KRIŽALJKE

Bičvić, Leopold

Undergraduate thesis / Završni rad

2021

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:251595

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-06**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA

Sveučilišni studij

KRIŽALJKE

Završni rad

Leopold Bičvić

Osijek, 2020

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

Obrazac Z1P - Obrazac za ocjenu završnog rada na preddiplomskom sveučilišnom studiju

Osijek, 01.09.2020.

Odboru za završne i diplomske ispite

Prijedlog ocjene završnog rada na preddiplomskom sveučilišnom studiju

Ime i prezime studenta:	Leopold Bičvić
Studij, smjer:	Preddiplomski sveučilišni studij Računarstvo
Mat. br. studenta, godina upisa:	R4036, 24.09.2019.
OIB studenta:	42922753826
Mentor:	Izv. prof. dr. sc. Alfonzo Baumgartner
Sumentor:	
Sumentor iz tvrtke:	
Naslov završnog rada:	Križaljke
Znanstvena grana rada:	**Informacijski sustavi (zn. polje računarstvo)**
Predložena ocjena završnog rada:	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene mentora:	01.09.2020.
Datum potvrde ocjene Odbora:	23.09.2020.

Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O ORIGINALNOSTI RADA

Osijek, 23.09.2020.

Ime i prezime studenta:	Leopold Bičvić
Studij:	Preddiplomski sveučilišni studij Računarstvo
Mat. br. studenta, godina upisa:	R4036, 24.09.2019.
Turnitin podudaranje [%]:	7

Ovom izjavom izjavljujem da je rad pod nazivom: **Križaljke**

izrađen pod vodstvom mentora Izv. prof. dr. sc. Alfonzo Baumgartner

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD 1
1.1 Zadatak završnog rada 1
2. UPOTREBLJENE TEHNOLOGIJE 2
2.1 Html 2
2.2 Css 2
2.3 JavaScript 3
3. ZAPIS I DINAMIČNOST 4
3.1 Zapis križaljke u datoteku 4
3.2 Učitavanje datoteke 5
3.3 Dinamičko stvaranje elemenata 6
4. UPRAVLJANJE I PROVJERA PODATAKA 7
4.1 Obuhvaćanje elemenata 7
4.2 Unos podataka u elemente 7
4.3 Postavljanje uvjeta nad elementima 8
4.4 Gumb s događajem 9
4.5 Funkcija provjere podataka 9
5. KORISNIČKI POGLED APLIKACIJE 11
5.1 Pristup i opcije odabira 11
5.2 Načini rješavanja križaljke 12
5.3 Prikaz točnosti unosa 13
5.4 Izgled gotovih križaljki 14
6. ZAKLJUČAK 17
LITERATURA: 18
SAŽETAK 19
ABSTRACT 20
ŽIVOTOPIS 21
PRILOZI 22

1. UVOD

Razvijanjem ljudskog uma sama ideja križaljki razvijala se zajedno s tiskanim novinama. S obzirom na potrebe ljudi novinska križaljka je služila za zabavu i razvitak. Naime, razvitkom same tehnologije i globalizacije potrebno je prilagoditi se civilizaciji i njihovom napretku pa je tako i nastala ideja digitalnih novinskih križaljki. Kako bi se olakšalo korištenje, a ujedno i pazilo na prirodu, ideja križaljki je provedena putem digitalizacije odnosno web aplikacije koja je dostupna skoro svima. Tehnologije korištene u izradi web aplikacije Križaljke su HTML, CSS i JavaScript. Svaka navedena tehnologija pridonijela je ostvarivanju web aplikacije kako bi se izgled novinske križaljke prilagodio današnjici. Problem digitalizacije križaljke je razrađen na više dijelova, počevši zapisom same križaljke, učitavanjem u određenoj tehnologiji te daljnjom provedbom križaljke u realnost i funkcionalnost. Struktura završnog rada sadrži više dijelova. Na početku je objašnjena ideja i početci praktičnog rada, korištenim tehnologijama te provedbom i opisom funkcionalnosti web aplikacije. Na kraju je iznesen osvrt na cjelokupan rad, rezultati rada i mogućnosti daljnjeg razvoja aplikacije.

1.1 Zadatak završnog rada

Definirati zapis kojim se u računalu može zapisati novinska križaljka. Napraviti aplikaciju koja je u stanju učitati ovakav zapis i omogućiti korisniku rješavanje križaljke. Na kraju rješavanja provjeriti križaljku i povratno prikazati korisniku koliko je bio uspješan u rješavanju.

2. UPOTREBLJENE TEHNOLOGIJE

U ovom poglavlju su opisane upotrebljene tehnologije za izradu web aplikacije Križaljke. To su opisni jezik HTML, jezik za uređivanje CSS i programski skriptni jezik JavaScript.

