

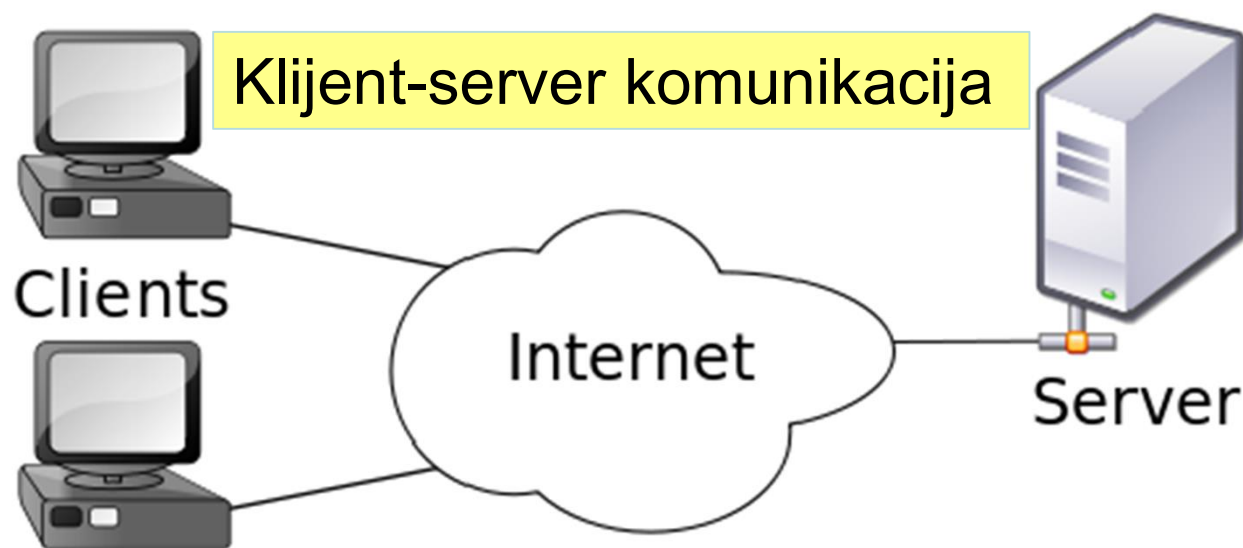
Telekomunikacioni servisi i tehnologije

Profesor dr Miroslav Lutovac

- **Lekcija 9: Internet klijent-server aplikacije, IP adresiranje**
(A. Zeković)

Klijent-server aplikacije

- U slučaju klijent-server aplikacija, server je taj koji obezbeđuje servise (DNS server, SMTP server, web server i slično) na zahtev klijenta
- Ilustracija klijent-server komunikacije data je na slici



DNS (Domain Name System), sistem koji pretvara imena računara (hostnames) u IP adrese

SMTP (Simple Mail Transfer Protocol) je osnovni protokol sloja aplikacija za elektronsku poštu, koristi TCP transfer podataka protokola

Telnet

- Telnet spada u najstarije klijent-server aplikacije (port 23)
- Tekstualno je orijentisana i namenjena za dvosmernu komunikaciju između klijenta i servera
- Ova aplikacija je namenjena za na primer logovanje na udaljeni server ili za pokretanje programa na udaljeni računar
- Po definicije Telnet ne koristi enkripciju, što znači da su u slučaju presretanja sve informacije vidljive i razumljive
- Protokol Secure Shell (SSH) (port 22) koristi kriptografiju za udaljeno logovanje pomoću linijskih komandi i omogućava sigurnije povezivanje mreže

Internet protokoli

Sloj aplikacije

BGP • DHCP • DNS • FTP • HTTP • IMAP •
IRC • LDAP • MGCP • NNTP • NTP • POP •
RIP • RPC • RTP • SIP • SMTP • SNMP •
SOCKS • SSH • Telnet • TLS/SSL • XMPP

Transportni sloj

TCP • UDP • DCCP • SCTP • RSVP • ECN

Internet sloj

IP • IPv4 • IPv6 • ICMP • ICMPv6 • IGMP •
IPsec

Sloj veze

ARP • NDP • OSPF • Tunneling protocol
• L2TP • PPP • Media access control • Ethernet
• DSL • ISDN • FDDI

Telnet je mrežni protokol unutar grupe internet protokola. Namena je uspostavljanje dvosmernog osmobitnog komunikacionog kanala između 2 umrežena računara. Koristi se da osigura korisniku jednog računara sesiju za korišćenje komandne linije na drugom računaru; naziv potiče od Telephone Network

Telnet predstavlja jedan od najstarijih protokola sloja aplikacije

Podaci koji se razmenjuju nisu zaštićeni; presretanjem se mogu protumačiti → SSH (Secure Shell) naslednik telneta

SMTP i POP3

- Za aplikaciju elektronske pošte koriste se odvojeni serveri za dolaznu i odlaznu poštu
- Odlazna pošta koristi SMTP (Simple Mail Transport Protocol) koji sluša na portu 25
- Najčešće korišćeni dolazni protokol za elektronsku poštu je POP3 (Post Office Protocol verzija 3) koji koristi port 110
- POP3 služi da preuzme mailove sa udaljenog mail servera i da ih smesti lokalno

DHCP i DNS

- Među posebno važne protokole aplikacionog sloja spadaju i DHCP (port 67/68) i DNS (port 53) serveri
- U najviše korišćene aplikacione protokole spada HTTP

DNS (Domain Name System), sistem koji pretvara imena računara (hostnames) u IP adrese

DHCP (Dynamic Host Configuration Protocol), protokol za dinamičko konfigurisanje računara je skup pravila koji omogućava uređajima na računarskoj mreži da traže i dobiju IP adresu od DHCP servera, da pribavi automatski deljenu adresu i sazna dodatne informacije kao što je adresa njegovog rutera za prvi skok i adresa njegovog DNS servera. DHCP je u stanju da automatizuje mrežne aspekte, nazvan plug-and-play protokol

HTTP

- HTTP je skraćenica za Hypertext Transfer Protocol
- Ovaj protokol koriste web pregledači (browser, internet pretraživači) prilikom komunikacije sa web serverima
- Prilikom posete web stranici, računar prvo prevodi ime stranice, na primer facebook.com u njegovu IP adresu, a zatim šalje serveru poruku, čiji sadržaj prate pitanja ili **zahtevi**
 - Pošalji mi sadašnju osnovnu stranu
 - Pošalji mi login stranu
 - Pošalji mi difoltni (podrazumevani) pregled (sadržaj)
- Na sledećoj slici ilustrovana je komunikacija između web klijenta koji traži informacije i servera koji odgovara sa zahtevanom web stranom

HTTP zahtev



Poruka koju klijent šalje izgleda slično sledećem tekstu:

```
GET / HTTP/1.1  
Host: www.google.com  
...
```

HTTP je asinhroni protokol: klijent šalje zahtev serveru; server vraća klijentu poruku sa odgovorom (sintaksa poruke je u HTTP specifikaciji)

- GET označava kako računari dobijaju informacije
- Oni kreiraju zahtev u tekstualnoj formi koji izgleda slično prethodnoj poruci
- Ova jednostavna poruka se otvara na strani servera, koji onda odgovara na zahtev
- Znak / odmah nakon GET označava da se traži koreni direktorijum, odnosno najviši direktorijum
- Da bi se sa kompletnom sintaksom posetio sajt potrebno je da se napiše `http://www.facebook.com/`, tj. adresa sa završnim znakom /, koji označava da se traži najviši direktorijum, odnosno difoltna (podrazumevana) stranica
- Deo poruke `HTTP/1.1` znači da je u pitanju verzija 1.1 za HTTP

