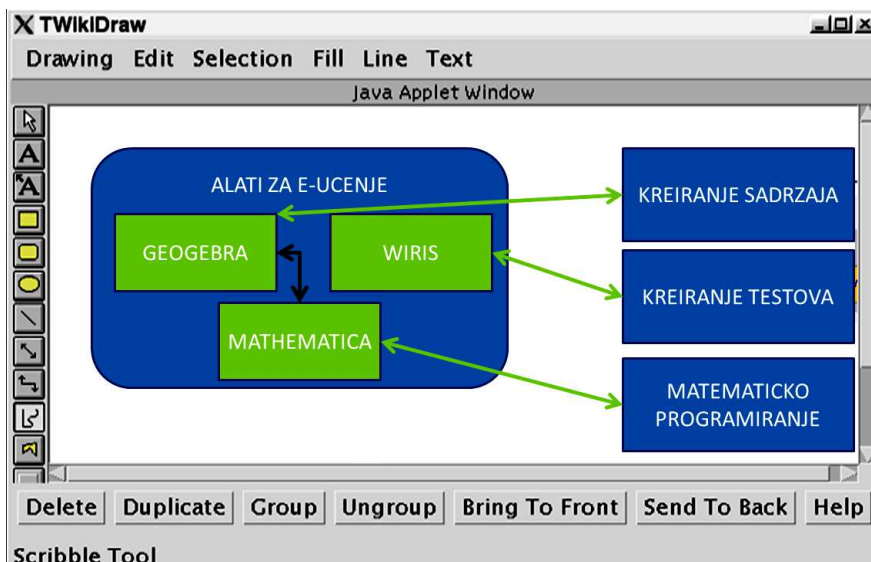
3.7 Mape uma i *brejnstorming* alati kao Java apleti

Mape uma jasno definišu ključne pojmove, odvajajući bitno od nebitnog, čime nam omogućavaju da se fokusiramo na bitne stvari i da ne gubimo vreme na nebitne detalje. Takođe predstavljaju način da se puno informacija stavi na jedan list papira, da se na dobar način organizuju beleške. Veoma dobar način da se uči, pa se ovaj način može primeniti i u nastavi matematike.

Brejnstorming alati (eng. BrainStrom Tools) su namenjeni kreiranju mapa uma za aktivnosti ili događaje, kao i za brzo hvatanje beležaka i ideja na sastancima ili predavanjima. Pogodne su i za kolaborativan rad. Mape uma su sredstvo za kreativnu organizaciju misli koje nam pomažu da bolje iskoristimo neograničenu moć mozga.

Pomoću mapa uma u matematici, koristeći slova, brojeve, boje, linije, crteže, sa lakoćom postizemo izuzetne rezultate u učenju, kreativnom razmišljanju, pripremanju predavanja i prezentacija, hvatanju i pravljenju beležaka, planiranju izrade zadataka i razvoj toka podataka, donošenju odluka, rešavanju problema, i naročito - u pamćenju i učenju na svim nivoima.

Alat *TWikiDraw* je izuzetno koristan za kolaborativan rad i rad sa mapama uma koji se mogu sačuvati u obliku apleta, pa se zato i često koristi zbog lakoće implementacije na webu ili u sistemima za e-učenje (više o mogućnostima implementacije apleta u Moodle sistemu za e-učenje u poglavlju 6).



Slika 24. Okruženje TWikiDraw

4. ALATI ZA OCENJIVANJE U NASTAVI MATEMATIKE

Postupak nastavnikovog ocenjivanja učenika je proces davanja mišljenja učeniku o tome koliko je ostvario postavljenje kriterijume. Pošto izveštaj treba da obuhvati precizne informacije o znanju, napredovanju i da savetuje učenika kako da nastavi učenje, ako se radi o e-učenju ili hibridnom načinu veoma je bitno odabrati i adekvatne alate koji to mogu postići.

Pre svega, nastavnik je neposredno odgovoran i nosilac je procesa školskog i razrednog ocenjivanja. Stoga je neophodno razvijati odgovarajuće veštine proveravanja, ocenjivanja i vrednovanja postignuća i razvoja učenika (Healy, Berger i ostali, 2004). Kompetencije iz područja ocenjivanja su važne profesionalne kompetencije nastavnika koje su osnova nastavnikovih uloga procenitelja, evaluatora itd.

4.1 Metode, postupci provere i ocenjivanje u matematici

Pri vrednovanju (procenjivanju) kvaliteta i nivoa znanja u nastavi matematike ocenom se iskazuje trajnost, korektnost, svesnost i upotrebljivost znanja (njegova funkcionalnost) na raznim stupnjevima, odnosno nivoima¹⁴.

¹⁴ U pogledu kvaliteta postoji nekoliko nivoa znanja (Miroslav B. Mladenović, Ocenjivanje učenika u nastavi matematike):

1. **Nivo prepoznavanja** kao najniži nivo, kad učenik nije u stanju da samostalno iskaže traženi podatak ili pravilo, ali ga se može setiti uz izvesnu pomoć nastavnika, ili ga može prepoznati u nizu ponuđenih odgovora.
2. **Nivo reprodukcije kao nivo sa malo kvalitetnim ali neophodnim znanjem.** To je nivo kada učenik može samostalno da reprodukuje naučeni sadržaj u pogledu poznavanja činjenica, termina, pravila, klasifikacija, postupaka itd.
3. **Nivo razumevanja** je nivo sa kvalitetnijim znanjem u odnosu na predhodne nivoe, kada učenik stvarno shvata i razume naučeni sadržaj i u stanju je da ga logički obrazloži. Gradivo izlaže i s razumevanjem (učenik je u stanju ne samo da prepozna i reprodukuje naučeno, već da vrši i misaonu preradu znanja – da razume i objasni činjenice, pojmove, pravila, definicije, da izdvoji bitno od nebitnog, povezuje činjenice i izvodi zaključke). Učenik koji je naučio gradivo na ovom nivou može verbalno iskazati zadatak da „prevede“ na matematički jezik (jezik simbola), i obrnuto, sa više apstraktnog (matematičkog) jezika može da „prevede“ na manje apstraktan (konkretniji, običan) jezik.
4. **Nivo primene** gde se podrazumeva vrlo kvalitetno znanje, kada je učenik u stanju da naučene sadržaje (pravila, algoritme, teoreme, metode) smostalno primenjuje u rešavanju raznih teorijskih ili praktičnih zadataka, sličnih onima koji su već rešavani. Učenik ume stečeno znanje da primenjuje pri učenju novog gradiva, u životu i praksi. U ovoj fazi značaju ulogu zauzimaju apleti i interaktivne simulacije. Konkretni primeri primene se baš mogu navesti u nastavi informatike ili matematike ali u većoj meri apleti se primenjuju i u nastavi tehničkog obrazovanja (Pavlović i ostali, 2010).

U radu (Bjelanović, 2005) se navode konkretni predlozi učenja istraživanjem posredstvom Java apleta prema modelu Georgea Polya. Takođe rad (Polya, 2003) pomenutog autora iznosi prednosti savremenih metoda u nastavi matematike i novih otkrića u ovoj oblasti.

4.2 Alati za ocenjivanje bazirani na Javi

Softverski alati mogu da analiziraju aktivnosti i pomognu nastavniku da pruži povratnu informaciju na osnovu ispisane komunikacije koja se nalazi na kursu. Novi softverski alati mogu biti korišćeni od strane instruktora a mogu ih koristiti učenici i za samoocenjivanje.

Ovakvi alati u nastavi matematike imaju široku primenu naročito kod kreiranja testova, sa nizom pitanja ili zadataka gde nastavnik želi proveriti koliko učenik zaista razume gradivo koje je rađeno na času. Veoma je bitno da se učenicima omoguće i tzv. *samoevolutivni testovi* gde će svaki učenik imati mogućnost da proveri koliko je naučio odnosno da li je potrebno detaljnije da pogleda pređeno gradivo.

Navedeni su najčešće korišćeni alati za ocenjivanje u e-učenju koji se baziraju na Java programskom jeziku (JavaScript, JavaApplets).

Tabela 2: Alati za ocenjivanje u e-učenju

Quiz Maker	ActivePresenter	QAEWT QuizMaker
Java Web Start	Whizlabs SCJP Exam Simulator	iQuiz Maker
Java DB	QuizFaber	Multiple Choice Quiz Maker
Milon Quiz Program	HotPotatoes	Quiz Builder
Greenfoot	Articulate Quizmaker	Multi-choice quiz gen.
Java Access Bridge	PollDaddy	Java 3D

Svrha ovakvih alata je omogućiti kreiranje interaktivnih vežbi koje se mogu izvoditi u raspoloživom web pregledniku. Bitno je da vežbe koriste HTML i **JavaScript** kako bi se implementirala interaktivnost. Korisnik ne mora imati znanja iz područja programiranja da bi koristio ovakve alate (Materijal : E-Lab TFC , L4).

-
5. **Nivo kreativnosti** ili stvaralačkog rešavanja problema se odnosi na najkvalitetnije znanje, kada je učenik (saglasno svom uzrastu) u stanju da stečeno znanje i poznate metode primenjuje u sasvim novim situacijama, da samostalno izdvaja bitne ideje i činjenice i pronalazi odgovarajuće postupke za rešavanje pojedinih problema, da stvaralački i samostalno reorganizuje gradivo koje izlaže, kritički analizira i procenjuje iznete tvrdnje ili teorije.

Sve što potrebno odraditi je da se unesu potrebni podaci, odnosno kreirati pitanja, odgovore, izveštaje i sl. Ovi alati na zahtev, samostalno kreira web stranicu koja se može postaviti na web server ili snimiti na računaru i po želji izvoditi bez pristupa na Internet.

Uglavnom, zajedničko za pomenute alate za ocenjivanje u e-učenju je da koriste JavaScript, a Java apleti se lako povezuju sa *JavaScript* naredbama, tako da Java apleti takođe nalaze primenu i mogu se iskoristiti u kreiranju testova za ocenjivanje.

Tabela 3: Primer upotrebe JavaScript naredbe i implementacije u Java apletima

APLET I JAVASCRIPT	HTML
<pre>import java.applet.*; import java.net.*; public class InJava4 extends Applet{ public void init(){ String msg = "Hello from Java (using javascrip alert)"; try { getAppletContext().showDocument (new URL("javascrip:doAlert(\"" + msg + "\"")); } catch (MalformedURLException me) { } } }</pre>	<pre><HTML><HEAD></HEAD><BODY> <SCRIPT> function doAlert(s) { alert(s); } </SCRIPT> <APPLET CODE="InJava4.class" NAME="myApplet" MAYSCRIPT HEIGHT=10 WIDTH=10> </APPLET> </BODY> </HTML></pre>

Ceo *JavaScript* kôd, koji se obično stavlja u zaglavlje vežbe (koji sadrži pitanja i odgovore), će pri izradi web stranice biti izdvojen u zasebnu datoteku, s istim imenom, ali nastavkom *.js* umesto *.htm*. Obe datoteke su neophodne da bi test radio, i obe moraju biti smeštene u istom folderu (sličan princip kao kod Java apleta). Ovo može biti korisno ako želimo učenicima malo otežati da pogledaju u izvorni *kôd* i pročitaju odgovore i povratne informacije dok rešavaju vežbu.

Kreiranje kvizova u alatima koji se baziraju na JavaScript jeziku, je vrlo brzo i kreativno jer uglavnom pomenuti alati poseduju korisnički interfejs koji je razumljiv i prilagođen korisniku. Zahvaljujući korisničkom interfejsu nekih alata za ocenjivanje, mogu se primeniti sledeće napredne karakteristike u nastavi (Materijal : E-Lab TFC, L4):

- Brzo grupisanje i nasumičan izbor pitanja iz grupe pitanja,
- Davanje specifičnih rezultata korisnicima kviza i davanje povratnih informacija na osnovu njihovih rezultata (veoma bitno u odsustvu nastavnika),

- Uključivanje slika, *Flash*-eva i audio zapisa i scenarija u okviru kviza,
- Grananje kviza koje vodi korisnika kroz pitanja u zavisnosti od odgovora (veoma bitan segment za pojedince, odnosno dostiže se veći nivo nastave kada je u pitanju individualizacija, odnosno prilagođenost testa učeniku u skladu sa njegovim sposobnostima), animiranje objekata i njihovo vremensko podešavanje itd.

4.3 Primeri upotrebe alata za ocenjivanje u nastavi matematike

Testovi znanja (testovi postignuća ili provere znanja) predstavljaju najobjektivnije sredstvo za “merenje” znanja. Njima se utvrđuje koliko je znanja stekao neki pojedinac tokom obavljanja određene aktivnosti (predavanja, vežbi, prakse) ili određenog perioda učenja.

Zadaci u testovima znanja prema obliku u kom se daju učenicima dele se na **testove reprodukcije** (prisećanja) i **testove rekognicije** (prepoznavanja)¹⁵. U svakoj od ovih grupa postoji nekoliko oblika zadataka koji se upotrebljavaju u testovima znanja, a obe grupe se mogu primeniti u ocenjivanju u nastavi matematike, prvenstveno se misli na proveru znanja kada je u pitanju teorijsko znanja učenika iz ovog predmeta (Kurnik, 2000, *Matematički zadatak*).

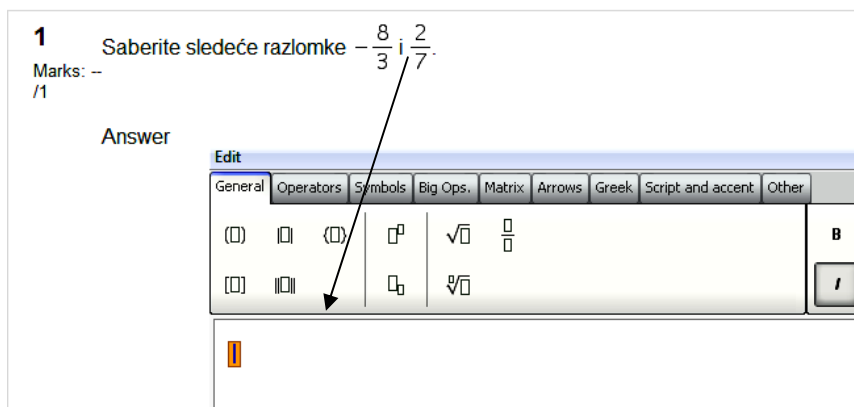
Na osnovu sopstvenog iskustva smatram da je korisnije upotrebiti testove reprodukcije (testove otvorenog tipa) u kontekstu e-procena znanja, jer e-učenik sam traži, reprodukuje i odgovara na postavljeno pitanje. Mnogo je realnije upotrebiti ovakav tip zadataka (kada je reč o e-učenju) jer i rezultat i nivo postignuća je realniji ako postoji više zadataka ovakvog tipa.

Osnovni tipovi ovakvog oblika znanja su:

1. esejski testovi ,
2. testovi dopunjavanja ili prisećanja.