2.1 Html

HyperText Markup Language ili HTML je opisni jezik kojim se određuje temelj web stranice.[1] HTML-om se prikazuju elementi web stranice kao što su npr. naslov, poveznica i slično.[1] Svaki HTML element sadrži svoju HTML oznaku.[1] HTML dokument sadrži tekst koji preko preglednika jasno i jednostavno pregledom se dobije uvid u način izrade dokumenta.[2] Specifičnost HTML-a je mogućnost ugnježđivanja elemenata.[1] Povezanost preglednika s HTML kodom je ostvarena na način da preglednik prikazuje cijeli HTML sadržaj unutar HTML oznaka u smislenom obliku.[1] Primjer jednog HTML elementa je odlomak koji se označava sa slovom p unutar znakova manje od i veće od   te se između zagrada nalazi prostor za odgovarajući tekst odlomka.[1] Slikom 2.1 prikazan je primjer odlomka.

```
<p> Tamo, tamo daleko, živio je jedan mladić</p>
```

Slika 2.1 *Primjer odlomka*

2.2 Css

CSS ili *Cascading Style Sheets* je jezik za uređivanje oblika web stranice.[3] Elementi CSS-a pišu se u posebnim datotekama kako bi se lakše manipuliralo kodom te uključivanje CSS-a u HTML dokument učinilo jednostavno.[3] Uređivanje elemenata stranice CSS-om moguće je opisati kao slojevito pozicioniranje i popunjavanje nedostataka HTML-a. Nasljeđivanje svojstava je moguće te je prisutno najviše među oblikovanjem teksta.[4] Slikom 2.2 prikazana je promjena visine i širine slike.

```
img{

    height: 50%;
    width: 50%;

  }
```

Slika 2.2 *Promjena visine i širine slike*

Prikazan primjer CSS elementa slikom 2.2 je mijenjanje visine i širine određenog HTML elementa odnosno slike. To se postiže tako da se navede oznaka elementa *img* te da se u vitičastim zagradama navedu atributi *height* i *width*.[3] Nakon toga, znak dvotočke i željenu veličinu određenu u npr. postotcima ili pikselima.[3]

2.3 JavaScript

JavaScript je programski skriptni jezik čiji je zadatak razvoj dinamičkih web stranica.[5] Jezgru JavaScript-a podržavaju mnogi trenutni preglednici poput Google Chrome-a i Mozilla Firefox-a.[5] Funkcionalnosti zbog kojih JavaScript nudi moderan izgled web stranice su interakcija s korisnicima, brze provjere, dinamičnost elemenata i slično.[5] Interakcija s korisnicima vrši se kroz događaje koji upotpunjuju funkcionalnost interakcije.[5] Događaji pridonose funkcionalnosti na način povezivanja HTML elemenata s JavaScriptom.[5] JavaScript podržava različite stilove programiranja.[5] Zbog toga se naziva i više paradigmatskim jezikom.[5] Time JavaScript pruža fleksibilnost korisniku.[5] JavaScript izvršava se u pregledniku.[5] JavaScript također omogućava manipulaciju nad CSS elementima i ostalim datotekama te time pruža cjelokupnu mogućnost izrade web stranice.[5] Specifičnost JavaScript-a su naredbe čija naznaka završetka nije potrebna, u ovom slučaju točka zarez.[5] Jednostavan primjer JavaScript elementa je varijabla.[5] Postoji globalna, čija je oznaka var te lokalna, čija je oznaka let.[5] Varijabla takvog tipa sadržava bilo kakvu vrijednost.[5] Kreirana varijabla sadrži npr. oznaku let, naziv i pridodanu vrijednost operatorom jednako. [5] Slikom 2.3 prikazan je opisan primjer varijable.

```
let naziv = "bojanka"
```

Slika 2.3 *Primjer varijable*

3. ZAPIS I DINAMIČNOST

U ovom poglavlju su detaljno opisani koraci izrade web aplikacije. To su zapis križaljke u datoteku, učitavanje datoteke i dinamičko stvaranje elemenata aplikacije.

3.1 Zapis križaljke u datoteku

Zapis u računalu kojim se novinska križaljka upisuje je datoteka. Ulazni format datoteke je tekstualni (*.txt*) čiji je sadržaj sastavljen sa ASCII znakovima. Takav format datoteke koristi se radi lakšeg učitavanja što omogućuje bolju manipulaciju nad elementima. Tekstualna datoteka je najčešći oblik zapisa u računalu kod zadataka čiji je problem pojednostavljenje bitnog sadržaja. Zapis novinske križaljke sadrži i točna rješenja. Sadržaj križaljke je postavljen na određena polja. Također, unutar datoteke nalazi se veličina svake križaljke. Polja su označena brojem retka i stupca, a podatak križaljke koji se unosi određen je svojom vrstom. Stoga, datoteka sadrži 3 tipa polja i tekstualni podatak čija se vrijednost unosi. Slikom 3.1 prikazan je zapis križaljke u datoteci.

```
10 8
0 0 1
0 1 1 RUŽAN RUKOPIS (razg)
0 2 1 NAČELO, SLOGAN
0 3 1 KANADSKA GLUMICA, SANDRA
0 4 1 OBLIK RAVNALA
0 5 1 SKICA, NACRT
0 6 1 PROKOP, ROV
0 7 1 ŽENSKO IME, AIŠA
1 0 1 DRŽAVA U EUROPI
2 0 1 VRHUNAC, VRŠAK
2 4 1 POZDRAV MLADIH U ŽARGONU ------ VELŠKI GLUMAC, ANTHONY NA SLICI
3 0 1 RIJEKA
3 3 1 NOVINARKA ERCEG ------ IME KRALJA ZVONIMIRA
4 0 1 ČAŠĆENJE NAKON OBAVLJENA POSLA, LIKOVO
5 0 1 GLAVNI GRAD UGANDE
6 0 1 LOPATICA ZA ČIŠĆENJE PLUGA
7 0 1 KONZUMIRATI TEKUĆINU
8 0 1 IVAN OD MILJA
9 0 1 NETIPIČNA UPALA PLUĆA
6 5 3 ../images/anthony.jpg
```

