

PROJEKTIRANJE INFORMACIJSKIH SUSTAVA

1. KOLOKVIJ

1. Što je informacijski sustav?

Informacijski sustav (Information System – IS System) je sustav za prikupljanje, pohranu, obradu i pružanje informacija. Premda takav sustav može biti implementiran i bez računala danas se pod pojmom informacijskog sustava podrazumijevaju računalni sustavi za prikupljanje, pohranu, obradu i pružanje informacija.

2. Koja su svojstva velikih informacijskih sustava?

- više ljudi organiziranih u timove
- pažljiva analiza, dizajn, implementacija i testiranje
- >100KLOC (LOC – line of code) (1KLOC = 1000 linija koda)
- cijena softvera nadmašuje cijenu hardvera
- sustav je u uporabi više godina i mijenja se s vremenom

3. Koje su 4 faze izrade informacijskog sustava?

- Planiranje
- Analiza
- Dizajn
- Implementacija

4. Kako se dijele metodologije razvoja informacijskih sustava?

- usmjerene na proces koji se informacijskim sustavom obuhvaća (sustav za obračunavanje plaća: proračun radnih sati za mjesec, obračun prethodnih plaća, izračun plaće za ovaj mjesec,)
- usmjerene na podatke koji obuhvaćeni sustavom (sustav za obračunavanje plaća: podaci o radnim satima, podaci o poreznim olakšicama,)
- objektno-orijentirane metodologije obično balansiraju između procesno orijentiranog i podatkovno orijentiranog pristupa informacijskom sustavu

5. Što je karakteristika metodologija razvoja informacijskih sustava orijentiranih na podatke?

- Bitno je definirati sadržaj spremnika podataka i način na koji su oni organizirani.
- Koriste se modeli podataka kao jezgra koncepta sustava.

6. Što je karakteristika metodologija razvoja informacijskih sustava orijentiranih na procese?

- Bitno je fokusirati se na definiranje aktivnosti povezanih sa sustavom
- Cilj je prikazati koncept sustava kao skup procesa s informacijom koja ulazi i izlazi iz njih (tok informacija = flow)

7. Kako se dijele metodologije razvoja informacijskih sustava s obzirom na formalne zahtjeve, brzinu i složenost razvoja sustava i što su svojstva tih metodologija?

1. Strukturirani dizajn (Structured design) (1980-tih) je uveo formalne modele i korištenje dijagrama. Prednosti su u pouzdanosti zbog strogih formalnih zahtjeva, a nedostaci u sporosti i složenosti.
2. RAD (Rapid Application Development) metodologije (1990-te) ubrzavaju razvoj sustava. Mogu i kao metodologije strukturiranog dizajna biti ili procesno orijentirane ili podatkovno orijentirane ili objektno-orijentirane.
3. Metodologija brzog razvoja (Agile Development) još više ubrzavaju razvoj sustava, ali na štetu formalnih zahtjeva. Ovo je najnovija metodologija.

8. Što su prednosti, a što nedostaci Metode Vodopada za razvoj informacijskih sustava?

- Prednosti:
 - Zahtjevi sustava su definirani dugo prije nego počme samo programiranje
 - Promjene zahtjeva su svedene na minimum kako se projekt razvija
- Nedostaci:

- Dizajn mora kompletno biti specificiran prije nego počme programiranje
- Predugo vremena protekne između završetka prijedloga sustava u fazi analize i same isporuke sistema

9. Što su prednosti, a što nedostaci Metode Paralelnog Razvoja informacijskih sustava?

- Prednosti:
 - Vrijeme koje protekne do isporuke se smanjuje, što dovodi do toga da je mogućnost promjene početnih zahtjeva smanjena
- Nedostaci:
 - Ako podsustavi nisu samostalni, dolazi do problema i integraciji ako planiranje nije dobro izvršeno

10. Koje metode se koriste za RAD razvoj informacijskih sustava?

RAD metodologije:

- Phased
- Prototyping
- Throwaway Prototyping

11. Opišite metodu Ekstremnog Programiranja za razvoj informacijskih sustava.

- Bazira se na 4 osnovne vrijednosti:
 - Komunikacija
 - Jednostavnost
 - Feedback (povratna informacija)
 - Hrabrost
- Najbolje koristiti za izradu manjih i srednje velikih aplikacija, zbog nedostatka dokumentacije.
- Glavni principi u izradi uspješnog sustava
- Kontinuirano testiranje
- Što jednostavnije programiranje, u parovima
- Što bolja interakcija s korisnicima (ponekad su dio razvojnog tima)
- Nakon površnog procesa planiranja, projektni tim iterativno provodi faze analize, dizajna te na kraju implementacije sustava.