Slede primeri kako se može alat **WIRIS QUIZ** primeniti u kreiranju testova za ocenjivanje u e-učenju. Učenici 6, 7 i 8 razreda su imali prilike da rade prikazane testove u toku razvijenog kursa za potrebe istraživanja.

¹⁵ U testovima reprodukcije od učenika se traži da sami nađu, reprodukuju odgovor na postavljeno pitanje. U testovima rekognicije ili prepoznavanja učenik treba od ponuđenih odgovora da odabere, da prepozna onaj koji je tačan (<http://www.cet.rs/cetcitaliste/CitalisteTekstovi/429.pdf>)

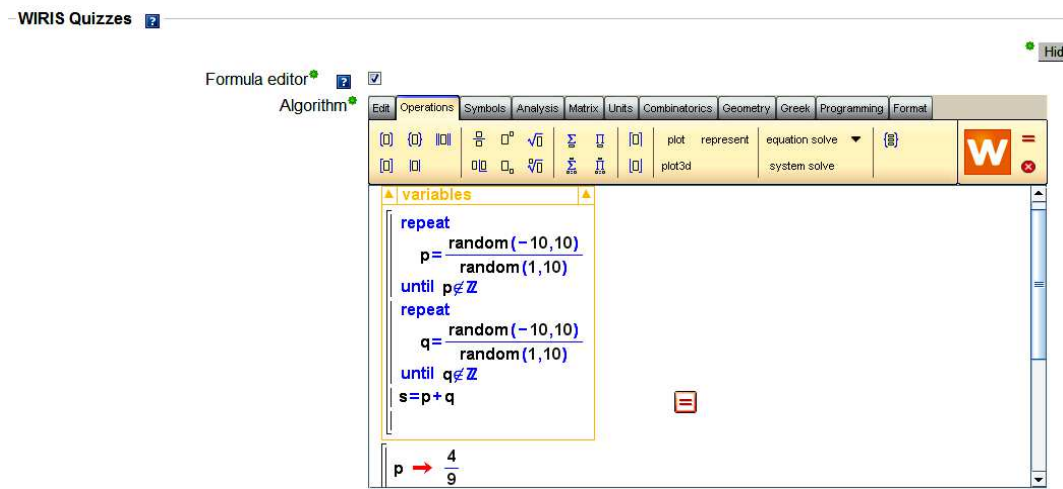


Slika 25. Primena Wiris alata u otvorenom tipu pitanja – esej sa kratkim odgovorom

Nivo ozbiljnosti u evaluaciji se može povećati i na taj način da se omogući automatsko menjanje pitanja, odnosno da uvedemo promenljive u postavljenom zadatku (brojevi, operacije, polinome, date vrednosti ulaznih parametara i sl.).

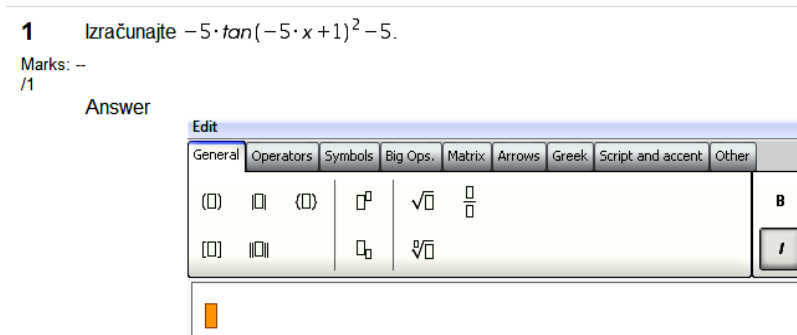
U prethodnom primeru koriste se promenljive, odnosno svakom se učeniku prilikom otvaranja testa nasumično menjaju vrednosti odnosno svako dobija različite razlomke u zadatku. Vrednost parametara za koji želimo da se vrednost menjao navodimo na sledeći način *#promenljiva*. To navodimo u tekst editoru u kome navodimo pitanje.

Sledi primer kako nastavnik definiše algoritam za promenljive u prethodnom primeru:

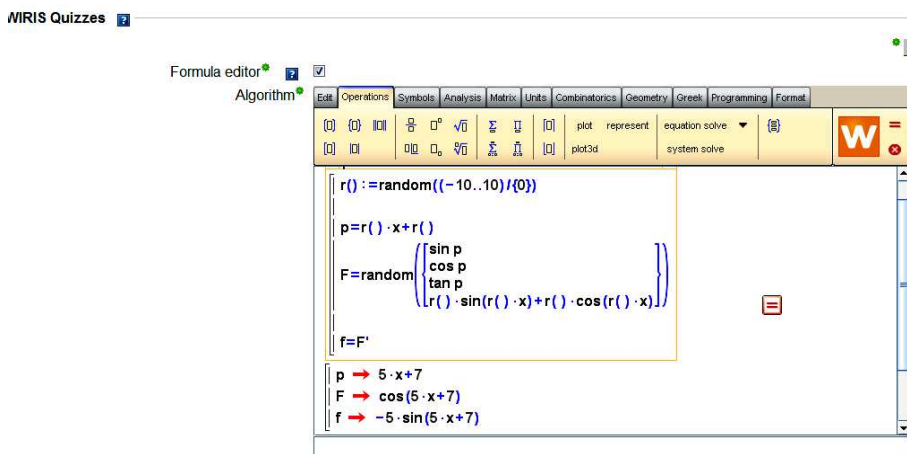


Slika 26. Primer kako nastavnik unosi algoritam za prethodni primer

Naveden je primer otvorenog pitanja sa složenim odgovorom (slika 27) i algoritam za promenljive u zadatku (slika 28):

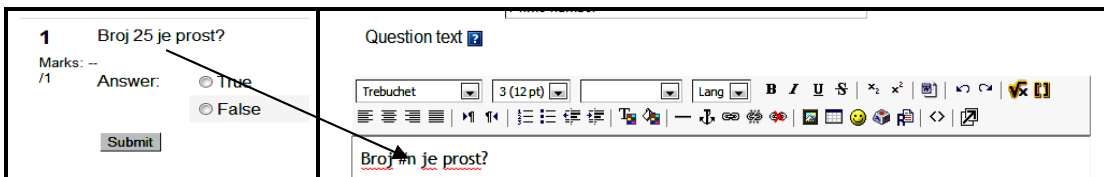


Slika 27. Primena Wiris alata u otvorenom tipu pitanja – esej sa slozenim odgovorom



Slika 28. Primer kako nastavnik algoritam za prethodni primer

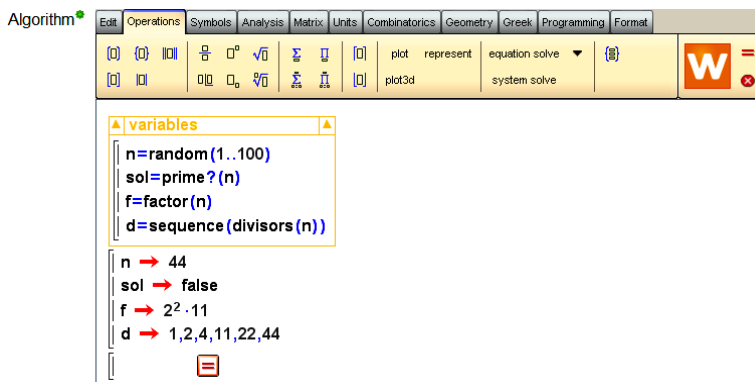
Navedeno je još jedno pitanje sa nasumičnim menjanjem datih parametara:



Slika 29. Primer pitanja

Slika 30. Primer navođenja promenljive u Wiris editoru

U delu za unošenje potrebnog algoritma navodimo naziv promenljive i naredbu *random* (opseg vrednosti):

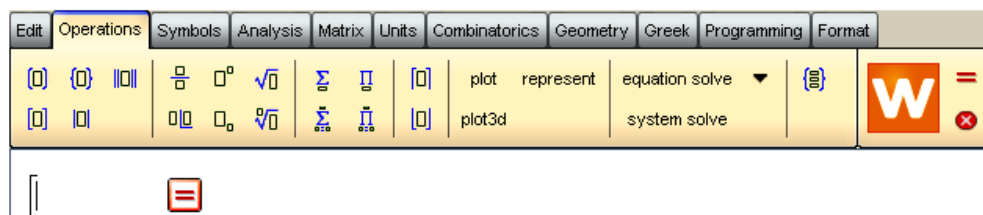


Slika 31. Primer definisanja algoritma za promenljivu

Broj 25 je vrednost koja se uzima prilikom jednog otvaranja testa. U narednom pokušaju se uzima druga proizvoljna vrednost. Na ovaj način nastavnik može obezbediti realnije ocenjivanje učenika, odnosno ima mogućnost da obezbedi da svaki učenik dobije drugačiji zadatak. Posebno je ovo korisno za rad sa razlomcima, za polinome, za algebarske izraze itd.

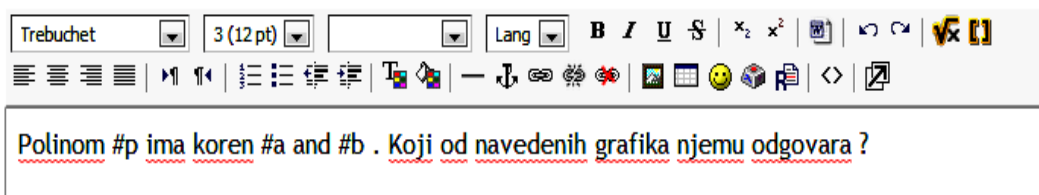
Sledi primer sa višestrukim odgovorom¹⁶ i proces kreiranja ovakvog tipa pitanja u dodatku *Wiris* za Moodle sistem. Pitanje sadrži ponuđene odgovore ali i *WIRIS* editor kako bi korisnik pre konačnog opredeljenja za jedan od ponuđenih odgovora mogao sam da proveri i dođe do rešenja. Pitanje izgleda ovako :

Polinom $x^2 - 1$ ima koren $- 1$ and 1 . Koji od navedenih grafika njemu odgovara ?



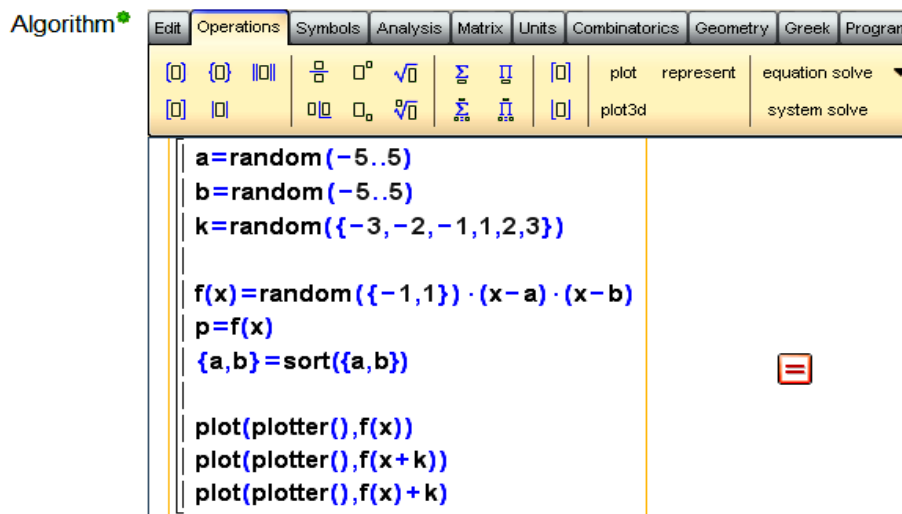
Slika 32. Primer pitanja sa višestrukim odgovorom

¹⁶ Test višestrukog izbora je fleksibilan format procene koji se može koristiti za merenje znanja, postignuća, veština, sposobnosti, veštine mišljenja itd. Takav test se sastoji obično od više stavki koje predstavljaju pitanje na koje ispitanici moraju izabrati ispravan odgovor između nekoliko opcija, ili to mogu biti izjave za koje ispitanici moraju pronaći ispravan završetak. Testovi višestrukog izbora spadaju u grupu testova rekognicije (prepoznavanja) gde ispitanici moraju da identifikuju tačan odgovor.



Slika 33. Primer navođenja pitanja sa više promenljivih

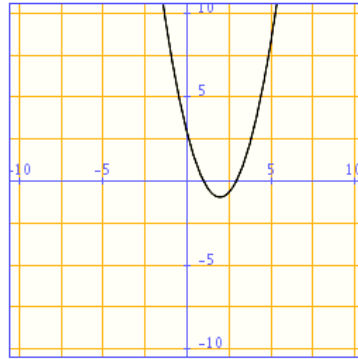
Možemo videti da nemamo konkretan polinom, već promenljivu p , kao i njemu pridružene promenljive a i b .



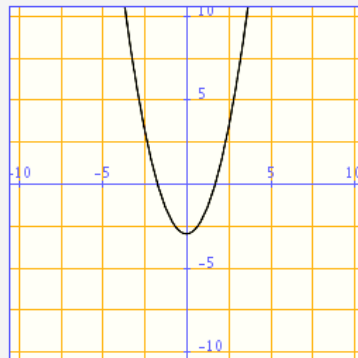
Slika 34. Primer definisanja algoritma za vise promenljivih

U zavisnosti od vrednosti navedenih promenljivih menjaju se i ponuđeni odgovori. Bitno je da nastavnik odradi algoritam koji će na proizvoljan način uvek menjati polinom i date promenljive p , a i b . U ovom primeru ponuđeni odgovori se takode menjaju u zavisnosti od proizvoljno generisanog polinoma:

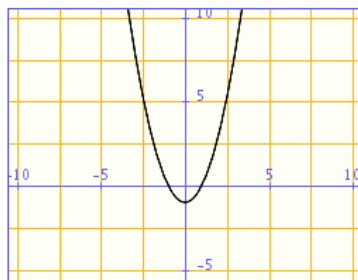
Choose one answer.



a.



b.



Slika 35. Ponudeni odgovori za pitanje sa višestrukim odgovorom

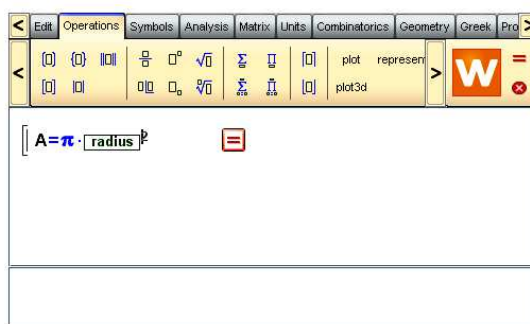
Bitno je napomenuti da Wiris dodatak za Moodle sistem ima mogućnost da svaku aktivnost koju odradimo u editor veoma jednostavno sačuva u vidu apleta, tako da uvek postoji mogućnost da isti modifikujemo i dalje nadgradujemo. Takođe se jednostavno implemetira.