Slika 3.1 *Zapis križaljke*

3.2 Učitavanje datoteke

Programskim skriptnim jezikom JavaScript provodi se učitavanje datoteke i manipulacija nad podacima, uz određene funkcije. Naredba *fetch* korištena je za realizaciju učitavanja križaljki koje se nalaze u tekstualnim datotekama. Naredba *fetch* koristi se upisivanjem željenog naziva dokumenta kojeg treba dohvatiti i izvršavanjem dva zahtjeva nad tim podacima. Prvi zahtjev vraća tip podatka koji dohvaća se u željenom obliku poput teksta, a drugi zahtjev omogućava manipuliranje nad podacima poput rastavljanja podataka na zasebna polja križaljke. Naredbom *split* i predanim argumentom, dobije se svaki redak u datoteci kao samostalno polje. Time je moguće izdvojiti veličine križaljke u redcima i stupcima iz podataka datoteke za daljnje upravljanje nad njima u npr. dinamičkom zauzimanju elemenata. Grupiranje podataka je dostupno naredbom *map* u kojoj se odredi željeni izgled svakog zasebnog polja, odnosno prilagođava se uvjetima za rad. Slikom 3.2 prikazano je učitavanje datoteke uz manipulaciju nad podacima s navedenim naredbama.

```
let data2 = fetch('krizaljkabr1.txt')
    .then((response) => {
        return response.text()
      }).then(someText =>{
        console.log(someText)
        var rows = someText.split('\r\n')
        var redak = parseInt(rows[0].split(" ")[0]);
        var stupac = parseInt(rows[0].split(" ")[1]);
        rows = rows.filter((row, index)=>{
            return index!==0;
        }).map(row =>{
            let dataRow=[];
            row = row.split(" ");
            dataRow[0]= parseInt(row[0]);
            dataRow[1]= parseInt(row[1]);
            dataRow[2]= parseInt(row[2]);
            dataRow[3]=row[3];
            row.forEach((element, index) => {
                if(index>3) dataRow[3] +=`\n${element}`
            });
            return dataRow
        })
```

Slika 3.2 *Učitavanje datoteke*

3.3 Dinamičko stvaranje elemenata

Kreiranje statičkog elementa HTML-a nije pogodno, stoga web aplikacija realizirana je dinamičkim stvaranjem elemenata. Kako bi se dobio izgled križaljke, korišten je HTML element tablica. Mogućnost manipulacije nad dokumentima i njihovim podacima, tablicu dohvaća metoda *getElementById*, koja je sastavni dio objekta *document* te se sprema u varijablu. Dinamičko stvaranje elemenata ostvaruje se pomoću petlji, poput *for* sa svojim uvjetima u kojima je veličina redaka i stupaca prema kojima će se *for* petlja ponavljati i pripadajućim metodama za kreiranje pojedinih HTML elemenata koji označuju element tablice. Elementi se pridodaju tablici metodom *appendChild* s predanim argumentom te se tako ujedini cjelokupna tablica. Elementima poput unosa i slike dodjeljuju se posebni atributi poput tipa unosa i širine te visine zauzimanja ćelija u križaljci. Slikom 3.3 prikazano je dinamičko kreiranje HTML elemenata aplikacije.

```
var element = document.getElementById("firstTable");
let caption=document.createElement("caption");
let text=document.createTextNode("SLAVNE OSOBE");
caption.appendChild(text);
element.appendChild(caption);

    for(var i=0;i<redak;i++){
        var tableRow = document.createElement("tr");
        for (var j=0; j<stupac; j++) {

            var cell = document.createElement("td");
            if(i===6 && j===5){
                let cellImage=document.createElement("img");
                cellImage.id="imageAnthony";
                cell.rowSpan="4";
                cell.colSpan="3";
                cell.appendChild(cellImage);
                stupac-=cell.colSpan;

            }
            else{
            let cellInput=document.createElement("input");
            cellInput.type="text";
            cell.appendChild(cellInput);
            }

            tableRow.appendChild(cell);
        }
        element.appendChild(tableRow);
    }
```

Slika 3.3 *Dinamičko stvaranje HTML elemenata*

4. UPRAVLJANJE I PROVJERA PODATAKA

U ovom poglavlju su detaljnije opisani koraci upravljanja podacima križaljke uz postavljanje uvjeta nad njima te kako se vrši provjera podataka križaljke.

4.1 Obuhvaćanje elemenata

Obuhvaćanje dinamičko stvorenih elemenata HTML-a ostvaruje se dohvaćanjem pomoću metoda manipulacije dokumenata. Svaki redak tablice dohvaćen je s metodom *querySelectorAll* s predanim argumentom HTML oznake nad objektom *document* koja kao rezultat vraća sve retke u tablici te se pridodaje varijabli. Slikom 4.1 prikazano je obuhvaćanje elemenata križaljke.

```
var htmlRows = document.querySelectorAll("tr");
var htmlImg=document.getElementById("imageAnthony");
```

Slika 4.1 *Obuhvaćanje elemenata*

4.2 Unos podataka u elemente

Unos podataka križaljke iz datoteke u određene elemente vrši se pomoću *forEach* petlje, čija je funkcionalnost slična *for* petlji, unutar koje se provjeravaju uvjeti vrste podataka za unos te na temelju njih se unosi odgovarajući podatak u elemente tablice. Moguće vrste podataka su tekst, označeno s brojem 1 i slika, označeno s brojem 3 u kodu na slici 4.2 u uvjetima s desne strane. Slikom 4.2 prikazan je unos podataka u elemente.

```
rows.forEach((row)=> {
    if(row[2]===1){
        htmlRows[row[0]].cells[row[1]].innerText = row[3];
        htmlRows[row[0]].cells[row[1]].style.backgroundColor="papayawhip";
    }
    else if(row[2]===3){
        htmlImg.src=row[3];


    }}
```

