

Online aplikacija za testiranje učenika za pripremu prijemnog ispita

Dragana Prokin, Dušan Čoko, Gabrijela Dimić, Slobodanka Đenić, Ana Savić, Boško Bogojević
Visoka škola elektrotehnike i računarstva strukovnih studija, Beograd, Srbija
dusan.coko@viser.edu.rs, gdemic@viser.edu.rs, dprokin@viser.edu.rs, bosko.bogojevic@viser.edu.rs

Sažetak— U ovom radu predstavljena je aplikacija koja omogućava budućim studentima da provere svoje znanje putem Interneta kako bi se adekvatno spremili za prijemni ispit koji se održava na Visokoj školi elektrotehnike i računarstva. U radu će biti opisane tehnologije koje su implementirane pri izradi aplikacije, statistički podaci koji su dobijeni na osnovu rezultata testiranja kao i opis interakcije korisnika sa aplikacijom..

Ključne riječi - e-učenje, obrazovanje, prijemni ispit, učenici srednjih škola

I. UVOD

Na tržištu informacionih tehnologija postoje različita komercijalna i besplatna softverska rešenja koja služe da generišu kvizove za čiju realizaciju poznavanje programskih jezika nije potrebno. Dok za predmete opšteg obrazovanja ovakav pristup zadovoljava osnovne zahteve pri proveri znanja, za predmete koji pripadaju prirodnim naukama, najpopularnije aplikacije nude da se tekstualni zapis kao što su formule uvoze kao slike ili flash elementi. Ove metode su zastarele i u vremenu kada sve više prenosni uređaji menjaju način na koji korisnici pristupaju internet sadržaju[1] neke od njih nisu upotrebljive.

LMS sistemi kao Moodle ili Blackboard već su adaptirali svoje interfejse kako bi bili mobile-ready ali ovo su sistemi čija sama struktura prelazi potrebe kviza koji može da sakuplja anonimne podatke. Ovakve vrste sistema zahtevaju registraciju korisnika za naprednije funkcionalnosti kvizova, što smanjuje nivo anonimnosti. Ako se i dozvoli pristup testovima u ulozi gosta, moduli za prikupljanje informacija moraju biti modifikovani ili je potrebno pronaći odgovarajuće rešenje u vidu spoljnih aplikacija koje služe za tu potrebu.

U skladu sa navedenim razmatranjima donesena je odluka da se kreira sopstvena aplikacija. Prethodni način pripreme za prijemni ispit bili su dostupni studentima u formi tekstualnih dokumenata na sajtu, sa određenim brojem pitanja iz informatike i matematike na osnovu kojih su se kandidati spremali za prijemni ispit. Sa implementacijom novih tehnologija u obrazovanju, odlučeno je da se u okviru postojećeg sajta doda sekcija za samostalno testiranje budućih studenata, tako što je dodata strana sa elektronskim testom, kome se lako pristupa sa sajta škole, pri čemu je broj pojedinačnih pristupa neograničen.

II. OBRAZLOŽENJE I ANALIZA ZAHTEVA APLIKACIJE

Test koji se polaže na prijemnom ispitu na Visokoj školi elektrotehnike i računarstva sastoji se od deset pitanja iz opšteg poznavanja pojmova iz informaciono-komunikacionih tehnologija i deset zadataka iz matematike. Svako pitanje ima identičnu formu koja se sastoji od:

- teksta pitanja
- četiri ponuđena odgovora

Među ponuđenim odgovorima, osim tačnog odgovora, nalaze se dva netačna, a kao četvrti je ponuđen odgovor „Ne znam“. Tačan odgovor donosi tri poena, netačni odgovori minus jedan, dok odgovor „Ne znam“ nosi nula poena. Pitanja formira komisija koja se imenuje za datu školsku godinu. Pitanja koja dolaze na testu su klasifikovana u dve glavne kategorije, od kojih svaka sadrži po deset podkategorija. Kandidati imaju sto dvadeset minuta za rešavanje zadataka.

Zahtevi javnog dela aplikacije su bili usmereni na kreiranje simulacije realnog okruženje za testiranje korisnika. Generisan je algoritam koji razvrstava pitanja u dve oblasti: matematika i informatika, a zatim kreira kategorije pitanja za svaku oblast, pri čemu je implementiran slučajan izbor pitanja iz svake kategorije. U cilju postizanje bolje fleksibilnosti ažuriranja sadržaja pitanja, omogućeno je unošenje matematičkih formula u formi teksta, a ne kao slika ili flash objekata. Ukoliko se korisnik testira više puta, pri svakom testiranju, generiše se različita kombinacija pitanja, bez mogućnosti višestrukog ponavljanja pitanja iz jedne iste kategorije.