4.3.1 Primena apleta u izradi testova sa višestrukim odgovorom

Nastavnik kao partner u pedagoškoj komunikaciji postavlja pitanja, podstiče interakciju, modeluje diskusije i povezuje sa vanškolskim iskustvom. Postavlja se pitanje kako pospešiti pedagošku interakciju sa akcentom na interakciju u ocenjivanju.

Sledi primer kako se **Wiris Quiz** alat može integrisati sa sistemom za e-učenje i kako je moguće omogućiti interakciju, odnosno da učenik prilikom rešavanja testa ima automatsku povratnu informaciju i dokaz zbog čega je tačan jedan odgovor a drugi nije. Sledi demonstracija na primeru pitanja sa višestrukim odgovorom.

1 Izračunajte površinu kruga prečnika 25 decimetara.
Marks: -/1



- Choose one answer.
- Potrebno je mnogo više informacija da bi se izvršio proračun
 - Približno 5 m²
 - 19.635 m²

Slika 36. Primena Wiris alata u zatvorenom tipu pitanja – pitanje sa višestrukim odgovorom

Bitno je navesti objašnjenje za svaki ponuđeni odgovor, odnosno kako sistem daje povratne informacije učeniku, zbog čega je taj odgovor netačan ili tačan i dokaz za tu tvrdnju:

Choose one answer.

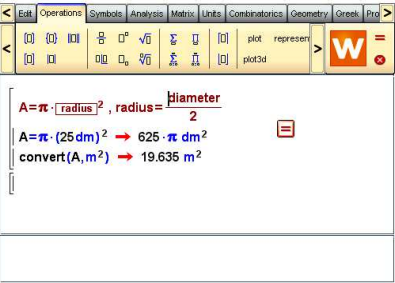
- Potrebno je mnogo više informacija da bi se izvršio proračun ✘ The area of a circle can be calculated using the formula $A = \pi \cdot radius^2 = \pi \cdot \left(\frac{diameter}{2}\right)^2$
- Približno 5 m²
- 19.635 m²

Slika 36.1 Prvi slučaj - Netačan odgovor

Choose one answer.

- Potrebno je mnogo više informacija da bi se izvršio proračun
- Približno 5 m²
- 19.635 m² ❌

You probably forgot to divide diameter length by 2 to obtain the radius length.



Submit

Incorrect

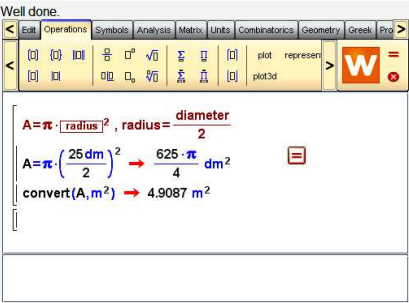
Marks for this submission: 0/1. This submission attracted a penalty of 0.1.

Slika 36.2 Drugi slučaj – Netačan odgovor

Choose one answer.

- Potrebno je mnogo više informacija da bi se izvršio proračun
- Približno 5 m² ✔️
- 19.635 m²

Well done.



Submit

Correct

Marks for this submission: 1/1. With previous penalties this gives 0.9/1

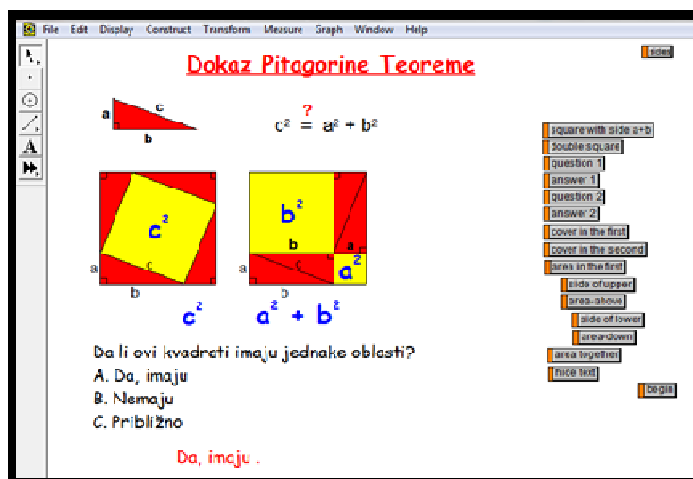
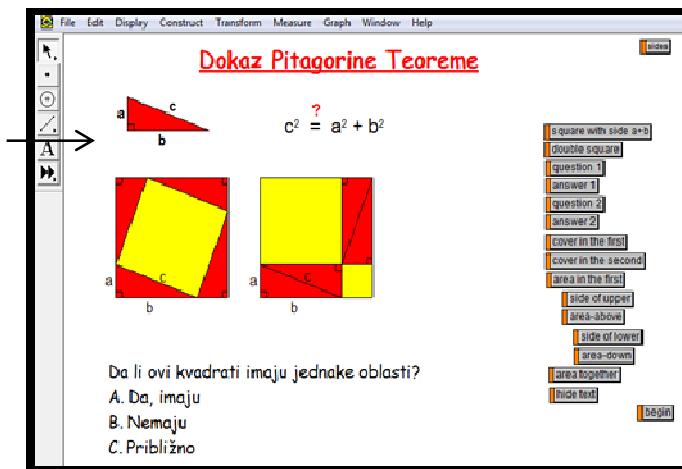
Slika 36.3 Treći slučaj – Tačan odgovor

4.3.2 Provera znanja učenika pomoću SketchPad alata

Kod ocenjivanja u matematici i pripreme za kontrolne ili pismene zadatke, veoma je bitno odabrati adekvatne alate za pripremanje učenika, odnosno omogućiti učeniku postupak samoocenjivanja. U *Geometer's SketchPad* alatu možemo na jednostavan način kreirati java aplete koji u sebi sadrže simulacije koje su namenjene za samoproveru znanja učenika uz dodatne mogućnosti (objasniti učeniku kako da dođe do rešenja, odnosno predstaviti mu postupak rešavanja, kao da se nalazi u učionici i ima ispred sebe nastavnika i tablu).

Na sledećim slikama je predstavljen postupak kako se može učenik pripremiti za kontrolni zadatak uz objašnjenje i analizu postavljenog pitanja. Ovakvi apleti se mogu posmatrati kao prateći materijali uz testove koji imaju ocenjivački karakter. Alat SketchPad pored

možnosti za grafičko predstavljanje rešenja ima opcije da se na jednostavan način dodaju komentari i dodatni funkcionalni dugmići koji omogućavaju učeniku da dalje samostalno istražuje. Aplet koji je prikazan je korišćen u toku kursa – *Primena Java apleta u nastavi matematike* (sedmi razred, nastavna jedinica – Pitagorina teorema).

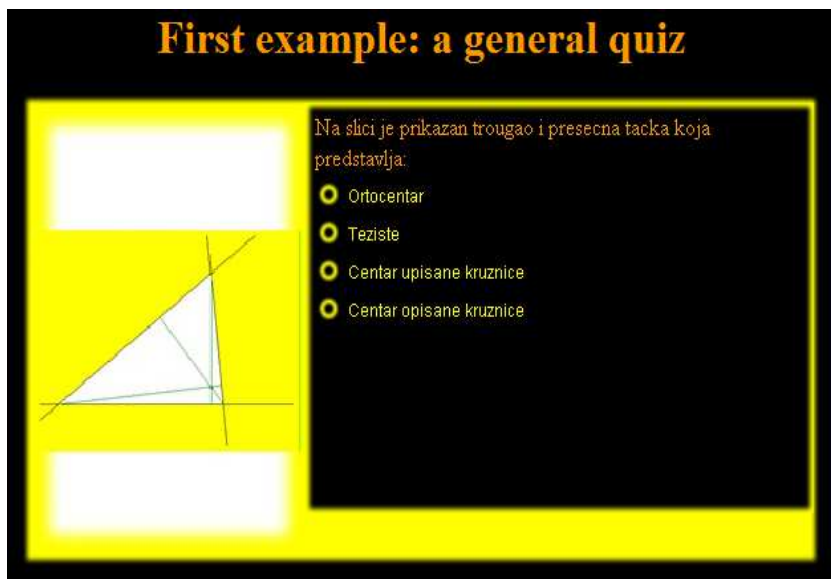


4.3.3 Primer kreiranja testova u obliku apleta - QuizMaster

Test znanja koji je namenjen za evaluaciju u nastavni čini niz pitanja iz obrađenih nastavnih sadržaja na koja učenici odgovaraju samo jednom rečju, zaokruživanjem ili podvlačenjem tačnog odgovora (Morales, 2009). Ako želimo unaprediti učenje, potrebno je osigurati da učenici znaju što se od njih očekuje, da se upoznaju sa standardima znanja i da imaju

moгуćnost tokom proveravanja učiti za buduće ocenjivanje. Nastavnik treba načinom svoga rada i prikladnim povratnim informacijama postići da učenici žele proveravanje.

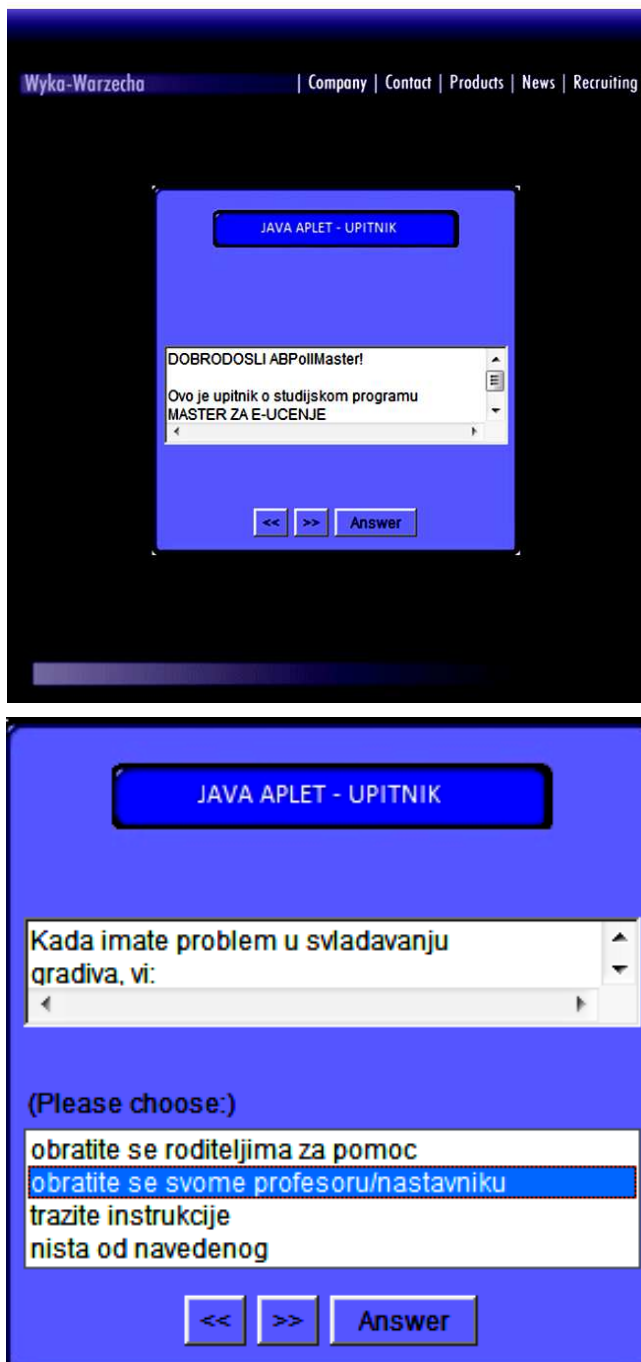
Sledi primer testa koji je rađen za kurs “Primena Java apleta u nastavi matematike”. Test je odrađen kao Java aplet i sadrži niz pitanja sa višestrukim odgovorima iz oblasti Geometrije. Postavlja se pitanje, *Zašto odraditi test kao Java aplet?*. Odgovor je jasan, zbog lakoće implementacije na internetu ili u nekom od sistema za e-učenje.



Slika 37. primer testa odrađen u obliku Java apleta

4.3.4 Primer kreiranja upitnika i anketa u obliku apleta - ABPoll Master

Upitnik je instrument koji je značajno koristiti u matematici za prikupljanje odgovora koji se sastoji od popisa niza pitanja. Ukoliko ispitanik sam popunjava upitnik onda je reč o anketi. Upitnik nema ocenjivački karakter već ima za cilj da prikupi potrebne podatke vezane za neki problem, predmet ili oblast. Za potrebe istraživanja kreiran je upitnik u vidu apleta sa nizom pitanja vezanih za predmet matematika gde su učenici u obliku pitanja sa višestrukim odgovorom navodili svoja mišljenja.



Slika 38. Primer kreiranog upitnika u obliku apleta

Pri kreiranju upitnika za potrebe nastave veoma je bitno da se uključe samo ona pitanja koja se neposredno odnose na sam problem. Pri kreiranju upitnika za učenike za potrebe istraživanja poštovana su sledeća pravila (Aberson, Berger i ostali, 2000):

- Postaviti samo ona pitanja za koje se može pretpostaviti da većina ispitanika zna odgovore (konkretno u ovom slučaju pitanja se odnose za predmet matematika i njegove metode izvođenja, za prednosti i nedostatke nastave, za unapređenje itd.),
- Isključiti pitanja na koja se odgovori mogu pribaviti iz drugih izvora ,
- Ponuditi učenicima jasne i razumljive odgovore,
- Uključiti sve moguće situacije,
- Osigurati uporedive podatke itd.

5. ALATI ZA KORISNIČKU PODRŠKU U SISTEMIMA ZA E-UČENJE

U tradicionalnoj organizaciji nastavnog procesa individualni pristup i njegovo ostvarenje nailaze na ozbiljne poteškoće. U tradicionalnom sistemu obrazovanja ne postoji mogućnost individualizacije nastave, odnosno ovde se prvenstveno misli na rad sa talentovanim učenicima. Postoji jedan vid dodatne nastave (ili sekcija slobodnih aktivnosti) koji nije dovoljan. Individualni pristup proizilazi iz potrebe talentovane dece za većim napredovanjem u odnosu na grupu ili obrnuto. Zato se konkretno u nastavi matematike kao prioritetan nameće primena inovativnih metoda tipa simulacija i drugih multimedijalnih sadržaja kao korisničke podrške učenicima.

Kao odgovor na ovaj problem navodimo podršku primenom Java apleta. Ova podrška nije ograničena mestom i vremenom a takođe postoje i nivoi složenosti koje diktira talentovani pojedinac. Pored apleta postoje i celokupni sistemi koji su namenjeni dodatnom obrazovanju ili kao podrška već tradicionalnim metodama obrazovanja. Konkretno predlog bi bio da se u svakoj školi implementira neki od sistema za e-učenje (ili portal sa velikom brojem apleta, interaktivnih sadržaja, simulacija itd.) koji će imati za cilj da talentovanoj deci pruži individualno napredovanje kroz dodatne aktivnosti, testove, diskusije (kontakt sa nastavnikom i drugom decom), rad na praktičnim i zanimljivim zadacima itd.