12. Planiranje informacijskog sustava uključuje dva osnovna koraka. Koji su i što obuhvaćaju?

1. Pokretanje projekta (project initiation) je faza u kojoj se procjenjuje poslovna vrijednost sustava (koliko će smanjiti troškove ili povećati primanja. Dobiveni dokumenti su studija provedivosti (feasibility analysis) i zahtjev sustava (system request)
2. Upravljanje projektom (project management) je faza u kojoj se izrađuje radni plan (workplan), određuje raspodjela posla te definiraju tehnike izrade, kako bi se pomoglo projektnom timu u kontroliranju i usmjeravanju projekta kroz cijeli ciklus SDLC-a. Dobiveni dokument je projektni plan (project plan).

13. Što sadrži studija izvedivosti projekta (feasibility study) koja se dobije analizom izvedivosti informacijskog projekta?

- Tehničku izvedivost (Technical feasibility)
- Ekonomski izvedivost (Economic feasibility)
- Organizacijsku izvedivost (Organizational feasibility)

14. Koja tri koraka obuhvaća analiza informacijskog sustava?

- Analiza treba uključiti identificiranje postojećeg sustava (As-Is system) (koji može već može biti računalni sustav, ali i ne mora).
- Sljedeći korak je identificiranje sustava koji se razvija (To-Be system) na osnovu prikupljenih zahtjeva.
- Rezultat ove faze SDLC treba biti Prijedlog Sustava (System Proposal) koji sadrži niz dokumenata uz revidiranu analizu izvedivosti i radni plan.

15. Što je zahtjev za informacijskim sustavom i koje ključne elemente sadrži?

- Zahtjev za sustavom je dokument koji opisuje razloge predloženog projekta kao i očekivane rezultate.
- Zahtjev za sustavom sadrži ključne elemente projekta:
 - Sponzor projekta
 - Poslovne potrebe (Business need)
 - Poslovni zahtjevi (Business requirements)
 - Očekivanu vrijednost (Business value)
 - Specijalne zahtjeve (Special issues or constraints)

16. Tehnička izvedivost treba dati odgovor na pitanje da li je moguće tehnički realizirati predloženi projekt. Na osnovu koja četiri parametra se procjenjuje tehnička izvedivost informacijskog sustava?

- Upoznatost s predloženom aplikacijom (Familiarity with Application)
- Upoznatost s predloženom tehnologijom (Familiarity with Technology)
- Veličina projekta
- Kompatibilnost s postojećim sustavima

17. Ekonomska izvedivost treba dati odgovor na pitanje da li je ekonomski isplativo realizirati predloženi projekt. Na osnovu čega se određuje ekonomska izvedivost?

Ekonomska izvedivost se određuje na osnovu identifikacije troškova i prihoda (costs and benefits), određivanje njihovih vrijednosti, te proračuna toka novca (cash flow) i povrata investicije (ROI – return on investment)

18. U koje se četiri grupe dijele troškovi i prihodi prilikom proračuna ekonomske izvedivosti informacijskog sustava?

- Troškovi razvoja (development costs) (npr. plaće programera, troškovi konzultanata, ...)
- Troškovi rada (operational costs) (npr. softverske licence, nadogradnja hardvera, komunikacijski troškovi (npr. zakupljeni telekomunikacijski vodovi HT-a))
- Materijalni prihodi (tangible benefits) (prihodi organizacije uslijed uvođenja sustava (npr. porast prodaje) ili smanjene troškova (npr. smanjenje potrebnog broja zaposlenih u tom odjelu pa su manji troškovi za plaće))
- Nematerijalni prihodi (intangible benefits) (to nisu financijski direktno opipljivi prihodi, ali utječu na njih – npr. porast zadovoljstva kupaca će indirektno utjecati na porast prodaje)

19. Na osnovu koja četiri proračuna se procjenjuje ekonomska izvedivost informacijskog sustava?

- Povrat investicije (Return on Investment – ROI ili Rate Return – ROR) je ukupni prihod ili ušteda na osnovu investicije u predloženi projekt.
- Trenutna vrijednost (Present Value – PV) je veličina investicije uspoređena s veličinom iste investicije u budućnosti s obzirom na vrijeme i inflaciju.
- Trenutna neto vrijednost (Net Present Value – NPV) je trenutna vrijednost prihoda umanjena za trenutnu vrijednost troškova.
- Poravnanje vrijednosti (Break-Even Point) se dešava kada se troškovi projekta izjednače s prihodom koji je projekt donio.

