

# Klijentske Web aplikacije

## Uvodno predavanje

Nikola Savanović

[nsavanovic@singidunum.ac.rs](mailto:nsavanovic@singidunum.ac.rs)

Dr. Mlađan Jovanović

[mjovanovic@singidunum.ac.rs](mailto:mjovanovic@singidunum.ac.rs)

# Sadržaj

## ◆ Klijentske Web aplikacije

- Principi rada i gradivni elementi-tehnologije

## ◆ Ciljevi predmeta

## ◆ Oblasti

## ◆ Smernice

## ◆ Ispit

## ◆ Literatura

# **Pre nego krenemo sa kursem ...**

Pitanje broj 1 koje studeni postavljaju:  
**Koju tehnologiju da izaberem ?**

# Kako držati korak sa tehnologijama ?

- ◆ Postoji veliki broj tehnologija ...
- ◆ Kako da znam koju da koristim ?
- ◆ Gde da učim o novim tehnologijama ?
- ◆ Da li će to što naučim uskoro biti zastarelo ?

**Odgovor: postavljate pogrešna pitanja !**

- ◆ Biti u koraku sa tehnologijama nije najvažnija stvar
- ◆ Tehnologija se ne menja suštinski vrlo često ili vrlo brzo

# Osnove se ne menjaju

- ◆ Tehnologije se ne menjaju tako brzo:
- ◆ Dobar deo Facebook sistema je još uvek napisan u PHP
- ◆ Većina Google sistema je napisana u C++/Java
- ◆ Nećete (i ne bi trebalo) ponovo pisati kod u drugoj tehnologiji svake ili svake 2. godine
- ◆ Nova tehnologija ne znači nužno i napredak
  - Greške, stabilnost, skalabilnost, pouzdanost, ...

# Pravo pitanje je:

- ◆ **Kako da izaberem tehnologiju koja će mi na najlakši i najefikasniji način pomoći da obavim zadatak koji imam ?**
- ◆ Ili:
  - ◆ Imate dovoljno znanja i iskustva da sami procenite da li je alat ili tehnologija korisna
  - ◆ Ili:
    - ◆ Nemate dovoljno znanja i iskustva da sami procenite
  - ◆ Konsultujete **validne** izvore
    - Pouzdane osobe sa iskustvom
    - Članke, knjige i slično (sa dobrom reputacijom)

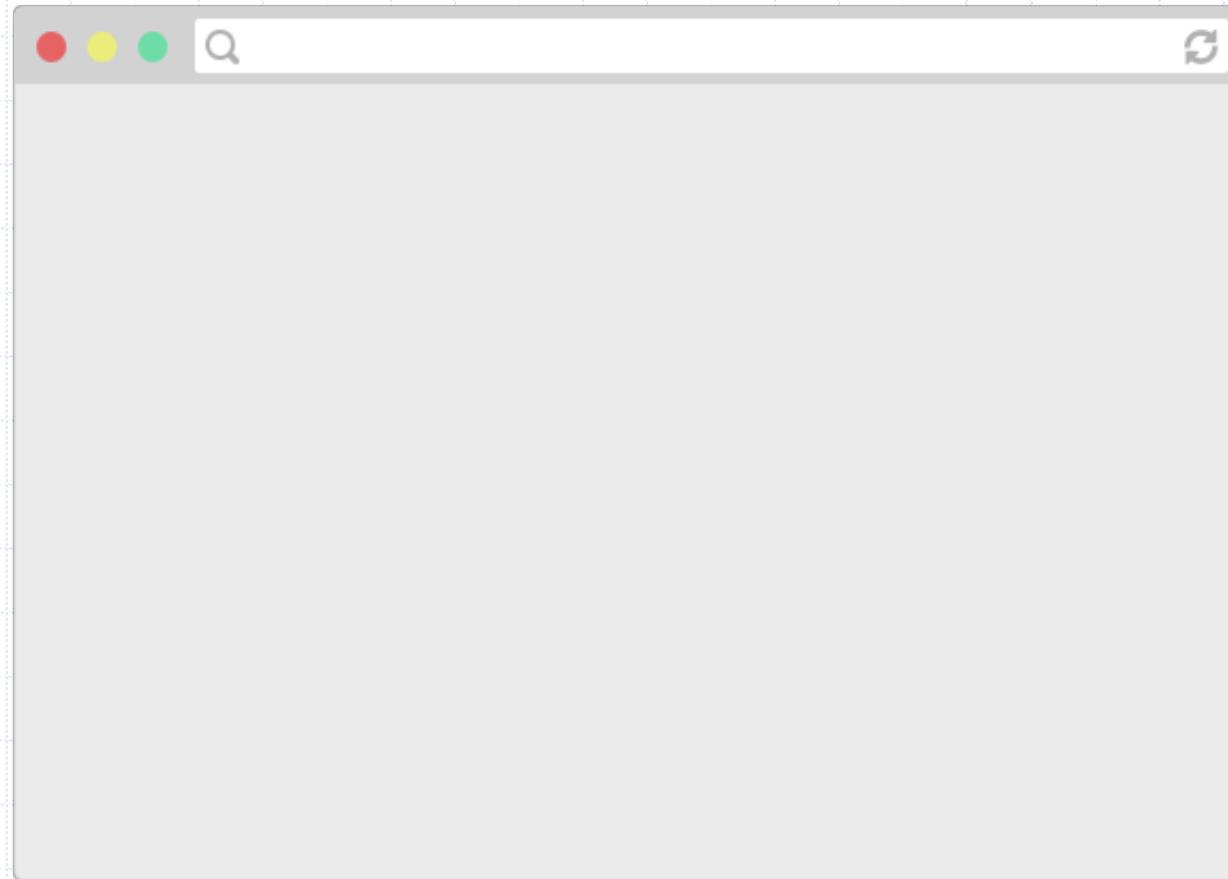
# Generalni savet

- ◆ Radite na tome da postanete dobar **inženjer**
- ◆ **Apstraktno** razmišljanje na nivou praktičnog **problema** koji rešavate
  - Problem-solving skills
- ◆ Naučite da rešite problem korišćenjem **različitih** tehnologija - frontend, backend, web, mobile, ...
  - Koje učite na studijama
- ◆ U praksi:
- ◆ Radite N godina kao full-time inženjer softvera sa drugim (dobrim) inženjerima
  - Čak i posle 1. godine rada vaše inženjerske veštine će zнатно uznapredovati
- ◆ Na ovaj način razvijate ne samo vaše znanje i veštine, već i sposobnost za procenu tehnologije

