

Programim Web

Leksioni 1

Aplikacion web

- Një aplikacion web është një program i cili rendohet në një browser.
- Nuk është një website ...
- Për të dalluar një aplikacion web dhe një website duhet të kemi parasysh tre karakteristika formale. Një aplikacion web:
 - Adreson një problem specifik, edhe pse mund të jetë thjesht kërkimi i një informacioni;
 - Është aq interativ sa një aplikacion desktop;
 - Ka një CMS – Content Management System.

Aplikacion web (2)

- Interneti dhe teknologjitet e programimit në web të cilat do t'i studiojmë në ketë lëndë, janë portabël, dmth të lejojnë të ndërtosh faqe web apo aplikacione që ekzekutohen në pajisje të ndryshme.
- Teknologjitet e programimit në anën klient (client-side) përdoren për të ndërtuar faqe/aplikacione web që ekzekutohen në klient (p.sh. në browser-in që ndodhet në pajisjen e përdoruesit).
- Programimi në anën server (server-side) përdoren për të ndërtuar aplikacione që i përgjigjen kërkesave nga browser-at në anën klient (client-side), si p.sh. kërkimi në internet, kontrolli online i llogarisë bankare, porosia e një produkti ne Amazon, etj.

Principet në krijimin e një aplikacioni web

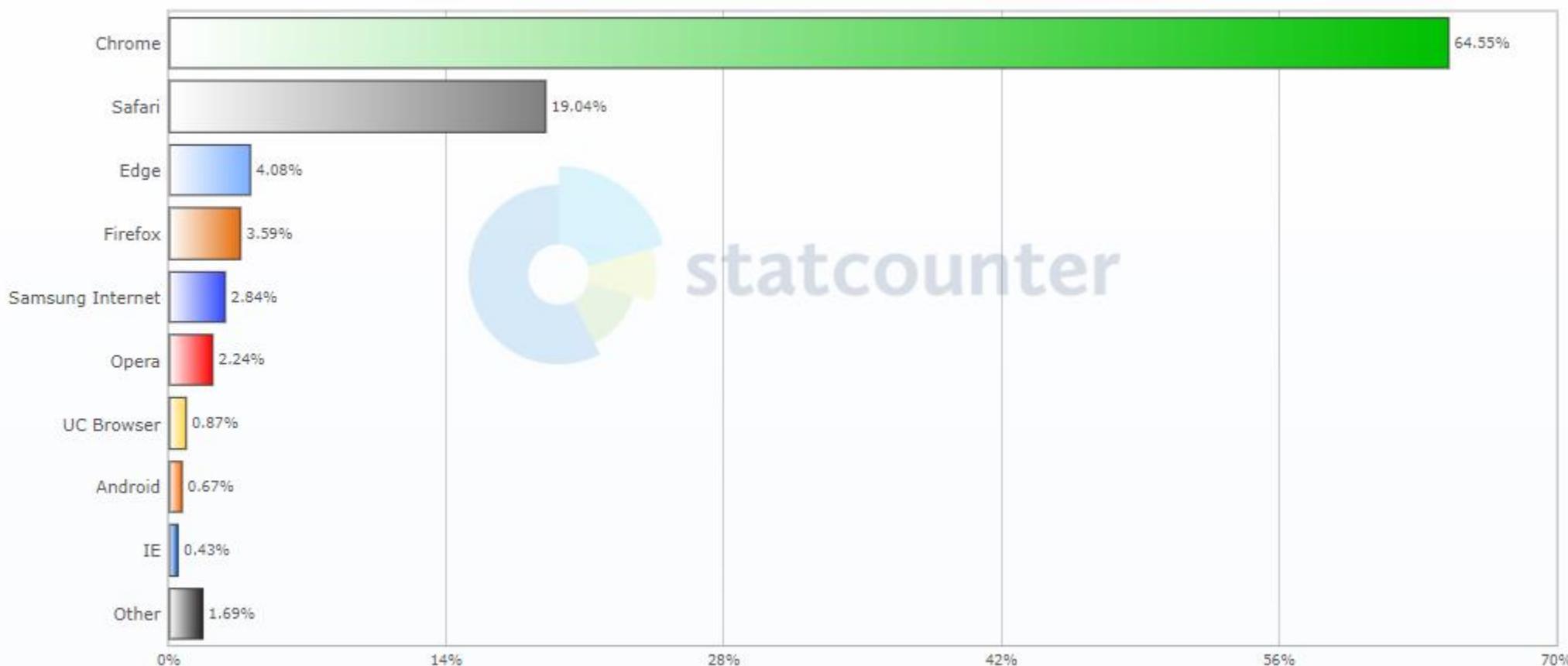
- Për klientin aplikacioni duhet të jetë:
 - I thjeshtë;
 - Estetikisht i këndshëm;
 - Të adresoj pjesën më të madhe të probleme të tyre.
- Nga aspekti i biznesit një aplikacion web duhet të jetë në përputhje me raportin produkt/treg.
- Nga perspektiva e një programuesi një aplikacion web duhet të jetë:
 - I shkallëzueshem;
 - Funksional;
 - Funksional dhe në rastet e aksesimit të shumëfishtë (shumë përdorues në të njëjtën kohë).