20. Kako se proračunava povrat investicije (formula)?

$$ROI = \frac{\text{Ukupni dobitak} - \text{Ukupni troškovi}}{\text{Ukupni troškovi}}$$

21. Kako se proračunava poravnanje vrijednost (formula)?

$$BEP = \frac{\text{Trenutni tok novca} - \text{Kumulativni tok novca}}{\text{Kumulativni tok novca}}, \text{Kumulativni tok novca} \geq 0$$

22. Kako se proračunava trenutna vrijednost (formula)?

$$PV = \frac{\text{iznos}}{(1 + \text{stopa})^n}, n - \text{broj godina}$$

23. Kako se definira upravljanje projektom?

- Upravljanje projektom (project management) je proces planiranja i kontroliranja razvoja sustava.
- Sustav se mora razviti unutar točno određenog vremenskog perioda i uz minimalnu cijenu.
- Gotov sustav treba ispunjavati svoje funkcionalne i nefunkcionalne zahtjeve.

24. Koja su četiri ključna koraka u upravljanju projektom?

- Procjena veličine projekta (project estimation)
- Stvaranje i održavanje plana rada (workplan)
- Odabir osoba koje će raditi na projektu.
- Koordiniranje aktivnosti vezanih uz projekt.

25. Koja se dva osnovna pristupa koriste za procjenu veličine projekta?

- Pristup orijentiran na planiranje (Planning Phase Approach)
- Funkcijski pristup (Function Point Approach)

26. Opišite Pristup orijentiran na planiranje za procjenu veličine projekta.

- Procjena vremena potrebnog za izradu projekta može se napraviti na osnovu vremena potrošenog na planiranje.
- Ukupno vrijeme može se izračunati korištenjem postotka industrijskog standarda po kojem se uobičajeno 15% vremena koristi za planiranje sustava, 20% za analizu, 35% za dizajn i 30% za implementaciju.
- Vrijeme potrebno za fazu planiranja dijeli se sa određenim industrijskim postotkom.
- Na ovaj način možemo izračunati vrijeme potrebno za razvoj ostalih faza (analiza, dizajn, implementacija).
- Ako nam za planiranje projekta trebaju 4 mjeseca, za razvoj čitavog projekta trebat će nam 22.66 mjeseci ako samo 1 osoba radi na njemu.

$$\frac{4}{15\%} = \frac{4}{\frac{15}{100}} = \frac{4}{0.15} = 22.26$$

27. Što je funkcijska točka u Funkcijskom pristupu za procjenu veličine projekta.

Funkcijska točka je mjera veličine programa koja se bazira na broju i kompleksnosti komponenti sustava:

- ulaza (input)
- izlaza (output)
- upita (query)
- datoteka
- programskih sučelja

28. Kako se određuje faktor prilagođene kompleksnosti programa (APC) u Funkcijskom pristupu za procjenu veličine projekta (formula)?

$$APC = 0.65 \cdot \frac{PC}{100}$$

29. Kako se određuje ukupan broj prilagođenih funkcijskih točaka (TAFP) u Funkcijskom pristupu za procjenu veličine projekta (formula)?

$$TAFP = APC \cdot TUFC \quad \text{TUFC} = \text{Total Unadjusted Function Points}$$

30. Kako se po COCOMO modelu procjenjuje količina truda potrebna za izradu informacijskog sustava?

$$TRUD = 1.4 \cdot \frac{\text{broj linija koda}}{1000} [\text{broj osoba/broj mjeseci}]$$

31. Što je radni plan (workflow) informacijskog sustava? Koji se tipovi dijagrama običajno koriste u radnom planu?

Radni plan je dinamički raspored svih zadataka koji se trebaju izvršiti tijekom trajanja projekta.

Zadaci se grafički prikazuju korištenjem Gantt i PERT dijagrama.

32. Prema modelu uragana (Hurricane Model) za koliko može odstupati cijena projekta predviđena projektnim planom u stvarnosti te duljina trajanja projekta?