# Gde je ovaj kurs u **svemu tome** ?

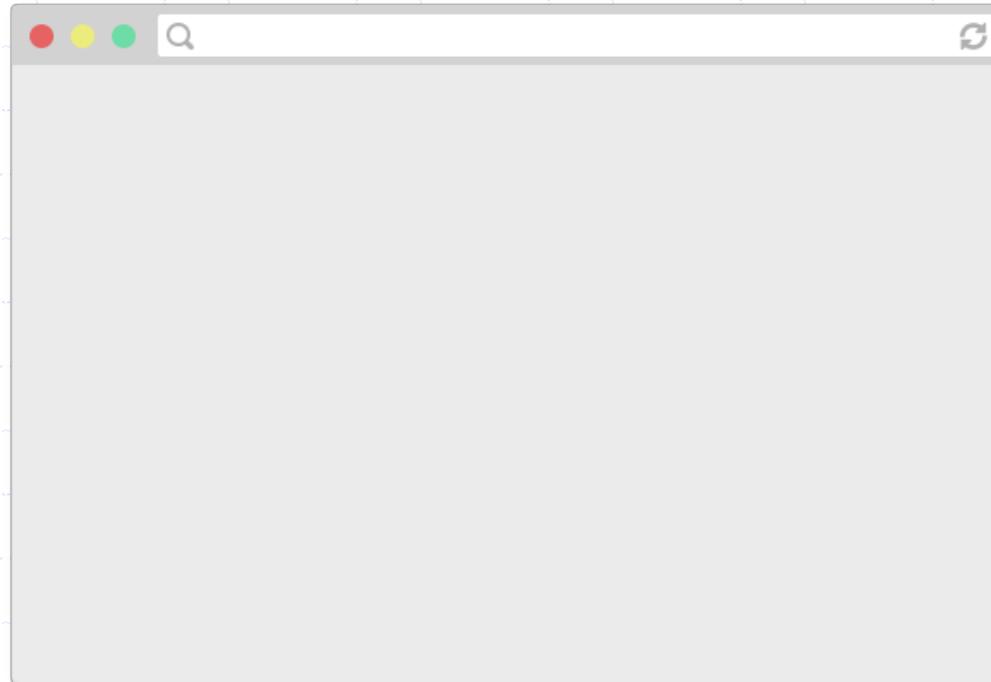
- ◆ Mi vas upoznajemo sa osnovnim **principima** i **konceptima** dobrih Web tehnologija za kreiranje klijentskih aplikacija
- ◆ Kako ?
- ◆ Kroz predavanja: **razumevanje** principa i koncepata
- ◆ Kroz vežbe: **primena** principa i koncepata
- ◆ Podjednako važni:
  - ◆ Bez razumevanja nema pravilne primene
  - ◆ **Bez primene i samostalnog rada nema učenja !**
  - ◆ Dugoročan cilj: lako ovladavanje novim tehnologijama zbog toga što će biti zasnovane na naučenim

# Kako rade Web stranice ?



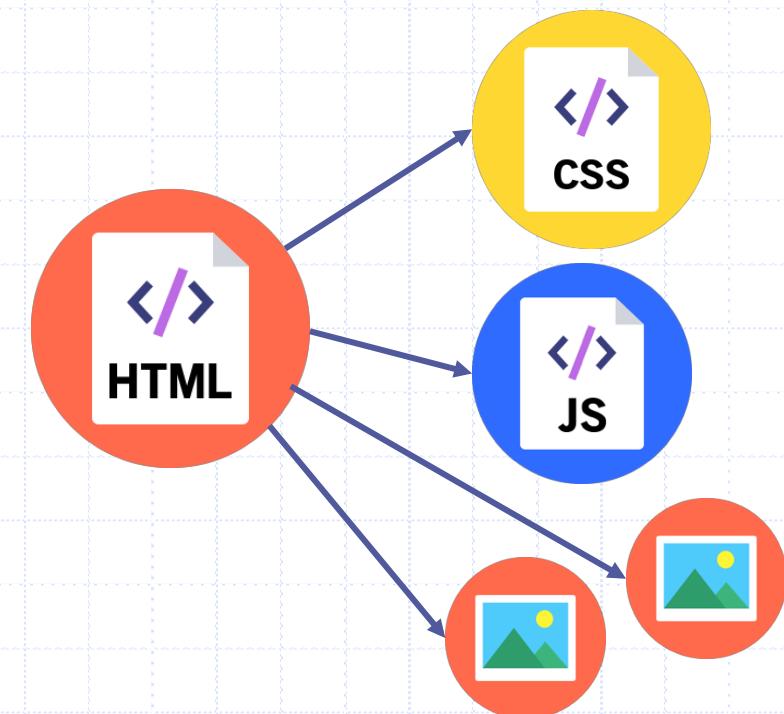
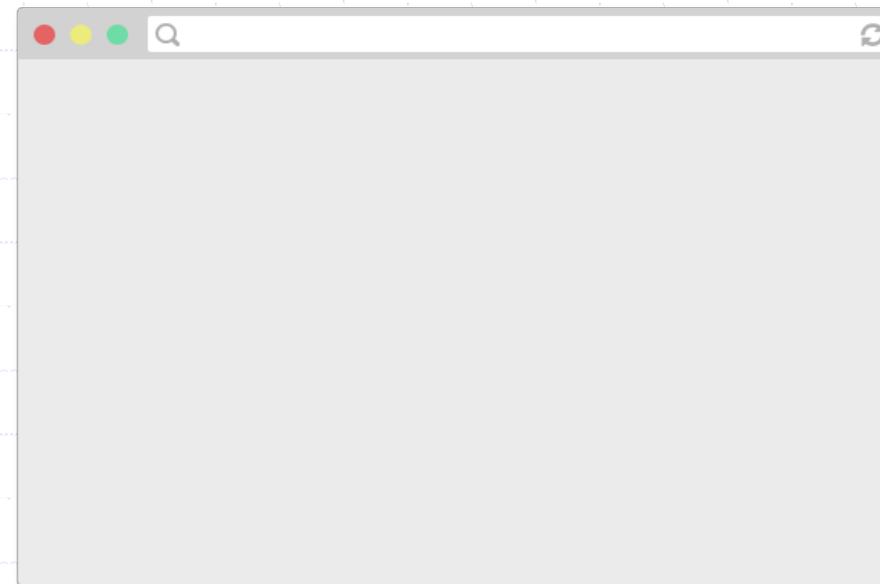
Pretraživači su aplikacije koje prikazuju Web stranice  
Na primer, Chrome, Safari, IE, Firefox, ....