Slika 4.2 *Unos podataka u elemente*

4.3 Postavljanje uvjeta nad elementima

Pomoću petlje *forEach*, postavljanje uvjeta za maksimalnu i minimalnu duljinu unosa podataka, postignuto je metodama *minLength* i *maxLength* kojima se pridodaje vrijednost cijeli broj. Vrste podataka određene su brojevima 2, 4, 5 s obzirom na tip rješenja križaljki. Slikom 4.3 prikazani su postavljeni uvjeti pomoću navedenih vrsti podataka i metoda.

```
rows.forEach((row)=> {
    if(row[2]===2){
        htmlRows[row[0]].cells[row[1]].firstChild.minLength=1;
        htmlRows[row[0]].cells[row[1]].firstChild.maxLength=1;


    }
    else if(row[2]===4){
        htmlRows[row[0]].cells[row[1]].firstChild.minLength=1;
        htmlRows[row[0]].cells[row[1]].firstChild.maxLength=1;
        htmlRows[row[0]].cells[row[1]].style.borderBottom="dashed";

    }
    else if(row[2]===5){
        htmlRows[row[0]].cells[row[1]].firstChild.minLength=1;
        htmlRows[row[0]].cells[row[1]].firstChild.maxLength=1;
        htmlRows[row[0]].cells[row[1]].style.borderRight="dashed";


    }
});
```

Slika 4.3 *Uvjeti nad elementima*

4.4 Gumb s događajem

HTML element gumb kreiran je radi jednostavnog korištenja za provjeru unesenih podataka korisnika. Gumb s događajem *click* nudi provjeru na svaki klik miša ako je povratna vrijednost funkcije korištene za provjeru istina. Dohvaćanje elementa gumb se izvršilo pomoću metode za manipulaciju *getElementById*. Događaj *click* dodan je pomoću metode *addEventListener* s odgovarajućim argumentima tipa događaja i funkcija. Slikom 4.4 prikazan je gumb za provjeru podataka.

```
var btnProv=document.getElementById("btnProvjeri");
btnProv.addEventListener('click',checkAllInputs)
```

Slika 4.4 *Gumb za provjeru podataka*

4.5 Funkcija provjere podataka

Pomoću petlje *forEach* uz uvjete provjere tipa podataka i unosa podataka korisnika s rješenjem križaljke, omogućena je provjera unesenih podataka koju izvršava funkcija na svaki poziv. Unos podataka prihvaća velika i mala slova, stoga i provjera podataka omogućava točnost za oba slučaja. Ako je podatak točan, sva polja križaljke će se obojati u zelenu boju. Ako je podatak netočan, sva polja će se obojati u crvenu boju te ako je podatak ostavljen ne unesen podloga će ostati bijela. Na temelju toga, slikom 4.5 prikazana je funkcija za provjeru podataka križaljke.

```
function checkAllInputs(){

    rows.forEach((row)=> {

        if(row[2]===2 && htmlRows[row[0]].cells[row[1]].firstChild.value.toUpperCase() === row[3]){
            htmlRows[row[0]].cells[row[1]].firstChild.style.backgroundColor='green';

        }
        else if(row[2]===2 && htmlRows[row[0]].cells[row[1]].firstChild.value.toUpperCase() !== row[3] && htmlRows[row[0]].cells[row[1]].firstChild.value !==""){
            htmlRows[row[0]].cells[row[1]].firstChild.style.backgroundColor='red';
        }
        else if(row[2]===2 && htmlRows[row[0]].cells[row[1]].firstChild.value ===""){
            htmlRows[row[0]].cells[row[1]].firstChild.style.backgroundColor='white';
        }
        else if(row[2]===4 && htmlRows[row[0]].cells[row[1]].firstChild.value.toUpperCase() === row[3]){
            htmlRows[row[0]].cells[row[1]].firstChild.style.backgroundColor='green';

        }
        else if(row[2]===4 && htmlRows[row[0]].cells[row[1]].firstChild.value.toUpperCase() !== row[3] && htmlRows[row[0]].cells[row[1]].firstChild.value !==""){
            htmlRows[row[0]].cells[row[1]].firstChild.style.backgroundColor='red';
        }
        else if(row[2]===4 && htmlRows[row[0]].cells[row[1]].firstChild.value ===""){
            htmlRows[row[0]].cells[row[1]].firstChild.style.backgroundColor='white';
        }
        else if(row[2]===5 && htmlRows[row[0]].cells[row[1]].firstChild.value.toUpperCase() === row[3]){
            htmlRows[row[0]].cells[row[1]].firstChild.style.backgroundColor='green';

        }
        else if(row[2]===5 && htmlRows[row[0]].cells[row[1]].firstChild.value.toUpperCase() !== row[3] && htmlRows[row[0]].cells[row[1]].firstChild.value !==""){
            htmlRows[row[0]].cells[row[1]].firstChild.style.backgroundColor='red';
        }
        else if(row[2]===5 && htmlRows[row[0]].cells[row[1]].firstChild.value ===""){
            htmlRows[row[0]].cells[row[1]].firstChild.style.backgroundColor='white';
        }
    })
}
```

Slika 4.5 *Funkcija za provjeru podataka*

5. KORISNIČKI POGLED APLIKACIJE

U ovom poglavlju prikazani su gotovi rezultati rada. Uz opisan pristup i opcije odabira križaljki, načine rješavanja križaljki te prikaz točnosti unosa podataka, rezultatima rada pripadaju i primjeri digitalnih križaljki odnosno izgled gotovih križaljki u web aplikaciji kroz korisnički pogled aplikacije.