Odgovor na HTTP zahtev

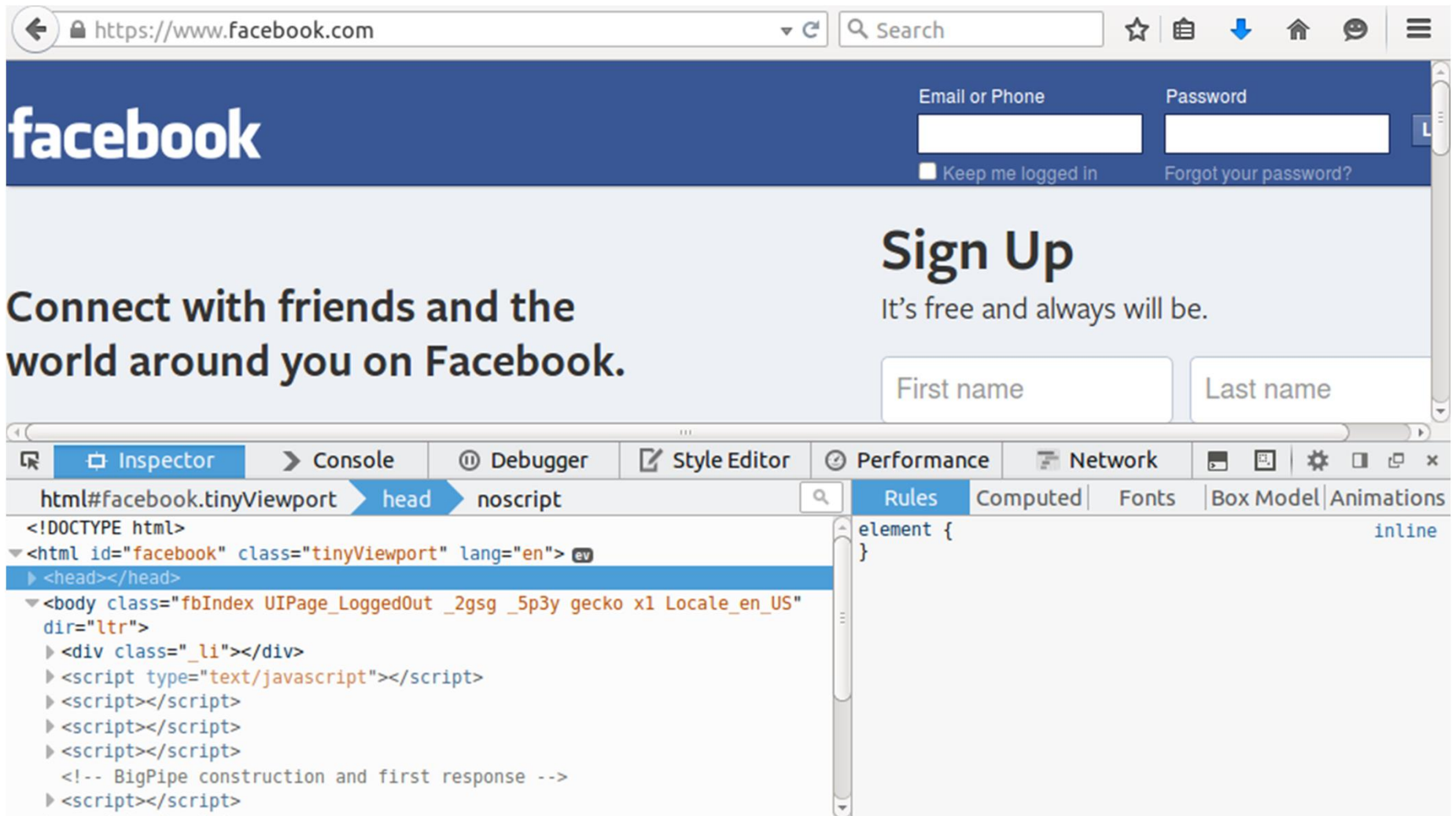


U odgovoru na zahtev web server šalje:

```
HTTP/1.1 200 OK
Content-type: text/html
. . .
```

- U prvoj liniji odgovora potvrđuje se verzija 1.1 za komunikaciju, dok je 200 statusni kod koji označava da je sve u redu sa komunikacijom i da server ima web stranicu koja je tražena
- U drugoj liniji poruke koja je dobijena kao odgovor od servera, označeno je da je odgovor tipa tekst (mogao je biti i slika, video)
- Poslednji deo ... zapravo označava samu poruku sa kojom server odgovara
- U web pratraživačima moguće je pogledati deo posvećen razvoju web strane i razmeni poruka između servera i klijenta
- Ovaj deo je obično označen u meniju pretraživača kao Developer Tools

Elementi web strane



- Na prethodnoj slici u donjem delu vide se stranica napisana u jeziku HTML (Hypertext Markup Language)
- Ovo nije programski jezik, pošto ne sadrži funkcije i petlje, već se označava kao markup jezik u smislu da sadrži tagove i attribute koji služe da kažu pretrživaču na koji način da prikaže sadržaj na ekranu (da li je boldovani tekst, koje je boje)
- Na sledećoj slici u donjem delu dat je Network segment u kome se mogu uočiti zahtevi
- Prvi je GET zahtev koji je dobio odgovor 200, koji označava da je stranica uspešno pronadena
- Kako je ovaj zahtev selektovan sa desne strane se vide detalji zahteva, na primer da je sadržaj tipa tekst

HTTP poruke

The screenshot shows a web browser displaying the Facebook sign-up page. The address bar shows the URL `https://www.facebook.com`. The page content includes the Facebook logo, login fields for "Email or Phone" and "Password", a "Keep me logged in" checkbox, a "Forgot your password?" link, and a "Sign Up" section with the text "Connect with friends and the world around you on Facebook." and "It's free and always will be." Below this are input fields for "First name" and "Last name".

The Network developer tool is open at the bottom, showing a list of requests. The first request is a 200 GET to `/` from `www.facebook.com`. The selected request details are shown on the right:

- Request URL:** `https://www.facebook.com/`
- Request method:** GET
- Status code:** 200 OK
- Response headers (1.356 KB):**
 - Cache-Control:** "private, no-cache, no-store, must-revalidate"
 - Content-Encoding:** "gzip"
 - Content-Type:** "text/html"
 - Date:** "Wed, 29 Apr 2015 20:38:09 GMT"

The bottom of the Network tool shows a summary: "26 requests, 1,163.23 KB, 387.63 s".

- Svaki put kada otvorimo neku web stranicu traži se njena adresa, a zatim joj se šalje GET zahtev u kome takode stoji i IP adresa onoga ko je napravio zahtev
- Zatim ako je stranica pronadena, dobija se traženi sadržaj.
- Neki od mogućih statusnih kodova su:
 - 200 OK
 - 301 trajno premešteno
 - 302 pronadeno
 - 401 autorizovani pristup
 - 403 zabranjen pristup
 - 404 nije pronadeno
 - 500 greška na serveru

HTML

- Hypertext Markup Language (HTML) je interpretatorski jezik koji definiše na koji način će biti prikazana web stranica
- Primer HTML koda ima za zadatak da prikaže poruku Zdravo svima!
- Ako se sadržaj koda smesti u editor dokument i pokrene pomoću web pretraživača poruka će biti prikazana
- Obavezno je da dokument ima ekstenziju .html da bi interpretacija bila moguća

```
<!DOCTYPE html>
<html>
  <head>
    Zdravo!
  </head>
  <body>
    Zdravo svima!
  </body>
</html>
```

Početak HTML koda je obeležen sa <html>, a zatvaranje ovog taga sa </html>

- Ostatak koda je podeljen u dva dela glava i telo, čiji tagovi takode imaju otvaranje i zatvaranje, po principu primenjenom u prethodnom kodu.
- U okviru koda može biti dodata i slika

```
<!DOCTYPE html>
<html>
  <head>
    Slika
  </head>
  <body>
    <img src= "mojaSlika.jpeg" />
  </body>
</html>
```

Potrebno je da slika sa istim imenom postoji u folderu u kome se nalazi kod kada se ovaj kod pokreće. Slika se zatvara samo pomoću />, a ne sa /img>

Dinamični web

- Za kreiranje dinamičnih web stranica potrebno je uključiti i jezike kao što su PHP, JavaScript, jezike za baze podataka SQL i slično
- U okviru koda mogu biti uključeni i linkovi

```
<!DOCTYPE html>
<html>
  <head>
    Zdravo!
  </head>
  <body>
    <a href="http://www.facebook.com/">
      http://www.google.com </a>
  </body>
</html>
```

U ovom primeru, na stranici je bilo ispisano www.google.com (moglo bi da bude bilo šta što unesemo na ovo mesto), dok bi link zapravo vodio na www.facebook.com (ovde bi mogao da bude bilo koji url). Očekivano je da se ova dva podatka poklapaju, a na ovom primeru je ilustrovano koliko je jednostavno staviti lažni link.