# Kako rade Web stranice ?



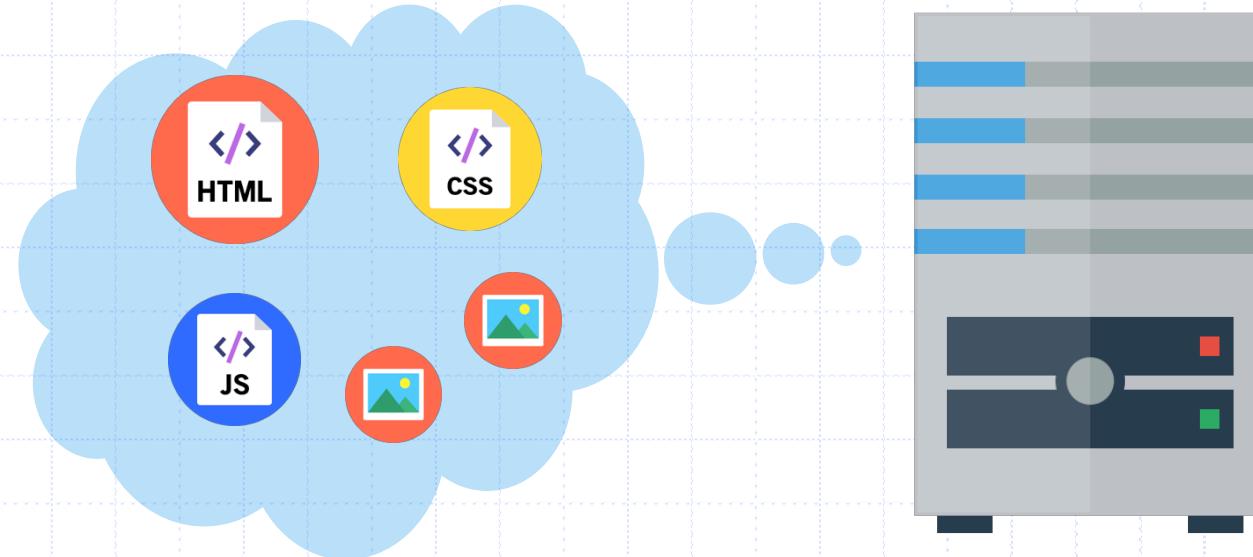
Web stranice su napisane u deklarativnom (markup) jeziku koji se zove HTML. Drugim rečima, pretraživač prikazuje Web stranicu tako što čita i interpretira HTML elemente

# Kako rade Web stranice ?



HTML datoteka može imati veze (linkove) ka drugim resursima, kao što su slike, video, kao i JavaScript i CSS datotekama koje pretraživač takođe učitava i interpretira

# Kako rade Web stranice ?



Web server je program koji se izvršava na računaru koji isporučuje Web stranice kao odgovore na zahteve klijenata.

Stranice mogu biti unapred pripremljene (statičke) ili generisane dinamički, po zahtevu

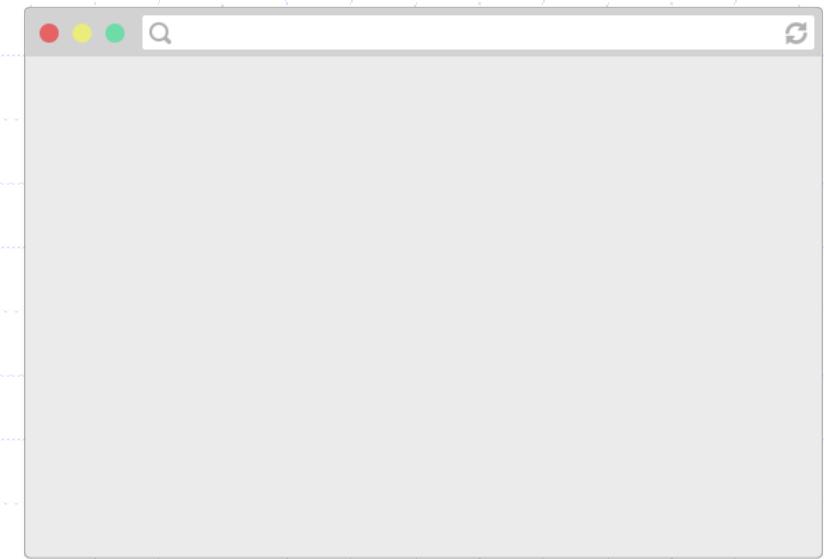
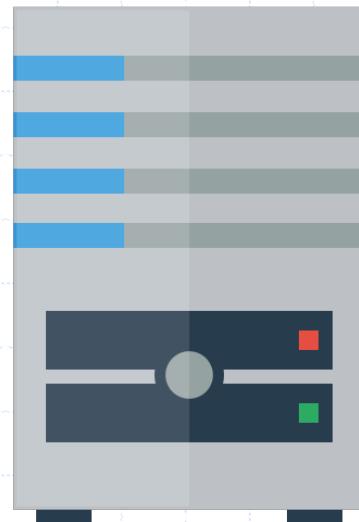
# Kako rade Web stranice ?