5.1 Praktični razlozi za korisničku podršku

Neke od ključnih prednosti integrisanja simulacija u korisničku podršku su (Materijal : E-Lab TFC, L3):

- simulacije pomažu u razvoju velikih ideja i koncepata,
- omogućavaju učenicima da donose odluke u bezbednom okruženju, pre nego se izlože potencijalno rizičnim i opasnim situacijama,
- omogućavaju da se u dosta realnim uslovima sagledaju neke situacije koje nije moguće lako doživeti u realnim uslovima,

Ukoliko obrazovna institucija želi da svoje talentovane učenike dodatno podstakne na rad sa eksperimentima to vrlo lako može postići bez preteranih ulaganja u laboratorijski potrošni materijal jer je za mnoge instrumente potrebno daleko više opreme i sredstava nego za korišćenje

računara. Naravno treba naglasiti da ni u kom slučaju upotreba računara ne sme da zameni praktični laboratorijski rad.

Simulacije omogućavaju korisnicima da se suoče sa složenim i kompleksnim zadacima koji zahtevaju da aktivno koriste znanje i veštine iz različitih oblasti, čime stiču konkretna iskustva u rešavanju problema. Simulacije se mogu koristiti u nizu različitih situacija bilo da se radi o online učenju ili učenju u učionici. Ovi interaktivni sadržaji se mogu koristiti da se predstavi tema koja se obrađuje ili da se obave eksperimenti, da se podstakne diskusija ili da se steknu istraživačka iskustva. Takođe se mogu koristiti za uvođenje u nova istraživanja.

5.2 Primena apleta u kreiranju virtuelnih učionica i laboratorija u nastavi matematike

Konkretno, dobar predlog za unapređenje nastave mogu biti virtuelne učionice (*eng. Virtual Classroom*) koje se u osnovi mogu definisati kao skup tehnologija, strategija za učenje, prezentacija i raznih aktivnosti za učenje kojima se podstiče i promovise interakcija u realnom vremenu između grupe korisnika i nastavnika. Bitno je naglasiti da virtuelne učionice mogu biti formirane kao skup apleta (interaktivnih simulacija i sadržaja) koji pružaju adekvatnu korisničku podršku učenicima u učenju.

Tabela 4: Besplatni alati koji se mogu iskoristiti za razvoj virtuelnih učionica

Virtual classrooms	Voice Chat Tool	Slide show	Text Chat	Application sharing
WizIQ	YackPack	Slide share	Meebo	Boscoss screen
Yugma	Skype Yahoo Messenger		Yahoo Messenger MSN Messenger	share

Kao drugi dobar predlog mogu se navesti i tzv. virtuelne laboratorije koje predstavljaju on-line interaktivnu demonstraciju procesa i sistema koja omogućava učenicima da vežbaju u kontrolisanom i sigurnom okruženju. Za talentovane učenike VI, VII i VIII razreda odrađena je virtuelna laboratorija za predmet Matematika.

Za razliku od prvog vida korisničke podrške, virtuelne laboratorije zahtevaju od učenika da na neki način interaktivno učestvuje u izvršavanju eksperimenta, bilo da se traži unošenje nekih podataka ili donošenje odluka ili izvršavanje nekih koraka eksperimenta. Virtuelne

laboratorije takođe omogućavaju korisnicima da više puta ponove isti eksperiment (Materijal : E-Lab TFC, L5).



Slika 39. Naslovna strana Virtuelne laboratorije namenjene talentovanim učenicima

Za potrebe nastave matematike i istraživanja u toku šest nedelja trajanja kursa odrađena je web strana [<http://muzafers.uninp.edu.rs/virtualLAB.html>, postupak i način izrade ove strane je opisan u radu (Saračević i Mašović, 2010)] u vidu virtuelne laboratorije sa mnoštvom apleta koji se direktno odnose na zadatke koji su rađeni u toku školskog časa. Pristup ovoj virtuelnoj laboratoriji je dostupan svima. Laboratorija se sastoji od 4 dela:

- Deo za 6 razred
- Deo za 7 razred

- Deo za 8 razred
- Linkovi za nastavnika

a = 40 a² = 1600

Primer br4 - Aplet ilustruje pitagorinu teoremu i odnos kateta nad hipotenuzom

Primer br5 - Aplet ilustruje odnos centralnog i perifernog ugla u trouglu.

Increase the angle at the centre

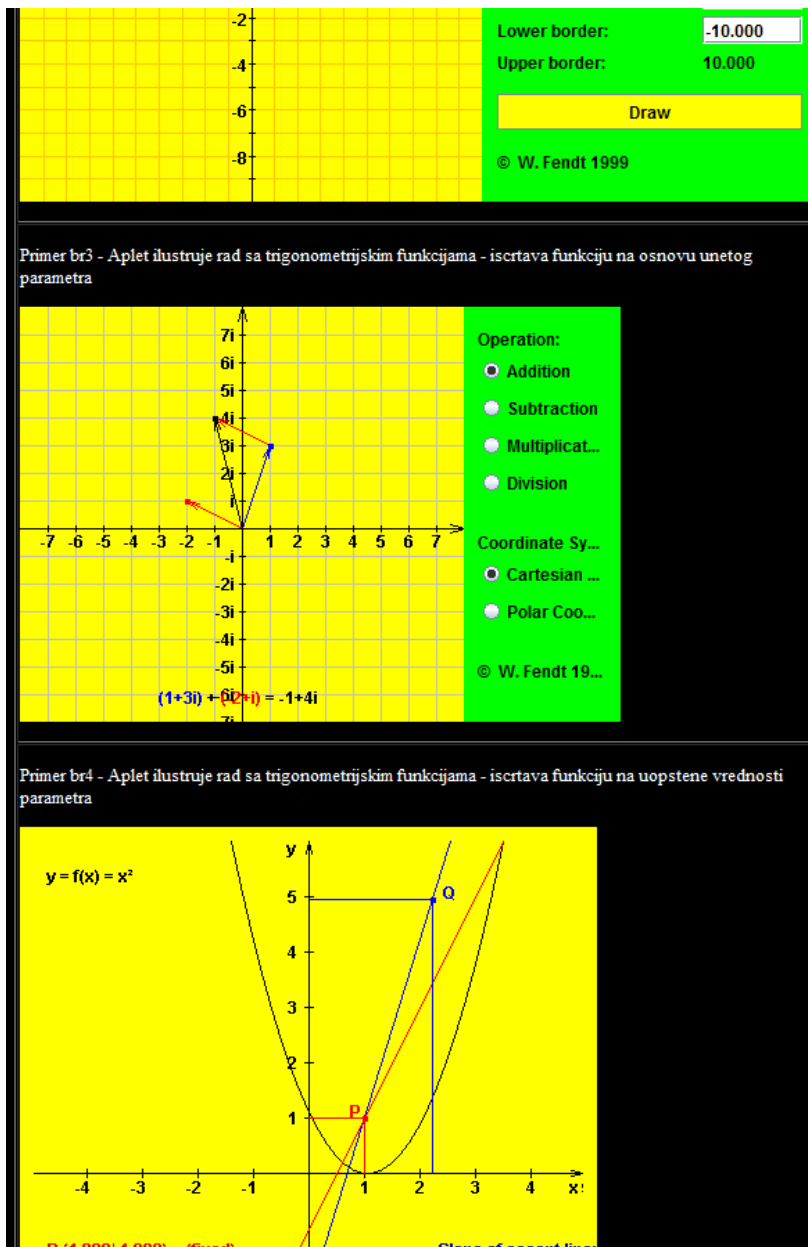
Decrease the angle at the circumference

Angle at the centre:

Angle at the circumference:

Angle between the chord and the tangent:

© W. Fendt 1997

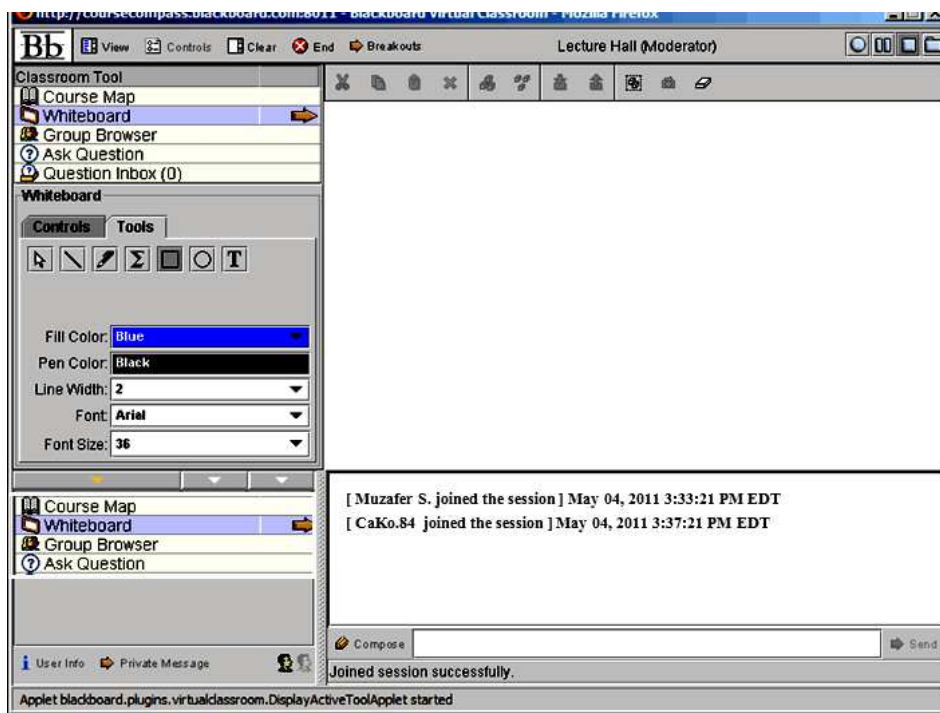


Slika 40. Delovi Virtuelne laboratorije sa velikim brojem apleta za eksperimentisanje

5.2.1 Primer virtuelne učionice u obliku apleta

Virtual Classroom je Java aplet koji zahteva svim učesnicima, koji žele da pristupe, instalaciju *Java plugin (dodatka)*. Za razliku od foruma ovaj je aplet sinhron, znači komunikacija

se odvija u realnom vremenu. Da ne bi došlo do problema u komunikaciji potrebno je učesnike podeliti u nekoliko manjih grupa.



Slika 41. Okruženje apleta Virtual Classroom

Ključne prednosti ovakve učionice, koja je kreirana u vidu apleta, su:

1. Kompletnu sesiju je moguće veoma jednostavno snimiti u obliku apleta,
2. Postoji jednostavna mogućnost da se ova virtuelna učionica implementira na webu ili nekom programu za kreiranje interaktivnog sadržaja,
3. Veoma se jednostavno uvozi u sisteme za e-učenje,
4. Može se veoma jednostavno implementirati i na mobilne uređaje i smart telefone (svaki telefon koji podržava Javu).


Veštine koje se podstiču kroz virtuelne učionice u nastavi matematike (Turoff, 1995) su:

- Rad u grupama i timski rad, tako što se definišu uloge u timu, rešavaju kritični problemi i diskutuje u timu. Virtuelne učionice imaju i posebne “sobe” gde korisnici mogu da se sastaju i da rade na rešavanju zadataka koji su im postavljeni. Kao predlog dodatnog alata u ovom slučaju bio bi opisan dodatak *WIRIS conference – math chat*.
- Rešavanje problema i razmatranje različitih solucija za rešenja, kroz diskusiju koja se odvija u grupi. U grupi se mogu razmotriti različite ideje za rešenje problema i isprobati različite strategije u rešavanju zadataka.
- Komunikacione sposobnosti, postavljanjem pitanja i davanjem odgovora, time što se od korisnika traži da zauzmu stav u određenim situacijama, da se slože ili da iskažu svoje neslaganje i da objasne svoj stav.
- Efikasno korišćenje novih tehnologija, korisnici se navikavaju da nove tehnologije koriste u okviru različitih aktivnosti.

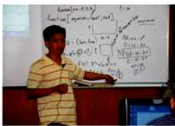
Mathematics and Multimedia

A blog about teaching and learning mathematics, software tutorials, technology integration, Web 2.0 technology, blogging and more.

[HOME](#) [POSTS LIST](#) [MATHEMATICS](#) [MULTIMEDIA](#) [GEOGEBRA](#) [ALL FOR FREE](#) [BLOGROLL](#) [ABOUT](#)




Applets



I work in a university in the Philippines and my interest is on the integration of ICT in teaching mathematics. You can contact me at mathandmultimedia@gmail.com

SEARCH BLOG



- April 2011 (28)
- March 2011 (29)
- February 2011 (19)
- January 2011 (17)
- December 2010 (16)
- November 2010 (15)
- October 2010 (16)
- September 2010 (17)
- August 2010 (17)
- July 2010 (18)
- June 2010 (17)
- May 2010 (17)
- April 2010 (14)
- March 2010 (14)
- February 2010 (12)
- January 2010 (12)

- [Chris Sangwin's GeoGebra Page](#)
- [GeoGebra Applet Central](#). My new blog. It will contain the GeoGebra applets of Math and Multimedia.
- [GeoGebra Files](#) by Ranger College
- [GeoGebra Gettys](#).
- [GeoGebra Applets](#) by Saint Louis University
- [GeoGebra Applets](#) by Montana Council of Teachers in Mathematics
- [GeoGebra Applets](#) used in PreCalculus and Calculus from Minnesota West.
- [GeoGebra Applets for Math](#) by dmnlearn
- [GeoGebra Math Interactive Lessons](#). Interactive GeoGebra Applets.
- [GeoGebra Resources](#) by William Emeny.
- [GeoGebra Dynamic Worksheets](#) at [The Random Mathematical Ponderations of a Math Geek](#)
- [Lindy's Archive](#)
- [GeoGebra Applets](#) from Math Maine

Slika 42. Primer virtuelne laboratorije na <http://mathandmultimedia.com>

6. IMPLEMENTACIJA U SISTEMIMA ZA E-UČENJE

Okruženja za e-učenje se javljaju u različitim varijantama od kojih su najčešće (Materijal : E-Lab TFC, L2):

- Sistemi za upravljanje učenjem (Learning Management Systems - LMS),
- Sistemi za računarom podržano podučavanje (Computer Managed Instruction Systems)
- Sistemi za upravljanje sadržajem učenja (Learning Content Management Systems – LCMS)
- Virtuelna okruženja za učenje (Virtual Learning Environments - VLE) ,
- Web zasnovani sistemi za obuku (Web Based Training Systems - WBT),
- Platforme za upravljanje učenjem (Learning Management Platforms - LMP).

