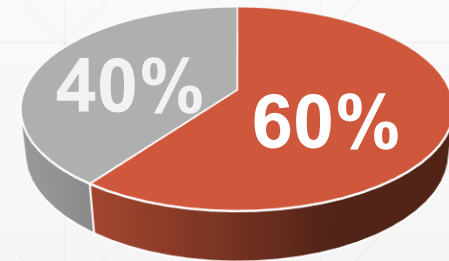


Uvod

Principi softverskog inženjerstva, *Elektrotehnički fakultet Univerziteta u Beogradu*

Organizacija kursa

- Fond časova: 2 P + 2 V + L (drže se blokovski, u računarskim laboratorijama, 4x u toku semestra)
- Veb stranica predmeta: <http://si3psi.etf.rs>
- Mejling lista: 13s113psi@lists.etf.rs (biće formirana nakon 2. nastavne sedmice)
- Predavači: Prof. dr Dragan Bojić, prof. dr Dražen Drašković, master inž. Mihajlo Ogrizović, dipl. inž. Luka Hrvačević, dipl. inž. Teodora Radaljac
- Kontakt za predavanja: bojic@etf.bg.ac.rs, drazen.draskovic@etf.bg.ac.rs
- Studenti treba da se upoznaju sa pravilima predmeta, koja se nalaze na veb stranici.
- Raspodela poena: **60% pismene provere znanja (kolokvijumi, ispit)**
40% timski projekat tokom celog semestra