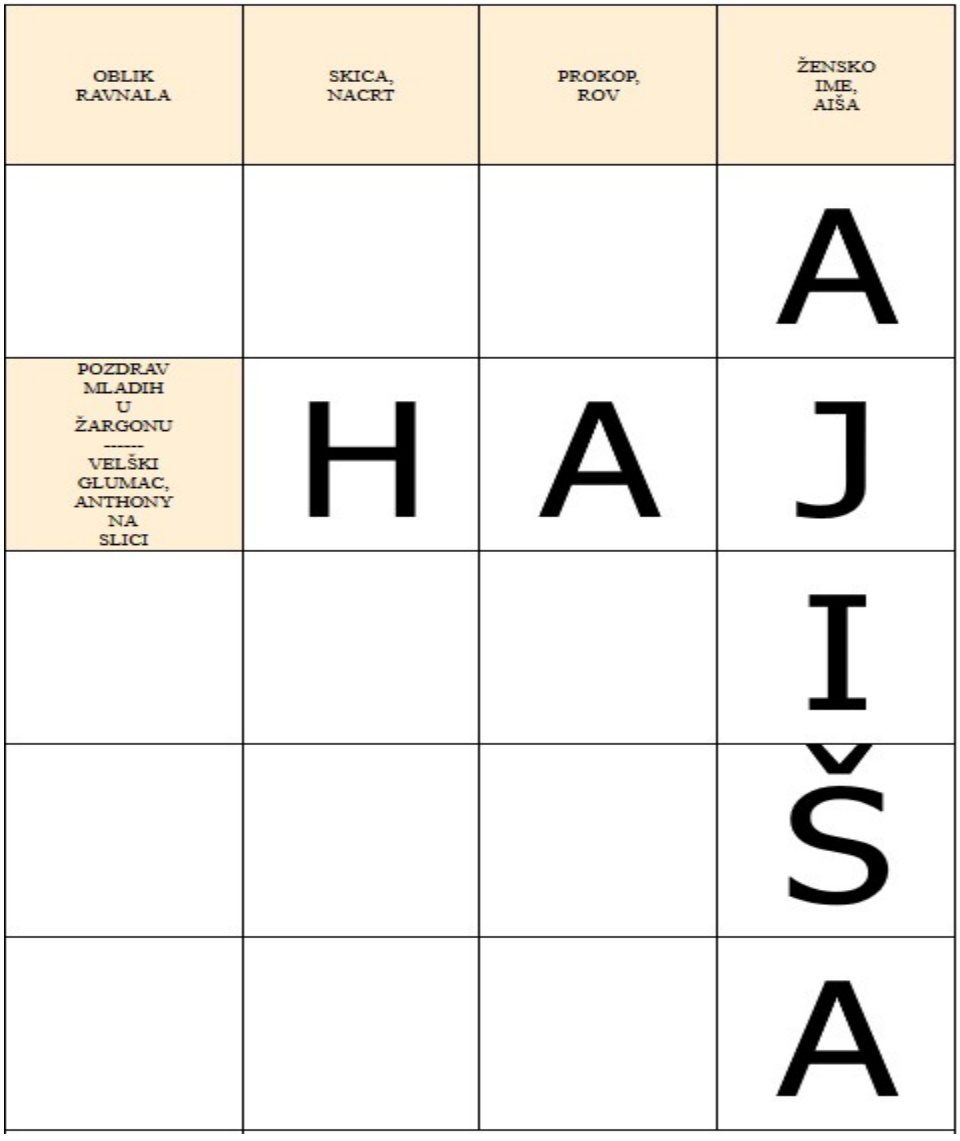
5.1 Pristup i opcije odabira

Korisnički pristup web aplikaciji vrši se preko poveznice koja vodi do naslovne stranice aplikacije. Naslovna stranica sadržava niz ponuđenih križaljki uz pozdravnu poruku i prikladnu pozadinsku sliku. Odabir križaljki osnovan je na temelju niza gumbova sa svojim poveznicama. Svaka poveznica vodi na određenu križaljku. Korisnik vrši pristup odabranoj križaljci klikom miša na određeni gumb unutar navigacijske trake. Rješavanjem zadanog problema zadatka stvorene su dvije križaljke. Omogućeni pristup i opcije odabira olakšavaju kretanje korisnika aplikacijom. Slikom 5.1 prikazana je naslovna stranica s ponuđenim odabirom križaljki.