Kreiran je administratorski interfejs zasnovan na CMS tehnologijama. Na taj način omogućen je jednostavan unos i izmena pitanja u bazi podataka. Administrator pri unosu bira oblast, a zatim kategoriju pitanja..

III. FRAMEWORK ZA APLIKACIJU

Za razvoj aplikacije implementiran je model troslojne arhitekture, što pruža mogućnost nezavisnog proširenja i unapređenja modula aplikacije. Implementirana arhitektura podrazumeva tri sloja: korisnički sloj, logički sloj i sloj baze podataka. Troslojna arhitektura rešava deo čestih problema vezanih za dizajn i razvoj, tako da je pristup razvoju olakšan a utrošeno vreme efektivnije. Korisniku je omogućen intuitivan način komunikacije sa aplikacijom, a logički sloj u pozadini obrađuje zahteve korisnika, kontroliše funkcionalnosti i

manipuliše podacima. Jedna od važnih funkcionalnosti je i asinhrona obrada rezultata, kako bi se onemogućio uvid u izvorni kod stranica aplikacije.

Model aplikacije po principima troslojne arhitekture prikazan je na slici (Sl. 1). Primjenjene su biblioteke JQuery i MathJax i MVC (Model-View-Controller). Korisnički interfejs predstavlja prvi sloj aplikacije koji upravlja prihvatanjem i prikazivanjem podataka. Zbog dostupnosti aplikacije na različitim uređajima, kao zajednička komponenta odabran je browser kao kontejner u okviru kog će biti prikazan interfejs testa. Korisnici mogu da pristupe korišćenjem bilo kog aktuelnog browser-a na tržištu. Korisnički interfejs zasnovan je na više komponenti za vizuelni prikaz: za strukturu dokumenta HTML5 standard, za stilizovanje stranice CSS3 standard, a kao jezik za interakciju na klijentskoj strani korišćen je Javascript. U cilju ostvarivanja što boljih performansi i ključnih strukturnih aspekata implementirane sledeće biblioteke: jQuery za interakciju korisnika sa testom i MathJax za konverziju Latex sadržaja u HTML kod. Biblioteke su open-source i dostupne su pod GPL (General Public License) licencom.

Na serverskoj strani podignut je Apache server otvorenog tipa koji je zastupljen na velikom broju veb platformi, a kao jezik za procesiranje podataka izabran je PHP. Za sistem za upravljanje bazama podataka izabran je MySQL DBMS. PHP biblioteka PDO (PHP Data Objects) služi za obradu zahteva iz logičkog sloja i baze podataka. PDO biblioteka omogućava keširanje upita [2] koji su iste strukture a različitim vrednostima. Ova funkcionalnost je važna zbog provere rezultata pitanja korisnika. Moguće je proveravati pitanje po pitanje bez da se troše serverski resursi na proveru pitanja. Specijalizovane korisničke klase koriste trajnu vezu koja je uspostavljena korišćenjem Singleton šablona [3] za slanje odgovora korisnika bazi podataka i na osnovu vraćenih rezultata formira se broj tačnih odgovora koje je dao korisnik. Broj odgovora „Ne znam“ se čuva u memoriji sesije nakon čega se sumira sa brojem tačnih odgovora. Broj netačnih odgovora je preostali broj pitanja iz testa.

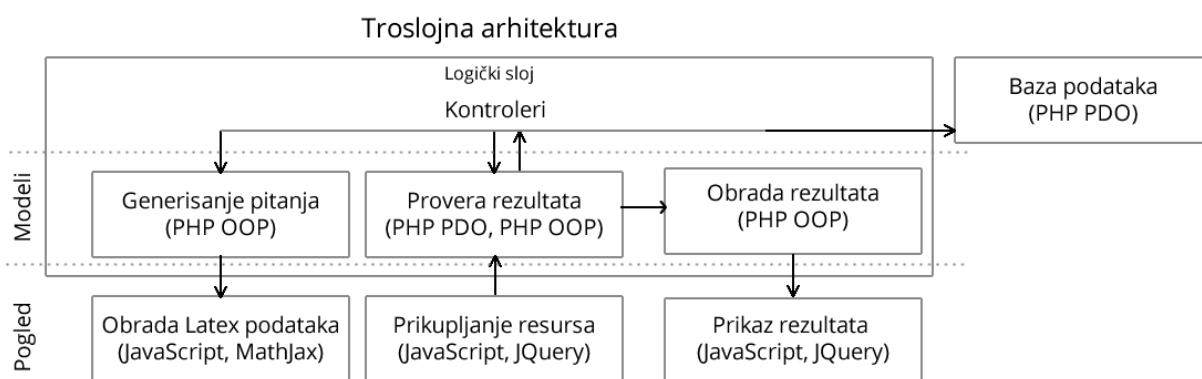
Aplikacija implementira MVC model, gde je glavna klasa "Test" koja kreira određenu kombinaciju. Kombinacija (kontroler i model) generiše pitanja za zadati broj kategorija, gde se pitanja dohvataju na osnovu glavne kategorije, a zatim se nasumično bira pitanje na osnovu podkategorije. U okviru testova dato je deset podkategorija i svaka podkategorija obeležena je rednim brojem. Na osnovu broja podkategorija redom se uzima indeks podkategorije, a na osnovu microtime funkcije operativnog sistema bira se nasumično pitanje. U cilju postizanja nasumičnog redosleda pitanja implementirana je php funkcija array_shuffle.