1. Unosite adresu Web stranice kao URL, koji predstavlja adresu HTML datoteke na Internetu



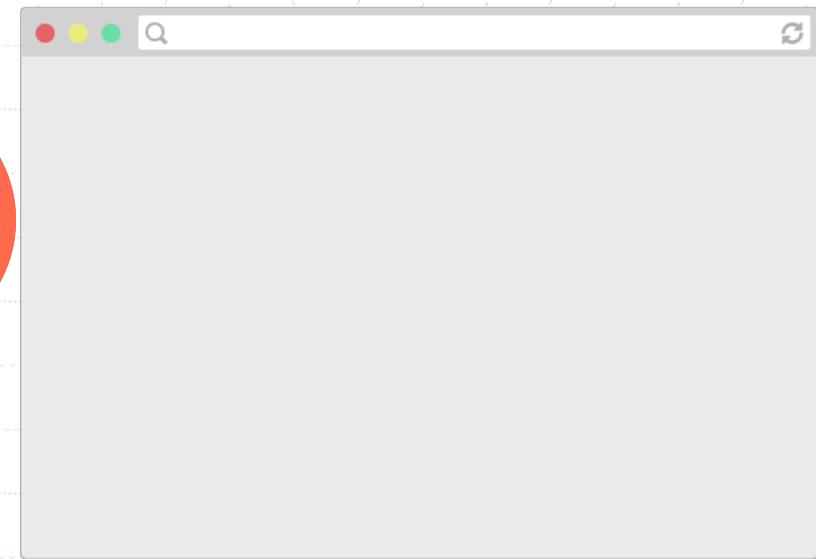
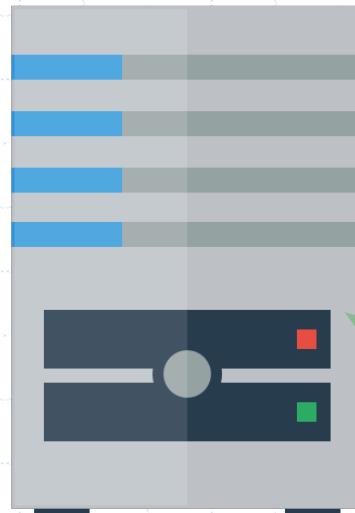
# Kako rade Web stranice ?

2. Pretraživač traži od web servera koji čuva dokument da isti pošalje



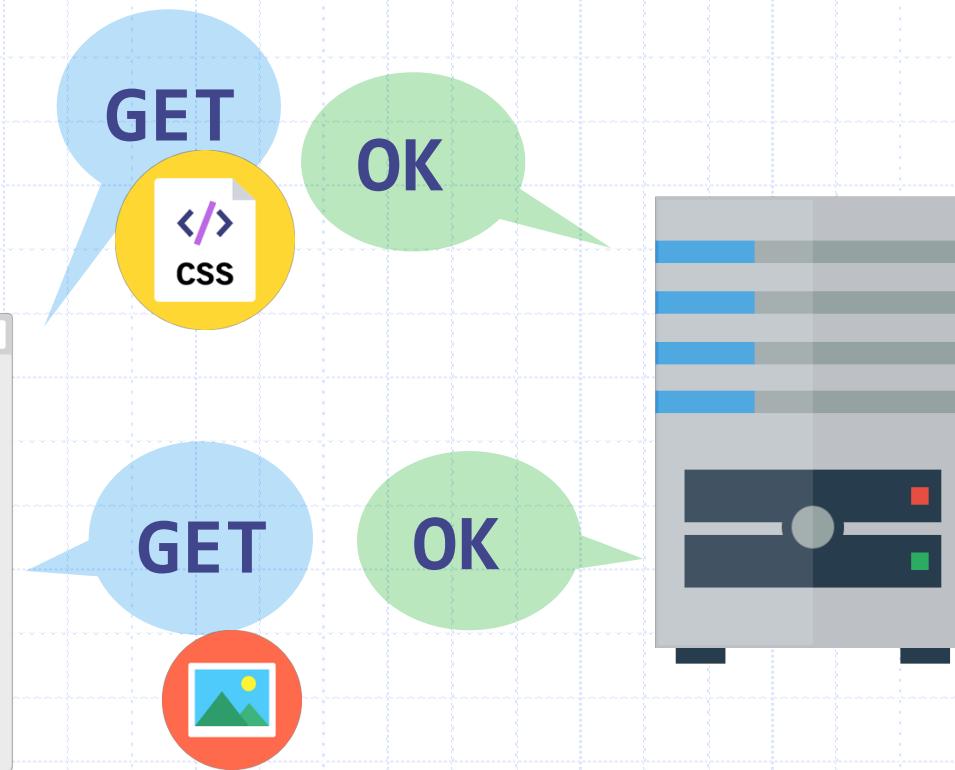
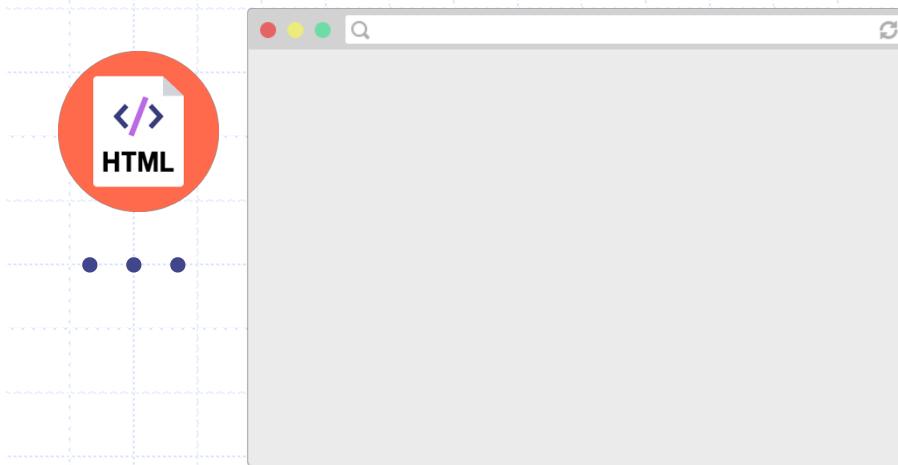
# Kako rade Web stranice ?

