

**SVEUČILIŠTE U ZAGREBU**  
**FAKULTET ORGANIZACIJE I INFORMATIKE**  
**VARAŽDIN**

**Sicut Oraculi**

**INTERNET TRAŽILICA**

**PROJEKT**

**Varaždin, 2019.**

**SVEUČILIŠTE U ZAGREBU**  
**FAKULTET ORGANIZACIJE I INFORMATIKE**  
**V A R A Ž D I N**

**Sicut Oraculi**

**Matični broj: 123454/07–R**

**Studij: Primjena informacijske tehnologije u poslovanju**

**INTERNET TRAŽILICA**

**PROJEKT**

**Mentor:**

Izv. prof. dr. sc. Markus Schatten

**Varaždin, siječanj 2019.**

### **Izjava o izvornosti**

Izjavljujem da je moj projekt izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

*Autorica*

---

## **Sažetak**

Izrađena je baza podataka i odgovarajuća aplikacija za internet tražilicu. Baza podataka omogućuje indeksiranje web stranica, pretraživanje po ključnim riječima i pregled logova. Projekt je izražen u alatu MS Access uz korištenje jezika SQL i Visual Basic for Applications.

**Ključne riječi:** baza podataka; internet tražilica; access; sql.

# Sadržaj

1.	Opis aplikacijske domene . . . . .	1
2.	Opis modela baze podataka . . . . .	2
3.	Opis obrazaca . . . . .	5
4.	Opis izvještaja i upita . . . . .	17
5.	Primjeri korištenja aplikacije . . . . .	19
6.	Zaključak . . . . .	21
	Popis literature . . . . .	22
	Popis slika . . . . .	23

# 1. Opis aplikacijske domene

Aplikacijska domena predstavlja dio stvarnosti koji se modelom odnosno aplikacijom želi modelirati. U konkretnom kontekstu riječ je o procesu pretraživanja Internet dokumenata koji se želi informatizirati te omogućiti korisnicima da pretražuju različite web stranice. U skladu s navedenim aplikacija treba omogućiti unos novih stranica koje se želi pretraživati, njihovo automatizirano indeksiranje, postavljanje upita u obliku ključnih riječi, prikaz rezultata te pretraživanje načina korištenja (logova) same aplikacije. Uzor aplikacije su različite web stražilice kao što su to Google<sup>1</sup> i sl.

Stoga su glavni koncepti koje aplikacija u ovom kontekstu koristi Web stranica, Upit, Rezultat i Indeks stranice. Web stranica predstavlja podatke o nekom web dokumentu kojeg identificiramo URL (engl. Unified Resource Locator) adresom. Upit je niz ključnih riječi koje je upisao korisnik, a možemo ga opisati i IP (engl. Internet Protocol) adresom korisnika te vremenom upita. Rezultat povezuje upit s određenom stranicom, tj. postavljanje upita rezultira određenom web stranicom koja zadovoljava taj upit. U konačnici indeks stranice je struktura podataka koja za svaku stranicu sadrži sve ključne riječi i broj pojavljivanja kako bi se olakšalo i ubrzalo pretraživanje dokumenata.

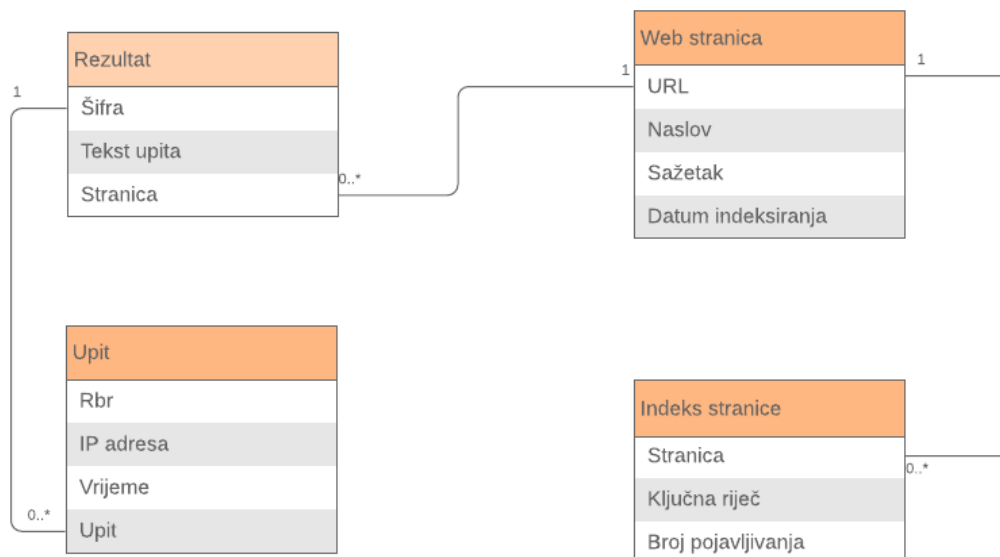
Ova je aplikacija ograničena na neke od najjednostavnijih metoda indeksiranja dokumenata, tj. izvorni tekst neke stranice analizira se statistički samo tako što se broje pojavljivanja svih riječi koje se u njemu pojavljuju. Također, pretraživanje je omogućeno samo u obliku popisa ključnih riječi, tj. nisu podržani nikakvi posebni operatori pretraživanja. Naprednije metode pretraživanja i indeksiranja nisu predmet ovog projekta.

---

<sup>1</sup><https://www.google.com>

## 2. Opis modela baze podataka

Na slici 1 prikazan je UML (engl. Unified Modelling Language) dijagram klasa koji prikazuje implementiranu bazu podataka. Model se sastoji od četiri prethodno navedena koncepta (klase) koji su Web stranica kao centralni entitet, Indeks stranice, Rezultat i Upit.



