

HTTP server aplikacija

CS101

Uvod u objektno orijentisano programiranje

Projektna dokumentacija

19.01.2019.

Student: Nikola Tasić

Mentor: Jovana Jović

Sadržaj

[1. Predlog teme 3](#_Toc491866984)

[2. Opis funkcionalnosti 3](#_Toc491866985)

[3. Struktura aplikacije 3](#_Toc491866986)

[4. Korisničko uputstvo](#_Toc491866987) 4

[5. Zaključak](#_Toc491866988) 5

[6. Literatura](#_Toc491866989) 5

# Predlog teme

Veoma prost HTTP server koji može služiti za šerovanje fajlove preko lokalne mreže ili čak preko interneta uz određena podešavanja.

# Opis funkcionalnosti

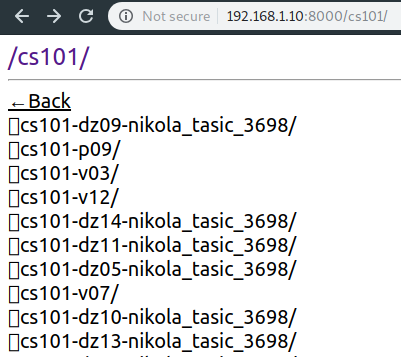
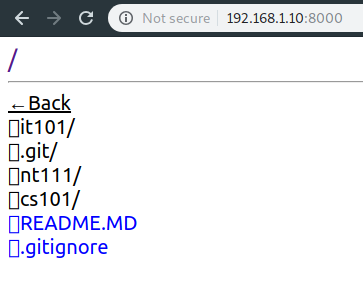
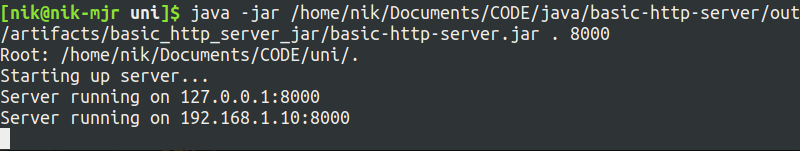
Aplikacija nakon pokretanja “servira” fajlove iz izabranog početnog foldera. Pristupon na IP adresu računara uz odgovarajući port vidimo strukturu fajlova i foldera u tom folderu. Klikom na fajl se pokreće njegovo preuzimanje a klikom na folder se ulazi u izabrani folder uz mogućnost vraćanja unazad.

# Struktura aplikacije

Server se pokreće preko Main klase koja ima ulogu da parsira argumente sa komandne linije i na osnovu toga set-uje odgovarajuce atribute u objektu servera koji kreira. Server klasa ima ulogu upravljanja "handler-ima" koji imaju ulogu davanja odgovora na zahteve klijenata. U ovoj osnovnoj implementaciji postoji samo jedan handler pod nazivom RootHandler koji prihvata sve zahteve koji dolaze na server. Ostavljenja je mogucnost pravljenja novih handlera koji mogu da budu speicijalizovani za odgovaranje na druge metode ili druge path-ove. Handler klasa ima zadatak da na osnovu path-a iz zahteva koji je upućen server pokuša da nađe odgovarajući fajl na sistemu i njega posalje klijentu. On to radi uz pomoc funkcija u HandlerUtils klasi koje citaju niz bajtova sa diska koji odgovara traženom fajlu i odredjuju njegov odgovarajući "Content-Type" header koji je neophodan za validan HTTP odgovor. Ako HTTP zahtev traži folder ova klasa ce uz pomoc statickih metoda iz ClientUtils klase ipisati korisniku u pretraživacu sadržaj foldera koji je zahtevan uz sve mogućnosti kao sto su vraćanje nazad prelaženje u neki drugi folder ili preuzimanje fajla.

# Korisničko uputstvo

Aplikaciju korisnik pokreće sa komandne linije ili uz podrazumevane vrednosti duplim klikom na .jar fajl aplikacije. Mogući parametri za pokretanje iz komandne linije su "--noindex" koji sprečava automatsko prikazivanje "index.html" stranice ukoliko je ona sadržana u zahtevanom folderu. "--cors" opcija dodaje Cross-Origin Request Sharing headere koji omogucavajuda vise aplikacija na istom domenu komunicira medjusobno. "--silent" opcija sprečava prkazivanje pristižućih zahteva na server na komandnoj liniji. Folder koji će biti "hostovan" od strane servera je prvi parametar iz komandne linije nakon čega se može uneti i specifični port na kojem želimo da se pokrene server. Podrazumevani parametri za folder i port su trenutni folder u kojem se nalazi aplikacija ili odakle je pozivamo i port 8000. Dobijanje help dialoga možemo postići unošenjem jednog od ovih parametara na komandnoj liniji. "-h", "-help", "--help". Pristupamo server tako što u, na primer, pretrživaču zahtevamo IP adresu računara na kojem server radi i port na kojem smo pokrenuli server. Na primer: "[http://192.168.1.10:8000/](http://192.168.200.10:8000/)".



\*na windowsu folderi i fajlovi imaju odgovarajuce ikonice

# Zaključak

Dizajnirana je aplikacija koje je prost HTTP server pravljen po uglednu na slične aplikacije koje postoje a napisane su u popularnim programskim jezicim tipa python, javascript, golang, etc. Obzirom na to da nisam nasao ni jedan primer sliče bazične aplikacije odlučio sam da to bude tema projketa. Aplikacija bi mogla da se unapredi dodavanjem broja podržanih Content-Type headera, dodavanje opcije za HTTPS(SSL), malo bolji client interfejs etc...

# Literatura

*1. StackOverflow.com*

*2.GitHub.com*