Slika 5.1 *Naslovna stranica*

5.2 Načini rješavanja križaljke

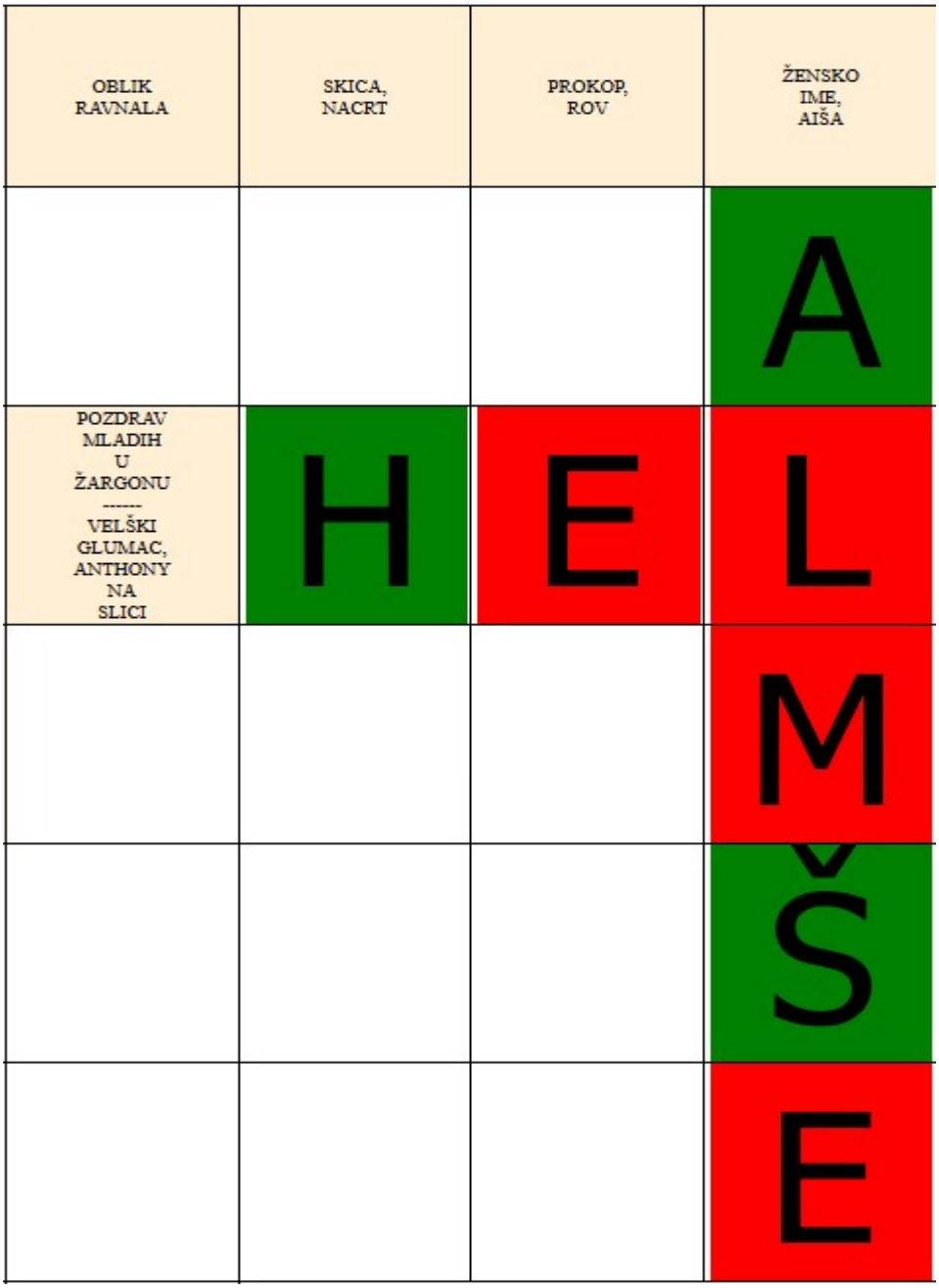
Nakon odabira križaljke, način rješavanja križaljke isti je kao u običnim novinskim križaljkama gdje se odabire polje križaljke te smjer rješavanja. Postoje dva smjera rješavanja, a to su vodoravni i okomiti. U svakom retku i stupcu križaljke, prvo polje označava opis koji navodi na točno rješenje. Opis se također može nalaziti na bilo kojem polju ovisno o sastavu križaljke. Odabirom smjera rješavanja korisnik upisuje u svako polje aktivnog unosa po jedno slovo. Odabir ostalih križaljki zajedno s gumbom za provjeru unesenih podataka je dostupan u svakom trenutku. Slikom 5.2 prikazana su oba načina rješavanja križaljki.



Slika 5.2 *Načini rješavanja križaljke*

5.3 Prikaz točnosti unosa

Na temelju funkcije provjere opisane u prijašnjem poglavlju, točnost upisa dobije se pritiskom na gumb. Točnost je određena s dvije različite boje, zelenom i crvenom, te na osnovi tih boja korisnik dobije povratnu informaciju. Prazna polja ostaju bijela. Slikom 5.3 prikazan je primjer točnih i netočnih unosa podataka.



Slika 5.3 *Točnost unesenih podataka*

5.4 Izgled gotovih križaljki

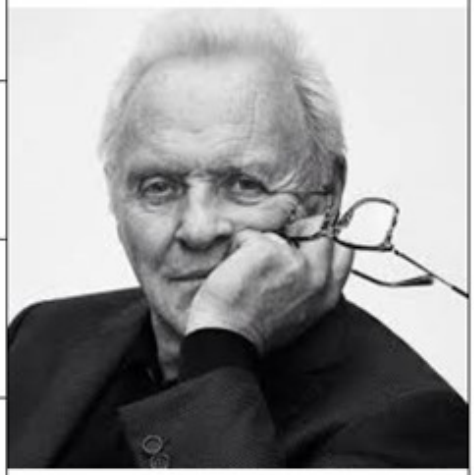
Izgled gotovih križaljki riješen je pomoću svih navedenih tehnologija u drugom poglavlju, ponajviše tehnologijom CSS. Križaljkama je stvoren ugođaj korisniku na način izgleda novinskih križaljki kako bi se lakše snašlo pri rješavanju te da gledanje križaljki nije zamorno. Prilagođenost oku ostvarena je kombinacijom boja CSS tehnologijom. Slikom 5.4 i slikom 5.5 prikazan je odabir prisutan pri svakoj križaljci, a slikom 5.6 i slikom 5.7 prikazan je izgled gotovih križaljki u web aplikaciji.