6.1 Pojam i podela sistema za e-učenje

Apleti nalaze svoju primenu u sistemima za e-učenje i mogu se koristiti u njihovom razvoju. U radu (Codolini i ostali, 2007) se apleti pominju u kontekstu centralnog rukovodioca prenosa podataka u sistemima za e-učenje. Posmatranjem aktuelnog stanja konfiguracija sistema za e-učenje primećuje se da se u fokusu nalaze:

1. Sistemi za upravljanje učenjem¹⁷
2. Sistemi za upravljanje sadržajem učenja

6.2 Moodle-sistem za e-učenje

Moodle (Modular Object-Oriented Dynamic Learning Environment)¹⁸ je jedan od najpopularnijih LCMS-ova. Značajan deo projekta Moodle je i veb-sajt *moodle.org*, koji služi kao

¹⁷ Sistem za upravljanje učenjem (LMS) predstavlja programsku podršku koja globalno omogućava potpuno administriranje procesa učenja i podučavanja. LMS obavlja registraciju studenta, omogućava praćenje kurseva u katalogu kurseva, opis podataka o studentu, i omogućava izveštavanje o obavljenom. Osim toga, LMS je obično oblikovan tako da može upravljati kursevima koje su isporučili različiti izdavači i provajderi usluga. Obično LMS ne uključuje u svojoj konfiguraciji autorske alate za stvaranje nastavnog sadržaja. Proizvođači LMS sistema obično nude i dodatne alate za stvaranje nastavnog sadržaja. Ponovna upotrebljivost je na nivou celog kursa (jedan kurs može se isporučiti većem broju studenta, omogućeno je praćenje postignuća).

centar za prikupljanje informacija, diskusije i saradnju korisnika Moodle projekta (sistem administratora, predavača, istraživača i naravno kreatora kurseva).

Opšte karakteristike Moodla su sledeće (Sasikumar, 2008):

- Ugrađena je podrška za ocenjivanje i praćenje aktivnosti studenata,
- Sistem uloga (roles) može se podešavati do nivoa aktivnosti,
- Podržava kolaboraciju preko foruma, četa, vikija i drugih modula,
- Podržana je izrada testova sa raznovrsnim oblicima zadataka,
- Podržan je uvoz standardizovanih paketa za objekte učenja,
- Na Internetu je dostupan veliki broj besplatnih dodataka (plug-ins),
- Moodle je lokalizovan na 78 jezika,
- Podržane platforme (operativni sistemi) su: UNIX/Linux, Windows, MacOS.
- Podržane baze podataka su: MySQL, Oracle, MS SQL, PostgreSQL itd.

6.3 Mogućnosti uvoza interaktivnog sadržaja i apleta u Moodle sistem

Iz ugla administratora školske računarske mreže treba omogućiti da nastavnik ima pristup alatima u kojima može da pravi lekcije koje poštuju aktuelne tehničke standarde i da ima na raspolaganju sistem za distribuciju takvih nastavnih sadržaja i aktivnosti ka svojim učenicima. Bitno je za ovakve sisteme obezbediti infrastrukturu u vidu potrebnog hardvera i softvera (Međedović, Saračević i ostali, 2011).

Mogućnosti uvoza apleta u Moodle sistem su višestruke. Aplet možemo ubaciti u tekstualnu stranicu, forum, natpis ili najjednostavnije kao link na web stranicu.

¹⁸ Nastao je kao doktorska disertacija australijskog istraživača Martina Dugimasa.



Slika 43. Načini implementacije apleta u Moodle okruženje

Slede koraci uvoza apleta u tekstualnu stranicu na Moodle sistemu:

<p>KORAK 1</p> <p>Kreirajte svoj aplet u nekom od ponuđenih alata i sačuvajte ga u obliku dinamičke web strane.</p> <p>Otvorite web stranu i birajte view source. To možete uraditi u nekom od programa za obradu web strana ili u Notepad-u.</p>	<p>KORAK 2</p> <pre><HTML> <HEAD> </HEAD> <BODY BGCOLOR="000000"> <CENTER> <APPLET code = "MojAplet.java" width = "500" height = "300" > </APPLET> </CENTER> </BODY> </HTML></pre>
<p>KORAK 3</p> <p>The screenshot shows the Moodle editor interface. The title "PRIMENA JAVA APLETA U NASTAVI MATEMATIKE" is entered. Below the title, there are logos for Java and GeoGebra. An arrow points from the "View source" icon in the editor toolbar to the source code in the next step.</p>	<p>KORAK 4</p> <p>The screenshot shows the browser's source code for the page. It displays the HTML code for the applet, including the <APPLET> tag with attributes for code, width, and height. An arrow points from the source code in the previous step to this screenshot.</p>

Drugi način implementacije je da jednostavno učitamo *html* stranicu na kojoj se nalazi java aplet. Postoji mogućnost i da se implementira aplet u okviru foruma, testa ili rečnika pojmova.

Primer ispravne implementacije java apleta i njegovo učitavanje:



Slika 44. Primer učitavanja apleta

6.4 Moodle dodaci za rad sa apletima

Moodle je modularan sistem, što znači da se sastoji od više manjih delova (*modula*) koji zajedno čine jednu celinu, a mogu se odvojeno dodavati ili menjati. Ovaj sistem za e-učenje poseduje veoma veliki broj dodataka (*modules & plugins*) koji se vrlo jednostavno mogu dodavati, brisati ili uređivati a takođe ih nastavnici na svom kursu mogu uključiti, uklanjati ili uređivati. Koliko će biti blokova i kako će biti raspoređeni zavisi od konfigurisanja sistema koje obavlja administrator. Svaki blok predstavlja određeni skup funkcionalnosti koje sistem pruža.

Upravo zbog gore navedenih stvari, glavne prednosti Moodle-a su interaktivnost, multimedijalnost i individualizacija, odnosno prilagodljivost svakom korisniku. Postoje dodaci (filteri, blokovi, integracije itd.) koji se mogu naknadno instalirati, poboljšava prilagodljivost specifičnim potrebama, a samim tim i pospešuje funkcionalnost i upotrebu sistema. Jednostavnost instalacije i pristupačnost dodataka je još jedna prednost. Tako da ako poželimo neku funkcionalnost koju bi u ulozi kreatora kursa (ili studenta) voleli da sistem za e-učenje ima, rešenje postoji, jer Moodle sistem zaista ima mnoge mogućnosti koje se mogu kroz instalaciju

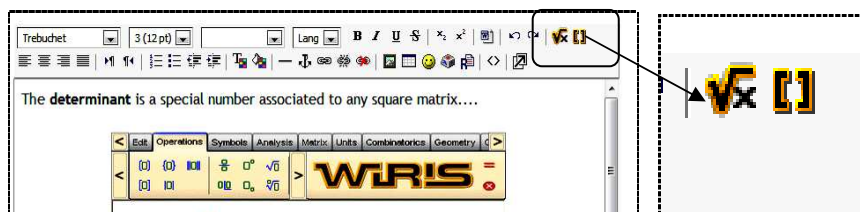
dodatnih modula (dodataka) uključiti u cilju ostvarivanja specifičnih potreba. Jedna od odličnih osobina Moodlea jeste njegova proširivost i integracija sa drugim sistemima. Postoji ogroman broj različitih aktivnosti koje se mogu dodati Moodleu u obliku tzv. modula. Postoji čak 25 različitih tipova modula (Materijal : E-Lab TFC, L6).

U nastavku ovog rada sledi opis i mogućnosti dodataka :

- **WIRIS plugins** (modul tipa *integration*)
- **GeoGebra plugins** (modul tipa *filter*)

6.4.1 WIRIS dodatak za Moodle sistem

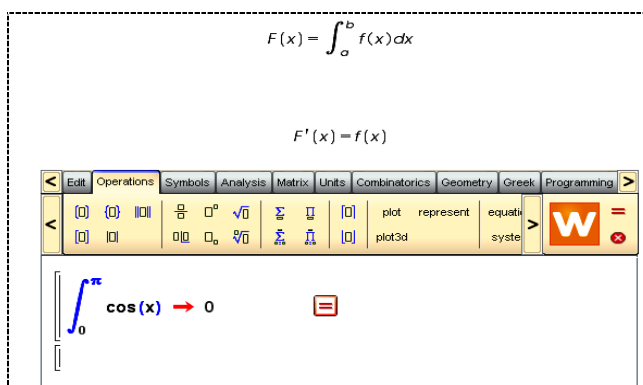
Integracija Wiris dodatka u Moodle sistemu je veoma jednostavna. Potrebno je samo preuzeti sa *Moodle-ovog sajta* ovaj dodatak i instalirati ga. Nakon instalacije desiće se određene promene u editor u sistemu za e-učenje.



Slika 45. Izgled editora posle instalacije Wiris plugin-a u Moodle sistemu

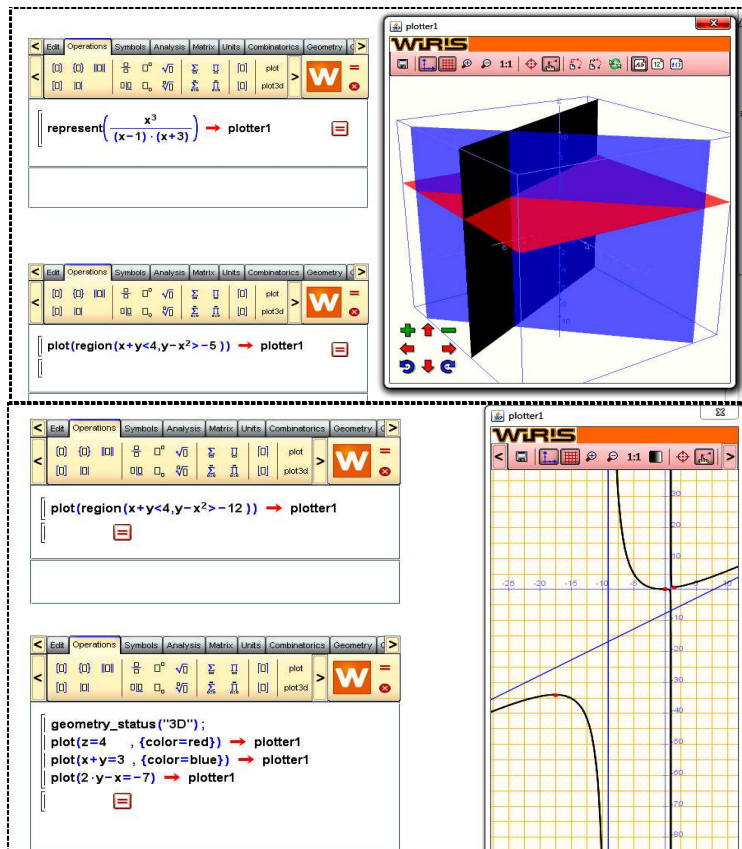
Slede mogućnosti i prednosti integracije *Wiris* alata sa Moodle sistemom:

- A) Mogućnost lakog dodavanja matematičkih formula i simbola u tekstualnim stranicama na Moodle sistemu:



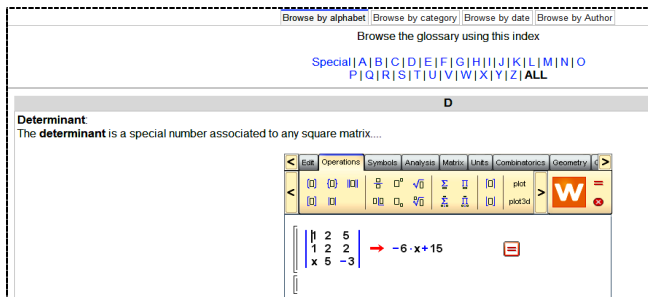
Slika 46. Importovanje u resurse - tekstualna strana

B) Dodavanje grafika i dinamičko menjanje na osnovu trenutnog unosa:

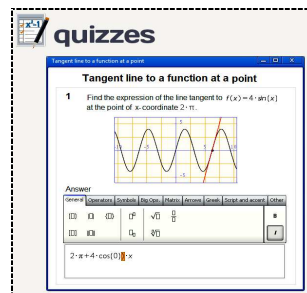


Slika 47. Mogućnost generisanja grafike u zavisnosti od sadržaja

C) Dodavanje formula, simbola i grafika u rečniku pojmova i izuzetne mogućnosti Wiris alata u kreiranju testova u Moodle sistemu (ovaj postupak je objašnjen u poglavlju 4):

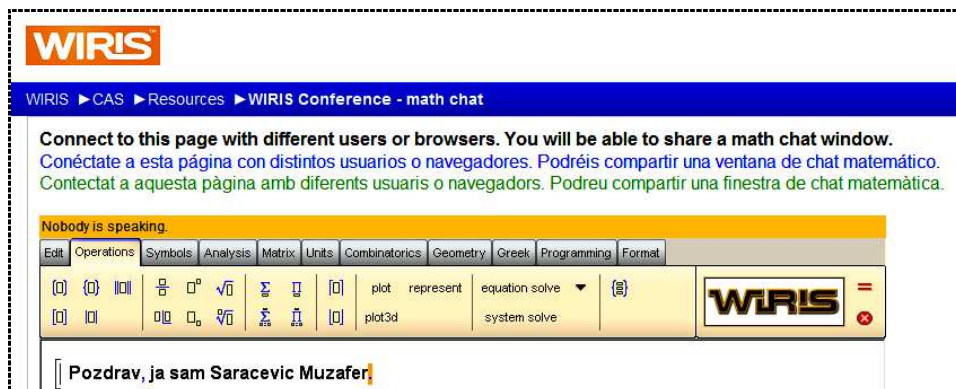


Slika 48. Importovanje u rečnik pojmova



Slika 49. Kviz

D) Wiris modul za kolaborativan rad – Wiris Konferencija i “čet za matematičare”:



Slika 50. chat

6.4.2 GeoGebra dodatak za Moodle sistem

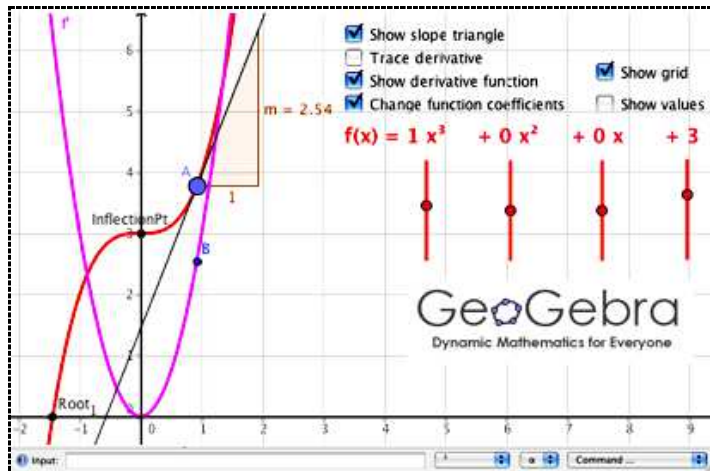
U poglavlju 3 je bilo reči o alatu *GeoGebra* koja omogućava kreiranje interaktivnih sadržaja u obliku apleta. Kao što smo mogli videti, taj aplet smo mogli na veoma jednostavan način ubaciti u sistem za e-učenje. Međutim, šta ako korisnik (nastavnik, kreator kursa ili čak učenik) bude želeo da taj već gotov aplet menja, odnosno poželi da ima *GeoGebra editor* u samom sistemu za e-učenje.