Browser-at

- Një nga sfidat më të mëdha e aplikacioneve web është të sigurojnë një pamje konsistente në browser-a të ndryshëm në anën klient.
- Nuk ekziston një standard për prodhuesit që krijojnë web browser-a.
- Pavarësisht se në përgjithësi browser-at kanë një bashkësi të përbashkët karakteristikash, secili i renderon faqet në mënyra të ndryshme.
- Browser-at janë të versioneve dhe platformave të ndryshme (Microsoft Windows, Apple Macintosh, Linux, UNIX, etj.).
- Shtimi i karakteristikave të reja sjellin probleme me kompatibilitetit në cross-platform.
- Është e vështirë të ndërtohen aplikacione web që renderohen siç duhet në të gjitha versionet e secilit browser.

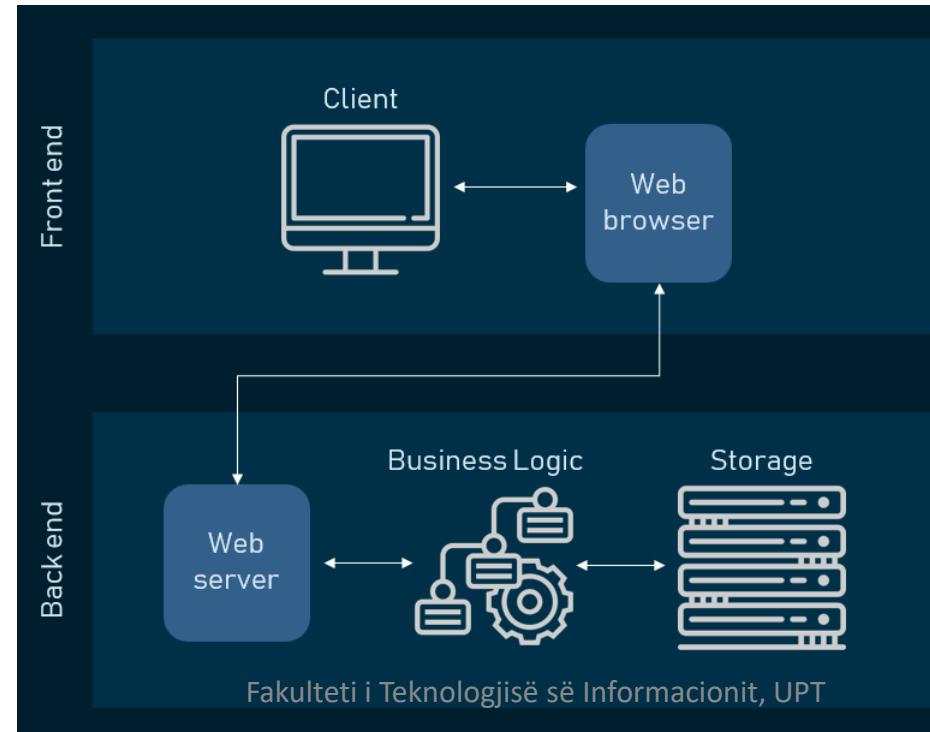
Browser-at (2)