3. Web server odgovara pretraživaču  
i šalje traženu HTML datoteku



# Kako rade Web stranice ?

4. Pretraživač čita i interpretira HTML datoteku, nalazi ugrađene resurse i iste takođe traži od servera



# Kako rade Web stranice ?

5. Web stranica je učitana kada su dobavljeni svi resursi i prikazana



Izvor: Holmes, S., (2019). Getting MEAN with Mongo, Express, Angular, and Node. Manning Publications, 2nd Ed.

# Kurs o klijentskim Web aplikacijama

- ◆ Tehnologije za kreiranje savremenih Web aplikacija sa naglasom na klijentsku stranu
- ◆ Stek tehnologija
  - Pretraživač
  - Web server
  - Skladište podataka
- ◆ Cilj: razvoj funkcionalne i upotrebljive Web aplikacije
- ◆ Kako se uče tehnologije koje će biti obrađene na kursu ?
  - ◆ Učenje pokušajima (learn by doing)
  - ◆ Samostalan rad van časova vežbi i predavanja !

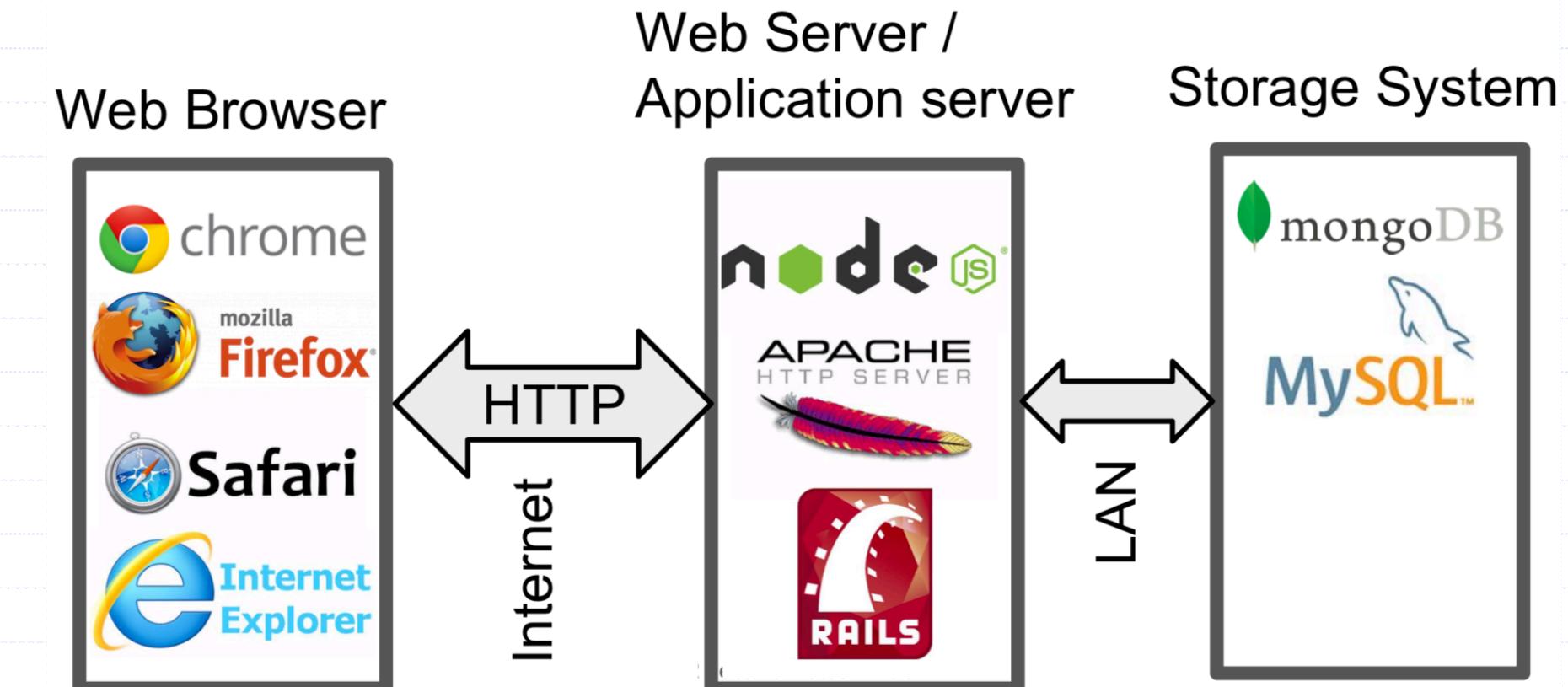
# Tehnologije i koncepti

- ◆ HTML/CSS/JavaScript – Sadržaj, izgled, ponašanje, ponovna upotrebljivost
- ◆ Document Object Model (DOM) – Struktura dokumenta, tj. Web stranice
- ◆ Angular.js – MVC (Model-View-Controller) obrazac, Single page aplikacije
- ◆ Node.js – Serverske funkcionalnosti
- ◆ HTTP/AJAX – Komunikacija
- ◆ Cookies/Sessions – Stanje
- ◆ DB (Database) – Skladište podataka