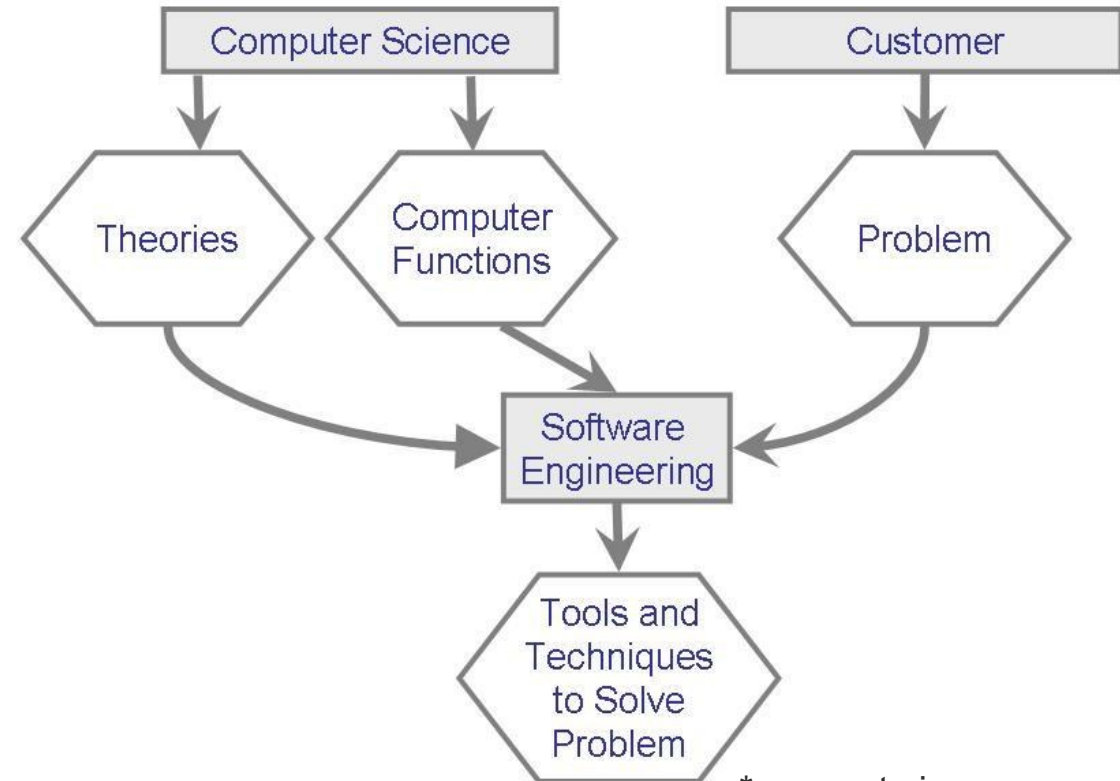


Softversko inženjerstvo - pojam i definicije

- Prvi pominjanje softverskog inženjerstva, 1968. godine, kao odgovor na krizu softvera
- „Sistematska primena naučnih i tehnoloških znanja, metoda i iskustva u projektovanju, implementaciji, testiranju i dokumentovanju softvera“ - Američki zavod za statistiku rada
- „Primena sistematskog, disciplinovanog, kvantifikovanog pristupa razvoju, radu i održavanju softvera“ - IEEE (standardni rečnik terminologije softverskog inženjerstva)
- „Inženjerska disciplina koja se bavi svim aspektima proizvodnje softvera“ - Prof. Ian Sommerville
- „Uspostavljanje i korišćenje proverenih inženjerskih principa kako bi se ekonomično proizveo softver koji je pouzdan i efikasno funkcioniše na stvarnim mašinama“ - Prof. Fritz Bauer

Veza softverskog inženjerstva i računarske nauke

- Računarska nauka bavi se teorijom i osnovama (apstrakcija).
- Softversko inženjerstvo bavi se praktičnim aspektima razvoja i isporuke korisnog softvera.



* preuzeto iz:
Software Engineering: Theory and Practice
4th edition, by S. Pfleeger, J. Atlee