- Statistikë rreth përdorimit të browser-ave në periudhën shtator 2021 – shtator 2022:
<http://gs.statcounter.com>



Arkitektura e një aplikacioni web

- Arkitektura e një aplikacioni web përcakton se si komponentët e aplikacionit komunikojnë me njëri-tjetrin.
- Përcakton mënyrën se si klienti dhe serveri lidhen me njëri-tjetrin.



Si funksionon një kërkesë në web?

- Shkruani URL → browser-i përgatitet për të njojur URL, sepse duhet të dije adresën e serverit ku gjendet faqja → dërgohet një kërkesë tek DNS, Domain Name Server → browser-i dërgon kërkesën në IP e gjetur duke përdorur protokollin HTTP.
 - DNS – mbajtës i emrave të domain dhe adresave IP.
 - Nëse faqja është klikuar më parë të dhënat merren nga cache.
- Web server merr kërkesën → e dërgon në vendndodhjen ku ruhen faqet dhe të dhënat. Ky rrugëtim kalon nëpërmjet Logjikës së Biznesit, BL → pas procesimit të kërkesës BL e dërgon atë në storage për të gjetur të dhënat.
 - BL për çdo të dhënë përcakton mënyrën e aksesimit dhe përcakton rrjedhjen e informacionit për çdo aplikacion.
- Përdoruesi merr të dhënat. Përgjigja kthehet tek përdoruesi dhe ai mund të shoh përbajtjen e faqes.

Protokollet

- Protokollet janë standarde dhe rregulla të cilat përcaktojnë mënyrën se si informacioni shkëmbhet midis kompjuterëve apo pajisjeve.
- HTTP
 - Me anë të këtij protokolli çdo faqe web mund të shfaqet në browser. Dërgohet një kërkesë në serverin përkatës dhe në përgjigje merren përgjigja që përban kodet HTML, CSS, dhe JavaScript të faqe web.
- DDP
 - Përdor websockets për të krijuar lidhjen konsistente midis klientit dhe serverit. Si rezultat kemi rifreskim në kohë reale të të thënave në një faqe web pa berë refresh në browser.
- REST
 - Përdoret zakonisht për API's, ky protokoll ka metodat standarde si GET, POST, dhe PUT që lejojnë shkëmbimin e informacionit midis aplikacioneve.

Web Server

- Web server-eri i përgjigjet kërkesave të klientit (që vijnë nga një browser) duke i kthyer burime të ndryshme si dokumenta HTML.
- Kur përdoruesi vendos një adresë URL (Uniform Resource Locator) në browser, në fakt ai po kërkon një dokument specifik nga web server-i.
- Web server-i e lidh URL-në me një file në rrjetin e vet dhe ia kthen atë përdoruesit.
- Web server-i dhe klienti komunikojnë duke përdorur një protokoll të pavarur nga platforma që quhet Hypertext Transfer Protocol (HTTP).
- HTTP është një protokoll që transferon kërkesa ose dokumenta në internet ose intranet.

Transaksionet HTTP (I)

- Uniform Resource Identifiers, URI, identifikon burimet në internet.
- URI që fillojnë me http:// quhen URL (Uniform Resource Locators).
- http:// tregon që duhet të përdoret protokolli HyperText Transfer Protocol (HTTP) për të marrë burimet.
- URL i referohet file-ve, direktorive ose kodeve në anën server, server-side, që kryejnë një detyrë të caktuar, si kërkime në bazën e të dhënave, kërkime në Internet, etj.
- URL ka informacion për ta drejtuar browser-in në burimet që përdoruesi do të aksesoj.