# Struktura kursa

- ◆ **Lekcije** su fokusirane na **objašnjenje** elemenata tehnologije
- ◆ **Vežbe** su fokusirane na **korišćenje** elemenata tehnologije
  
- ◆ Kolokvijum 1 – praktičan zadatak iz oblasti obrađenih do prve kolokvijumske nedelje
- ◆ Kolokvijum 2 i Završni ispit – semestralni projekat na zadatu temu

# Stek tehnologija Web aplikacije



# MEAN Stek



express



# MEAN Stek



# MEAN



A Client-Side (Browser) Framework which allows you to build Single-Page-Applications (SPA)

Render UI with Dynamic Data

Handle User Input

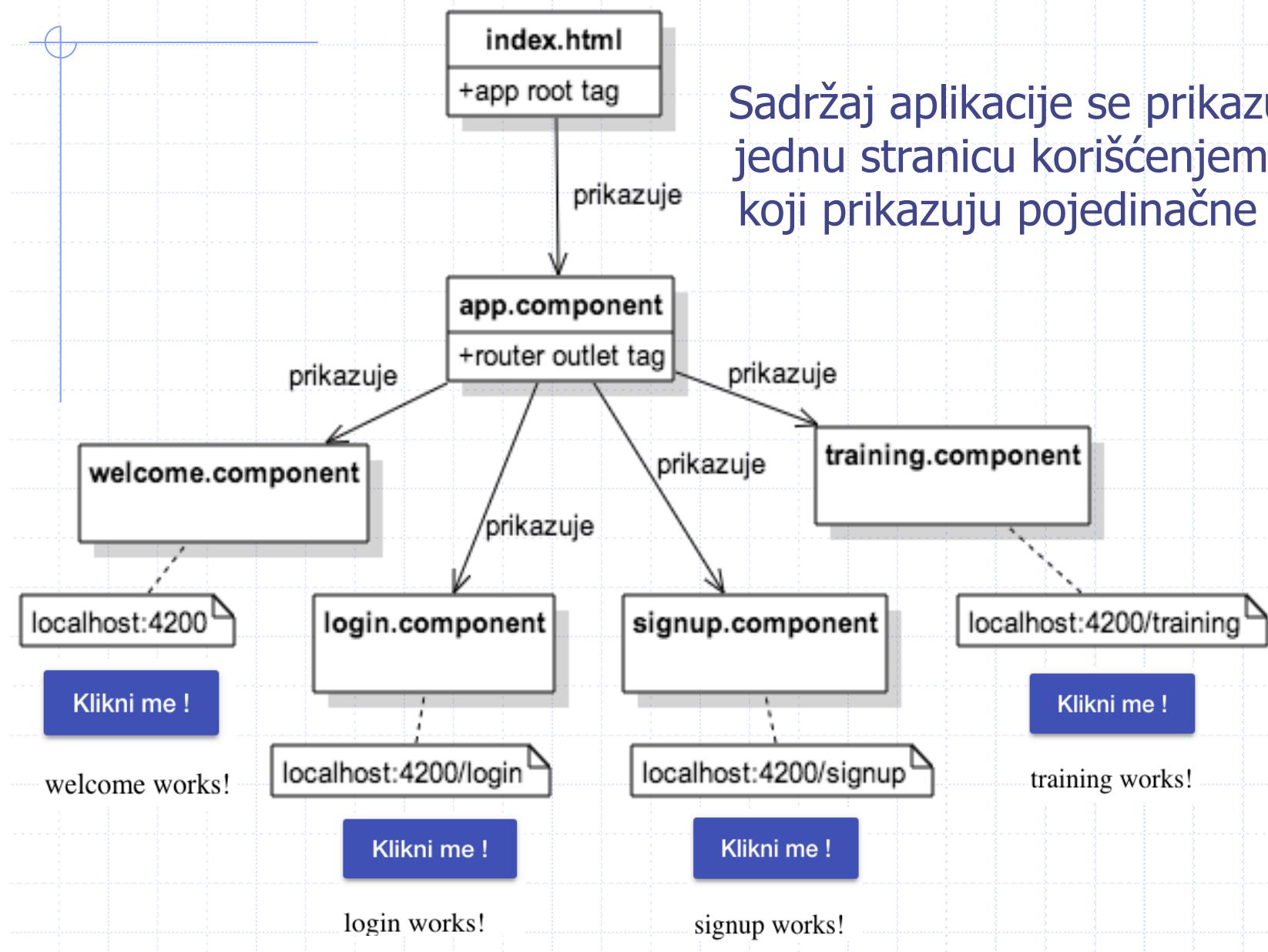
Communicate with Backend Services

Provides a “Mobile App”-like User Experience

# Angular

- ◆ Izvršava se na strani klijenta, u pretraživaču
- ◆ Okvir za razvoj Web stranica zasnovan na TypeScript jeziku
- ◆ Prikaz korisničkog interfejsa sa dinamičkim podacima
  - Podaci koji se menjaju u realnom vremenu, tj. u toku korišćenja aplikacije
- ◆ Prijem i obrada ulaznih akcija i podataka korisnika
- ◆ Prosleđivanje na dalju obradu i prijem rezultata za prikaz od pozadinske logike (backend)
- ◆ *Mobile-like* doživljaj korisnika – sadržaj se menja dinamički, bez ponovnog učitavanja čitave stranice, već samo delova koji su zahtevani ili promenjeni