IP adresiranje

- Osnove IP adresiranja biće ilustrovane na primeru adrese 192.168.60.0/24
- IP adresa u verziji 4 sadrži 32 bita, podeljena u 4 okteta (8 bita) koji se u decimalnom zapisu razdvajaju tačkom
- Prvi oktet u primeru je 192
- Za njegovo binarno predstavljanje koriste se biti 0 ili 1, pri čemu biti vrede $2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0$, (kaže se i težinski faktor), odnosno 128 64 32 16 8 4 2 1
 - Ako je potrebno predstaviti vrednost 192 binarno važiće 1100000, pošto je $1 \times 128 + 1 \times 64 = 192$
- Ostali članovi bi u zbiru bili jednaki nula
 - Na sličan način bi se za decimalnu vrednost 168 dobilo binarno 10101000

Adresa 192.168.60.0/24

- Oznaka /24 predstavlja netmask-u i označava da netmask-a za prvih 24 bita ima binarne vrednosti 1, dok za preostale bite ima vrednost 0
- Ukoliko bi se ove binarne vrednosti zapisale decimalno rezultat bi bio 255.255.255.0
- Netmask-a obeležava koji biti u okviru jedne mreže ne mogu da se menjaju
- U ovom slučaju to je prvih 24 bita. što znači da za sve adrese u ovoj mreži prva 24 bita moraju da budu, zapisano decimalno 192.168.60, dok su poslednjih 8 bita promenljivi

- Dobra praksa je oktete koji imaju sve ili deo promenljivih bita zapisati binarno.
- U ovom slučaju je to ceo poslednji oktet, koji binarno glasi 00000000.
- Ako su svi promenljivi biti jednaki nula u pitanju je adresa mreže i ona se ne može koristiti za korisničke adrese
- Takođe, za korisničke adrese se ne može koristiti ni broadcast adresa koja se dobija kada se za sve promenljive bite dodeli vrednost 1 (u ovom slučaju 11111111)
- Ovde je kompletna decimalna adresa koja se koristi kada se poruka šalje svima u mreži (broadcast)
192.168.60.255

- Sve preostale kombinacije između mrežne i broadcast adrese predstavljaju korisničke adrese
- Prva korisnička adresa se dobija kada su promenljivi biti 00000001 (kompletna IP adresa decimalno 192.168.60.1), dok se poslednja korisnička adresa dobija kada su promenljivi biti 11111110 (kompletna IP adresa decimalno 192.168.60.254)
- Kako je broj promenljivih bita u ovoj mreži 8 broj mogućih adresa je 2^8 , dok je broj korisničkih adresa $2^8 - 2$ zbog mrežne i broadcast adrese

Servlet i korišćenje Apache Tomcat servera



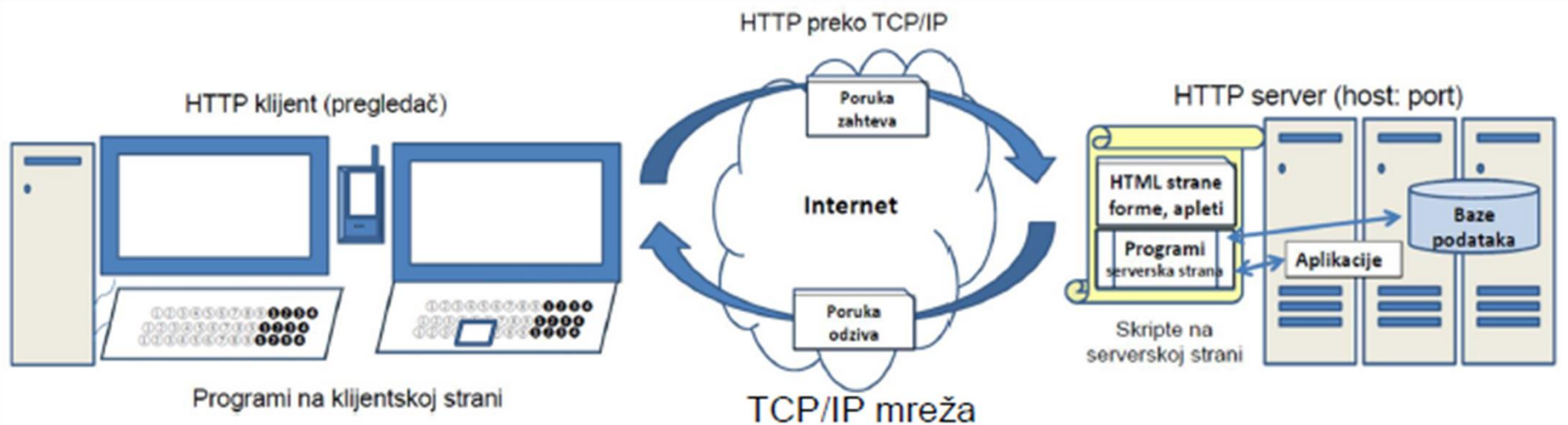
- Servlet tehnologija se koristi **za kreiranje veb aplikacija**
- Postoji **na serveru** i generiše **dinamičnu veb** stranicu
- Servlet tehnologija je **robustna** i **skalabilna** (Java jezik)
- Postoji mnogo interfejsa i klasa u API servletima (Application Programming Interface) kao što su Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponse

- HTTP Server, 2. HTTP Client (web pregledač), 3. Database (baza podataka), 4. Client-Side programi, 5, Server-Side programi
- Tomcat je HTTP aplikacija koja radi preko TCP/IP
- HTTP protokol aplikacijskog sloja se pokreće preko TCP/IP; IP za usmeravanje i adresiranje jedinstvenom IP adresom za mašine na Internetu); TCP podržava multipleksiranje preko portova
- Tomcat server radi na specifičnim TCP portovima sa specifičnom IP adresom; podrazumevani TCP porta za HTTP protokol je 80
- Za testiranje HTTP servera, može se izabrati bilo koji broj neiskorišćenog porta između 1024 i 65535

Apache Tomcat je Java-orjentisan HTTP server koji može da izvršava specijalne java programe, "Java Servlet" i "Java Server Pages" (JSP), "Apache Software Foundation"

Tomcat 8.x: RI for **Servlet 3.1**, **JSP 2.3**, EL 3.0, & Java WebSocket 1.0

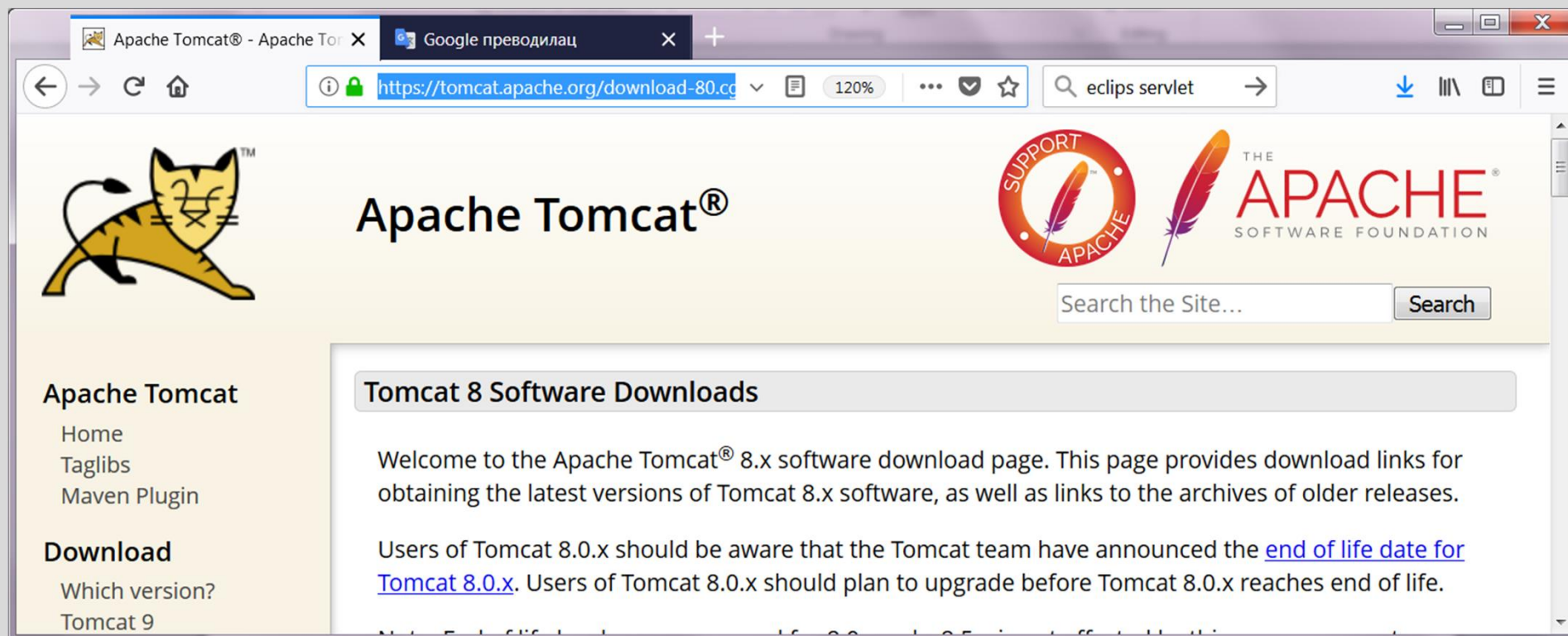
1. HTTP Server: Apache HTTP Server, Apache Tomcat Server, Microsoft Internet Information Server, Google Web Server...
2. HTTP Client (web pregledač): Chrome, FireFox, Internet Explorer...
3. Database (baza podataka): MySQL, Apache Derby, mSQL, SQLite...
4. Client-Side programi: napisani u HTML, JavaScript, VBScript, Flash...
5. Server-Side programi: napisani u Java Servlet/JSP, ASP, PHP, Python...