Slika 1: UML Class dijagram baze podataka

Iz UML modela proizlazi relacijski model koji je kao što slijedi:

**Web stranica** ( URL, naslov, sažetak, datum indeksiranja )

**Indeks stranice** ( stranica, ključna riječ, broj pojavljivanja )

**Rezultat** ( šifra, tekst upita, stranica )

**Upit** ( rbr, IP adresa, vrijeme, upit )

Atributi pojedinih relacija obzirom na njihovu suštinu i semantiku (značenje) opisani su u nastavku:

**URL** - jedinstvena adresa web stranice (ujedno i ključ relacije Web stranica) - znakovni niz

**naslov** - naslov web stranice kako je naveden u HTML (engl. HyperText Markup Language) dokumentu (između <title>...</title> oznaka - znakovni niz

**sažetak** - kratki sažetak dokumenta (prvih 255 znakova teksta web stranice) - znakovni niz

**datum indeksiranja** - datum i vrijeme kada je dokument zadnji puta indeksiran s Interneta - trenutak u vremenu

**stranica** - vanjski ključ koji se referencira na URL adresu web stranice - znakovni niz

**ključna riječ** - jedna riječ koja se pojavljuje u dokumentu (zajedno sa stranicom čini primarni ključ relacije Indeks stranice) - znakovni niz

**broj pojavljivanja** - broj pojavljivanja riječi u odgovarajućem dokumentu - cijeli broj

**šifra** - šifra konkretnog rezultata (artificijalni primarni ključ uveden iz praktičnih razloga - tablica se referencira u drugoj tablici pa je praktičnije imati jedan ključ nego kombinaciju radi referencijalnog integriteta) - cijeli broj vezan uz sekvencu (engl. sequence, autoincrement)

**tekst upita** - tekst upita kako ga je upisao korisnik - znakovni niz

**stranica** - vanjski ključ koji se referencira na relaciju Web stranica - znakovni niz

**rbr** - redni broj upita (artificijalni primarni ključ uveden iz praktičnog razloga što ne postoji prirodni ključ - postoje potencijalna ponavljanja redaka ostalih atributa) - cijeli broj vezan uz sekvencu

**IP adresa** - IP adresa korisnika koji je postavio upit - znakovni niz

**vrijeme** - vrijeme postavljanja upita - trenutak u vremenu

**upit** - vanjski ključ koji se referencira na relaciju Rezultat - cijeli broj.

Iz relacijskog modela dobiven je implementacijski model realiziran u SQL-u. Tablica Web stranica kreirana je kao što slijedi:

```
1 CREATE TABLE [Web stranica]
2 (
3     [URL] TEXT PRIMARY KEY,
4     naslov TEXT NOT NULL,
5     [sažetak] TEXT,
6     [datum indeksiranja] DATETIME
7 )
```

Tablica Indeks stranice realizirana je na sljedeći način:

```
1 CREATE TABLE [Indeks stranice]
2 (
3     stranica TEXT REFERENCES [Web stranica] ( [URL] ),
4     [ključna riječ] TEXT,
5     [broj pojavljivanja] INT,
6     PRIMARY KEY ( stranica, [ključna riječ] )
7 )
```

Tablica Rezultat implementirana je u SQL-u ovako:

```
1 CREATE TABLE [Rezultat]
2 (
3     [šifra] AUTOINCREMENT PRIMARY KEY,
```



```

4      [tekst upita] TEXT NOT NULL,
5      stranica TEXT REFERENCES [Web stranica]( [URL] )
6  )

```

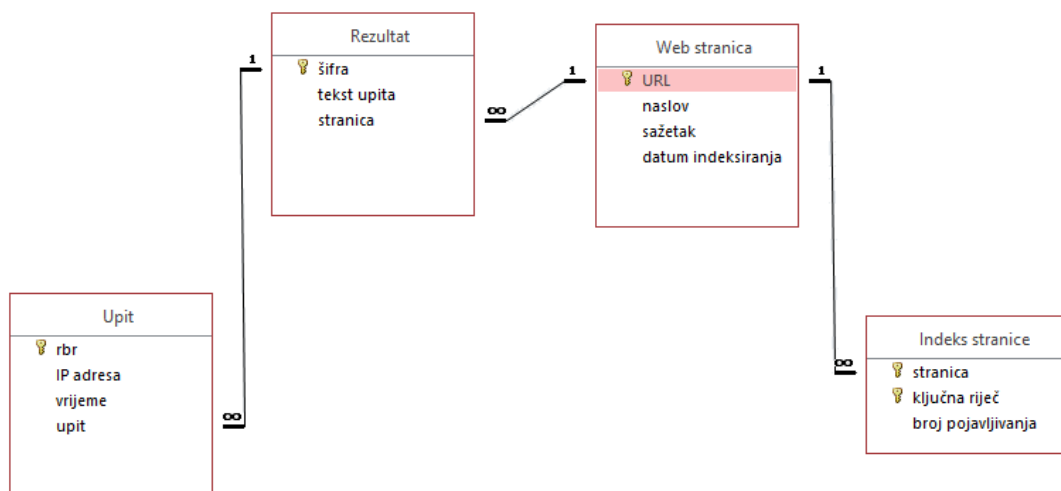
I na kraju, tablica Upit kreirana je putem sljedeće SQL CREATE naredbe:

```

1  CREATE TABLE [Upit]
2  (
3      rbr AUTOINCREMENT PRIMARY KEY,
4      [IP adresa] VARCHAR( 50 ),
5      vrijeme DATETIME,
6      upit INT REFERENCES [Rezultat]( [šifra] )
7  )

```

Konačni implementacijski model u alatu MS Access prikazan je na slici 2.

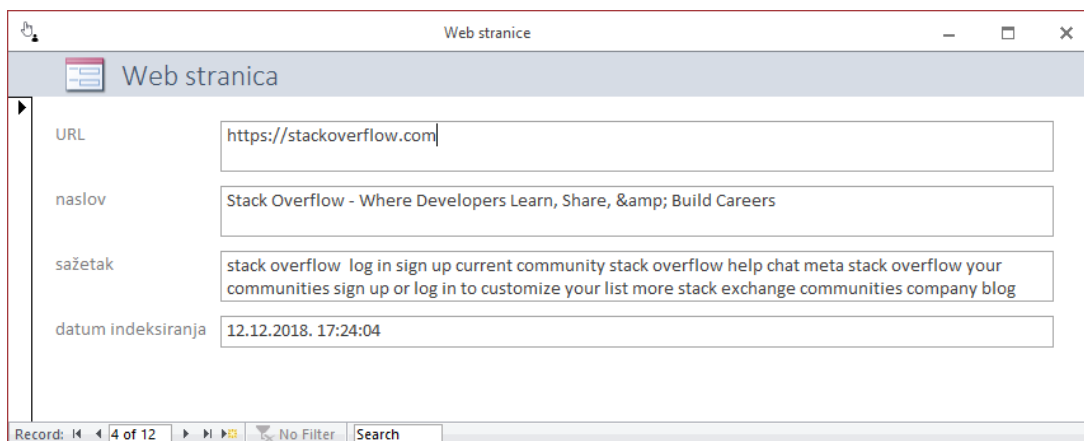


Slika 2: Implementacijski model

### 3. Opis obrazaca

Za potrebe aplikacije kreirana su dva obrasca: (1) Obrazac Web stranice - za unos i pregled indeksiranih web stranica i (2) Tražilica - glavni meni aplikacije koji omogućuje pretragu dokumenata, pregled logova, indeksiranje i unos web stranica.