Transaksionet HTTP (II)

- Në URL gjendet hostname i serverit, emri i kompjuterit web-server, në të cilin ndodhen burimet e kërkuara.
- Hostname përkthehet në një adresë IP (Internet Protocol).
- IP është një vlerë numerike që identifikon në mënyre unike serverin.
- Server DNS mban bazën e të dhënavë të hostname me IP e tyre koresponduese dhe sipas saj bën automatikisht përkthimin.
- Pjesa tjeter e URL pas hostname përcakton pozicionin e burimeve, p.sh. (/views) dhe emrin e tyre në web server.
- Path-i i vendodhjes është një direktori virtuale, kjo për arsyen sigurie.
- Direktoria virtuale përkthehet nga web serveri në pozicionin real.

Transaksionet HTTP (III)

- Metoda get e HTTP-se tregon që klienti dëshiron të marrë një burim nga serveri.
- Pjesa tjeter e kërkesës ka path-in e burimit, emrin e protokollit dhe numrin e versionit (HTTP/1.1).
- Çdo server që kuption HTTP mund të pranojë kërkesa get dhe t'i përgjigjet atyre në mënyrën e duhur.
- Status code 200 i HTTP tregon sukses.
- Status code 404 i HTTP tregon mos sukses dhe informon klientin që web server-i nuk mundi të gjej burimin e kérkuar.
- Linku i kodeve të statusit të HTTP: www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

Transaksionet HTTP (IV)