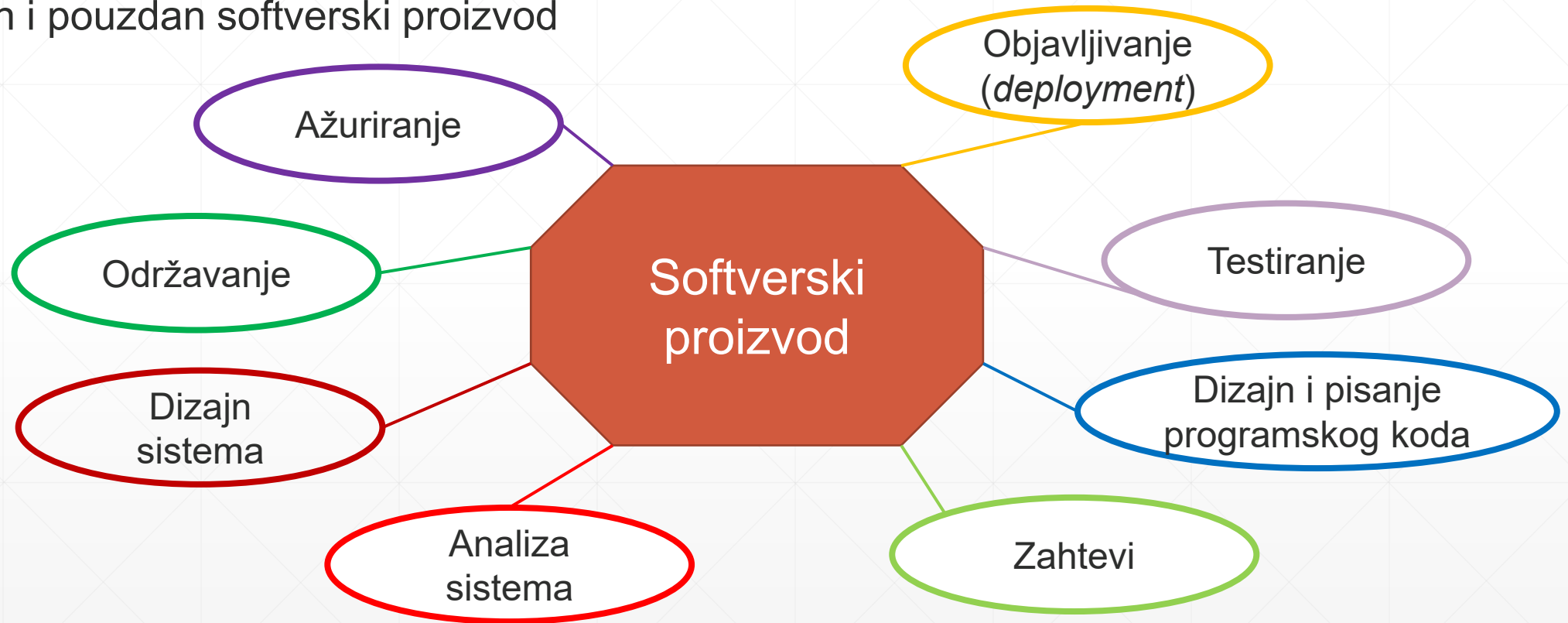
Alternativna definicija

- *Alistair Cockburn*: „Softversko inženjerstvo je **kooperativna igra**“
- Ako bi razvoj softvera zaista bio nauka, mogli biste na njega primeniti naučni metod. Ako je to zaista inženjerstvo, onda možete primeniti poznate inženjerske tehnike. Ako je razvoj softvera stvar proizvodnje formalnog modela, onda možete uložiti svoj novac u razvoj modela.
- Međutim, nije ništa od toga.
Razvoj softvera je „igra“, igra brzine i saradnje unutar vašeg tima, u konkurenciji protiv drugih timova. To je igra protiv vremena i igra da se napravi popularan brend.
Trebalo bi da uložiš svoj novac na pobjedu.

Rezultat softverskog inženjerstva

Ovo su i glavne discipline softverskog inženjerstva koje ćemo uskoro upoznati.

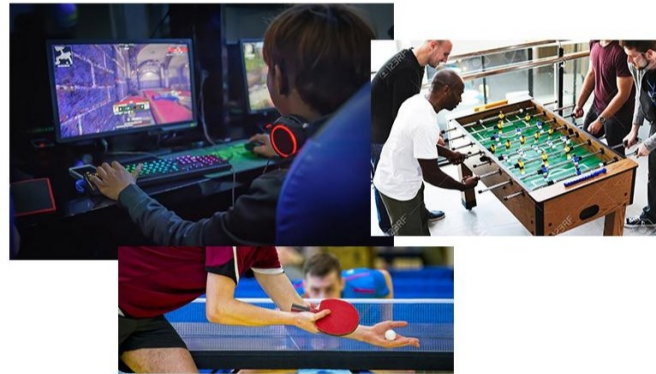
- Efikasan i pouzdan softverski proizvod