U Moodle sistemu postoji dodatak u vidu *filtera*. Ovaj modul nam može omogućiti da imamo editor unutar sistema i da svaki postavljeni aplet koji je rađen u ovom alatu možemo menjati (ovo se može poistovetiti sa open source sistemima). Dodatak može doprineti boljem i kvalitetnijem učenju u sistemu kao što je Moodle.

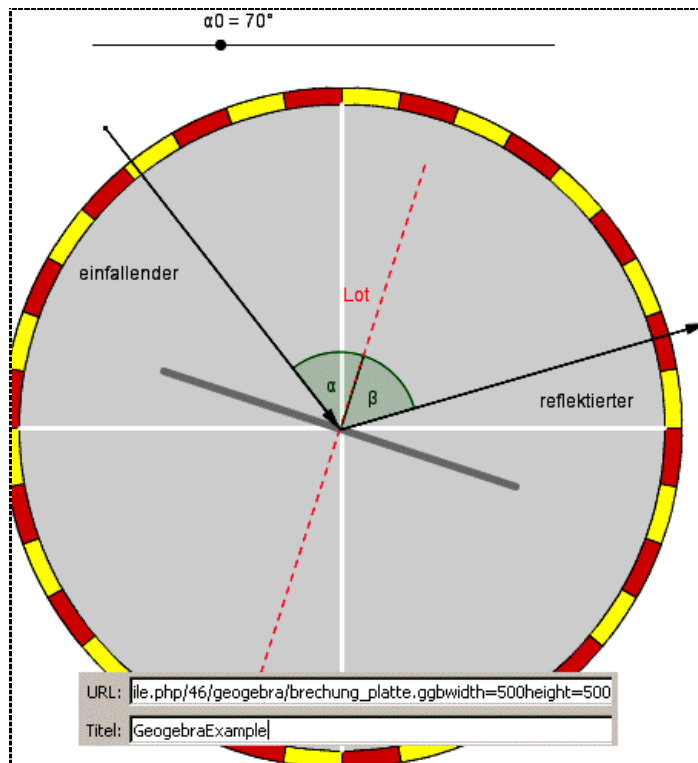


Slika 51. Primer rada sa filterima u Moodle sistemu za e-učenje

Sa *GeoGebra filterom* možemo da uključujemo dodatne elemente u već postojeći aplet ili da odradimo potpuno novi projekat (aplet). Znači, sa ovim dodatkom dobijamo novi editor, odnosno novo okruženje koje nam omogućava da radimo u alatu *GeoGebra*.



Slika 52. GeoGebra editor u sistemu za e-učenje



Slika 53. Geogebra filter (Math Applets)

ZAKLJUČAK

U ovom praktikumu analizirane su prednosti primene interaktivnih sadržaja u nastavi a generalno razmatrana je i integracija informaciono-komunikacionih tehnologija u školama koja može pružiti realnu šansu za napredovanje učenika i nastavnika. Prvenstveno se navodi izvođenje nastave matematike uz upotrebu apleta, animacija, simulacija i individualizacija nastave posredstvom apleta i sistema za e-učenje. Apleti su predstavljeni kao alati za kreiranje interaktivnih sadržaja za e-učenje i kao alati koji se mogu primeniti u ocenjivanju odnosno evaluaciji u nastavi matematike. Naravno ovakav metod bi trebao da postane imperativ, šta više standard u našem obrazovnom sistemu, i ne samo kod nastave matematike, već i svih predmeta gde postoji potreba za ovakvim vidom nastave.

U ovom praktikumu akcenat je stavljen na prirodne nauke, stoga autor ovog rada navodi svoja iskustva (u toku 3 godine rada) i predloge kako bi se unapredila nastava matematike. Pre svega, navedeni su interaktivni sadržaji u nastavi kao poseban vid nastave pomoću računara koji je primenljiv u svim disciplinama a posebno u prirodnim i tehničkim naučnim oblastima u kojima postoji realna potreba za vizuelizacijom procesa.

Kratak pregled svih izloženih poglavlja, njihovih ishoda kao i ideja za dalja istraživanja, biće prikazani u ovom zaključnom delu rada. Rad može da posluži svi nastavnicima matematike, informatike, fizike i tehničkog obrazovanja, kao i svim profesorima razredne nastave, a prvenstveno se odnosi na unapređenje nastave matematike.

Ishod prvog dela rada bi trebao biti da svi nastavnici znaju šta je interaktivna nastava odnosno da upoznaju pojam interaktivnih sadržaja i njihovih mogućnosti i primena u nastavi. Takođe i da se upoznaju sa konkretnim alatima za izradu interaktivnog sadržaja koji se mogu koristiti za e-učenje.

Drugi deo rada bi trebao da nastavnike upozna sa pojmom apleta, sa njegovom strukturom, implementacijom i osnovnim java naredbama. Nakon toga bi trebalo da svi nastavnici imaju konkretan uvid u mogućnosti apleta kao alata za kreiranje interaktivnih sadržaja tj. simulacija, da upoznaju mogućnosti apleta kao alata za kolaborativan rad, za planiranje učenja kroz odgovarajuće mape uma, a i kao alata za kreiranje obrazovnih igara. Veoma je bitno da svi koji pročitaju ovaj rad delimično upoznaju i programske alate za kreiranje apleta kao što su *GeoGebra*, *JavaSketchPad*, *Geometer's SketchPad*, *Wiris*, *Mathematica*, *Java View* i *Easy Java Simulations*.

Ishod trećeg dela rada bi trebao biti da nastavnici znaju da primene aplete kao alate za ocenjivanje i evaluaciju u e-učenju kroz testove, ankete, upitnike i sl. Naravno sve ove mogućnosti Java apleta je bitno povezati sa metodama i postupcima ocenjivanja u nastavi matematike. Dati su i predlozi kako se mogu kreirati interaktivni testovi (sa višestrukim odgovorima i drugim tipovima) pomoću alata za interaktivne sadržaje, a ishod je da nastavnici razumeju način primene apleta i kreiranje automatskog povratnog odgovora koji je veoma koristan za učenike.

Četvrti deo rada bi trebao da upozna nastavnike i učenike sa korisničkom podrškom u vidu java apleta, kroz tzv. virtuelne laboratorije i učionice. Odnosno da se predstavi jedan vid nastave koji se može realizovati kroz aplete koji se mogu grupisati u virtuelne učionice.

Ishod petog dela je da nastavnici spoznaju mogućnosti implementacije java apleta u sistemima za e-učenje i njihovu konkretnu primenu u nastavi matematike. Zadatak ovog dela je i da se predstave mogućnosti sistema za e-učenje kao što je Moodle i da se objasni postupak uvoza apleta kroz module (dodatke).

Sve ovo navodi na zaključak da era digitalne nastave, koja će biti vizuelizovana i podržana sistemom e-učenje, tek treba da doživi svoju pravu ekspanziju. Naravno ovaj proces bi trebao da se ostvari i uvede u našem obrazovnom sistemu, jer je bitno istaći da kvalitetni interaktivni sadržaji omogućavaju svakom učeniku da ima aktivnu ulogu u procesu učenja i da se u nastavi prvenstveno mora staviti akcenat na primenu teorijskog znanja u praksi.

10. Literatura

1	Aberson, C. L., Berger, D. E., Healy, M. R., Kyle, D., & Romero, V. L. (2000). Evaluation of an interactive tutorial for teaching the Central Limit Theorem, <i>Teaching of Psychology</i> , 27, 289-291.
2	Ardito C. , De Marsico M., Lanzilotti R. (2004), Usability of E-learning tools, <i>AVI '04 Proceedings of the working conference on Advanced visual interfaces ACM New York, NY, USA, Gallipoli, Italy, May 25-28</i> (pp. 80-84).
3	Arnold K., Gosling J., Holmes D. (2001.), <i>Programski jezik Java</i> , treće izdanje, CET, Beograd.
4	Beene G., (2004), <i>3d on the web - pure java applets</i> , internet članak - http://www.garybeene.com/3d/3d-pure.htm
5	Bjekić, D. (2008), <i>Psihologija učenja i nastave u e-obrazovanju</i> , materijal za studijski program- master za e-učenje, Tehnički fakultet Čačak – E-LAB.
6	Bjelanović D. Ž., (2005), <i>Učenje istraživanjem u java apletima prema modelu Georgea Polya</i> , internet članak - http://www.normala.hr/wp-content/uploads/2010/09/Bjelanovic.pdf
7	Blachman N.(1992), <i>MATHEMATICA: A Practical Approach</i> , Englewood Cliffs, New Jersey: Prentice-Hall.
8	Cekuš G., Namestovski Ž. (2005.): Primena računara na nastavnim časovima. <i>Međunarodna naučno-stručna konferencija: Savremene informatičke i obrazovne tehnologije i novi mediji u obrazovanju</i> , Učiteljski fakultet u Somboru, ISBN: 86-83097-31-5, 16-29, Sombor.
9	Codolini P., Davolini F., Marescotti G., Maryni P. (2007), <i>Developing a distance laeraning system using java apalets</i> . ISBN: 0-7803-3336-5, 63 – 66.
10	Đukic M. (1995.), Individualizacija procesa usvajanja znanja u nastavi, <i>Nastava i vaspitanje</i> , vol. 44, br. 3, str. 197-206 Beograd.
11	Francisco E., Jose S.M, (2005), <i>Easy java simulations-using ejs to run simulink models in an interactive way for version, e-book</i> .
12	Francisco E. (2005), <i>Easy java simulations- manual for version 3.4, e-book</i> .
13	Gray T., Glynn J. (1991.), Exploring Mathematics in MATHEMATICA, <i>Redwood City, California: Adisson-Wesley</i> .
14	Healy, Berger, Michael R., Dale E..., (2004), Evaluating java applets for teaching on the Internet, <i>Java Applets in Education</i> , (2, Pt. 1), pp. 279-285.
15	Horthon W., Horton K. (2003), <i>E-learning tools and technologies</i> , internet portal - http://www.horton.com/toolsworkshop.htm
16	Knežević O. (2004), Interaktivna nastava iz matematike, <i>Obrazovna Tehnologija</i> , br. 2/2004.
17	Kurnik Z. (2000), Matematički zadatak, <i>Matematika i škola</i> 7, 51–58.
18	Kurnik Z. (2000), Suvremena metodika i nastava matematike, <i>Zbornik radova 1. kongresa nastavnika matematike Republike Hrvatske</i> , 187–201, Zagreb.
19	Kurnik Z. (2001), Matematičke sposobnosti, <i>Matematika i škola</i> , 10 , 195–199.
20	Lemay W., Cadenhead R., (1998), <i>Sams teach yourself java 1.2 in 21 days</i> , sams publishing.

21	Maeder R.(1996), <i>Programming in MATHEMATICA</i> , Third Edition, Redwood City, California: Adisson-Wesley.
22	Markus H., Hohenwarter J. (2009), <i>GEOGEBRA, zvanično uputstvo 3.2</i> , praktikum za predmet – primena računara u matematici, PMF, Novi Sad.
23	Mašović S., Saračević M.(2010), Modelovanje poslovnih procesa i primena Data Mining tehnika u e-učenju, <i>Univerzitet METROPOLITAN - Elektronsko učenje na putu ka društvu znanja 2010</i> , vol1., 107-111, Beograd.
24	Mašović S., Saračević M., Kamberović H. (2011), Procesni pristup razvoja sistema za elektronsko učenje i UML modelovanje, <i>YUINFO 2011 – XVII međunarodna konferencija o računarskim naukama i informacionim tehnologijama</i> , Kopaonik.
25	Mašović S., Saračević M., Kamberović H. , Međedović E. (2011), Integrated management web-based applications on JAVA platform, <i>5th International quality conference, Center for Quality</i> , Volume 2 str.573-580, Serbia.
26	Mašović S., Saračević M., Kamberović H.(2010), Objektno-orijentisani pristup u simulaciji i metodologija simulacionog modeliranja, <i>Festival informatičkih dostignuća - INFOFEST 2010 Budva</i> , pp.118-129, Crna Gora.
27	Mašović S., Saračević M.,Kamberović H., Međedović E. (2011), Modern trends in higher education and the future of e-learning, <i>ITRO-conference 2011</i> , Zrenjanin (rad je prihvaćen).
28	Međedović E., Saračević M., Mašović S. (2011), Infrastruktura sistema za e-učenje Univerziteta u Novom Pazaru, <i>X međunarodni naučno-stručni Simpozijum INFOTEH - Jahorina</i> , Vol. 10, Ref. E-V-21, p. 842-845, March 2011.
29	Milosavljević B., Vidaković M., <i>Java i Internet programiranje</i> , Praktikum, Novi Sad, 2001.
30	Mitrović B., Devetakovic M., Petruševskic Lj.(2007), Integrisanje eksplorativnih alata u e-learning sisteme primena u oblasti arhitektonske geometrije (java view), <i>YuInfo2010</i> , r.br. 068, 77-84.
31	Morales R. A., (2009), Evaluation of Mathematics Achievement Test: A Comparison between CTT and IRT, <i>The International Journal of Educational and Psychological Assessment</i> , Vol. 1, Issue 1, pp. 19-26
32	Moreno, R., Mayer, R. (2007). Interactive multimodal learning environments: Special issue on interactive learning environments: Contemporary issues and trends. <i>Educational Psychology Review. Special Issue: Interactive Learning Environments: Contemporary Issues and Trends</i> , 19(3), 309–326.
33	Namestovski Ž. (2008), <i>Uticao primene savremenih nastavnih sredstava na povećanje efikasnosti nastave u osnovnoj školi</i> , Magistrarski rad, Univerzitet u novom sadu, Tehnički fakultet „Mihajlo Pupin”, Zrenjanin.
34	Pavlović V., Dragičević S., Papić Ž. (2010.), Metodologija primene apleta i animacija u nastavi Tehničkog i informatičkog obrazovanja, <i>Tehnika i informatika u obrazovanju</i> , 3. Internacionalna Konferencija, Tehnički fakultet Čačak.
35	Polya, G. (2003), <i>Matematičko otkriće</i> , HMD, Zagreb.
36	Polya G. (1957), <i>How to Solve It</i> , 2nd ed., Princeton University Press.
37	Popović N., Naumovic M. (2010), Razvoj interaktivnih simulacija pomoću easy java simulations i Simulink modela kao eksterne aplikacije, <i>INFOTEH-JAHORINA</i> Vol. 9, Ref. A-4, p. 22-25.
38	Radosav D. (2005.): <i>Obrazovni računarski softver i autorski sistemi</i> . Tehnički fakultet “Mihajlo Pupin”, Zrenjanin.