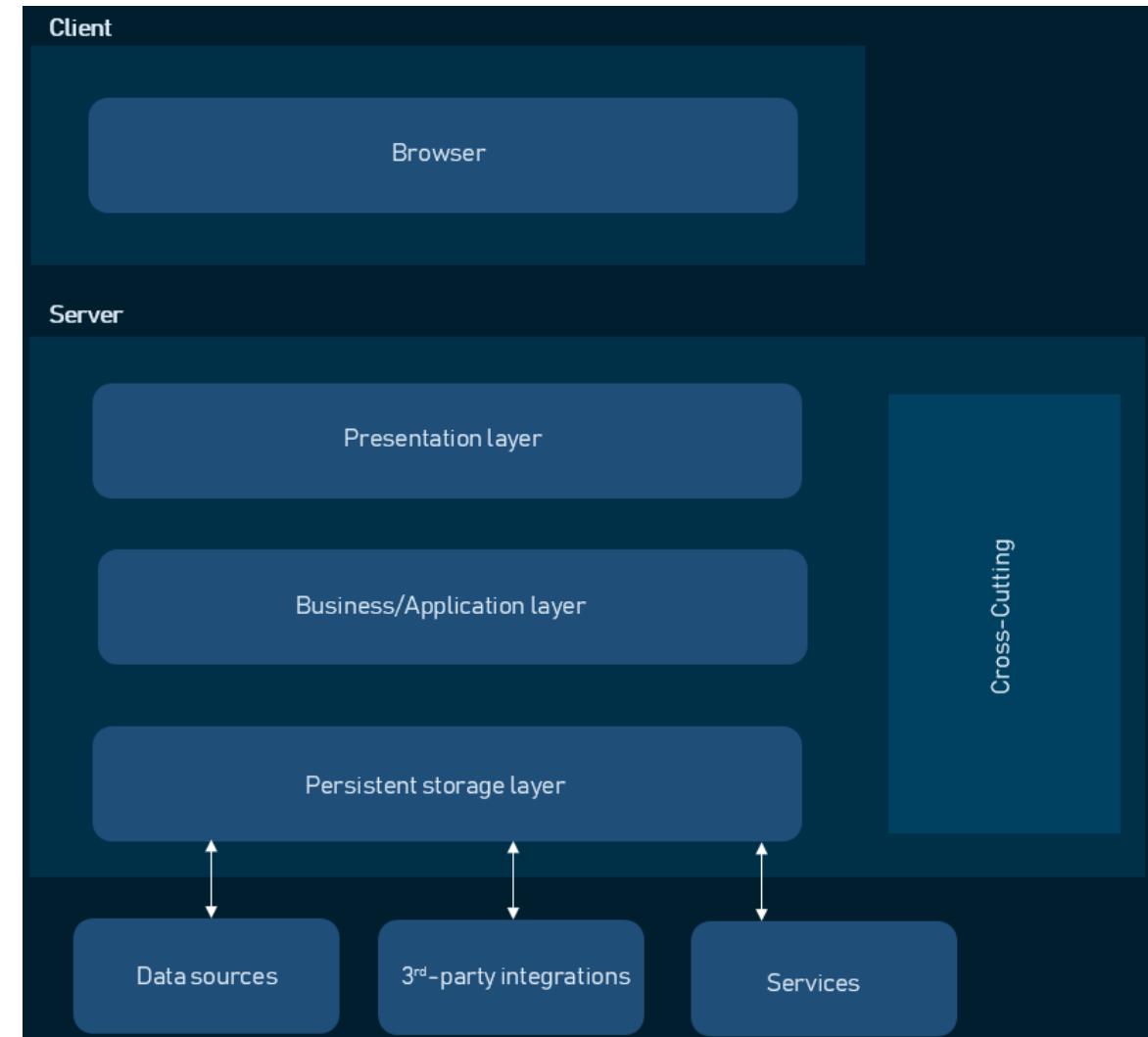
- Tipat e kërkesave HTTP
 - I. **get** - zakonisht merr informacion nga serveri. Përdoret për të marrë dokumenta HTML, imazhe ose rezultate të kërkimeve bazuar në terma të caktuara të dhëna nga përdoruesi.
 - II. **post** - zakonisht dërgon të dhëna në server. Përdoret për dërgim informacioni në server, si p.sh. informacion autentikimi ose të dhëna të formës që merr input nga përdoruesi.
- Zakonisht kërkesa HTTP poston të dhëna ne një form handler në anën e serverit, server-side, që i proceson ato.
- Kërkesa **get** i ngjit të dhënat në query string të URL-së.
 - Përdoret ? për të ndarë query string nga pjesa tjeter e URL-së.
 - Çiftet *emër/vlerë* i kalohen serverit në formën *emër=vlerë*.
 - & përdoret për të ndarë çiftet *emër/vlerë* kur ka më shumë se 1 çift.
 - Realizohet duke berë submit një forme HTML atributi method i së cilës ka vlerën “get”, ose duke shkruar në adresën e browser-it direkt URL-në bashkë me query string.
 - Janë të limituara në një numër të caktuar karakteresh të gjatësisë së query string.
- Kërkesa **post** i dërgon të dhënat në një mesazh HTTP dhe jo si pjesë e URL-së.
 - Dërgohen sasi të mëdha informacioni.
 - Përdoret për të dërguar të dhëna sensitive.

Transaksionet HTTP (V)

- Browser-at ruajnë në cache faqet web të vizituara së fundmi në mënyre që t'i ringarkojnë shpejt ato në një kërkese të dytë.
- N.q.s. nuk ka ndryshime midis versionit të ruajtur në cache dhe versionit të faqes në web, përmirësohet shpejtësia e browser-it duke përdorur versionin në cache.

Arkitektura Multi ose Three-Tier (I)

- Shumica e aplikacioneve web zhvillohen duke i ndarë funksionet kryesore të tij në shtresa, ose nivele.
- Kjo lejon ndryshimin e çdo shtresë në mënyre të pavarur.
- Shtesat mund të ndodhen në të njëjtin kompjuter ose në kompjutera të ndryshëm.
- Ky model quhet Arkitektura Multi ose Three-Tier.