Šta rade softverski inženjeri? 😊



What my boss thinks I'm doing



What my friends think I'm doing



* preuzeto sa:
<https://www.reddit.com/r/ProgrammerHumor>



What my mom thinks I'm doing



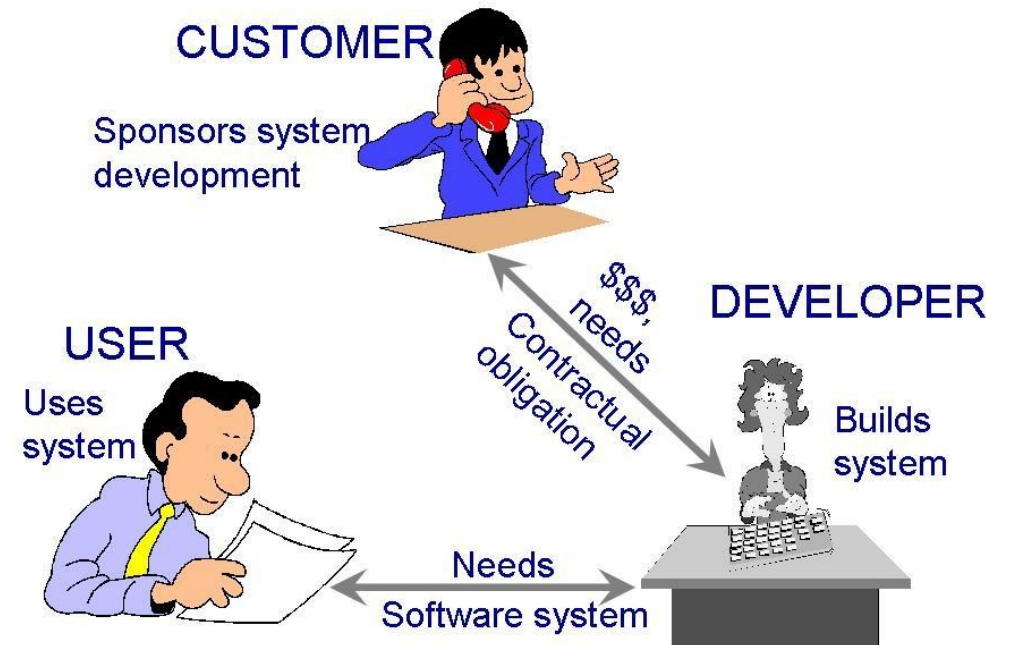
What I'm actually doing

Discipline u okviru softverskog inženjerstva

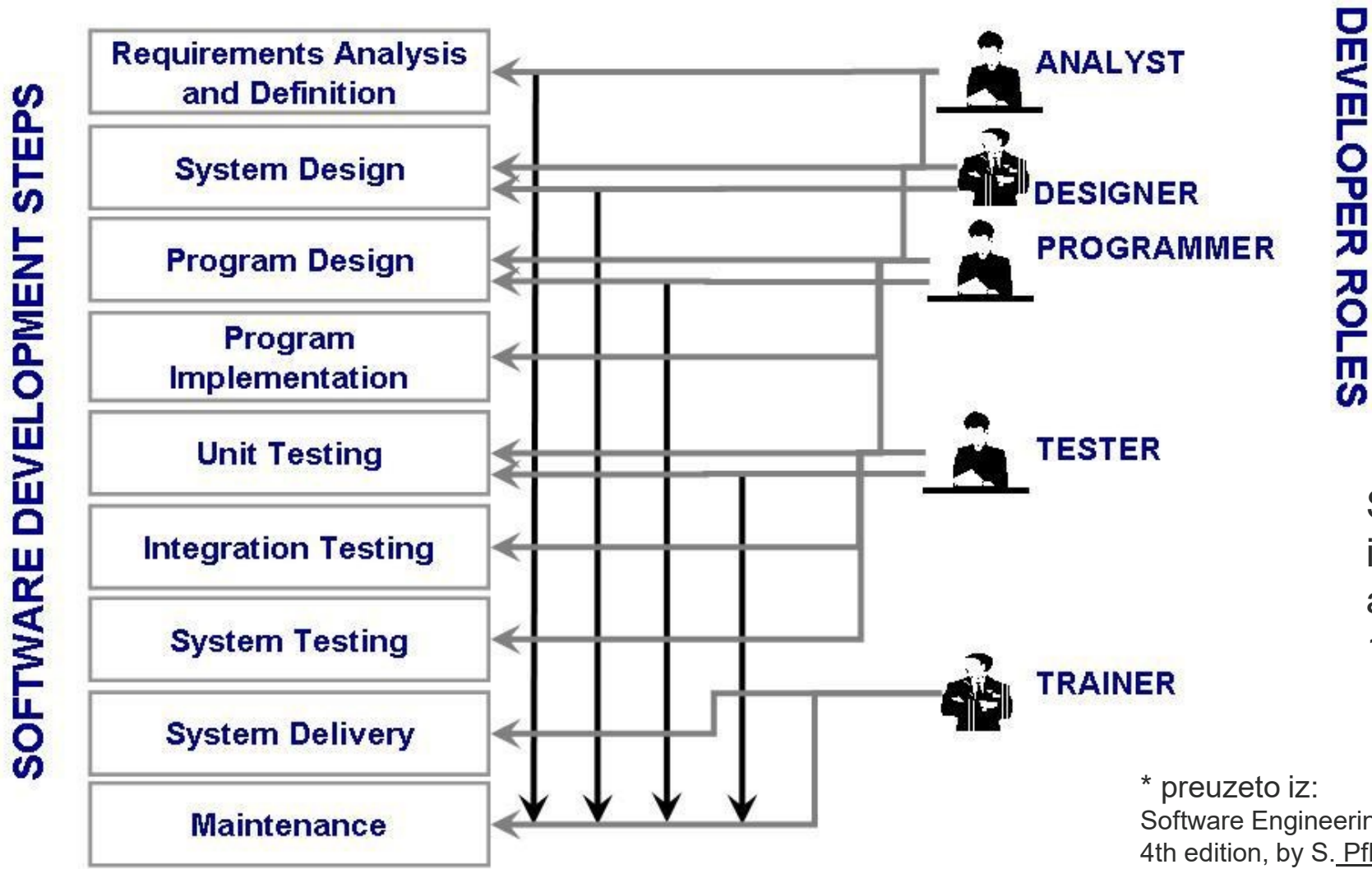


Karakteristike softverskog projekta i softvera

- Uvek pravi kupac.
 - Pre programiranja: pregovori sa kupcem da bismo dogovorili sve korisnički zahteve.
 - Česte su promene želja kupca, a i nesporazumi. Šta onda?
