

# RASSUS odgovori na pitanja iz prezentacije i knjige (1. predavanje)

B25

22. studenoga 2022.

## 1. Čime je definirana otvorenost weba?

Definirana je normiranim pravilima te definiranoj sintaksi i semantici.

## 2. Kojim aspektima transparentnosti pridonosi sustav imenovanja domena (DNS)?

Migracijska transparentnost, konkurencijska transparentnost i lokacijska transparentnost.

## 3. Kako replikacija pridonosi otpornosti na kvarove i skalabilnosti?

Otpornosti na kvarove doprinosi tako što u slučaju kvara jednog čvora možemo "posao" podijeliti na ostale čvorove.

Skalabilnosti doprinosi tako što se povećava propusnost sustava.

## 4. Kakvi bi se problemi pojavili kada bi se više repliciranih poslužitelja weba priključilo na mrežu izravno, a ne posredstvom zastupnika ( proxy )?

Korisnik ne bi znao kojem poslužitelju pristupa i sa svakim sljedećim pristupanjem ne bi bilo zajamčeno pristupanje sadržaju kojem je prethodno pristupio.

## 5. Što sve utječe na vrijeme odziva poslužitelja weba?

Procesor, memorija i mreža. \*

## 6. Na primjeru weba objasnite razliku između horizontalnog i vertikalnog skaliranja sustava.

**Vertikalno skaliranje** - poslužitelj ostaje se na jednom čvoru, ali mu se povećavaju resursi (memorija, procesor)

**Horizontalno skaliranje** - imamo više manjih čvorova na koji se onda dijele zahtjevi

## 7. Zašto se koriste perzistentne konekcije u HTTP-u?

Zbog toga što je HTTP protokol bez očuvanja stanja.

## 8. Kako se definira uvjetni GET i kako funkcionira? Kakve prednosti donosi za performance HTTPa?

Uvjetni GET služi za provjeru je li došlo do promjene sadržaja na izvornom poslužitelju, ako je njegovo vrijeme valjanosti isteklo. S ciljem smanjenja mrežnog prometa i šalje ga priručno spremište. Premda se zahtjev šalje do poslužitelja, štedi se na mrežnom prometu jer se sadržaj resursa ne prenosi ako nije došlo do promjene .

Uvjetni GET uključuje zaglavlja *If-Modified-Since*, *If-Unmodified-Since*, *If-Match*, *If-None-Match*, ili *If-Range*. Sadržaj će biti prenesen u odgovoru samo u slučaju da je uvjet iz zaglavlja zadovoljen, inače poslužitelj odgovara s 304 (*Not Modified*).

**9. Objasnite zašto današnja rješenja za IoT predstavljaju raspodijeljene sustave velikih razmjera?**

Zbog toga što postoje sustavi koji sadrže veliki broj uređaja (desetke tisuća) čije podatke treba na neki način obraditi.

**10. Objasnite ulogu programskog posredničkog sloja za raspodijeljene sustave.**

Programski posrednički sloj omogućava povezivanje, komunikaciju i suradnju aplikacija, sustava i uređaja te omogućuje interakciju programa na aplikacijskoj razini. Nalazi se između operacijskog sustava i aplikacijskih programa u programskoj arhitekturi sustava. Cilj je olakšati korisnicima i aplikacijama pristup i korištenje udaljenih sredstava na kontroliran i učinkovit način.

**11. Usporedite migracijsku i relokacijsku transparentnost raspodijeljenog sustava. Koje svojstvo (ne)zadovoljava web poslužitelj i zašto?**

**Migracijska transparentnost**- prikrivanje promjene lokacije sredstva na način da ta promjena ne utječe na način pristupa sredstvu

**Relokacijska transparentnost** - prikrivanje premještanja / kretanja sredstva

Web poslužitelj **zadovoljava migracijsku transparentnost**. Mijenjanjem lokacije i/ili mrežnog sučelja mijenja se mrežna adresa, ali se ne mijenja simbolička adresa.

Web poslužitelj **ne zadovoljava relokacijsku transparentnost** jer je poslužitelj stacionaran i ne krede se tijekom pružanja usluge.

**12. Objasnite pojam skalabilnosti raspodijeljenog sustava**

Raspodijeljeni sustav je skalabilan ukoliko posjeduje sposobnost prilagodbe povedanom broju korisnika i sredstava, njihovoj rasprostranjenosti te načinu upravljanja sustavom.

**13. Napravite analizu transparentnosti raspodijeljenog sustava elektroničke pošte.**

Primjer gmaila: Zadovoljava lokacijsku transparentnost jer korisnik pristupa usluzi preko simboličke adrese.

Zadovoljava konkurencijsku transparentnost jer uslugu koristi puno korisnika, a individualan korisnik ima dojam kao da je on jedini. Transparentnost pristupa, prikrivaju se razlike u pristupu sredstvu korisnicima s različitim OS-ima (npr. Mac, MS Windows itd.).

Transparentnost na kvar i replikacijska transparentnost, kopije podataka se čuvaju na drugim poslužiteljima za slučaj kada se jedan poslužitelj pokvari, da drugi može preuzeti posao bez da korisnik primjeti da je došlo do kvara.

**14. Prikažite i objasnite primjer troredne arhitekture weba.**



Primjer su aplikacije weba, gdje korisnički program koji se izvodi na klijentskom računalu nikad ne pristupa direktno bazi podataka, već posredno preko aplikacije weba. Klijentski program prikazuje korisničko sučelje i komunicira s aplikacijom weba koja obavlja cjelokupnu logiku usluge i pristupa potrebnim podacima.

Programski poslužitelj djeluje kao *poslužitelj* klijentu, a kao *klijent* poslužitelju baze podataka.

**15. Kakvi bi se problemi pojavili kad bi se više repliciranih poslužitelja weba priključilo na mrežu izravno, a ne putem zastupnika (proxy)?**

Korisnik ne bi znao kojem poslužitelju pristupa i sa svakim sljedećim pristupanjem ne bi bilo zajamčeno pristupanje sadržaju kojem je prethodno pristupio.

**16. Što je vrijeme odziva za poslužitelja weba?**

Vrijeme odziva je vrijeme koje je potrebno da poslužitelj procesira zahtjev i da odgovor stigne do korisnika.