Slojevi	Jedinica	Protokoli
Aplikacija	Podatak	HTTP
Prezentacija	Podatak	SSL
Sesija	Podatak	
Transport	Segment Datagram	TCP
Mreža	Paket	IP
Sloj veze	Frejm (Okvir)	IEEE 802.11x
Fizički sloj	Bit	

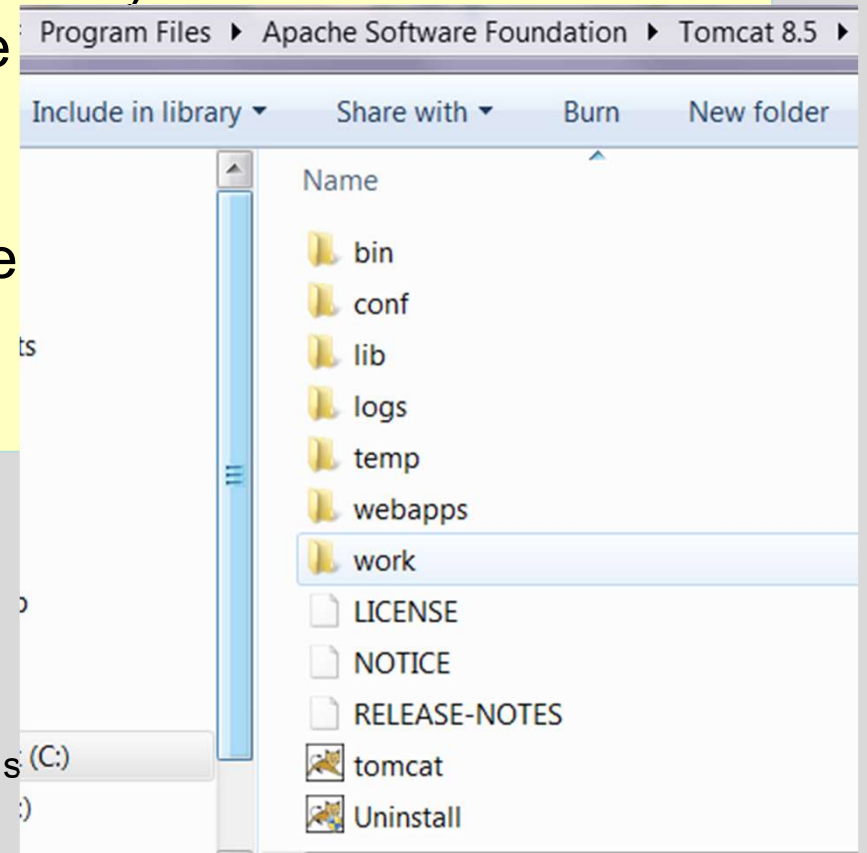
Multipleksiranje (port), re-transmisija
Adresiranje (IP adresa), rutiranje

Preuzeti instalaciju sa sajta Apache Tomcat
<https://tomcat.apache.org/download-80.cgi>



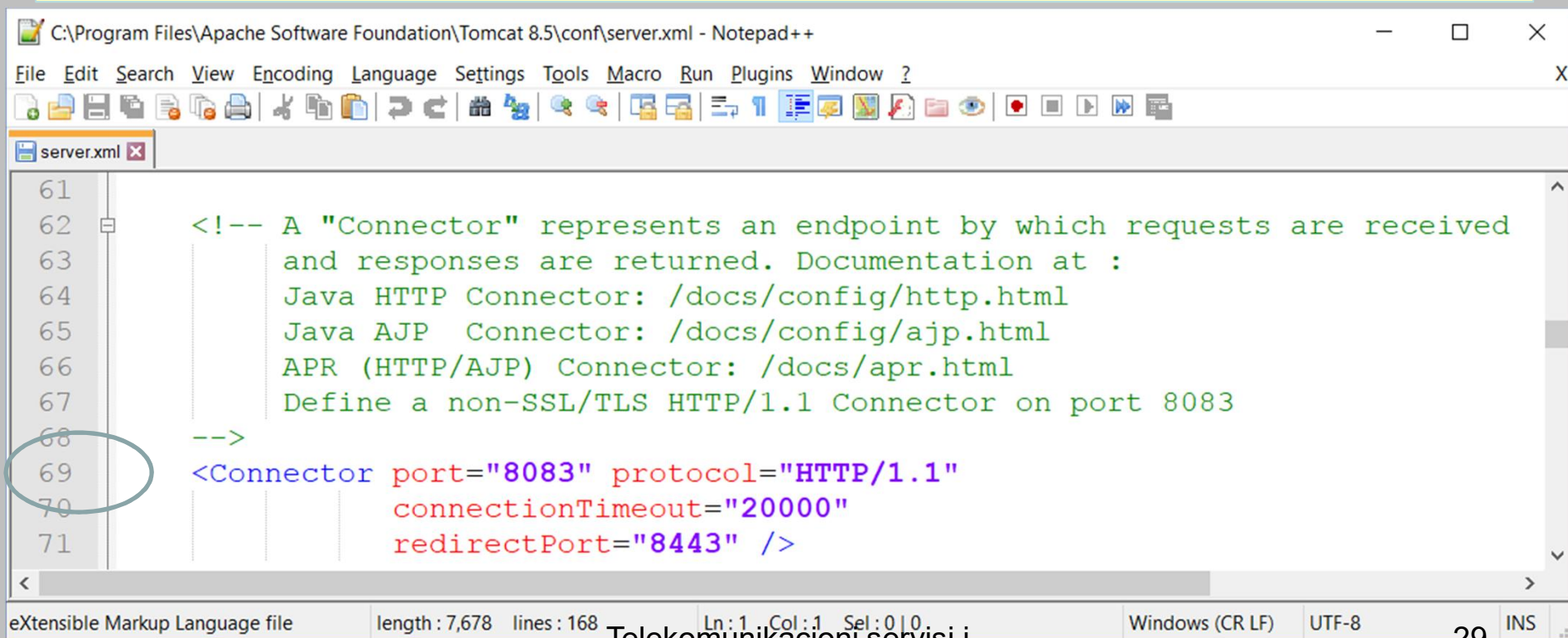
Sadržaj Tomcat foldera

- **bin**: sadrži binarne i skript fajlove, startup script (startup.bat for Windows), shutdown script (shutdown.bat for Windows)
- **conf**: sadrži konfiguracione fajlove; server.xml, web.xml, context.xml, tomcat-users.xml
- **lib**: sadrži Tomcat system JAR files, kojima pristupaju sve vebap; mogu da budu i eksterni JAR fajlovi (MySQL JDBC Driver)
- **logs**: Tomcat log fajlovi; error poruke
- **webapps**: sadrži vebap koje će se razvijati ; Webapp Archive
- **work**: Tomcat radni folder koji koriste JSP, JSP-to-Servlet konverzija
- **temp**: privremeni fajlovi