Na slici 3 prikazan je obrazac Web stranica. Kao što se vidi sa slike, obrazac omogućuje pregled svih unešenih web stranica, promjenu njihovih podataka te unos novih. Prilikom unosa nove web stranice potrebno je samo unesti URL adresu, dok se ostali podaci automatski prikupljaju sa same stranice putem VBA (engl. Visual Basic for Applications) procedure indeksiraj() koja je prikazana u nastavku.



Slika 3: Obrazac Web stranica

```
1 Sub indeksiraj()  
2     Dim adrese As DAO.Recordset ' DAO.Recordset predstavlja rezultate  
        upita (tablicu)  
3     Dim azuriraj As DAO.QueryDef ' DAO.QueryDef predstavlja  
        definiciju upita  
4     Dim URL As String, html As String, tekst As String ' Pomoćne  
        varijable  
5  
6     ' Učitaj upit "Indeksiraj web stranicu" koji je definiran u  
        Access-u  
7     Set azuriraj = CurrentDb.QueryDefs("Indeksiraj_web_stranicu")  
8     ' Pokreni upit "Adrese stranica" te njegove rezultate pohrani u  
        varijablu adrese  
9     Set adrese = CurrentDb.OpenRecordset("Adrese_stranica")  
10    ' Postavi se na prvi red  
11    adrese.MoveFirst  
12    ' Vrti petlju dok nismo na kraju tablice (EOF je oznaka End Of  
        File - kraj datoteke)  
13    Do While Not adrese.EOF  
14        ' U varijablu URL pohrani vrijednost atributa URL u trenutnom
```

```

        redu
15     URL = adrese("URL")
16     ' U varijablu html pohrani HTML izvorni kod stranice s tim
        URL-om
17     html = getHTML(URL)
18     ' U varijablu tekst izvudi tekst iz HTML-a te ga očisti
19     tekst = ocistiTekst(getText(html))
20     ' Postavljanje parametara kako su definirani u upitu "
        Indeksiraj web stranicu":
21     ' Pod URL unesi URL
22     azuriraj.Parameters("Unesi_URL").Value = URL
23     ' Pod naslov unesi izvučeni naslov iz HTML koda
24     azuriraj.Parameters("Unesi_naslov").Value = getTitle(html)
25     ' Pod sažetak unesi prvih 255 znakova teksta
26     azuriraj.Parameters("Unesi_sazetak").Value = Left(tekst, 255)
27     ' Pokreni upit
28     azuriraj.Execute
29
30     ' Pokreni proceduru izvuciRijeci i proslijedi joj argumente
        teksta stranice i URL adresu
31     izvuciRijeci tekst, URL
32
33     ' Pomakni se na sljedeći redak i nastavi petlju
34     adrese.MoveNext
35 Loop
36 End Sub

```

Procedura prvo učitava sve adrese stranica kako su dobiveni upitom "Adrese stranica" definiranim u alatu MS Access koji je implementiran kao što slijedi:

```

1 SELECT [URL]
2 FROM [Web stranica]
3 ORDER BY 1;

```

Zatim redom dohvaća podatke o svakoj stranici (HTML izvorni kod) putem procedure getHTML (preuzeta sa [1]) prikazane u nastavku:

```

1 Function getHTML(URL As String) As String
2     ' Prima URL adresu i vraća izvorni
3     ' HTML kod te stranice
4     With CreateObject("MSXML2.XMLHTTP")
5         .Open "GET", URL, False
6         .Send
7         getHTML = .ResponseText
8     End With

```

## 9 End Function

Onda taj kod čisti (izvlači naslov, tekst, izbacuje nepotrebne znakove, sve pretvara u mala slova i sl.) putem funkcija `getText()` (preuzeta sa [2]), `ocistiTekst()` i `getTitle()` (implementirane samostalno) prikazane u nastavku:

```
1 Function getText(html As String) As String
2     ' Vraća tekst HTML dokumenta (bez tag oznaka)
3     With CreateObject("htmlfile")
4         .Open
5         .write html
6         .Close
7         getText = .body.outerText
8     End With
9 End Function
10
11 Function ocistiTekst(tekst As String) As String
12     ' Čisti tekst od raznih nepotrebnih znakova.
13     ' Ovdje se koriste pravilni izrazi (regular
14     ' expressions) koji su slični LIKE izrazu u SQL-u
15     ' ali puno moćniji. Konkretno, brišu se svi
16     ' znakovi koji nisu slova abecede, brojevi odnosno
17     ' hrvatska slova i space znak (tj. svaki drugi znak
18     ' se zamjenjuje praznim stringom "").
19     With CreateObject("VBScript.RegExp")
20         .Pattern = "[^A-Z0-9čćžšđČĆŽŠĐ_]"
21         .IgnoreCase = True
22         .Global = True
23         tekst = .Replace(tekst, "")
24     End With
25     tekst = LCase(tekst) ' Pretvori tekst u mala slova.
26     ' Obriši znakove novog reda (10 i 13 su ASCII kodovi znakova \r i
27     ' \n
28     ' koji predstavljaju novi red.
29     ocistiTekst = Replace(Replace(tekst, Chr(10), "_"), Chr(13), "_")
30 End Function
31
32 Function getTitle(html As String) As String
33     ' Vraća naslov iz HTML dokumenta
34     Const strStart = "<title>" ' Naslov u HTML dokumentu je zadan <
35     title>...</title> oznakama
36     Const strStop = "</title>"
```

```

37     Dim pStart As Long, pStop As Long
38     pStart = InStr(1, html, strStart) + Len(strStart) ' Pronađi
        lokaciju teksta <title>
39     pStop = InStr(pStart, html, strStop) ' Pronađi lokaciju teksta </
        title>
40     getTitle = Mid(html, pStart, pStop - pStart) ' Izvuci teksti izme
        đu tih dviju lokacija
41 End Function