- Cijena projekta predviđena projektnim planom u stvarnosti može biti i 100% skuplja i projekt može kasniti za otprilike 25% vremena predviđenog za njegov razvoj prema Boehmu (Costs Models for Future Software Engineering). To su margine greški (margin of error).
- Ako smo procijenili da će projekt koštati 100 000 dolara i da će trajati 20 tjedana, on u stvarnosti može koštati između 0 i 200 000 dolara i trajati između 15 i 25 tjedana.

33. Koja se metoda koristi kod upravljanja dosegom projekta? Opišite je ukratko.

To je metoda Timeboxing:

1. Postavite rok do kojeg sustav mora biti gotov.
 - Rok ne bi trebao biti nemoguć
 - Trebao bi ga postaviti razvojni tim.
2. Poredajte funkcije sustava po važnosti.
3. Isprogramirajte "jezgru sustava", odnosno najvažnije funkcije.
4. Funkcije koje se ne mogu isprogramirati unutar zadanog roka odgodite za sljedeću verziju sustava.
5. Dostavite sustav korisniku.
6. Ponavljajte korake od 3 do 5, da bi implemenirali nove zahtjeve i poboljšanja.

34. Kako se dijele zahtjevi informacijskog sustava?

Zahtjevi se dijele na funkcionalne i nefunkcionalne zahtjeve:

- Funkcionalni zahtjevi odnose se na proces pokriven sustavom, na njegovu funkcionalnost i podatke.
- Nefunkcionalni zahtjevi se odnose na ponašanje sustava u smislu performansi sustava, sigurnosti sustava i mogućnosti korištenja.

35. Koje se tri tehnike koriste za određivanje zahtjeva i kako se te tri tehnike razlikuju?

- Automatizacija poslovnog procesa (BPA – Business process automation) – uvođenje malih promjena
- Unapređenje poslovnog procesa (BPI – Business process improvement) – uvođenje umjerenih promjena postojećeg sustava
- Reorganizacija poslovnog procesa (BPR – Business process reengineering) – uvođenje značajnih promjena

	Business Process Automation	Business Process Improvement	Business Process Reengineering
Potential business value	Low-moderate	Moderate	High
Project cost	Low	Low-moderate	High
Breadth of analysis	Narrow	Narrow-moderate	Very broad
Risk	Low-moderate	Low-moderate	Very high

36. Što su funkcionalni zahtjevi informacijskog sustava?

Funkcionalni zahtjevi odnose se na proces pokriven sustavom, na njegovu funkcionalnost i podatke. Npr. sustav mora provjeriti narudžbu kupca s obzirom na raspoloživost kupljenog artikla na skladištu, sustav mora kupcu omogućiti pregledavanje arhive kupnje unazad tri godine,....

37. Što su nefunkcionalni zahtjevi informacijskog sustava?

Nefunkcionalni zahtjevi se odnose na ponašanje sustava u smislu performansi sustava, sigurnosti sustava i mogućnosti korištenja. Npr. sustavu treba biti omogućen pristup sa mobilnih uređaja (Windows CE ??), sustav treba biti stalno dostupan (backup serveri ??), odgovor sustava ne smije biti sporiji od 10 sekundi, sustav treba biti dostupan na različitim jezicima korisnika.... Nefunkcionalni zahtjevi se koriste u fazi dizaja prilikom definiranja korisničkog sučelja, određivanja hardvera i softvera i sl. Ukratko nefunkcionalni zahtjevi ne opisuju sam poslovni proces, ali itekako opisuju budući informacijski sustav za taj poslovni proces.

38. Kojih se 5 osnovnih tehnika koristi za prikupljanje zahtjeva informacijskog sustava?

- Intervjui,
- JAD (Joint Application Development),
- Upitnici (Questionnaires),
- Analiza dokumenata (Document Analysis),
- Promatranje (Observation).