TCP Port Number

fajl "server.xml", u "conf" folderu Tomcat instalacionog direktorijuma
Podrazumevani broj TCP porta je 8080, može se izabrati bilo koji broj 1024 - 65535, koji ne koriste postojeće aplikacije
Izabere se **9999** u ovom slučaju; za proizvodnju se koristi port 80, koji je predefinisani za HTTP server kao podrazumevani broj porta



```
C:\Program Files\Apache Software Foundation\Tomcat 8.5\conf\server.xml - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
server.xml x
61
62 <!-- A "Connector" represents an endpoint by which requests are received
63      and responses are returned. Documentation at :
64      Java HTTP Connector: /docs/config/http.html
65      Java AJP  Connector: /docs/config/ajp.html
66      APR (HTTP/AJP) Connector: /docs/apr.html
67      Define a non-SSL/TLS HTTP/1.1 Connector on port 8083
68      -->
69 <Connector port="8083" protocol="HTTP/1.1"
70           connectionTimeout="20000"
71           redirectPort="8443" />
```

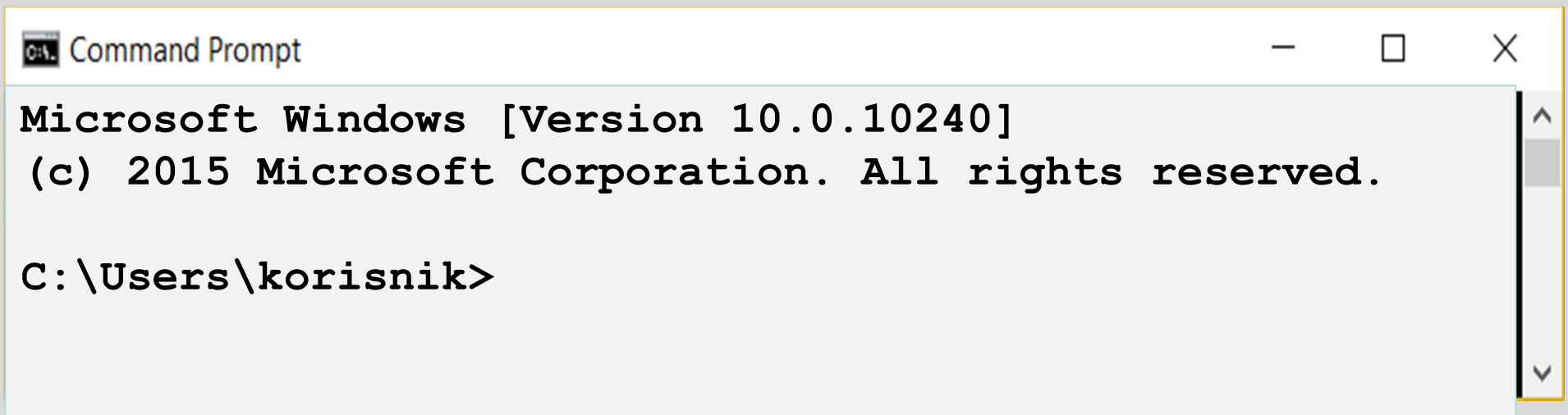
eXtensible Markup Language file length : 7,678 lines : 168 Ln : 1 Col : 1 Sel : 0 | 0 Windows (CR LF) UTF-8 INS

konfiguracija Tomcat servera

Izvršni i skript programi su u folderu "bin"
u Tomcat **instalacionom direktorijumu** (for Windows)

"d:\myProject\tomcat\bin"

Staruje se CMD shell (Command Prompt)



```
Command Prompt
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\korisnik>
```

Postavi se radni direktorijum na "<TOMCAT_HOME>\bin",
i startuje "startup.bat":

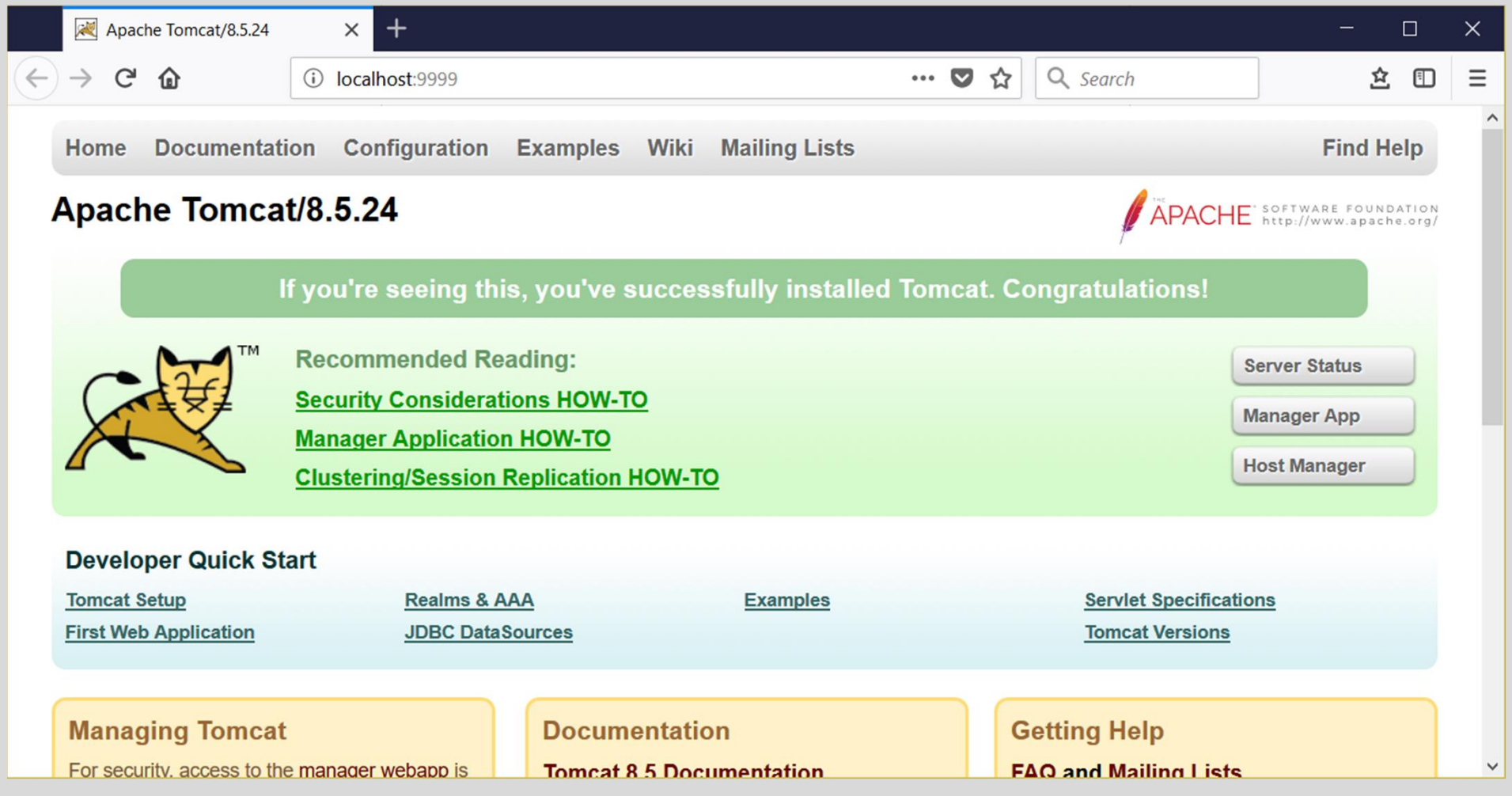
```
// promeni trenutni direktorijum u Tomcat "bin"  
// Podrazumevani Tomcat je instaliran u  
// "d:\myProject\tomcat"  
// 1. promeni drajver u d:  
// 2. promeni direktorijum u Tomcat "bin"  
// 3. startuj Tomcat server
```

```
d:  
cd \myProject\tomcat\bin  
startup
```

Start Server

Start a Client to Access the Server

Start a browser (as HTTP client); uneti URL
"http://localhost:9999" da bi se dobila Tomcat server welcome page



The screenshot shows a web browser window with the title "Apache Tomcat/8.5.24". The address bar displays "localhost:9999". The page content includes a navigation bar with links: Home, Documentation, Configuration, Examples, Wiki, Mailing Lists, and a "Find Help" button. The main heading is "Apache Tomcat/8.5.24". A green banner states: "If you're seeing this, you've successfully installed Tomcat. Congratulations!". Below this, on the left, is the Tomcat logo (a cat) and a section titled "Recommended Reading:" with links to "Security Considerations HOW-TO", "Manager Application HOW-TO", and "Clustering/Session Replication HOW-TO". On the right, there are three buttons: "Server Status", "Manager App", and "Host Manager". A "Developer Quick Start" section contains links to "Tomcat Setup", "First Web Application", "Realms & AAA", "JDBC DataSources", "Examples", "Servlet Specifications", and "Tomcat Versions". At the bottom, there are three yellow boxes: "Managing Tomcat" (with subtext "For security, access to the manager webapp is"), "Documentation" (with subtext "Tomcat 8.5 Documentation"), and "Getting Help" (with subtext "FAQ and Mailing Lists").