Arkitektura Multi ose Three-Tier (II)

- Shtresa e prezantimit (presentation layer)
 - Aksesohet nga përdoruesi nëpërmjet browser-it.
 - Konsiston në komponentët e ndërfaqes së përdoruesit dhe të proceseve UI që mbajnë ndërveprimin në sistem.
 - Zhvillohen duke përdorur tre teknologji kryesore HTML, CSS, dhe JavaScript:
 - HTML është kodi që përcakton çfarë do të përbaj faqja web;
 - CSS kontrollon pamjen/paraqitjen;
 - JavaScript e bën faqen web interaktive për veprimet e përdoruesit dhe dinamike.

Arkitektura Multi ose Three-Tier (III)

- Shtesa e biznesit (business layer/domain Logic/application layer)
 - Merr kérkesat e përdoruesit nga browser-i, i proceson ato dhe përcakton mënyrën se si mund të aksesohen të dhënat.
 - Rregullat e biznesit përcaktojnë si mund dhe si nuk mund të aksesohen të dhënat e aplikacionit nga klienti dhe si do të procesohen ato të dhëna.
 - Në ketë shtesë enkodohet mënyra si të dhënat dhe kérkesat transmetohen në back-end.
 - Sigurohet që të dhënat të janë konsistente me rregullat përpara se të modifikoje bazën e të dhënavë ose para se t'ia paraqesë të dhënat përdoruesit.
 - Megjithëse rregullat e biznesit mund të janë një manifest i logjikës së biznesit, ato nuk janë të njëjta.
 - Ndonjëherë rregullat e biznesit nxirren dhe menaxhohen veçmas, duke përdorur një Sistem të Menaxhimit të Rregullave të Biznesit.
- Nëse kemi një aplikacion rezervimi hoteli kjo shtesa do të jetë përgjegjëse për sekuencën e veprimeve që një përdorues do të përshkojë gjatë realizimit të rezervimit të një dhoma.

Arkitektura Multi ose Three-Tier (IV)

- Shtresa e të dhënavë (persistence layer/the storage layer/data access layer)
 - Kjo shtresë është një vendodhje e centralizuar që merr të gjitha thirrjet për të dhëna dhe siguron qasje të vazhdueshme në burimet e një aplikacioni.
 - Lidhet ngushtë me shtesën e biznesit, sepse logjika e biznesit përcakton se më cilën bazë të dhënash duhet të komunikohet duke optimizuar procesin e marrjes së të dhënavë.
 - Infrastruktura përfshin një server dhe një Database Management System, DMS, një software për të komunikuar dhe menaxhuar me bazën e të dhënavë.

Arkitektura Multi ose Three-Tier (V)