39. Koji su osnovni koraci u tehnici Intervjua za prikupljanje zahtjeva informacijskog sustava?

- Odabir osoba za intervjuiranje
- Definiranje pravila
- Priprema intervju
- Izvođenje intervju
- Obrada prikupljenih podataka

40. Opišite JAD tehniku za prikupljanje zahtjeva informacijskog sustava.

- Grupna tehnika prikupljanja informacija
- Razvijena '70-ih unutar IBM-a
- Suradnja projektnog tima, korisnika i menadžmenta u cilju identifikacije zahtjeva
- Veoma korisna tehnika, strukturirana u njoj sudjeluje od 10 do 20 sudionika

JAD Etape:

- Odabir sudionika na sličan način kao i kod intervju
- Osmišljavanje JAD rasprave
- Pripremanje za JAD sesije
- Vođenje JAD sesije
- Post JAD Follow-Up

41. Koji se kriteriji koriste prilikom odabira tehnike za prikupljanje zahtjeva informacijskog sustava.

- Potencijalna poslovna vrijednost
- Cijena projekta
- Širina analize
- Rizik

42. Koje informacije trebaju biti navedene u slučaju korištenja?

Use case name: Patient makes, changes, or cancels appointment	ID: 2	Importance level: High
Primary actor: Patient		
Short description: This use case describes how we make a new appointment as well as change or cancel an existing appointment		
Trigger: Patient calls and asks for a new appointment or asks to change or cancel an existing appointment		
Type: <input checked="" type="radio"/> External <input type="radio"/> Temporal		

43. Navedite 4 osnovna koraka u definiranju slučajeva korištenja.

- Prepoznavanje najvažnijih slučajeva korištenja
- Prepoznavanje glavnih koraka za svaki slučaj korištenja
- Prepoznavanje elemenata unutar koraka
- Potvrđivanje slučajeva korištenje





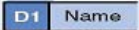
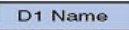
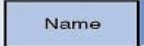
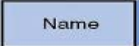
44. Što je model procesa informacijskog sustava?

- Formalni način predstavljanja kako poslovni sustav radi.
- Opisuje aktivnosti koje se izvode i kako podaci teku među njima.

45. Kada se koristi dijagram toka podataka (DFD)? Koji su osnovni elementi u DFD dijagramu?

- Dijagram toka podataka (Data Flow Diagrams, DFD) je uobičajena tehnika za stvaranje modela procesa.
- DFD dijagram se oslanja na tzv. DFD jezik koji definira četiri osnovna elementa DFD-a prikazana grafičkim simbolom:

1. proces
2. tok podataka
3. spremište podataka
4. vanjski entitet

Gane and Sarson Symbol	DeMarco and Yourdan Symbol
	
	
	
	

46. Što je proces, a što tok podataka u DFD dijagramu i koje ograničenje mora biti ispunjeno za proces, tj. za tok podataka da bi DFD dijagram bio sintaktički ispravan?

Proces:

- Aktivnost ili funkcija obavljena zbog specifičnog poslovnog razloga, bilo da je ručni ili kompjuteriziran.
- Preporučeno imenovanje procesa je sa glagolom i imenicom (npr. "Naruči proizvod", "Otvori kredit", "Prijavi ispit",....)
- Svaki proces mora imati barem jedan ulazni tok i barem jedan izlazni tok podataka.

Tok podataka:

- Jedinstveni podatak (npr. ime proizvoda) ili logička kolekcija podataka (npr. podaci o proizvodu – šifra proizvoda, ime proizvoda, jedinična količina, ...).
- Uvijek počinje ili završava unutar procesa, jer tokovi podataka služe za povezivanje procesa.
- Tok podataka pokazuje koji ulazi idu u proces tj. koje izlaze proces kreira.
- Preporučeno imenovanje toka podataka je sa imenicom (npr. "Ime", "Šifra", "Ispit",...)

47. Što je dekompozicija, a što uravnoteženje DFD dijagrama? Što treba biti ispunjeno prilikom dekompozicije DFD dijagram, a što treba biti ispunjeno prilikom uravnoteženja DFD dijagrama?

- Dekompozicija (Decomposition) je postupak predstavljanja procesa kroz hijerarhiju DFD dijagrama pri čemu svaki child dijagram prikazuje dio dijagrama roditelja, ali sa više detalja.
- Uravnoteženje (balancing) osigurava da informacije predstavljene na jednoj razini hijerarhije DFD dijagrama odgovaraju informacijama predstavljenim na sljedećoj razini hijerarhije DFD dijagrama.

48. Koji su koraci u validacija procesa u DFD dijagramu?

Za svaki dijagram provjeriti da li svaki proces ima:

- Jedinstveno ime te broj i opis
- Najmanje jedan ulazni tok podataka (miracle error)
- Najmanje jedan izlani tok podataka (black hole error)
- Imena izlaznih tokova podataka obično drugačija od imena ulaznih tokova podataka.
- Između 3 do 7 procesa u dijagramu.