# SPA – Single Page Application



Sadržaj aplikacije se prikazuje kroz jednu stranicu korišćenjem tagova koji prikazuju pojedinačne prikaze

# MEAN



A Server-side Library: JavaScript on the Server-side

Listen to Requests and  
Send Responses

Execute Server-side  
Logic

Interact with  
Databases and Files

An Alternative to PHP, Ruby on Rails, Java etc. Is rarely used Standalone!

# Node

- ◆ Serverska JavaScript biblioteka
  - Serverska: obavlja funkcije servera
  - Napisana u JavaScript-u
- ◆ Rad sa fajl sistemom, tj. pristup lokalnim datotekama
- ◆ Rad sa bazom podataka
  - Teorijski moguće i korišćenjem Angular-a, ali nije bezbedno
  - Parametri za pristup se mogu videti u developer prozoru pretraživača
- ◆ Obrada HTTP zahteva
  - Prijem zahteva i slanje odgovora(resultata)
- ◆ Autentikacija i autorizacija

# MEAN

# express

A Node Framework which simplifies writing Server-side Code and Logic.

Based on Node, offers  
same Functionalities

Middleware-based:  
Funnel Requests  
through Functions

Includes Routing,  
View-rendering &  
More

# Express

- ◆ Node nadogradnja
- ◆ Korisne, tzv. utility funkcije koje olakšavaju korišćenje Node servisa
- ◆ Kako ?
- ◆ Obrada zahteva kroz sekvencu koraka ili lanac u kojem se u svakom koraku obavlja specifična, atomična funkcija
  - Prijem zahteva
  - Čitanje podataka
  - Specifična obrada
  - Slanje odgovora
- ◆ Analogija: Java *wrapper* funkcije za obradu tokova
- ◆ Zbog čega ?
- ◆ Struktuiran kod koji je lak za razumevanje i održavanje
- ◆ Mehanizam rutiranja
  - Zahtevi koji potiču iz različitih domena

# MEAN



Mogu biti korišćene i  
druge BP (npr. MySQL)



A NoSQL Database which stores “Documents” in “Collections” (instead of “Records” in “Tables” as in SQL).

Store Application Data  
(Users, Products, ...)

Enforces no Data  
Schema or Relations

Easily connected to  
Node/ Express (NOT to  
Angular!)

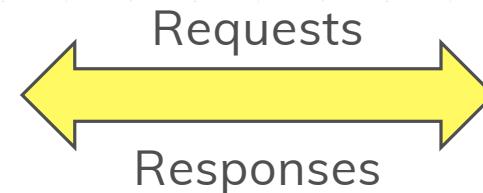
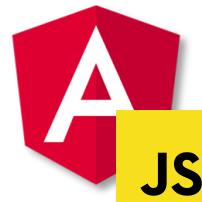
A powerful Database which can easily be integrated into a Node/ Express Environment.

# MongoDB

- ◆ Ne-relaciona baza podataka
- ◆ Čuva kolekcije dokumenata
- ◆ Relaciona baza: tabele, zapis, relacije
- ◆ Ne postoji unapred definisana šema, tj. model podataka koji će se skladištiti
- ◆ Jednostavna integracija sa Node/Express
- ◆ Pokazuje bolje performanse od relacione baze:
- ◆ Veliki broj upisa i pristupa u bazu podataka (logovi, notifikacije)
  - read per second, write per second
- ◆ Relacione baze pokazuju bolje performanse kod veće količine složenije struktuiranih poslovnih podataka sa relacijama i izveštavanja

# MEAN stek – funkcionalni pogled

Client (Browser)



Server



Presentation/ UI

Single-Page- Application

Data

JSON Format

Business Logic

Persistent Data Storage

Authentication Logic

Single Page is not necessarily served by Node Backend!

# Ispit

- ◆ Redovno pohađanje nastave i aktivnost na času (max 10 poena)
- ◆ Kolokvijum 1 (max 30 poena)
  - Zadatak iz oblasti obrađenih do prve kolokvijumske nedelje
- ◆ Kolokvijum 2 (max 30 poena)
  - Projekat prvi deo
- ◆ Završni ispit (max 30 poena)
  - Projekat drugi deo

# Vrednovanje

- ◆ Završna ocena se formira na osnovu skale:
- ◆ 10: 100 => poeni => 91
- ◆ 9: 90 => poeni => 81
- ◆ 8: 80 => poeni => 71
- ◆ 7: 70=> poeni => 61
- ◆ 6: 60 => poen => 51
- ◆ nije položila/o: poeni <= 50