Pitanja za kategoriju "Matematika" imaju izraze koji se ne mogu predstaviti tekстом. Komercijalni alati za kreiranje testova kao Articulate Quiz maker, Wondershare QuizCreator, iSping i drugi koji omogućavaju korisnicima kreiranje kvizova bez znanja programiranja, koriste posebne alate za prikaz matematičkih formula. Moodle LMS od verzije 2. je uveo podršku za predstavljanje Latex izraza [3] kao vizuelni sadržaj za korisnike.

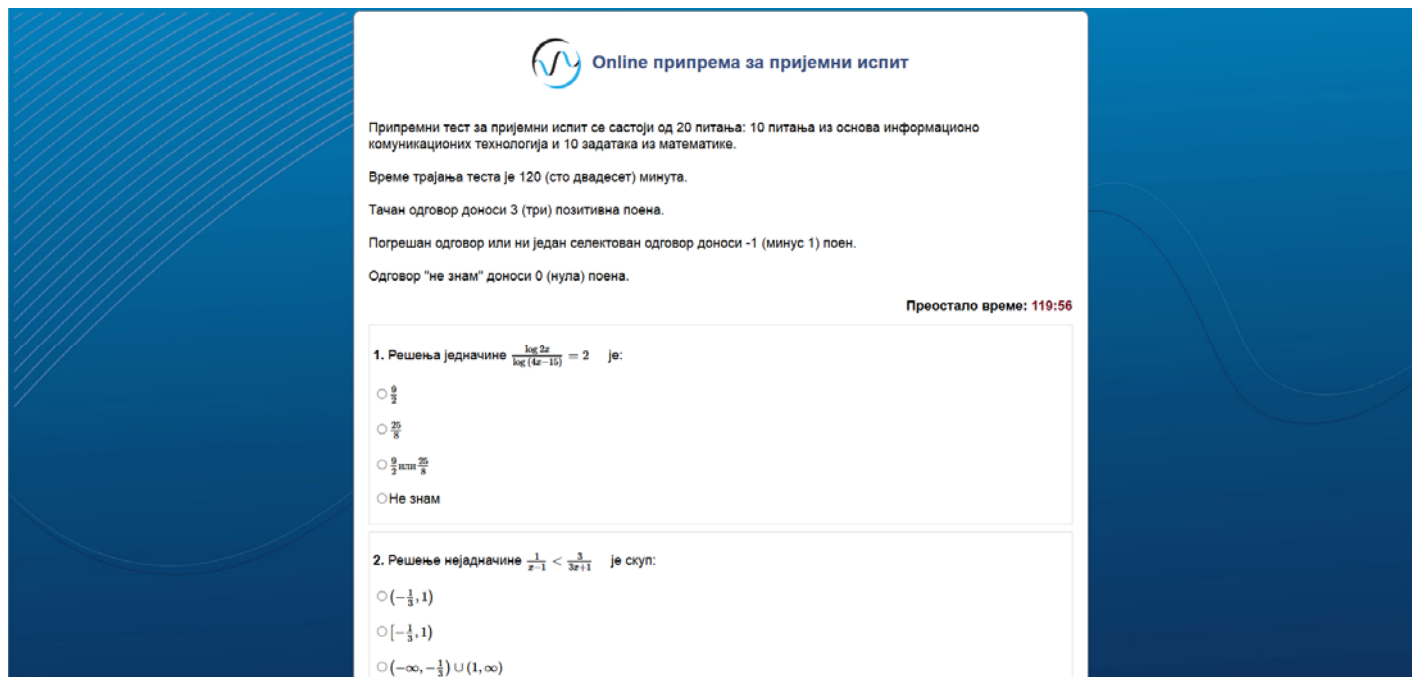
MathJax[4] je open-source Javascript biblioteka koja prevodi sadržaj na stranici u MathML kôd ukoliko pretraživač podržava standard odnosno menja Latex sa SVG grafikom ukoliko pretraživač ne podržava MathML. MathML standard je definisani markup standard koji Latex prevodi u sadržaj koji pretraživač može da interpretira u vizuelni prikaz koji prosečni korisnik može da razume. Kreirana je funkcija koja čeka da MathJax prevedu sadržaj i da se na kraju sadržaj prikaže. U zavisnosti od pretraživača i uređaja sa kog korisnik pristupa varira vreme odziva pa je postavljen i loader koji signalizira učitavanje sadržaja.