Apache Tomcat/8.5.24

localhost:9999

Home Documentation Configuration Examples Wiki Mailing Lists Find Help

Apache Tomcat/8.5.24

If you're seeing this, you've successfully installed Tomcat. Congratulations!



Recommended Reading:

- [Security Considerations HOW-TO](#)
- [Manager Application HOW-TO](#)
- [Clustering/Session Replication HOW-TO](#)

Server Status
Manager App
Host Manager

Developer Quick Start

- [Tomcat Setup](#)
- [First Web Application](#)
- [Realms & AAA](#)
- [JDBC DataSources](#)
- [Examples](#)
- [Servlet Specifications](#)
- [Tomcat Versions](#)

Managing Tomcat
For security, access to the manager webapp is

Documentation
Tomcat 8.5 Documentation

Getting Help
FAQ and Mailing Lists

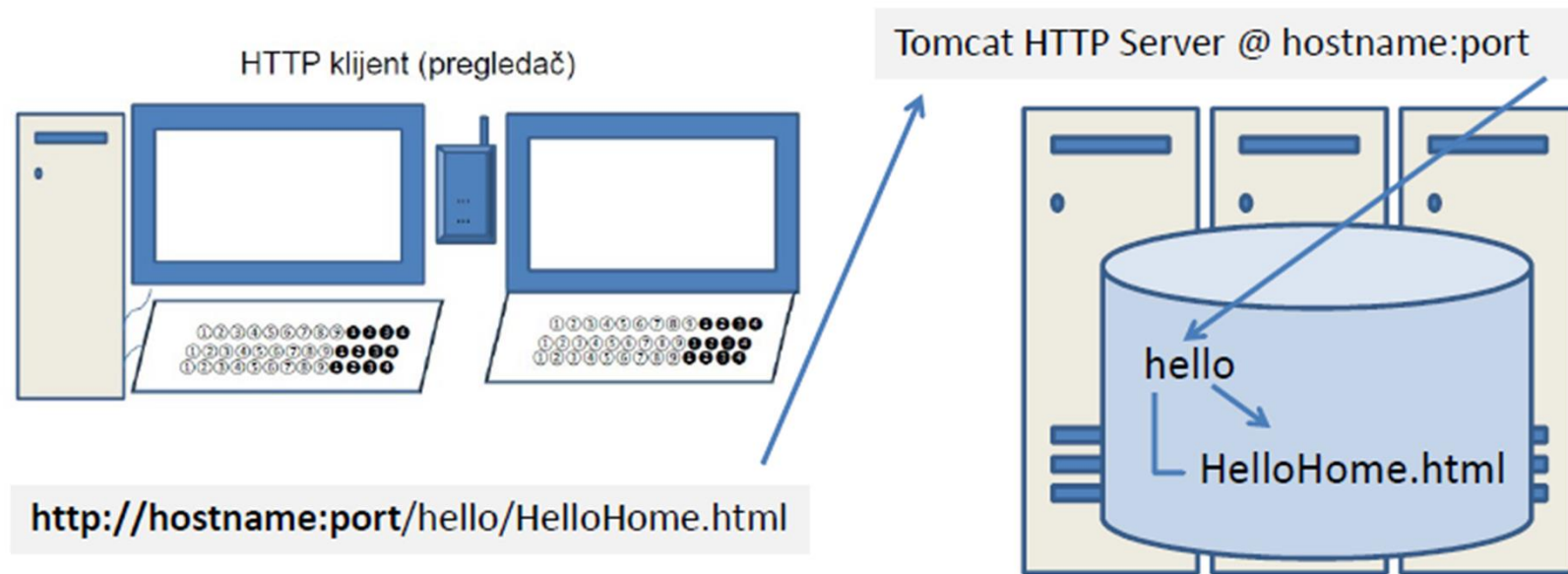
Client to Access the Server

URL "http://localhost:9999" za Tomcat server's welcome page
Hostname "localhost" (sa IP adresom 127.0.0.1)
je namenjen za testiranje lokalnih petlji unutar istog računara

Za korisnike sa drugih računara preko Interneta, koristi se
serverova IP adresa ili DNS ime domena ili hostname u formatu
"http://serverHostnameOrIPAddress:9999"

Na strani klijenta u pregledaču
unosí se adresa servera
hostname:port
naziv foldera (**hello**) gde se
traži fajl **HelloHome.html**

Na strani servera
(**hostname:port**) u osnovnom
direktorijumu se traži folder
(**hello**) gde se
traži fajl **HelloHome.html**
Taj fajl se šalje pregledaču