```

Zatim takve očišćene podatke pohranjuje putem pospremljenog upita "Indeksiraj web stranicu" koji je implementiran kao što slijedi:

```

1  UPDATE [Web stranica] SET naslov = [Unesi naslov], sažetak = [Unesi
    sažetak]
2  WHERE [URL] = [Unesi URL];

```

Na kraju pokreće proceduru izvuciRijeci() koja iz teksta izvlači ključne riječi.

```

1  Sub izvuciRijeci(tekst As String, URL As String)
2      ' Iz zadanog teksta i URL adrese izvlači
3      ' ključne riječi i pohranjuje ih u indeks.
4      ' Prije toga briše sve postojeće ključne riječi
5      ' za tu stranicu, obzirom da se funkcija pokreće
6      ' prilikom ažuriranja indeksa stranica.
7
8      ' rijeci je polje stringova (znakovnih nizova)
9      ' koja se koristi za pohranu ključnih rijeci
10     Dim rijeci() As String
11     ' Pobrojane je dinamička varijabla kojoj nije
12     ' dodjeljen tip, već se on dodjeljuje naknadno
13     ' prilikom inicijalizacije.
14     Dim pobrojane
15     ' Varijable obrisi i dodaj su definicije upita
16     Dim obrisi As DAO.QueryDef
17     Dim dodaj As DAO.QueryDef
18
19     ' U varijablu pobrojane pohranjujemo novi objekt scripting.
        dictionary
20     ' (riječ je o riječniku odnosno asocijativnom nizu koji se
        sastoji od
21     ' ključeva i vrijednosti). U njega ćemo pod ključeve unositi klju
        čne
22     ' riječi, a pod vrijednosti ukupan broj pojavljivanja te riječi
        na
23     ' trenutnoj web stranici.

```

```

24  Set pobrojane = CreateObject("scripting.dictionary")
25
26  ' U varijablu obrisi učitavamo pospremljeni upit iz Access-a,
27  ' postavljamo parametar kako je definiran u upitu te pokrećemo
28  ' upit. U stvari brišemo sve trenutno pohranjene ključne riječi
29  ' za trenutnu stranicu, kako bismo unesli nove, ažurirane.
30  Set obrisi = CurrentDb.QueryDefs("Obriši_ključne_riječ_i_stranice"
    )
31  obrisi.Parameters("Unesi_URL").Value = URL
32  obrisi.Execute
33
34  ' U varijablu dodaj učitavamo pospremljeni upit za dodavanje
35  ' ključne riječi.
36  Set dodaj = CurrentDb.QueryDefs("Dodaj_ključnu_riječ_za_stranicu"
    )
37
38  ' Putem funkcije Split dijelimo tekst po znaku space " " (svaki
39  ' element polja rijeci je sada po jedna riječ. One se naravno
    mogu
40  ' i ponavljati, a naš je cilj da ih pobrojimo.
41  rijeci() = Split(tekst)
42
43
44  ' Petlja se vrti od prvog do zadnjeg elementa polja rijeci
45  For i = LBound(rijeci) To UBound(rijeci)
46      ' Ako trenutna riječ u polju još nije u
47      ' riječniku pobrojane, dodaj ju kao ključ u
48      ' riječnik te mu postavi inicijalnu vrijednost
49      ' nula.
50      If Not pobrojane.exists(rijeci(i)) Then
51          pobrojane(rijeci(i)) = 0
52      End If
53      ' Povećaj broj pojavljivanja trenutne riječi u
54      ' riječniku za 1.
55      pobrojane(rijeci(i)) = pobrojane(rijeci(i)) + 1
56  Next i
57
58  ' Petlja prolazi kroz sve ključeve u riječniku
59  ' pobrojane, unosi parametre gore pripremljenog
60  ' upita dodaj (adresa, ključna riječ i broj
61  ' pojavljivanja), te pokreće upit koji pohranjuje
62  ' zadanu ključnu riječ.
63  For Each kljuc In pobrojane

```

```

64         dodaj.Parameters("Unesi_URL").Value = URL
65         dodaj.Parameters("Unesi_kljucnu_riječ") = kljuc
66         dodaj.Parameters("Unesi_broj_pojavljivanja") = pobrojane(
            kljuc)
67         dodaj.Execute
68
69     Next kljuc
70 End Sub

```

Ova procedura koristi upit "Obriši ključne riječi stranice" koji je implementiran ovako:

```

1  DELETE *
2  FROM [Indeks stranice]
3  WHERE stranica = [Unesi URL];

```

Upit služi za brisanje trenutno indeksiranih ključnih riječi web stranice kako bi se mogle unesti nove. Naime, prilikom reindexiranja stranice, nema smisla da se pohranjuju podaci iz prethodne inačice stranice, ako je došlo do ažuriranja podataka (što je primjerice s različitim portalima vrlo često slučaj).

Također, koristi se upit "Dodaj ključnu riječ za stranicu" koji dodaje po jednu ključnu riječ za neku stranicu, a implementiran je na sljedeći način:

```

1  INSERT INTO [Indeks stranice]
2  VALUES ([Unesi URL], [Unesi kljucnu rijec], [Unesi broj pojavljivanja]
    );

```

Obrazac Tražilica prikazan je na slici 4. Kao što se vidi na slici, ovaj je obrazac ujedno i glavni meni aplikacije te se pokreće pri pokretanju MS Access-a.

Obrazac se sastoji od tri dijela: (1) polje za unos upita, (2) administracijski dio za pregled logova i (3) administracijski dio za ostale akcije (indeksiranje i pregled odnosno unos web stranica).