39	Saračević M., Međedović E., Ahmeti J., Mustafić F. (2011), Comparative analysis of success studying at the department of computer science in the traditional way and on the internet, <i>ITRO-conference 2011</i> , Zrenjanin (rad je prihvaćen).
40	Saračević M., Milošević D., Međedović E., Novalić F. (2011), Neki predlozi unapređenja nastave matematike primenom alata za e-učenje, <i>Tehnologija, informatika i obrazovanje za društvo učenja i znanja</i> , 6. Međunarodni simpozijum, Tehnički Fakultet Čačak (rad je prihvaćen).
41	Saračević M., Hadžiahmetović A., Milošević D., Međedović E. (2011), Apleti u nastavi kao podrška u funkciji motivacije talentovanih učenika, <i>Tehnologija, informatika i obrazovanje za društvo učenja i znanja</i> , 6. Međunarodni simpozijum, Tehnički Fakultet Čačak (rad je prihvaćen).
42	Saračević M., Mašović S.(2010), Primena UML modelovanja i PHP jezika u izradi web aplikacije za e-učenje, <i>Univerzitet METROPOLITAN - Elektronsko učenje na putu ka društvu znanja 2010</i> , vol1., str. 131-136, Beograd.
43	Saračević M. (2011), <i>Objektno-orijentisano programiranje i modelovanje – JAVA i UML</i> , monografija, Univerzitet u Novom Pazaru.
44	Saračević M., Mašović S., Međedović E.(2011), Infrastruktura za realizaciju i razvoj e-učenja u obrazovnom sistemu, <i>YUINFO 2011 – XVII međunarodna konferencija o računarskim naukama i informacionim tehnologijama</i> , Kopaonik.
45	Sasikumar M. (2008), <i>Moodle Your Way to Elearning</i> , CDAC Mumbai.
46	Street, S., Goodman, A. (1998). <i>Some experimental evidence on the educational value of interactive Java applets in Web-based tutorials</i> , Proceedings of the ACM.
47	Turoff M. (1995), <i>Designing a Virtual Classroom</i> , March 7-10.
48	Venkataraman R. (2001), <i>Java and its supporting technologies, architectures for e-business systems</i> , auerbach publications.
49	Vilotijević M. (1999.), <i>Od tradicionalne ka informacionoj didaktici</i> , Pedagoško društvo Srbije, Beograd.
50	Weiss, R. E., Knowlton, D. S., & Morrison, G. R. (2002). Principles for using animation in computerbased instruction: Theoretical heuristics for effective design, <i>Computers in Human Behavior</i> , 18, 465–477.
51	Wolfram S. (1996.), <i>MATHEMATICA Book</i> , Version 3.0, Addison-Wesley Publishing, Co, Redwood City, California,

Internet izvori - linkovi:

L1	Klasifikacija alata za e-učenje	http://e-lab.tfc.kg.ac.rs/moodle/mod/book/view.php?id=5102
L2	Vrste sistema za e-učenje	http://e-lab.tfc.kg.ac.rs/moodle/mod/book/view.php?id=6236
L3	Razlozi za korisničku podršku	http://e-lab.tfc.kg.ac.rs/moodle/mod/resource/view.php?id=6486
L4	Ocenjivanje u e-učenju	http://e-lab.tfc.kg.ac.rs/moodle/mod/resource/view.php?id=6319
L5	Virtual Interactive Classroom	http://www.campus-technology.com/article.asp?Id=11046
L6	Modules and plugins,	http://moodle.org/mod/data/view.php?Id=6009

Lista priloga

- P1. Izgled kursa za učenike
- P2. Struktura kursa za nastavnike

Lista izvornih kodova u Javi

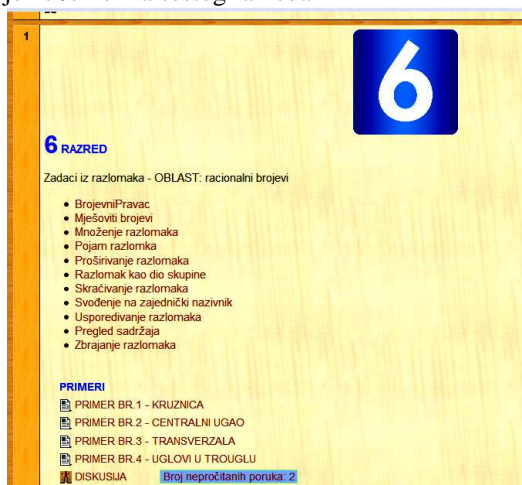
- S1. Primer 1 - aplet u matematici, rad sa grafičkim elementima
- S2. Primer 2 - aplet u matematici, rad sa događajima
- S3. Primer upitnika u obliku java apleta – Forma1
- S4. Primer upitnika u obliku java apleta – Forma2

P1. Izgled kursa za učenike

Naslovna strana kursa, ime kursa i autor



Deo kursa koji je namenjen učenicima šestog razreda



Deo kursa koji je namenjen učenicima sedmog razreda



Deo kursa koji je namenjen učenicima osmog razreda

3



8 RAZRED

PRIMERI - OBLAST FUNKCIJE

- Eksponencijalna funkcija
- Elementarne funkcije
- Stroj funkcija
- Graf kvadratne funkcije
- Rješavanje kvadratne nejednadžbe
- Polinomi
- Opća potencija
- Praktične Primjene Funkcija
- Tijek funkcije
- Funkcije

PRIMERI

- 📄 PRIMER BR.1 - CETVOROUGAO
- 📄 PRIMER BR.2 - upisana kruznica - CETVOROUGAO
- 📄 PRIMER BR.3 - GEOMETRIJSKI OBLICI
- 📄 PRIMER BR.3 - GEOMETRIJSKI OBLICI-2
- 🗨️ DISKUSIA [Broj nepročitalih poruka: 2](#)

Deo kursa koji se odnosi na kolaborativan rad učenika (radionice, diskusije) *

Deo kursa koji se odnosi na java applete (osnovni i napredni nivo), namenjen nastavnicima **

4



RADIONICE

- 🗨️ TIMSKI RAD - 6 R.
- 🗨️ TIMSKI RAD - 7 R.
- 🗨️ TIMSKI RAD - 8 R.

5



DEO ZA NASTAVNIKE - OSNOVNI NIVO
Java appleti

- 🗨️ Java appleti - osnovni koncepti
- 📄 Java appleti - deo skripte
- 📄 Java appleti sa primerima - prezentacija
- 🔍 Analiza Java appleta
- 🗨️ Primena appleta u nastavi
- 🗨️ Rečnik pojmova - Java appleti

6



DEO ZA NASTAVNIKE - NAPREDNI NIVO
Java appleti i java aplikacije.

- 📄 Applets vs. Applications - 1
- 📄 Applets vs. Applications - 2
- 🗨️ Praktican rad - video tutorijal za Java applete

P2. Struktura kursa za nastavnike

Ciljevi

- *kvalitetno izvođenje nastavnog procesa upotrebom besplatnog softvera i apleta,*
- *da se olakša priprema nastave i da se pojednostave nastavni sadržaji;*
- *da se učenicima i nastavnicima približe procesi u prirodnim naukama i tehnici,*
- *upoznavanje nastavnika sa mogućnostima i načinom kreiranja časa pomoću interaktivnih sadržaja - apleta, simulacija, animacija itd.*
- *upoznavanje nastavnika sa metodičkim i didaktičkim osobinama teorijske i laboratorijske nastave koja je podržana JAVA apletima,*
- *pronalaženje, čuvanje i obrada apleta,*
- *korišćenje nekih gotovih besplatnih programa koji su namenjeni nastavi predmeta iz prirodnih nauka.*

Teme

1. Čas

- Nastava u virtuelnoj učionici, upoznavanje sa pojmom simulacija,
- interaktivni sadržaji i apleti (*Zašto i kako se mogu apleti primeniti u nastavi*).

2. Čas

- Kako izabrati pravi aplet u nastavi, domet apleta (*S kim aplet može komunicirati*),
- Parametri i atributi apleta. Ubacivanje apleta u WEB-stranicu,

3. Čas

- Obrada apleta, prevođenje apleta.
- Demonstracija apleta iz raznih oblasti/predmeta,

4. Čas

- Demonstracija časa sa korišćenim apletom,
- Upoznavanje sa gotovim programima i alatima.

IZVORNI KOD (source code)

S1. Primer 1 - Java aplet, rad sa grafičkim elementima

```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
```

```
// Autor apleta : Muzafer Saracevic
```

```
public class MojAplet extends Applet implements ActionListener, MouseListener,  
KeyListener{
```

```
    Button tri=new Button("tri");
    Button cetiri=new Button("cetiri");
    Button sest=new Button("sest");
        Color nova_boja=(Color.blue);
        int aktivno=1;
    int tri_aktivno=0;
    int cetiri_aktivno=0;
    int sest_aktivno=1;

    public void mouseExited(MouseEvent e){ }
    public void mouseEntered(MouseEvent e){ }
    public void mouseReleased(MouseEvent e){ }

    public void mousePressed(MouseEvent e){
        int x=e.getX(); int y=e.getY();
        if(x>60 && x<150 && y>100 && y<220)
            if(aktivno==1) aktivno=0;
            else aktivno=1;
            repaint();
    }

    public void mouseClicked(MouseEvent e){ }
    public void keyReleased(KeyEvent e){ }
    public void keyTyped(KeyEvent e){ }

    public void keyPressed(KeyEvent e){
        switch (e.getKeyChar()){
            case 'b':nova_boja=Color.blue;break;
            case 'B':nova_boja=Color.blue;break;
```

```
        case 'y':nova_boja=Color.yellow;break;
        case 'Y':nova_boja=Color.yellow;break;
        case 'G':nova_boja=Color.green;break;
        case 'g':nova_boja=Color.green;break;}
        repaint();
    }

public void init() {
    tri.addActionListener(this);
    cetiri.addActionListener(this);
    sest.addActionListener(this);

    add(tri);
    add(cetiri);
    add(sest);

    this.addMouseListener(this);
    this.addKeyListener(this);
}

public void paint(Graphics g) {
    g.drawString("GEOMETRIJSKA TELA",325, 50 );
    g.drawRoundRect(320,30,140,40,12,12);
    g.drawRect(360,130,140,140);
    g.drawOval(60,60,150,150);
    g.setColor(nova_boja);

    if(aktivno==0)
        g.fillOval(100,100,70,70);else g.fillOval(60,60,150,150);
    int niz1[] = { 100,300,350,300,100,50};
    int niz2[] = { 20,20,170,300,300,170};

    if(tri_aktivno==1)g.drawPolygon(niz1,niz2,3);
        else if(cetiri_aktivno==1)g.drawPolygon(niz1,niz2,4);
        else if(sest_aktivno==1)g.drawPolygon(niz1,niz2,6);
}

public void actionPerformed(ActionEvent e){
    if(e.getSource()==tri) {
        tri_aktivno=1;
        cetiri_aktivno=0;
        sest_aktivno=0;
    }
    else
```

```
if(e.getSource()==cetiri){
    tri_aktivno=0;
    cetiri_aktivno=1;
    sest_aktivno=0;
}
else if(e.getSource()==sest){
    tri_aktivno=0;
    cetiri_aktivno=0;
    sest_aktivno=1;
}
    repaint();}
}
```

S2. Primer 2 – Java aplet, rad sa događajima

```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
```

```
// Autor apleta : Muzafer Saracevic
```

```
public class Prvi extends Applet implements ActionListener,MouseListener,KeyListener{
```

```
    Button JKT=new Button("P_190");
    Button JST=new Button("P_240");
    Button RST=new Button("P_290");
```

```
    int Dimenzija_poligona=6;
    int aktivno=1;
    int j_kt=0;
    int j_st=0;
    int r_st=1;
```

```
    public void mouseExited(MouseEvent e){}
    public void mouseEntered(MouseEvent e){}
    public void mouseReleased(MouseEvent e){}
    public void mousePressed(MouseEvent e){
        int x=e.getX();
        int y=e.getY();
        if(x>60 && x<150 && y>100 && y<220)
            if(aktivno==1) aktivno=0;
```

```
        else aktivno=1;
        repaint();
    }
    public void mouseClicked(MouseEvent e){ }
    public void keyReleased(KeyEvent e){ }
    public void keyTyped(KeyEvent e){ }
    public void keyPressed(KeyEvent e){

switch (e.getKeyChar()){
    case '4':Dimenzija_poligona=4;break;
    case '5':Dimenzija_poligona=5;break;
    case '6':Dimenzija_poligona=6;break;
    case '0':Dimenzija_poligona=0;break;}
        repaint();
    }

public void init() {
    JKT.addActionListener(this);
    JST.addActionListener(this);
    RST.addActionListener(this);
    add(JKT);add(JST);add(RST);
    this.addMouseListener(this);
    this.addKeyListener(this);
}

public void paint(Graphics g) {
    g.drawRect(360,130,140,180);
    g.drawString("GRAFICKI ELEMENTI",325, 50 );
    g.drawRoundRect(320,30,140,40,12,12);
    g.drawRect(60,60,150,150);
    g.fillOval(60,60,150,150);
    if(aktivno==0)
        {g.setColor(Color.red);
        g.drawString("PLAY",160,160);}
    else    g.fillOval(60,60,150,150);
    int P1[] = { 100,300,350,300,100,50};
    int P2[] = { 20,20,170,300,300,170};
    if(j_kt==1)                g.translate(50,190);
    else if(j_st==1)          g.translate(50,240);
    else if(r_st==1)          g.translate(50,290);

    if(Dimenzija_poligona==4)
    g.drawPolygon(P1,P2,4);
    else if(Dimenzija_poligona==5)
    g.drawPolygon(P1,P2,5);
```



```
else if(Dimenzija_poligona==6)
g.drawPolygon(P1,P2,6);
else if(Dimenzija_poligona==0)
g.clearRect(60,60,150,150);}
```

```
public void actionPerformed(ActionEvent e){
    if(e.getSource()==JKT)
    {j_kt=1;j_st=0;r_st=0;}
    else if(e.getSource()==JST)
    {j_kt=0;j_st=1;r_st=0;}
    else if(e.getSource()==RST){
    j_kt=0;j_st=0;r_st=1;}
    repaint();}
}
```