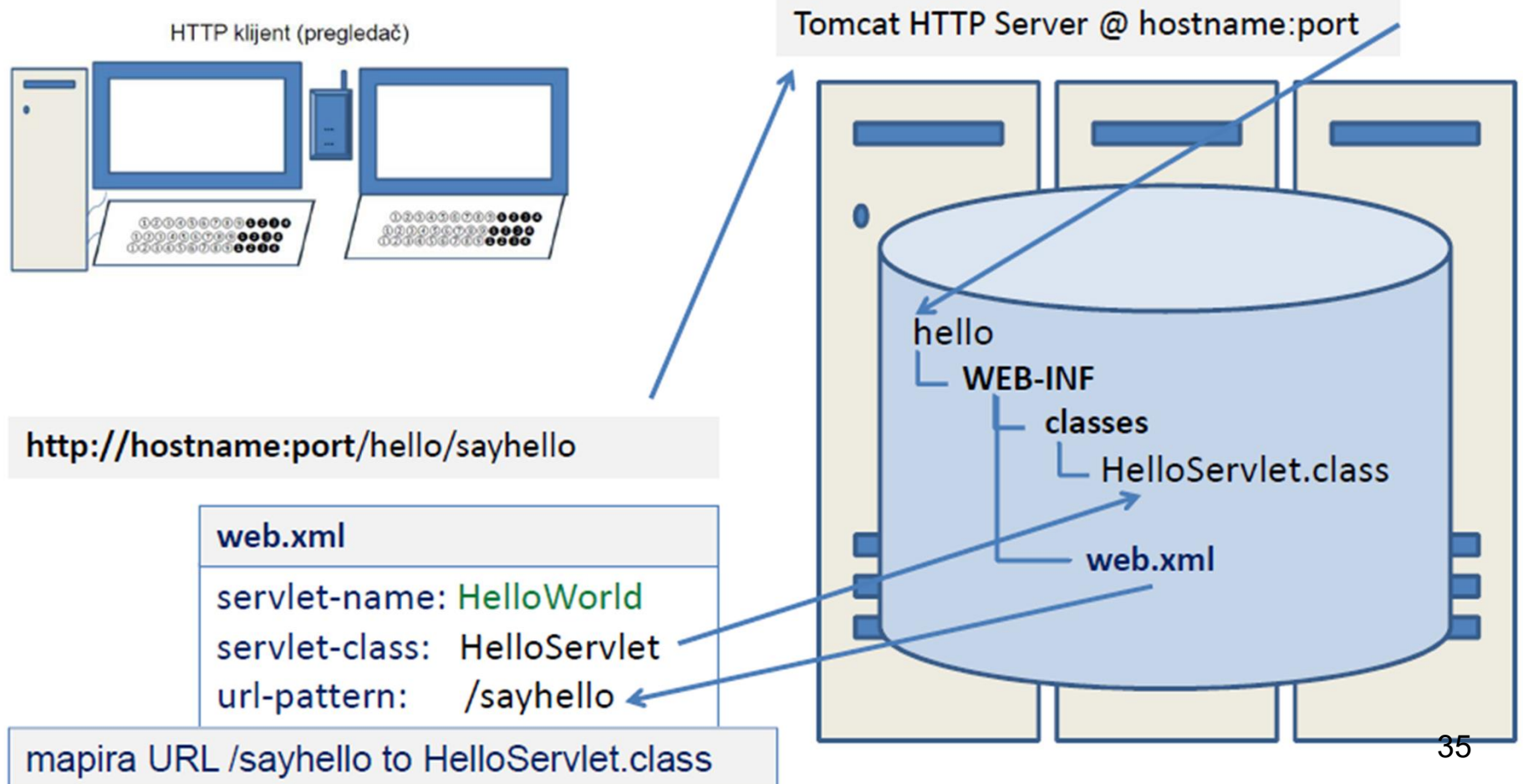


ako se ne unese naziv fajla, podrazumevani fajl koji se traži je **index.html**

ako nema fajla **index.html**, prikazuje se listing fajlova

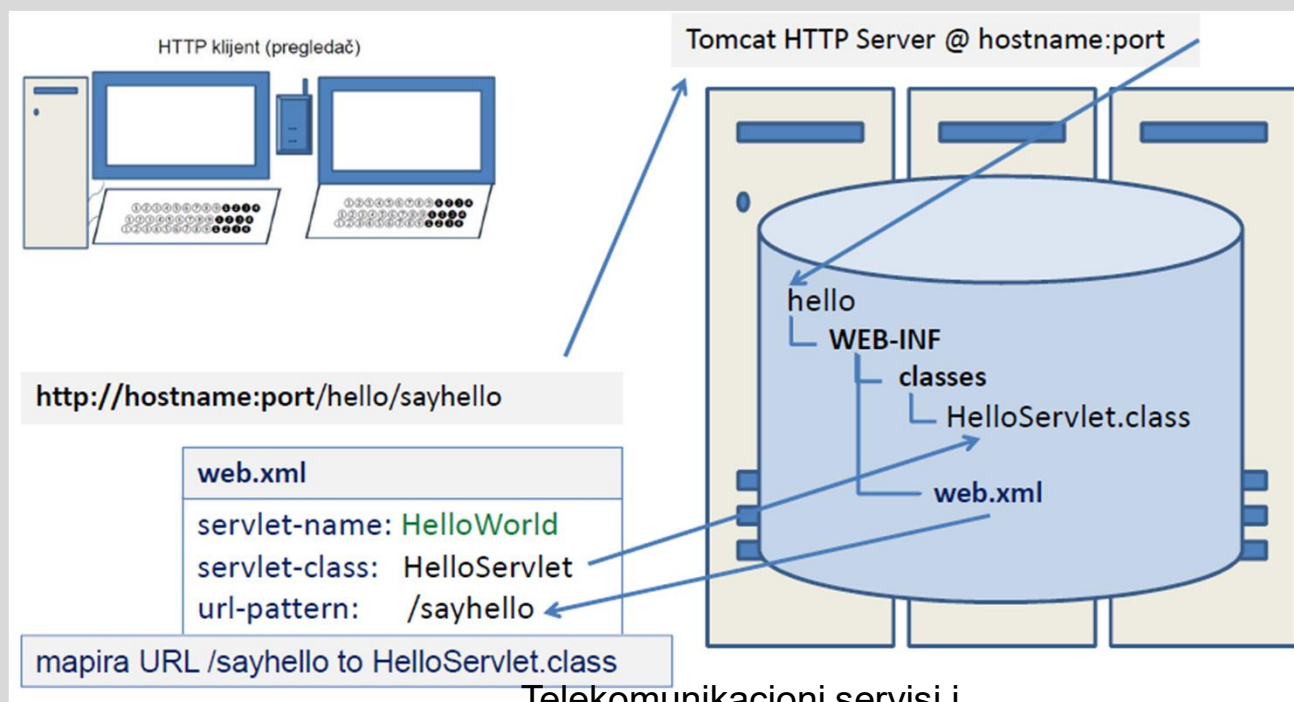
Na strani klijenta u pregledaču unosi se adresa servera **hostname:port** naziv foldera (**hello**) gde se traži servlet **sayhello**

Na strani servera (**hostname:port**) u osnovnom direktorijumu se traži folder (**hello**) u kome je folder **WEB-INF** i u njemu fajl **web.xml**, koji preusmerava **sayhello** servlet na **HelloServlet.class** java klasu.

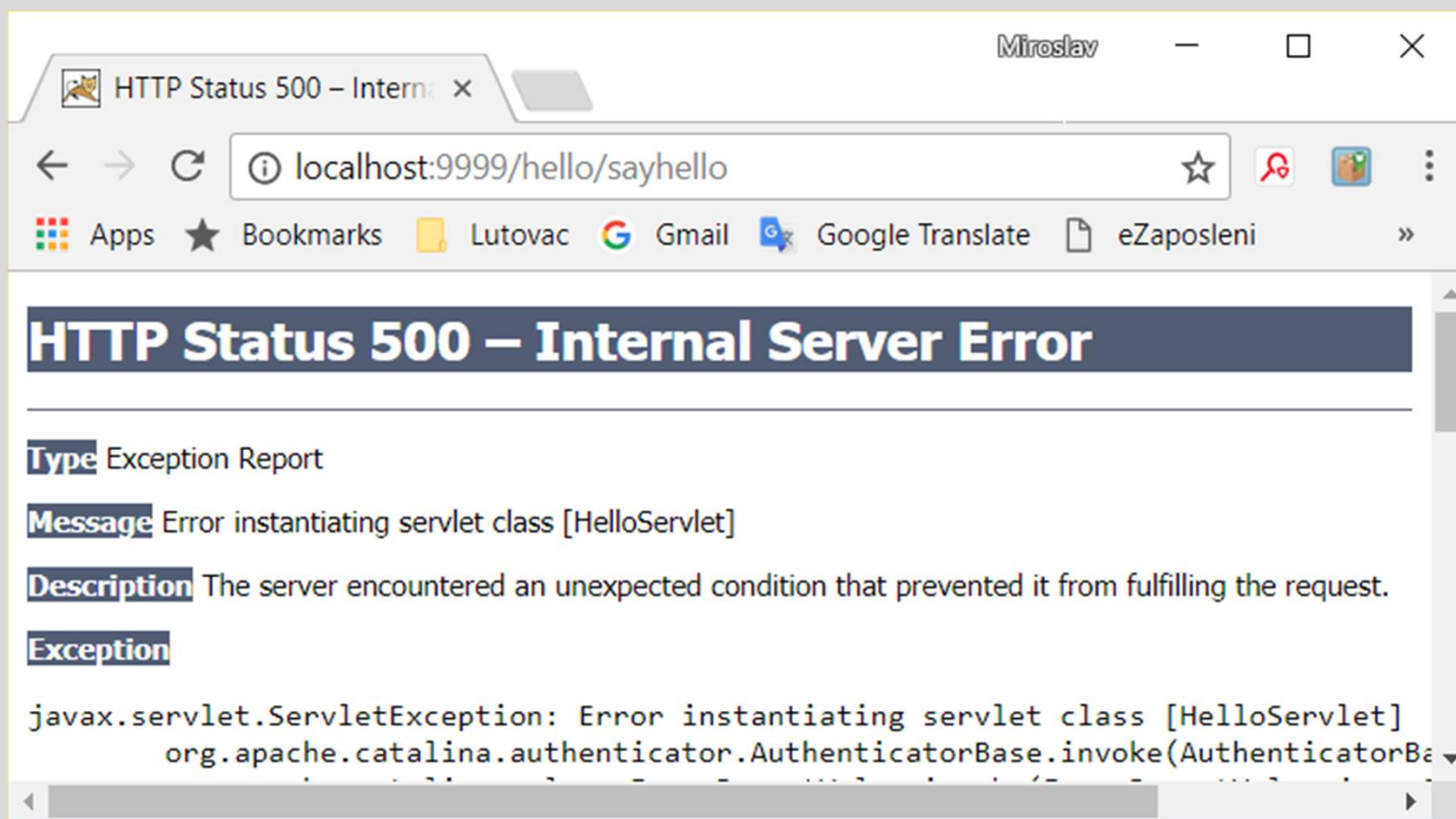


Nazivi koji se proizvoljno biraju su **hello**, **sayhello**, **HelloServlet**(.java, i .class), **HelloWorld**, a službene reči su **WEB-INF**, **classes**, **web.xml**, gde **web.xml** koji preusmerava servlet na java klasu.