Prvi dio omogućuje korisniku da unese ključne riječi te klikom na gump pretrage pokreće proceduru trazi() koja je implementirana na sljedeći način:

```

1  Sub trazi()
2      ' Varijable log i rez su definicije upita
3      Dim log As DAO.QueryDef
4      Dim rez As DAO.QueryDef
5
6      ' Varijable rs i zadnji su rezultati upita (tablice)
7      Dim rs As DAO.Recordset
8      Dim zadnji As DAO.Recordset
9
10
11     Dim upit_rezultatil As String

```

# foi Tražilica

Upiši ključne riječi



## Administracija

### Pretraži log

Pregledaj sve

Po datumu

Po dijelu IP  
adrese

### Ostale akcije

Indeksiraj  
stranice

Pregled i unos  
stranica

Slika 4: Obrazac Tražilica

```
12 Dim upit_rezultati2 As String
13 Dim upit As String
14 Dim naslov As String
15 Dim URL As String
16
17 Dim kljucne_rijeci() As String
18 Dim rpt As Report
19 Dim report2Copy As Report
20
21 On Error GoTo ERROR_NemaRezultata
22
23 upit_rezultati1 = "SELECT_[Web_stranica].URL,_" & _
24 "[Web_stranica].naslov,_" & _
25 "[Web_stranica].[sažetak],_" & _
26 "SUM(_[Indeks_stranice].[broj_pojavljivanja]_)_AS_score_" & _
27 "FROM_[Web_stranica]_INNER_JOIN_[Indeks_stranice]_" & _
28 "ON_[Web_stranica].[URL]=_[Indeks_stranice].stranica_" & _
29 "WHERE_[ključna_riječ]_In_"
30
31 upit_rezultati2 = ")__" & _
32 "GROUP_BY_[Web_stranica].URL,_" & _
33 "[Web_stranica].naslov,_" & _
```



```

34 "[Web_stranica].[sažetak]_" & _
35 "ORDER_BY_SUM(#[Indeks_stranice].[broj_pojavljivanja]_)_DESC"
36
37 DoCmd.OpenForm "Tražilica"
38
39 upit = Forms![Tražilica]!txtTrazi
40 naslov = "Rezultati_pretrage:_" & upit
41 kljucne_rijeci = Split(upit)
42 upit = ""
43 For i = LBound(kljucne_rijeci) To UBound(kljucne_rijeci)
44     upit = upit & " " & kljucne_rijeci(i) & " ",_"
45 Next
46 upit = Left(upit, Len(upit) - 2)
47
48 upit = upit_rezultati1 & upit & upit_rezultati2
49
50 Set rez = CurrentDb.QueryDefs("Unesi_rezultat")
51 rez.Parameters("Unesi_tekst_upita") = Forms![Tražilica]!txtTrazi
52
53 Set rs = CurrentDb.OpenRecordset(upit)
54 rs.MoveFirst
55
56 Do While Not rs.EOF
57     URL = rs("URL")
58
59     rez.Parameters("Unesi_stranicu") = URL
60     rez.Execute
61     rs.MoveNext
62
63     Set zadnji = CurrentDb.OpenRecordset("Zadnji_uneseni_rezultat
        ")
64     zadnji.MoveFirst
65
66     Set log = CurrentDb.QueryDefs("Unesi_log")
67     log.Parameters("Unesi_IP").Value = getIPAddress()
68     log.Parameters("Unesi_upit") = zadnji("zadnji")
69     log.Execute
70
71 Loop
72
73 kreirajIzvjestajRezultata upit, naslov
74 Exit Sub
75

```

```

76 ERROR_NemaRezultata:
77     MsgBox Error
78     MsgBox "Pretraga_nije_našla_niti_jednu_stranicu!", vbOKOnly, "
        Nema_rezultata"
79
80 End Sub

```

Procedura pretražuje bazu podataka za upisanim ključnim riječima (upit koji je postavio korisnik), zatim ispisuje rezultat u obliku izvještaja te zapisuje u log rezultate u log kako bi ih administrator mogao pregledavati.

Pri tome se koristi dinamički generirani upit koji izgleda ovako:

```

1 SELECT [Web stranica].URL,
2     [Web stranica].naslov,
3     [Web stranica].[sažetak],
4     SUM( [Indeks stranice].[broj pojavljivanja] ) AS score
5 FROM [Web stranica] INNER JOIN [Indeks stranice]
6     ON [Web stranica].[URL] = [Indeks stranice].stranica
7 WHERE [ključna riječ] In ( [POPIS KLJUČNIH RIJEČI] )
8 GROUP BY [Web stranica].URL,
9     [Web stranica].naslov,
10    [Web stranica].[sažetak]
11 ORDER BY SUM( [Indeks stranice].[broj pojavljivanja] ) DESC

```

Taj je upit najkompliciraniji, a ujedno i najvažniji za aplikaciju obzirom da se putem njega dobivaju rezultati pretraživanja. Dinamički se dio odnosi na [POPIS KLJUČNIH RIJEČI] u koji se putem VBA programskog koda upisuju sve ključne riječi koje je korisnik upisao u obrazac.

Osim tog upita koristi se i pohranjeni upit "Unesi rezultat" koji izgleda ovako:

```

1 INSERT INTO Rezultat ( [tekst upita], stranica )
2 VALUES ([Unesi tekst upita], [Unesi stranicu]);

```

Taj upit služi za unos podataka u log kako bi se kasnije mogao pretraživati.

Također koristi se upit "Zadnji uneseni rezultat" koji služi za pronalaženje zadnjeg rezultata koji je unesen, e kako bi se on mogao povezati kao vanjski ključ prilikom unosa u log u prethodnom upitu. Upit je jednostavan, a izgleda ovako:

```

1 SELECT MAX([šifra]) AS zadnji
2 FROM Rezultat;

```

Na kraju procedure trazi() pokreće se funkcija kreirajIzvjestajRezultata() (preuzeta djelomično sa [3]) koja dinamički kreira izvještaj temeljem upita, a koristi se za generiranje liste rezultata pretrage. Ta funkcija izgleda ovako:

```

1 Function kreirajIzvjestajRezultata(strSQL As String, title As String)