 - Izmene softvera nakon isporuke (nove želje, novi hardver, i slično).
 - Složeni zadaci -> složen program.
 - Softver podložan greškama.
 - Strogi rokovi (*deadlines*).
 - Timski rad.
- Programiranje nekad čini samo 20% svih aktivnosti u razvoju softverskog produkta.
 - Dugogodišnje korišćenje.
A koliko dugo treba da pružamo podršku?



Razvoj softvera je timska aktivnost



Svaki član tima ima svoju ulogu, ali ne mora biti 1:1 odnos.

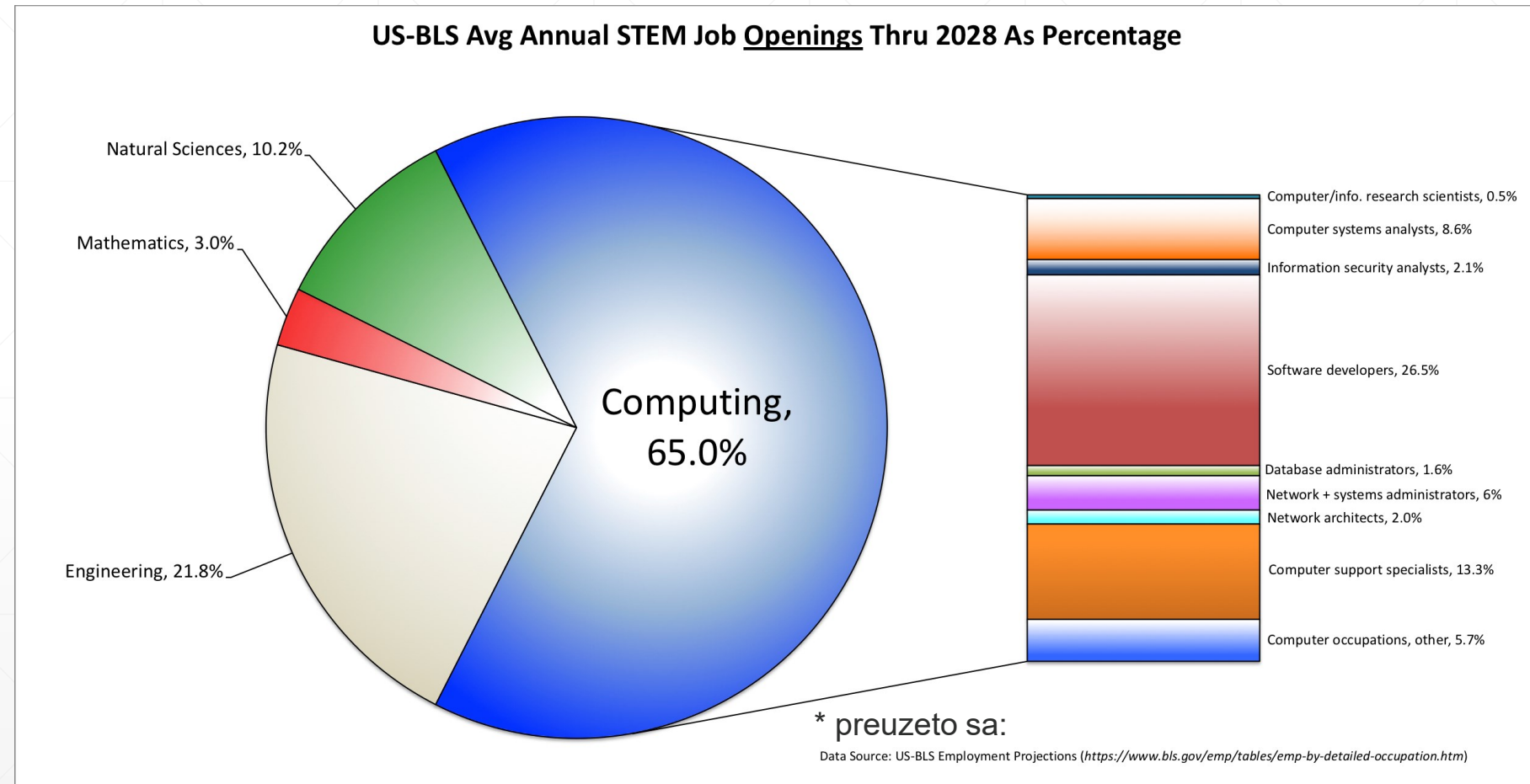
* preuzeto iz:
Software Engineering: Theory and Practice
4th edition, by S. Pfleeger, J. Atlee