S3. Primer upitnika u obliku java apleta – formal

```
import java.awt.event.*;
import javax.swing.*;
```

```
// Autor apleta : Muzafer Saracevic
```

```
public class Quiz1 extends Applet implements ActionListener{
```

```
    GridLayout glavna=new GridLayout(4,2);
    Panel p=new Panel();
    Label Naslovna =new Label("PROGRAMSKI JEZICI")
        Checkbox prvi=new Checkbox("java",true);
        Checkbox drugi=new Checkbox("c", false);
        Checkbox treci=new Checkbox("c++", false);
        Checkbox cetvrti=new Checkbox("pascal", false);

        Button nazad=new Button("Nazad");
        Button napred=new Button("Napred");
        TextArea Komentar = new TextArea(10,10);
        Panel panel_podaci=new Panel();

        GridLayout mreza_podaci=new GridLayout(3,2);
        Panel panel_check=new Panel();
        BorderLayout BL=new BorderLayout();

        Panel panel_pitanje=new Panel();
        GridLayout mreza_pitanje=new GridLayout(3,1);
```

```
Panel panel_radio=new Panel();
GridLayout mreza_radio=new GridLayout(2,1);

Panel panel_dugmad=new Panel();
GridLayout mreza_dugmad=new GridLayout(1,2);
Button prijava=new Button("prijava");
Label ime=new Label("prezime");
TextField ime_polje = new TextField();

Label prezime =new Label("prezime");
TextField prezime_polje=new TextField();
Label Sistem =new Label("sistem");
TextField sistem_polje=new TextField("administrator");

TextField password = new TextField(10);
Label Pitanje =new Label("Apleti su posebna vrsta Java programa koji su namenjeni za
ugradjivanje u :");

Button odg1=new Button("HTML stranice");
Button odg2=new Button("Pascal kodovi");
CheckboxGroup grupa=new CheckboxGroup();
Checkbox rb1=new Checkbox("web dizajn",grupa,true);
Checkbox rb2=new Checkbox("e-poslovanje",grupa,false);
int r_b1=1;
int r_b2=0;

public void init() {
    p.setLayout(glavna);
    add(Naslovna);add(panel_podaci);
    add(password);add(prijava);
    add(panel_pitanje);add(panel_check);
    add(panel_radio);add(panel_dugmad);
    panel_podaci.setLayout(mreza_podaci);
    panel_podaci.add(ime);
    panel_podaci.add(ime_polje);
    panel_podaci.add(prezime);
    panel_podaci.add(prezime_polje);
    panel_podaci.add(Sistem);
    panel_podaci.add(sistem_polje);
    password.setEchoChar('*');
    sistem_polje.setEnabled(false);
    odg1.addActionListener(this);
    odg2.addActionListener(this);
}
```

```

panel_check.setLayout(BL);
panel_check.add(prvi, BorderLayout.SOUTH);
panel_check.add(drugi, BorderLayout.NORTH);
panel_check.add(treci, BorderLayout.WEST);
panel_check.add(cetvrti, BorderLayout.EAST);
panel_check.add(Komentar, BorderLayout.CENTER);

```

```

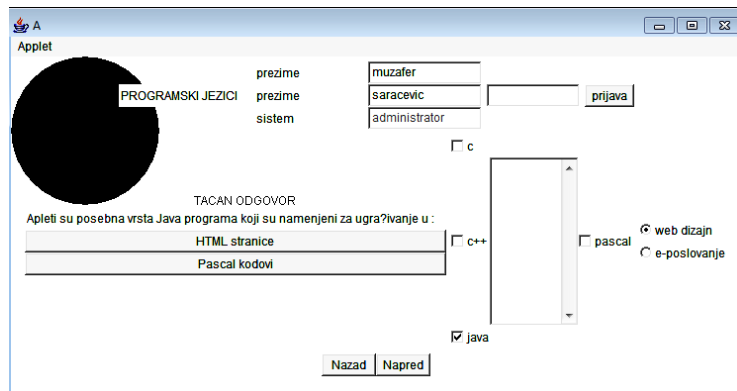
panel_radio.setLayout(mreza_radio);
panel_radio.add(rb1); panel_radio.add(rb2);
panel_pitanje.setLayout(mreza_pitanje);
panel_pitanje.add(Pitanje);
panel_pitanje.add(odg1); panel_pitanje.add(odg2);
panel_dugmad.setLayout(mreza_dugmad);
panel_dugmad.add(nazad);
panel_dugmad.add(napred);

```

```

public void actionPerformed(ActionEvent e){
    if(e.getSource()==rb1)
        {r_b1=1;r_b2=0;}
    else
        if(e.getSource()==rb2){
            r_b1=0;r_b2=1;}
        repaint();
}
public void paint(Graphics g) {
    if(r_b1==1)
        g.drawString("TACAN ODGOVOR",185, 150);
    else if (r_b2==1)
        g.drawString("NETACAN ODGOVOR",185, 150);
    g.fillOval(0,0,150,150);
}
}

```



S4. Primer upitnika u obliku java apleta – forma2

```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import javax.swing.*;

// Autor apleta : Muzafer Saracevic

public class Quiz2 extends Applet implements ActionListener{

    GridLayout glavna=new GridLayout(3,3);
    Panel p=new Panel();
    Label Naslovna =new Label("PRIJAVA");

    CheckboxGroup grupa=new CheckboxGroup();
    Checkbox prvi=new Checkbox("prvi",grupa,true);
    Checkbox drugi=new Checkbox("drugi",grupa,false);
    Checkbox treci=new Checkbox("prvi",grupa,false);
    Checkbox cetvrti=new Checkbox("cetvrti",grupa,false);

    Button back=new Button("back");
    Button next=new Button("next");
    TextArea Komentar = new TextArea(10,10);

    Panel panel_prijava=new Panel();
    GridLayout mreza_prijava=new GridLayout(2,2);

    Panel panel_izbor=new Panel();
    GridLayout mreza_izbor=new GridLayout(3,1);
    Panel panel_pitanje=new Panel();
    GridLayout mreza_pitanje=new GridLayout(3,1);
    Panel panel_radio=new Panel();
    BorderLayout polarna=new BorderLayout();

    Panel panel_dugmad=new Panel();
    GridLayout mreza_dugmad=new GridLayout(1,2);

    Button prijava=new Button("prijava");
    Label username=new Label("username");
    TextField username_polje = new TextField();
    Label password =new Label("password");
    TextField password_polje=new TextField();
```

```
Button java=new Button("java");
Button applet=new Button("applet");
Button aplikacija=new Button("aplikacija");
```

```
Label Pitanje =new Label("ZA IZRADU GRAFICKOG INTEFEJSA
KORISTIMO PROGRAM:");
```

```
Checkbox odgovor1=new Checkbox("JAVA",true);
Checkbox odgovor2=new Checkbox("PASCAL",false);
int jav=1;int app=0;int apl=0;
```

```
public void init() {
```

```
p.setLayout(glavna);
```

```
add(Naslovna);
add(panel_prijava);
add(prijava);
```

```
add(Komentar);
add(panel_izbor);
add(panel_pitanje);
add(panel_radio);
add(panel_dugmad);
```

```
panel_prijava.setLayout(mreza_prijava);
panel_prijava.add(username);
panel_prijava.add(username_polje);
panel_prijava.add(password);
panel_prijava.add(password_polje);
```

```
password_polje.setEchoChar('*');
```

```
java.addActionListener(this);
applet.addActionListener(this);
aplikacija.addActionListener(this);
```

```
panel_izbor.setLayout(mreza_izbor);
panel_izbor.add(java);
panel_izbor.add(applet);
panel_izbor.add(aplikacija);
```

```
panel_pitanje.setLayout(mreza_pitanje);
panel_pitanje.add(Pitanje);
panel_pitanje.add(odgovor1);
panel_pitanje.add(odgovor2);

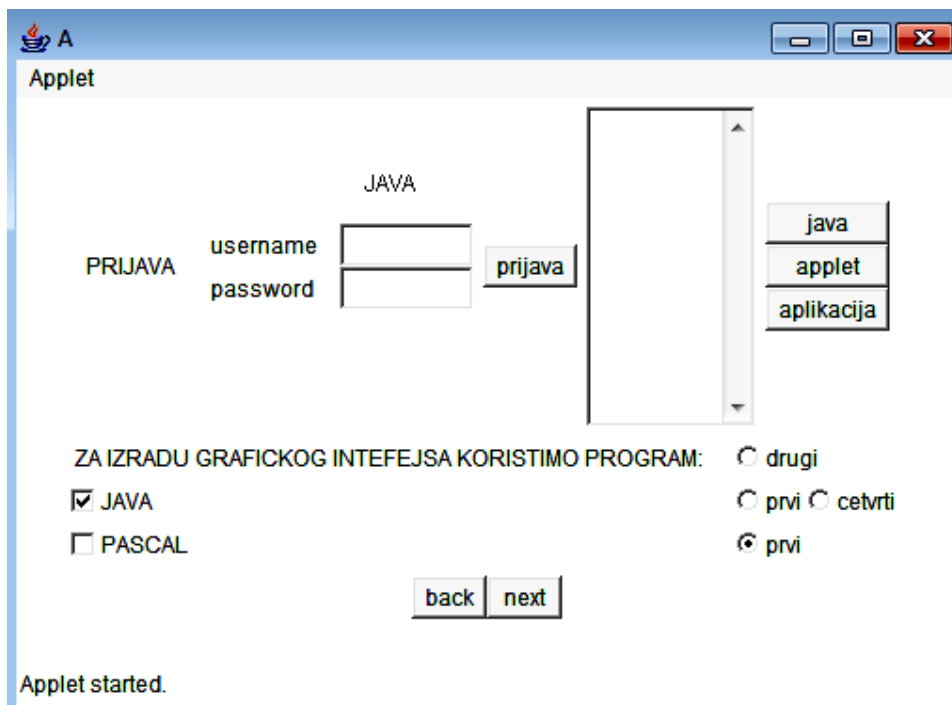
panel_radio.setLayout(polarna);
panel_radio.add(prvi, BorderLayout.SOUTH);
panel_radio.add(drugi, BorderLayout.NORTH);
panel_radio.add(treci, BorderLayout.WEST);
panel_radio.add(cetvrti, BorderLayout.EAST);

panel_dugmad.setLayout(mreza_dugmad);
panel_dugmad.add(back);
panel_dugmad.add(next);
}
```

```
public void actionPerformed(ActionEvent e){
    if(e.getSource()==java) {
        jav=1;
        app=0;
        apl=0;
    }
    else if(e.getSource()==applet){
        jav=0;
        app=1;
        apl=0;
    }
    else if(e.getSource()==aplikacija){
        jav=0;
        app=0;
        apl=1;
    }
    repaint();
}
```

```
public void paint(Graphics g) {
```

```
    if(jav==1)                g.drawString("JAVA",185, 50 );  
    else if (app==1) g.drawString("APLET",185, 50 );  
    else if (apl==1) g.drawString("APLIKACIJA",185, 50 );  
    }  
  
}
```



Lista slika

Slika 1 – aplet sa nizom Geometrijskih oblika	19
Slika 2 – okruženje JCreator-a	22
Slika 3 – Izbor kreiranja apleta u NetBeans-u	23
Slika 4 – aplet sa događajima na taster i na klik miša	24
Slika 4.1 – aplet sa mogućnošću menjanja pozicije oblika	24
Slika 5 – okruženje programa GeoGebra	25
Slika 5.1 – primer dinamičkog ažuriranja podataka u GeoGebri	26
Slika 6 - Radno okruženje alata Geometer's Sketchpad	29
Slika 6.1 - Jednostavan primer u SketchPad-u i Motion Controller	30
Slika 6.2 - Način realizacije i implementacija apleta u html editor	30
Slika 7 – editor Wiris alata	31
Slika 8 – Wiris u obliku onlajn okruženja	31
Slika 9 – Wiris alat za izradu kvizova	32
Slika 10 – Wiris - Bela tabla	32
Slika 11 – glavni odeljci Wiris alata	32
Slika 12 – mogućnost snimanja Wiris sadržaja	32
Slika 13. Korisnički interfejs Easy Java Simulations	33
Slika 13.1 Deo korisničkog interfejsa Easy Java Simulations	34
Slika 14. Primer apleta u Easy Java Simulations	34
Slika 15 - rad sa 3D grafikom u alatu Mathematica	35
Slika 16. - demonstracija izračunavanja rastojanja tačaka u koordinatnom početku	36
Slika 17– implementacija Mathematice na smart telefone	36
Slika 18. Primer apleta i njegovih podešavanja u alatu Java View	37
Slika 19 - Primer integrisanja Java View okruženja u funkcionalnost Forum sistema za e-učenje	38
Slika 20 - rad sa 3d grafikom u Mathematici i izvoz u Applet	38
Slika 21 – Okruženje WIRIS Conference - math chat	39
Slika 22 – primer obrazovne igre za nastavu matematike u obliku apleta	40
Slika 23. Primeri edukativnih java igrica u vidu apleta namenjenih za nastavu matematike	40
Slika 24. Okruženje TWikiDraw	41
Slika 25. Primena Wiris alata u otvorenom tipu pitanja – esej sa kratkim odgovorom	45
Slika 26. Primer kako nastavnik unosi tačan odgovor za prethodni primer	46
Slika 27. Primena Wiris alata u otvorenom tipu pitanja – esej sa složenim odgovorom	46
Slika 28. Primer kako nastavnik unosi tačan odgovor za prethodni primer	46
Slika 29. – primer navođenja promenljive u Wiris editoru	46
Slika 31. Primer pitanja promenljivom	47
Slika 30. Primer definisanja algoritma za promenljivu	47
	101

Slika 32. Primer navođenja pitanja sa vise promenljivih	47
Slika 33. Primer definisanja algoritma za vise promenljivih	47
Slika 34. Primer pitanja sa višestrukim odgovorom	48
Slika 35. Ponuđeni odgovori za pitanje sa višestrukim odgovorom	48
Slika 36. Primena Wiris alata u zatvorenom tipu pitanja – pitanje sa visestrukim odgovorom	49
Slika 36.1 Prvi slučaj	49
Slika 36.2 Drugi slučaj	49
Slika 36.3 Treći slučaj	50
Slika 36.4 Provera znanja u SketchPad-u	50
Slika 37. primer testa odrađen u obliku Java apleta	51
Slika 38. primer kreiranog upitnika u obliku apleta	52
slika 39. Virtuelna laboratorija namenjena talentovanim učenicima za predmet matematika	55
Slika 40. Virtuelna laboratorija sa velikim brojem apleta za eksperimentisanje	55
Slika 41. aplet Virtual Classroom	56
Slika 42. Primer virtuelne laboratorije na http://mathandmultimedia.com	57
Slika 43. Načini implementacije apleta u Moodle okruženje	59
Slika 44. Primer učitavanja apleta	60
Slika 45. Izgled editora posle instalacije Wiris plugin-a u Moodle sistemu	61
Slika 46. importovanje na resurse, tekstualna strana	62
Slika 47. mogućnost generisanja grafike u zavisnosti od sadržaja	62
Slika 48. import u rečnik pojmova	62
Slika 49. kviz	62
Slika 50. chat	62
Slika 51. Primer rada sa filterima u Moodle sistemu za e-učenje	63
Slika 52. GeoGebra editor u sistemu za e-učenje	63
Slika 53. Geogebra filter (Math Applets)	63

Lista tabela

Tabela 1 – lista alata za kreiranje interaktivnog sadržaja za e-učenje	10
Tabela 2 – Alati za ocenjivanje u e-učenju	43
Tabela 3 – primer upotrebe JavaScript naredbe i implementacije u Java apletima	45
Tabela 4 - Besplatni alati koji se mogu iskoristiti za razvoj virtuelnih učionica	54