Na osnovu adrese unete u pregledač (**sayhello**) u osnovnom direktorijumu prema definisanoj strukturi se pronalazi gde je fajl koji definiše gde se šta nalazi i šta treba da se izvrši



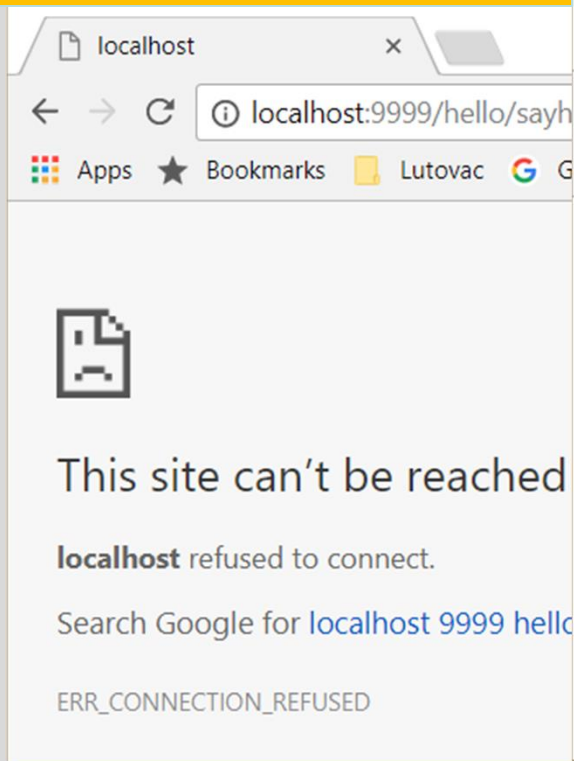
Ako postoji **HelloServlet.java** ali ne i **HelloServlet.class**,
pojaviće se greška
potrebno je kompajlirati .java program da bi se dobio .class fajl



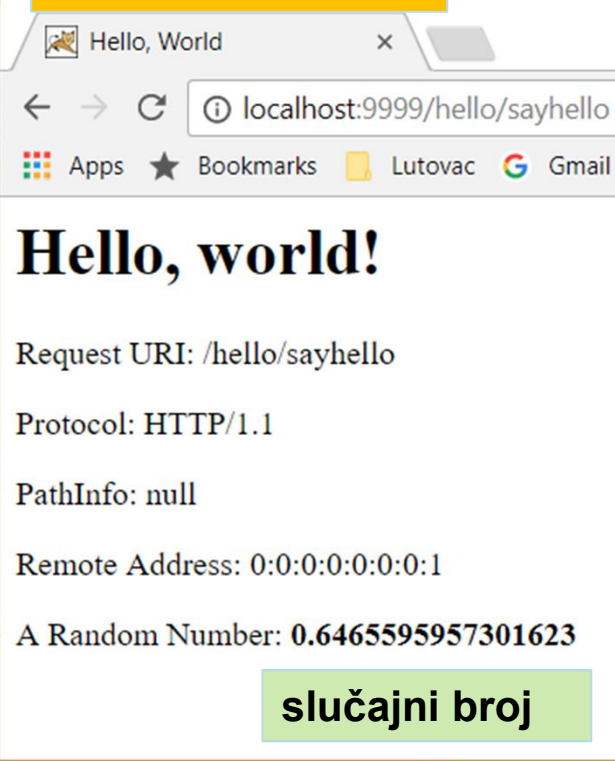
Da bi se pozvao servlet, koji je konfigurisan na strani servera, na strani klijenta se pokrene pregledač i unese adresa za zahtevani servlet na primer <http://hostname:port/hello/sayhello> , potrebno je da postoji konfiguracioni fajl na toj adresi servera

Pozivanje servleta

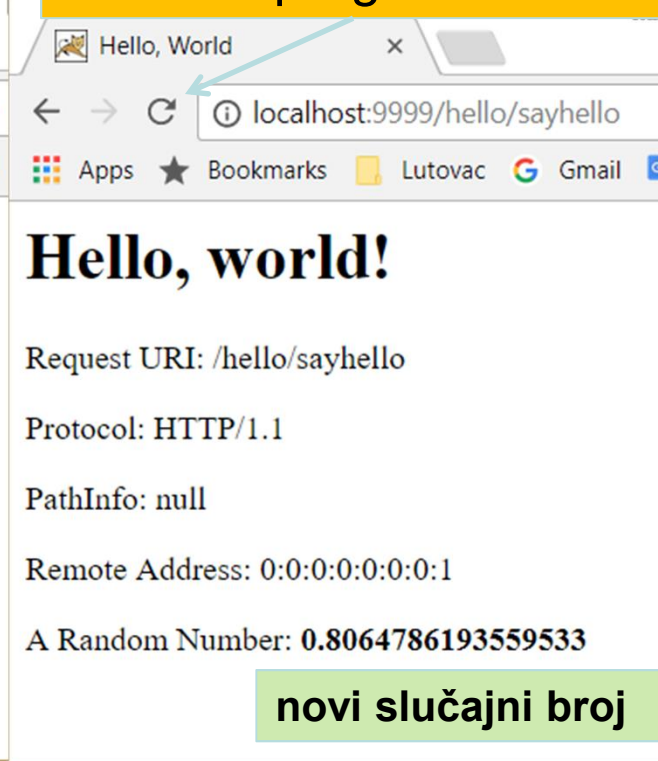
ako nije startovan server



startovan server



osvežen pregledač



U pregledaču Chrome, klikni na F12 da se omogući "developer tool". Izabрати "Network" paletu. Uneti URL ...

Accept: text/html,application/xhtml+xml...

Accept-Encoding: gzip, deflate, br

Accept-Language: en-US,en;q=0.9

Cache-Control: max-age=0

Connection: keep-alive

Host: localhost:9999

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Win64; x64) ...

```
sayhello x
```

Serving from the file system? Add your files into the workspace. [more](#)

```
<html>
<head><title>Hello, World</title></head>
<body>
<h1>Hello, world!</h1>
<p>Request URI: /hello/sayhello</p>
<p>Protocol: HTTP/1.1</p>
<p>PathInfo: null</p>
<p>Remote Address: 0:0:0:0:0:0:0:1</p>
<p>A Random Number: <strong>0.24954749523022168</strong></p>
</body></html>
```

▼ General

Request URL: http://localhost:9999/hello/sayhello
Request Method: GET
Status Code: 200
Remote Address: [::1]:9999
Referrer Policy: no-referrer-when-downgrade

▼ Response Headers [view source](#)

Content-Length: 287
Content-Type: text/html; charset=ISO-8859-1
Date: Fri, 16 Feb 2018 19:35:46 GMT

▼ Request Headers [view source](#)

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Cache-Control: max-age=0
Connection: keep-alive
Host: localhost:9999
Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.167 Safari/537.36

Za više detalja

```
GET http://localhost:9999/hello/sayhello HTTP/1.1
Host: localhost:9999
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.5
Cache-Control: max-age=0
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101 Firefox/52.0
```

**request message header;
no request message body**

Inspect request and response header and body

```
<html>
<head><title>Hello, World</title></head>
<body>
<h1>Hello, world!</h1>
<p>Request URI: /hello/sayhello</p>
<p>Protocol: HTTP/1.1</p>
<p>PathInfo: null</p>
<p>Remote Address: 0:0:0:0:0:0:0:1</p>
<p>A Random Number:
<strong>0.4480280769255568</strong></p>
</body></html>
```

response message header

```
HTTP/1.1 200 OK
Date: Wed, 22 Mar 2017 09:30:23 GMT
Content-Length: 286
Content-Type: text/html; charset=ISO-8859-1
```

response message body

Profesor dr Miroslav Lutovac

mlutovac@viser.edu.rs

Ova prezentacija je nekomercijalna.

Slajdovi mogu da sadrže materijale preuzete sa Interneta, stručne i naučne građe, koji su zaštićeni Zakonom o autorskim i srodnim pravima.

Ova prezentacija se može koristiti samo privremeno tokom usmenog izlaganja nastavnika u cilju informisanja i upućivanja studenata na dalji stručni, istraživački i naučni rad i u druge svrhe se ne sme koristiti –

Član 44 - Dozvoljeno je bez dozvole autora i bez plaćanja autorske naknade za nekomercijalne svrhe nastave:

(1) javno izvođenje ili predstavljanje objavljenih dela u obliku neposrednog poučavanja na nastavi;

- ZAKON O AUTORSKOM I SRODNIM PRAVIMA

("Sl. glasnik RS", br. 104/2009 i 99/2011)