Slika 5.4 *Odabir na prvoj križaljci*

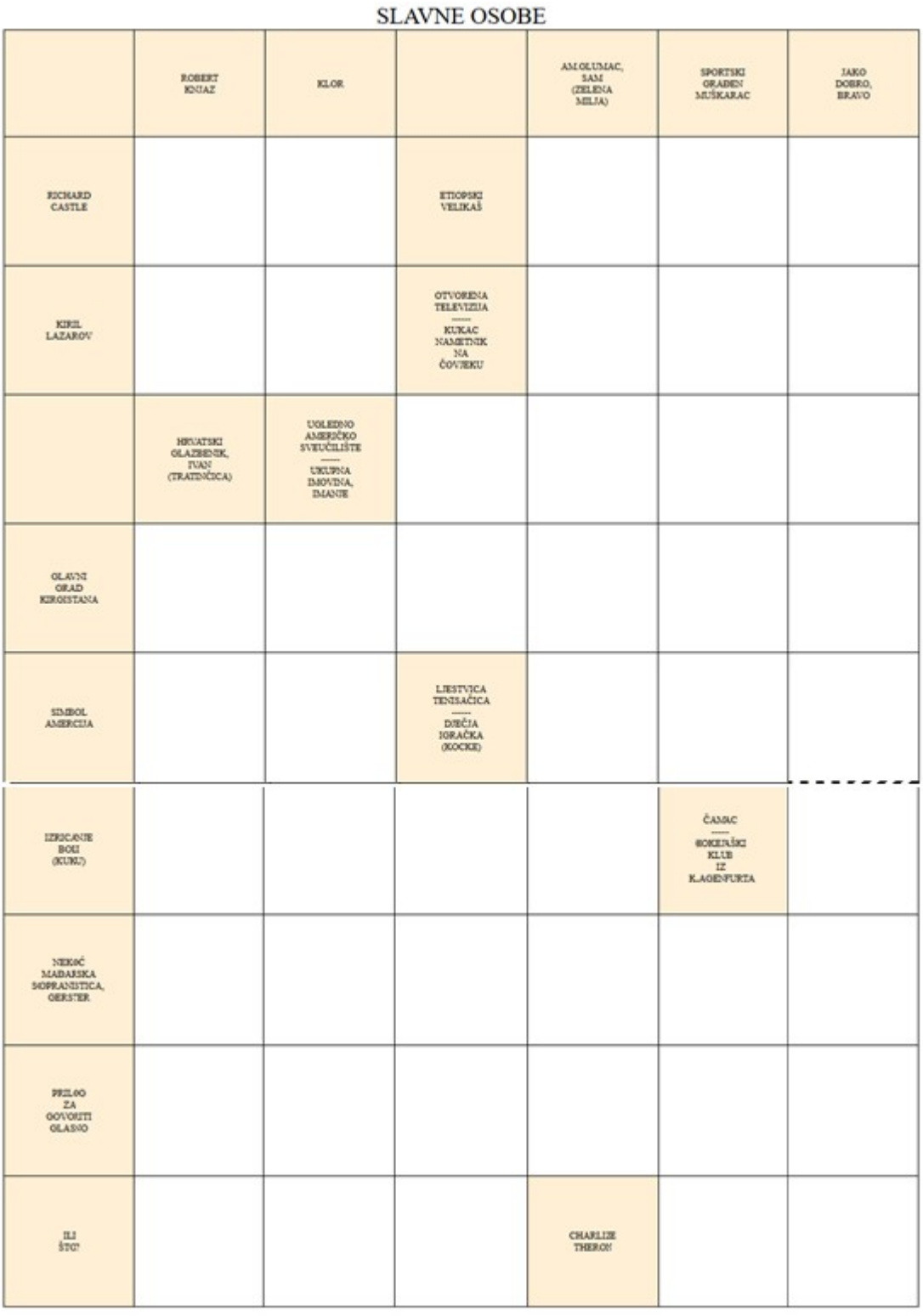
Slika 5.5 *Odabir na drugoj križaljci*

SLAVNE OSOBE

	RUŽAN RUKOPIS (razg)	NAČELO, SLOGAN	KANADSKA GLUMICA, SANDRA	OBLIK RAVNALA	SKICA, NACRT	PROKOP, ROV	ŽENSKO IME, AIŠA
DRŽAVA U EUROPI							
VRHUNAC, VRŠAK				POZDRAV MLADIH U ŽARGONU ----- VELŠKI GLUMAC, ANTHONY NA SLICI			
RIJEKA			NOVINARKA ERCEG ----- IME KRALJA ZVONIMIRA				
ČAŠĆENJE NAKON OBAVLJENA POSLA, LIKOVO							
GLAVNI GRAD UGANDE							
LOPATICA ZA ČIŠĆENJE PLUGA							
KONZUMIRATI TEKUĆINU							
IVAN OD MILJA							
NETIPIČNA UPALA PLUĆA							