# Saveti za slušanje kursa



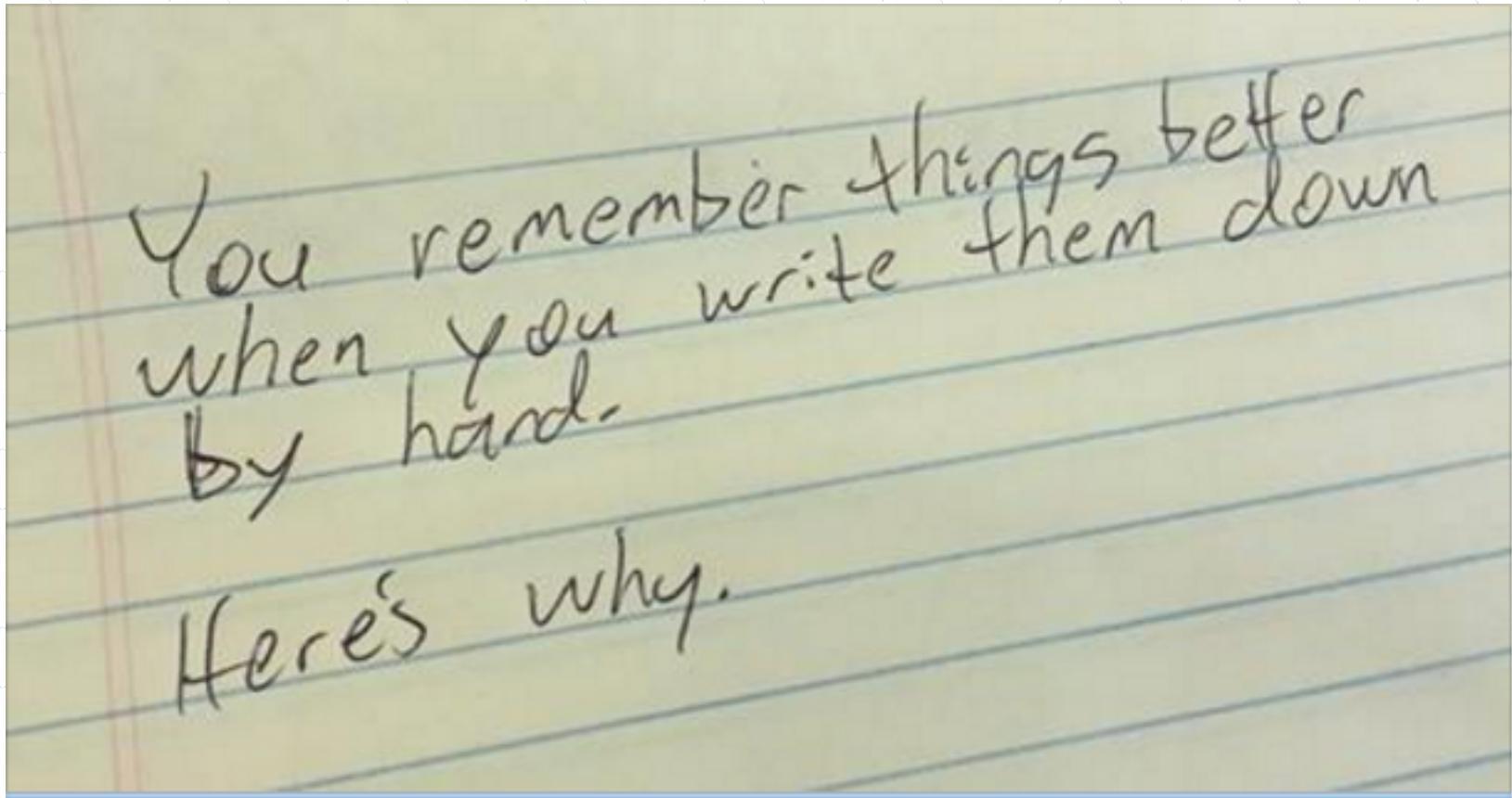
- ◆ Da bi Vašim kolegama i Vama obezbedili uslove za **održavanje pažnje i koncentracije** u toku nastave
1. Izbegavajte korišćenje mobilnih telefona u toku nastave (osim ako nije u vezi sa nastavom)
  2. Dođite na vreme na čas (ukoliko kasnите nemojte uznemiravati ostale kolege i na pauzi ih pitajte šta je rađeno dok ste bili odsutni)
  3. Svedite priču i šaputanje u pozadini na minimum
  4. Izbegavajte zabavne aktivnosti na vašim ili univerzitetskim računarima (gleđanje videoa u toku časa)

# Saveti za slušanje kursa



- ◆ Da bi poboljšali učenje, tj. **usvajanje** materijala u toku izvođenja nastave
1. Izbegavajte više zadatak u isto vreme (pisanje mejlova i slušanje nastave nisu kompatibilni)
  2. Ostavite Vaš telefon u torbi (osim ako se koristi za potrebe nastave)
  3. Hvatajte beleške u toku nastave i prelistajte u toku večeri
  4. Ukoliko nešto ne razumete, prekinite me kako bih pojasnio
  5. Ukoliko imate generalnije ili složenije pitanje, sačekajte pauzu da bi pitali
  6. **Učestvujte i budite proaktivni**

# Ručno vodenje beleški



Mueller (2014) The pen is mightier than the keyboard: Advantages of longhand over laptop note taking

# Zbog čega su beleške korisne ?

- ◆ **hipoteza enkodovanja:** kada osoba hvata beleške, procesiranje koje se dešava će poboljšati učenje i zadržavanje materijala
- ◆ **hipoteza eksternog skladišta:** ljudi uče gledajući ponovo u svoje zapise, ili čak zapise drugih ljudi

# Dodatna literatura

- ◆ Holmes, S., (2016). Getting MEAN with Mongo, Express, Angular, and Node. Manning Publications, 1st Ed.
- ◆ HTML, CSS i DOM документација:  
<https://developer.mozilla.org/en-US>
- ◆ JavaScript документација:  
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>
- ◆ Angular документација: <https://angular.io/docs>
- ◆ Angular Material документација:  
<https://material.angular.io>
- ◆ Node.js документација: <https://nodejs.org/en/docs/>

# Klijentske Web aplikacije

## Uvodno predavanje

Nikola Savanović

[nsavanovic@singidunum.ac.rs](mailto:nsavanovic@singidunum.ac.rs)

Dr. Mlađan Jovanović

[mjovanovic@singidunum.ac.rs](mailto:mjovanovic@singidunum.ac.rs)