```

```

2  Dim db As DAO.Database ' database object
3  Dim rs As DAO.Recordset ' recordset object
4  Dim txtNew As Access.TextBox ' textbox control
5  Dim lblNew As Access.Label ' label control
6  Dim rpt As Report ' hold report object
7  Dim lngTop As Long ' holds top value of control position
8  Dim lngLeft As Long ' holds left value of controls position
9
10 ' initialise position variables
11 lngLeft = 0
12 lngTop = 0
13
14 'Create the report
15 Set rpt = CreateReport
16
17 ' set properties of the Report
18 With rpt
19     .Width = 8500
20     .RecordSource = strSQL
21     .Caption = title
22     .CloseButton = True
23
24 End With
25
26 ' Open SQL query as a recordset
27 Set db = CurrentDb
28 Set rs = db.OpenRecordset(strSQL)
29
30 ' Create Label Title
31 Set lblNew = CreateReportControl(rpt.Name, acLabel, _
32     acPageHeader, , title, 0, 0)
33 lblNew.FontBold = True
34 lblNew.FontSize = 12
35 lblNew.SizeToFit
36
37 Set txtNew = CreateReportControl(rpt.Name, acTextBox, _
38     acDetail, , "naslov", lngLeft, lngTop, 6000)
39 lngTop = lngTop + 500
40 txtNew.OnClick = "=otvoriURL(CurrentRecord,'" + strSQL + "')"
41 txtNew.BorderStyle = 0 ' Transparentni okvir
42 txtNew.ForeColor = RGB(255, 0, 0)
43
44 Set txtNew = CreateReportControl(rpt.Name, acTextBox, _

```

```

45         acDetail, , "sažetak", lngLeft, lngTop, 6000, 1800)
46 lngTop = lngTop + 2000
47 txtNew.OnClick = "=otvoriURL(CurrentRecord,'" + strSQL + "')"
48 txtNew.BorderStyle = 0 ' Transparentni okvir
49
50
51 Set txtNew = CreateReportControl(rpt.Name, acTextBox, _
52     acDetail, , "URL", lngLeft, lngTop, 6000)
53 lngTop = lngTop + 500
54 txtNew.OnClick = "=otvoriURL(CurrentRecord,'" + strSQL + "')"
55 txtNew.BorderStyle = 0 ' Transparentni okvir
56 txtNew.ForeColor = RGB(0, 0, 255)
57
58
59 Set txtNew = CreateReportControl(rpt.Name, acTextBox, _
60     acDetail, , "score", lngLeft + 1500, lngTop, 400)
61 txtNew.OnClick = "=otvoriURL(CurrentRecord,'" + strSQL + "')"
62 txtNew.BorderStyle = 0 ' Transparentni okvir
63
64 Set lblNew = CreateReportControl(rpt.Name, acLabel, acDetail, _
65     txtNew.Name, "Score:", lngLeft, lngTop, 200, txtNew.Height)
66 lngTop = lngTop + 500
67
68
69 ' Create datestamp in Footer
70 Set lblNew = CreateReportControl(rpt.Name, acLabel, _
71     acPageFooter, , Now(), 0, 0)
72
73 ' Create page numbering on footer
74 Set txtNew = CreateReportControl(rpt.Name, acTextBox, _
75     acPageFooter, , "='Stranica_&[Page]&_od_&[Pages]", rpt.
76         Width - 1000, 0)
77 txtNew.SizeToFit
78
79 rpt.Section(0).Height = 4200
80
81 ' Open new report.
82 DoCmd.OpenReport rpt.Name, acViewReport
83
84 'reset all objects
85 rs.Close
86 Set rs = Nothing
Set rpt = Nothing

```

```

87      Set db = Nothing
88
89 End Function

```

Kako funkcija kreira različite objekte na izvještaju, kreira i naslove web stranica koji koriste funkciju `otvoriURL()` koja se pokreće klikom na naslov stranice i otvara stranicu u Internet pregledniku računala [4]. Ona je implementirana ovako:

```

1 Public Function otvoriURL(red As Integer, upit As String)
2     ' Otvara URL adresu u pregledniku
3     Dim rs As DAO.Recordset
4     Set rs = CurrentDb.OpenRecordset (upit)
5     rs.Move red - 1
6     Application.FollowHyperlink rs("URL")
7 End Function

```

Gumbi za pretraživanje loga pokreću odgovarajuće izvještaje koji su opisani u sljedećem poglavlju.

Gumb Indeksiraj stranice pokreće prethodno opisanu proceduru indeksiraj() koja indeksira web stranice. Na kraju, gumb Pregled i unos stranica pokreće obrazac Web stranice.

## 4. Opis izvještaja i upita

U aplikaciji su kreirana tri izvještaja, koji su međusobno vrlo slična. Prvi je Pregledaj log koji ispisuje čitavi log temeljem upita Pregledaj log (sve):

```
1 SELECT [Upit].rbr, [Upit].[IP adresa], [Upit].vrijeme, [Rezultat].[  
   tekst upita] AS upit  
2 FROM Upit INNER JOIN Rezultat ON [Upit].upit = [Rezultat].[šifra]  
3 ORDER BY 1;
```

Log				16. siječanj 2019.
				17:28:49
	rbr	IP adresa	vrijeme	upit
1	1	127.0.0.1	12.12.2018.	test
2	2	255.255.255.0	12.12.2018. 15:43:26	test
3	3	10.128.116.248, fe80::a900:8a00:25c0:a42	12.12.2018. 15:46:05	test
4	4	10.128.116.248, fe80::a900:8a00:25c0:a42	12.12.2018. 15:46:26	test
5	5	10.128.116.248, fe80::a900:8a00:25c0:a42	12.12.2018. 15:46:27	test
6	6	10.128.116.248, fe80::a900:8a00:25c0:a42	12.12.2018. 15:47:50	test
7	7	10.128.116.248, fe80::a900:8a00:25c0:a42	12.12.2018. 15:50:05	test
8	8	10.128.104.13, fe80::59f9:e0b4:e043:97a3	12.12.2018. 15:36:35	test
9	9	10.128.104.13, fe80::59f9:e0b4:e043:97a3	12.12.2018. 17:23:19	test
10	10	10.128.114.10, fe80::25fd:e88b:9bd4:7f11	12.12.2018. 19:25:57	test
11	11	10.128.116.248, fe80::a900:8a00:25c0:a42	19.12.2018. 13:22:52	test