Slika 5.6 *Prva križaljka*



Slika 5.7 *Druga križaljka*

6. ZAKLJUČAK

Svakim danom sve više se koristi tehnologija kao rješenje postavljenih problema. Razvijanje web aplikacija postalo je dio svakodnevnog života. Prednost web aplikacije je pristupačnost i fleksibilnost. Kako bi se ispunili zahtjevi koje postavljaju ljudi današnjice, cilj digitalizacije novinskih križaljki proveden je u obliku web aplikacije. Aplikacija pruža korisniku ugodnost pri rješavanju ponuđenih križaljki te time upotpunjuje postavljeni cilj. Korištenjem popularnih tehnologija današnjice ostvarena je web aplikacija Križaljke. Popularne tehnologije su HTML, CSS i JavaScript. Izrada aplikacije zahtjevala je dosta proučavanja materijala. Počevši putem kako zapisati podatke križaljke te daljnje upravljanje podacima i vršenje provjere nad njima. Trenutno aplikacija nudi osnovne značajke potrebne za rješavanje digitalnih novinskih križaljki. Dodatnim znanjima stečenim tokom izrade aplikacije pruženo je bogato iskustvo za daljne razvijanje aplikacije i konkurentnost u današnjici. Daljnji razvitak aplikacije je moguć kroz mnoštvo dodataka poput kretanja poljima križaljke, izdvajanja pojedinih polja tokom unosa uz prethodni odabir smjera i slično. Najbitnije je da tijekom izrade dodataka, korisnički pogled aplikacije bude što jednostavniji kako bi se korisnici lakše snašli i zabavili. Aplikacija je dostupna svima na korištenje uz sigurnost dobre zabave.

LITERATURA:

[1] G., Kurtović, 2016. *Uvod U HTML*. [online] Srce.unizg.hr. Dostupno na: <https://www.srce.unizg.hr/files/srce/docs/edu/osnovni-tecajevi/c201_polaznik.pdf> [21. kolovoza 2020]

[2] D., Grdić i S., Fajković, 2015. *HTML5 I CSS3*. [online] Carpediem.hr. Dostupno na: <http://carpediem.hr/PublikacijeCarpeDiem/Publikacije/HTML5%20i%20CSS3.pdf> [21. kolovoza 2020]

[3] E., Mujadžević, 2014. *Uvod U CSS*. [online] Srce.unizg.hr. Dostupno na: <https://www.srce.unizg.hr/files/srce/docs/edu/osnovni-tecajevi/c220_polaznik.pdf> [22. kolovoza 2020]

[4] Webtech.com.hr. n.d. *Web Tech · CSS*. [online] Dostupno na: <http://www.webtech.com.hr/css.php> [23. kolovoza 2020].

[5] D., Stančer, 2015. *Osnove Javascripta*. [online] Srce.unizg.hr. Dostupno na: <https://www.srce.unizg.hr/files/srce/docs/edu/osnovni-tecajevi/c501_polaznik.pdf> [23. kolovoza 2020].

SAŽETAK

Realizacija web aplikacije *Križaljke* provedena je upotrebom HTML-a, CSS-a i JavaScript-a. Glavni problem u izvedbi bio je zapis podataka križaljke u točnom formatu te upravljanje i provjeravanje podataka. Dobivenim rezultatima korisnicima je pruženo iskustvo rješavanja križaljke slično novinskim križaljkama prijašnjih generacija. Zapis križaljke proveden je kroz tekstualnu datoteku.

Upravljanjem podacima u okruženju CSS tehnologije dobiven je moderniji izgled križaljke. Provjerom podataka osigurana je točnost unesenih podataka korisnika. Provedbom rješenja navedenih problema, cjelokupna aplikacija korisnicima nudi lakše snalaženje pri korištenju.

Održavanje križaljki u digitalnom obliku jednako je važno kao i održavanje novinskih križaljki u fizičkom obliku. U radu su objašnjeni svi koraci izrade aplikacije, a realizacija je prikazana slikovnim prikazom gotovih rezultata križaljke. Omogućen je pristup korisnicima svih uzrasta, a postoji i zabavna komponenta. Aplikacija je prilagođena za daljnje dorađivanje.

Ključne riječi: aplikacija, korisnik, križaljka, podatak, tehnologija

ABSTRACT

CROSSWORDS

The web application *Crosswords* was designed using HTML, CSS, and JavaScript technology. The main problem was to document crossword data in the right format, manage and verify them. The obtained results provided users with the experience of solving crosswords similar to the experience previous generations had when doing newspaper crossword puzzles. The crosswords were written using a text file document.

Data management ensured a modern look of the crosswords. Data verification provided a constant security level in terms of accurate data entering. By implementing solutions to these problems, we managed to ensure an easier application navigation.

Maintaining crosswords in a digital form is just as important as maintaining newspaper crossword puzzles in a physical form. This thesis contains all application realization steps accompanied by completed crossword puzzles. The application is appropriate for all-age users who can test their knowledge and have fun simultaneously. The application can be upgraded.

Key words: application, user, crossword, data, technology

ŽIVOTOPIS

Leopold Bičvić rođen datuma 18.05.1998. u Osijeku, Republika Hrvatska. Pohađao Osnovnu školu „Vladimir Nazor“ Čepin-Područna škola „Briješće“ prva četiri razreda, te Osnovnu školu Svete Ane u Osijeku. Pohađao srednju školu Tehnička škola i prirodoslovna gimnazija Ruđera Boškovića Osijek, smjer prirodoslovna gimnazija. Upisao preddiplomski studij Računarstva 2017. godine na Fakultetu Elektrotehnike, Računarstva i Informacijskih Tehnologija Osijek.

PRILOZI

Na priloženom CD/DVD-u nalazi se cijeli projekt zajedno sa dokumentom završnog rada u obliku formata .docx i formata .pdf.