IV. OPIS APLIKACIJE

Aplikacija radi sa dva tipa korisnika: administratorom i kandidatom. Administrator može da unese ili ažurira pitanja u bazi podataka. On unosi Latex tekst za pitanja koja imaju matematičku formulu i tri odgovora, a odgovor „Ne znam“ se generiše u aplikaciji za prikaz sadržaja testa.



Slika 1. Prikaz MVC modela primenom principa troslojne arhitekture



Slika 2. Prikaz testa na dekstop računaru

Korisnik pristupa stranici za testiranje (Sl. 2) gde se prikazuje sadržaj. Sadržaj stranice je skalabilan i sadržaj je vidljiv na dekstop računarima kao i na prenosnim uređajima. Korisniku su napisana pravila bodovanja testa i vreme koje mu je preostalo da završi testiranje. Svako pitanje ima četiri ponuđena odgovora koja su predstavljena radio dugmadima. Na kraju testa nalazi se dugme za predaju testa. Klikom na dugme, korisniku se pojavljuje dijalog prozor za potvrdu da li zaista želi da preda test. Davanjem potvrdnog odgovora aplikaciji se šalje JSON string u formatu pitanjeId: odgovor. PitanjeId je id pitanja iz baze, a ne redni broj pitanja iz teksta. Id pitanja se nalazi u kôdu i nije vidljiv korisniku testa. Na serverskoj strani izvršava se sumiranje poena i korisniku se vraća prikaz osvojenih poena kao i informacije o tome koliko je tačnih odgovora dao na pitanja iz matematike, a koliko iz informatike. Rezultati o uspehu korisnika se paralelno snimaju u bazu podataka, kako bi se mogli generisati statistički izveštaji.

U periodu od 1.6.2015. do 1.7.2015. kada je postojalo najveće interesovanje učenika oko upisa na visoke škole i fakultete, prema mehanizmu za praćenje, ostvareni su rezultati dati u tabeli 1.

Ukupan broj pokušaja:	31 575
Ukupan broj pokušaja sa jedinstvenih IP adresa:	2 753
Prosečan broj pokušaja po danu:	976
Maksimalan broj pokušaja po danu (slika 4):	7067 (1.7.2015.)
Prosečan broj poena za	8.21

pitanja iz opšteg poznavanja informatike:	
Prosečan broj poena iz matematike:	4.36

TABELA I. STATISTIKE POSTIGNutih USPEHA

Ovi podaci daju opšte podatke o interesovanju učenika za ustanovu i kvalitet njihovog znanja.

V. ZAKLJUČAK

Cilj aplikacije opisane u radu je da se budućim potencijalnim studentima omogući na što realniji način simulacija polaganja testa za prijemni ispit. Naredni korak je usmeren u pravcu razvoja aplikacije kako bi se omogućilo marketing timovi i nastavnici prikupljanje podataka od značaja a da se pri tome sačuva privatnost korisnika aplikacije. Trenutni podaci koje aplikacija sakuplja su od opšteg karaktera.

Razvoj i unapređivanje aplikacije biće usmereno ka generisanju izveštaja vezanim za tačnost datih odgovora, izdvajanje podataka o završenim srednjim školama, kao i o interesovanju za određene studijske programe visokoškolske ustanove.

LITERATURA

- [1] <http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/>
- [2] <http://php.net/manual/en/pdo.connections.php>
- [3] <http://www.oodesign.com/singleton-pattern.html>
- [4] https://docs.moodle.org/23/en/Algebra_filter
- [5] <https://www.MathJax.org/>

ABSTRACT

In this paper application for entry exam test has been described. It allows students to test their knowledge in order to be fully prepared for entry exam which School of Engineering and Computer Science holds as yearly event. In this paper technologies which were implemented were described, as well as description of interaction between users and application, and statistical data.

ONLINE APPLICATION FOR TESTING PUPILS FOR ENTRY EXAM

Dragana Prokin, Dusan Coko, Gabrijela Dimic, Ana Savic,
Slobodanka Djenic, Bosko Bogojevic