Slika 5: Izvještaj - pregled loga

Drugi izvještaj Pregledaj Log (po datumu) omogućuje filtriranje loga po nekom vremenskom razdoblju, a koristi podatke upita Pregledaj Log (po datumu):

```
1 SELECT [Upit].rbr, [Upit].[IP adresa], [Upit].vrijeme, [Rezultat].[  
   tekst upita] AS upit  
2 FROM Upit INNER JOIN Rezultat ON [Upit].upit = [Rezultat].[šifra]  
3 WHERE vrijeme >[Datum od:] AND vrijeme < [Datum do:]  
4 ORDER BY 1;
```

Zadnji izvještaj je izvještaj Pregledaj Log (po IP adresi) koji omogućuje unos dijela IP adrese korisnika čije pretrage se želi pregledati. Koristi podatke upita Pregledaj Log (po IP adresi):

```
1 SELECT [Upit].rbr, [Upit].[IP adresa], [Upit].vrijeme, [Rezultat].[  
   tekst upita] AS upit
```

```

2 FROM Upit INNER JOIN Rezultat ON [Upit].upit = [Rezultat].[šifra]
3 WHERE [IP adresa] LIKE "*" + [Unesi (dio) IP adrese] + "*"
4 ORDER BY 1;

```

Svi izvještaji oblikovani su izgledom jednako, a primjer prvog izvještaja prikazan je na slici 5.

Uz navedene postoji i dinamički izvještaj za prikaz rezultata pretrage koji je opisan u prethodnom poglavlju, a prikazan je na slici 6.

#### Rezultati pretrage: vijesti

##### Index.hr

u skladu s novom europskom regulativom indexhr je nadgradio politiku privatnosti i korištenja kolačića kolačiće upotrebljavamo kako bismo vam omogućili korištenje naše online usluge što bolje korisničko iskustvo i funkcionalnost naših web stranica prikaz

<https://www.index.hr>

Score: 9

##### Najnovije vijesti iz Hrvatske i svijeta | 24sata

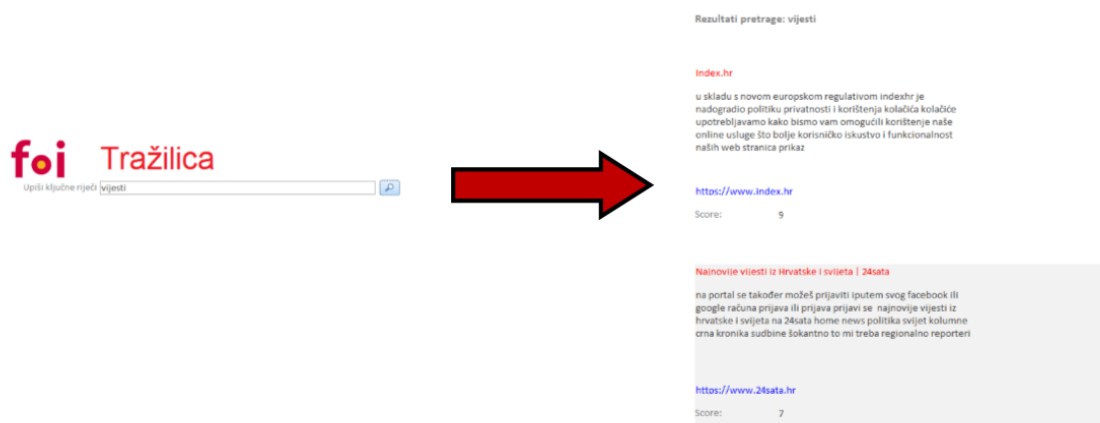
na portal se također možeš prijaviti i putem svog facebook ili google računa prijava ili prijava prijavi se najnovije vijesti iz hrvatske i svijeta na 24sata home news politika svijet kolumne crna kronika sudbine šokantno to mi treba regionalno reporteri

<https://www.24sata.hr>

Score: 7

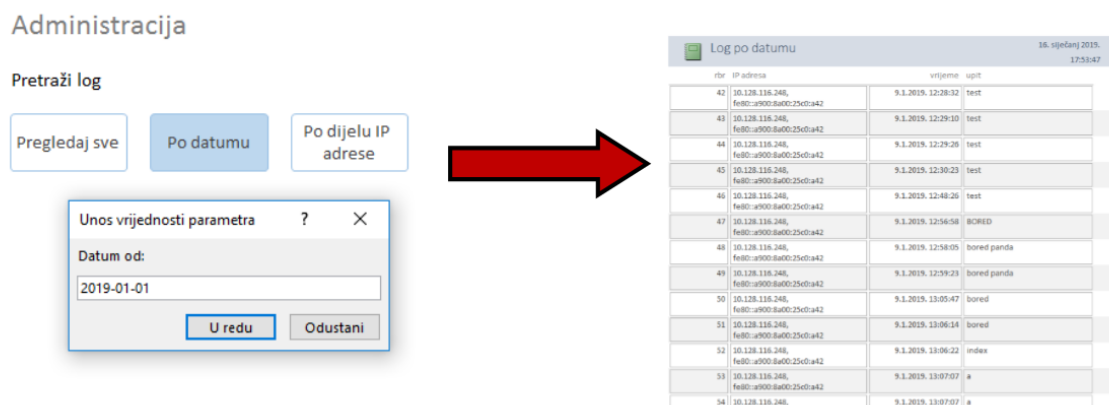
Slika 6: Dinamički izvještaj - rezultati pretrage

## 5. Primjeri korištenja aplikacije



Slika 7: Primjer korištenja - pretraživanje

Aplikacija se može koristiti na dva načina: (1) korisnički način (svodi se na pretragu indeksiranih stranica putem ključnih riječi) i (2) administrativni način (svodi se na administrativne zadatke). Radi jednostavnosti oba su načina implementirana na obrascu Tražilica (slika 4).