Šta ćemo još učiti kroz projekat, osim rada u timu?

- *Software Prototyping* – realizacija nepotpune verzije softverskog produkta koji se razvija
- *Formal Review* – proces formalne inspekcije karakteriše pregled različitih vrsta dokumenata na osnovu strogo definisanih procedura i zahteva.
- *Version Control Systems* – sistemi za verzionisanje (kontrolu verzija), alati kojim pratimo promene u kodu, ali i upravljamo verzijama dokumenata, modela, dijagrama, itd. Primeri: GIT, SVN, Mercurial,...
- *Code Review* – procena kvaliteta koda (metodologija kojom osoba koja radi reviziju koda, identifikuje greške, povećava kvalitet koda, radi refaktorisanje, i pomaže timu da uvek radi sa najčistijim i proverenim kodom).
- *Software Design patterns* – primena projektnih uzoraka u okviru softverskog projekta.
- *Software Documentation* – proces dokumentovanja softvera: pisanje projektne dokumentacije, pisanje komentara u programskom kodu, beleške sa sastanaka i izveštaji (logovi) iz sistema, itd.

Perspektive posla u softverskoj industriji (svet)

- Procene američkog biroa za statistiku rada (US-BLS) do 2028. godine
- Oblast STEM
 - Science
 - Technology
 - Engineering
 - Mathematics



Perspektive posla u softverskoj industriji (Srbija)

poslovi infostud.com

Poslovi ▾ Upoznajte poslodavce Dok tražite posao ▾ Blog

PRIJAVITE SE ▾ ZA POSLODAVCE Objavite oglas

Pozicija, poslodavac, ili ključna reč

Izaberite grad ▾

Odabrano: 2

PRETRAŽITE

Popularni gradovi

Beograd (2203) Novi Sad (765) Niš (439) Su

NAJNOVIJI OGLASI ▾

IT (629)

trgovina, prodaja (516)

mašinstvo (350)

elektrotehnika (327)

ekonomija (opšte) (236)

administracija (206)

arhitektura (42)

I svakog dana sve više novih poslova i pozicija 😊

OSTANITE NA RADARU POSLODAVCA

Omogućite poslodavcima da vide vaš profil i kontakti ponudama za zaposlenje.

Literatura

Autor(i)	Knjiga	Izdavač, broj i godina izdanja
Roger S. Pressman, Bruce Maxim	Software Engineering, A Practitioner's Approach	9. izdanje, McGraw-Hill, 2020.
Ian Sommerville	Software Engineering	10. izdanje, Pearson Education, 2016.
Shari Pfleeger, Joanne Atlee	Software engineering, theory and practice	4. izdanje, Prentice Hall, 2010.
Jovan Popović	Osnove softverskog inženjerstva	1. izdanje, CET, 2019.
<i>druge knjige su takođe pomenute u okviru slajdova gde su korišćene</i>		

Pitanja?

Hvala na pažnji.