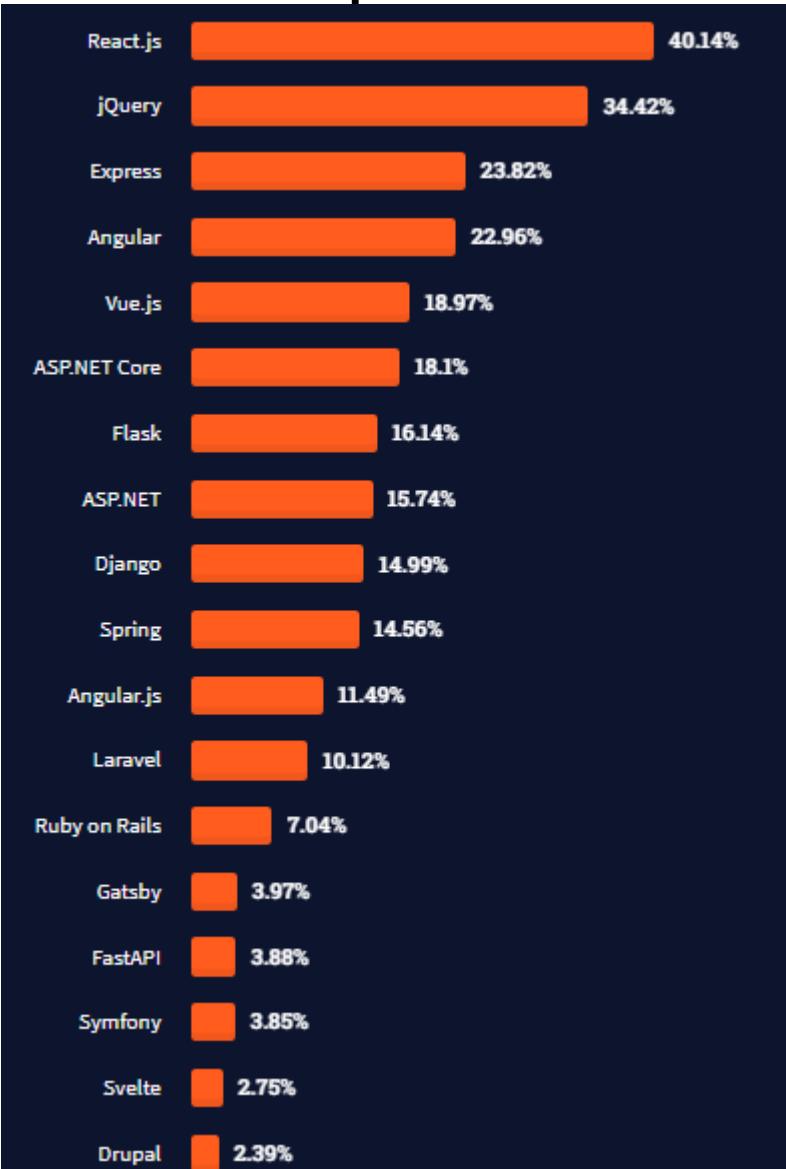
- Ka disa komponentë që janë të ndarë nga shtesat kryesore:
 - **Cross-cutting code:** trajton komponentë të tjerë të aplikacionit si komunikimet, menaxhimin operacional dhe sigurinë. Mund të afektoj të gjithë pjesët e tjera të sistemit por nuk mund të shkrihet me asnjë nga ato.
 - **Third-party integrations:** Integrimi me sistemet e pagesave, logimin në mediat sociale, integrimi me aplikacione të ndryshme nëpërmjet APIs. Lejon aplikacionin të marri të dhëna nga software të tjerë dhe të zgjerojë funksionalitetet pa pasur nevojë për vatkodim të tyre.

Skriptet në anën klient kundrejt skripteve në anën server (Skriptet Client-Side vs Skriptet Server-Side)

- Skriptet në anën klient:
 - Mund të përdoren:
 - për të validuar të dhënat e përdoruesit;
 - për të ndër vepruar me browser-in;
 - për të zgjeruar funksionalitetet e browser-it;
 - për të realizuar komunikimin klient/server midis browserit dhe serverit.
 - Kanë kufizime nga browser-i, browseri duhet të suportojë gjuhën e skriptimit.
 - Mund të shihen nga klienti duke përdorur browser's source-viewing.
 - Nuk duhet të ruhen dhe të validohen informacione sensitive, si fjalëkalime, ose të dhëna të tjera personale.
 - Probleme në siguri kur vendoset një sasi e madhe informacioni në JavaScript.
- Skriptet në anën server
 - Kodi i ekzekutuar në server shpesh gjeneron përgjigje të ndryshme për klientët.
 - Gjuhet e skriptimit në anën server ofrojnë mundësi më të mëdha nga ana e programimit në krahasim me ato në anën klient.

10 Web Framework-et më të përdorur në 2020

- Reference:
<https://insights.stackoverflow.com/survey/2021#most-popular-technologies-webframe>



Programet e nevojshëm për lëndën

- Editor
- Browser
- XAMPP përmban:
 - **Apache**: aplikacion web server, open source. Një makine që ekzekuton pjesë software, quhet web server dhe ruan dokumenta XHTML.
 - **MySQL/MariaDB**: server për bazën e të dhënave.
 - **PHP**: gjuhë skriptimi server-side dizenuar për web por që mund të përdoret edhe si general purpose. Open source.
 - **phpMyAdmin**: free dhe open source i implementuar në PHP për të menaxhuar administrimin e bazës së të dhënave nëpërmjet një web browser-i.