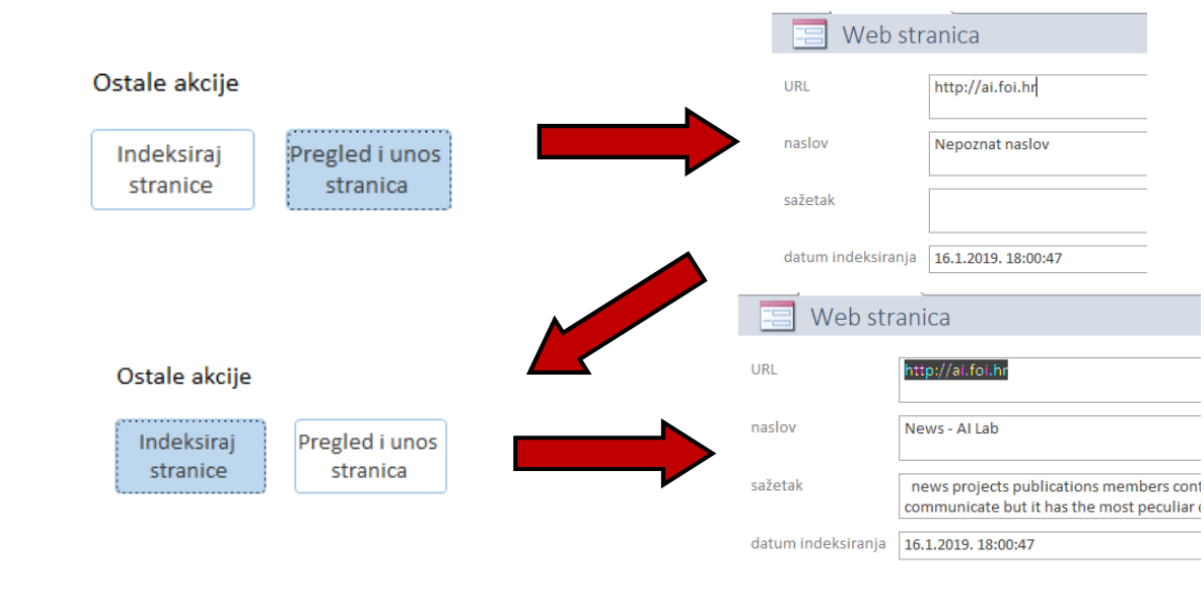


Slika 8: Primjer korištenja - pretraživanje loga

Iz korisničke perspektive, dovoljno je u polje za pretragu unesti ključne riječi pretrage i kliknuti na gumb za pretraživanje kako bi se prikazale sve web stranice koje sadrže upravo zadane ključne riječi. Taj je način korištenja vizualno prikazan na slici 7.

Iz perspektive administratora moguća su dva slučaja korištenja. Prvi je pretraživanje





Slika 9: Primjer korištenja - unos i indeksiranje stranica

loga kojeg je moguće pretraživati u cijelosti (klikom na gumb Pregledaj sve), prema određenom vremenskom razdoblju (klikom na gumb Po datumu i unosom raspona datuma) te prema dijelu IP adrese (klikom na gumb Po dijelu IP adrese) nakon čega se u sva tri slučaja prikazuje izvještaj o pretragama u ovisnosti o zadanom filtru. Takav je način korištenja (za primjer pretrage po datumu) vizualno prikazan na slici 8.

Drugi administrativni način je unos novih web stranica te njihovo indeksiranje. To se postiže klikom na gumb Pregled i unos stranica s kojim se pokreće obrazac Web stranice, unosom URL adresa stranica (ostale podatke nije nužno upisivati jer će se one dobiti automatski indeksiranjem), te na kraju klikom na gumb Indeksiraj stranice na obrascu Tražilica kojim će se novounesene stranice indeksirati. Nakon indeksiranja (postupak može potrajati u ovisnosti o broju unešenih web stranica) ponovnim otvaranjem obrasca Web stranice može se vidjeti da su za sve unešene URL adrese unešeni i ostali podaci koji su dinamički dohvaćeni s web stranice. Ovaj je način korištenja vizualno prikazan na slici 9.

## 6. Zaključak

Aplikacija FOI tražilica i odgovarajuća baza podataka pokušaj su izrade web tražilice korištenjem MS Access alata uz SQL i VBA. Izrađeni su odgovarajući model baze podataka koji je implementiran u SQL-u te su oblikovani odgovarajući obrasci, izvještaji i upiti koji podržavaju vrlo jednostavnu tražilicu.

Aplikacija naravno nije usporediva s profesionalnim tražilicama te su mnoge funkcionalnosti pojedostavljene, poput indeksiranja koje se svodi na puko zbrajanje pojavljivanja pojedinih riječi.

Također, aplikacija ne omogućuje izravan višekorisnički rad, nego su korisničko i administrativno sučelje implementirani na istom obrascu. Višekorisnički rad bi se mogao postići dodavanjem početnog obrasca za autentikaciju pa prema ulozi korisnika preusmjeriti na različite obrasce za rad. To bi povlačilo dodavanje dodatnih tablica za korisnike i njihove uloge te niz upita i funkcija koje bi vodile računa o tome da svaka vrsta korisnika ima drukčije sučelje.

Takve i slične funkcionalnosti mogu se implementirati u nekim budućim inačicama aplikacije FOI tražilica.

# Popis literature

- [1] StackOverflow. (2018). VBA: getHTML as custom function - Avoid several HTML retrivals, **adresa:** <https://stackoverflow.com/questions/52513408/vba-gethtml-as-custom-function-avoid-several-html-retrivals> (pogledano 10. 1. 2019).
- [2] —, (2012). Stripping HTML From A String, **adresa:** <https://stackoverflow.com/questions/12804067/stripping-html-from-a-string> (pogledano 11. 1. 2019).
- [3] Bytes. (2007). Create Dynamic Report using VBA, **adresa:** <https://bytes.com/topic/access/insights/696050-create-dynamic-report-using-vba> (pogledano 12. 1. 2019).
- [4] Microsoft. (2017). Office VBA Reference, **adresa:** <https://docs.microsoft.com/en-us/office/vba/api/access.application.followhyperlink> (pogledano 13. 1. 2019).

# Popis slika

1.	UML Class dijagram baze podataka . . . . .	2
2.	Implementacijski model . . . . .	4
3.	Obrazac Web stranica . . . . .	5
4.	Obrazac Tražilica . . . . .	11
5.	Izvještaj - pregled loga . . . . .	17
6.	Dinamički izvještaj - rezultati pretrage . . . . .	18
7.	Primjer korištenja - pretraživanje . . . . .	19
8.	Primjer korištenja - pretraživanje loga . . . . .	19
9.	Primjer korištenja - unos i indeksiranje stranica . . . . .	20