49. Koji su koraci u validaciji tokova podataka u DFD dijagramu?

Za svaki dijagram provjeriti da li svaki tok podataka ima:

- Jedinstveno ime i opis
- Spaja se na najmanje jedan proces
- Pokazuje samo u jednom smjeru (ne dvosmjerne strelice)
- Minimalan broj križanih linija

50. Što je model podataka informacijskog sustava i koji se dijagram koristi za prikazivanje modela podataka?

- Model podataka je formalni način predstavljanja podataka koji se koriste ili stvaraju u poslovnom procesu.
- U fazi analize razvija se logički model podataka koji se prebacuje u fizički model podataka tijekom faze dizajna.
- Logički model podataka opisuje podatke bez da sugeriraju kako se oni pohranjuju, kreiraju ili mijenjaju.
- Model podataka treba biti balansiran sa modelom procesa.
- Najčešće korištena tehnika za modeliranje podataka je ERD (entity-relationship diagram).
- ERD dijagram grafički prikazuje informacije koje se kreiraju, pohranjuju ili koriste u sustavu.
- Sastoji se od entiteta koji predstavljaju slične informacije.
- Linije između entiteta definiraju relacije među podacima.

51. Koji su osnovni elementi ERD dijagrama? Opišite ih.

Entitet:

- je osoba, mjesto ili stvar
- ima jedinstveno ime pisano velikim slovima
- ima identifikator
- bi trebao sadržavati više od jedne instance podatka

Atribut:

- je svojstvo entiteta
- bi trebao biti korišten od strane najmanje jednog radnog procesa
- se razlaže do najkorisnije razina detalja

Veza:

- pokazuje vezu između 2 entiteta
- ima roditelj entitet i dijete entitet
- opisana je glagolom
- ima kardinalnost (1:1, 1:N ili M:N)

- ima modalnost ("nula" ili "ne nula")
- je ovisna ili neovisna

52. Koja su dva svojstva ERD relacija? Što ta svojstva određuju u relaciji.

- Kardinalnost – određuje koliko instanci jednog entiteta je asocirano sa instancom drugog entiteta, moguće kardinalnosti relacije su 1:1, 1:N ili M:N
- Modalnost – određuje da li instanca child entiteta može postojati bez instance parent entiteta, moguće modalnosti su "nula" ili "ne nula"

53. Da li svaki entitet ERD-a mora imati identifikator? Koje smo tipove identifikatora naveli?

- Jedan ili više atributa entiteta može poslužiti kao identifikatori entiteta (entity identifier), koji jedinstveno identificiraju svaku instancu entiteta
- Spojni identifikator (concatenated identifier) se sastoji od nekoliko atributa
- Identifikatori se točno određuju u fazi dizajna sustava

54. Koja su tri tipa ERD entiteta? Opišite ih.

- Nezavisni entitet:
 - entitet koji može postojati bez pomoći nekog drugog entiteta
 - relacija koja uključuje nezavisni dijete entitet zove se neidentificirajuća relacija
- Zavisni entitet:
 - u slučaju kada dijete entitet zahtjeva attribute od entiteta roditelj da bi jedinstveno identificirali instancu onda se dijete entitet naziva zavisni entitet
 - relacija koja uključuje zavisni dijete entitet naziva se identificirajuća relacija
- Presjecajući entitet:
 - postoji u svrhu prikupljanja informacija o relaciji između dva entiteta
 - dodaju se u podatkovni model da bi pohranili informacije o dvoje entiteta koji dijele M:N relaciju
 - nazivaju se još i asocijativni entiteti
 - proces dodavanja presjecajućeg entiteta naziva se razdvajanje M:N relacije
 - postoje tri koraka za dodavanje presjecajućeg entiteta:
 - uklanjanje M:N relacije i umetanje novog entiteta između dva postojeća
 - dodavanje dvije 1:N relacije u podatkovni model
 - dodjela imena presjecajućem entitetu

55. Što je CRUD matrica, čemu služi, koje podatke sadrži?

- Koristan alat za jasno prikazivanje relacija između procesnog modela i modela podataka je CRUD (Create, Read, Update, Delete) matrica. To je tablica koja opisuje kako procesi koriste podatke unutar sustava.
- Od velike koristi je razviti CRUD matricu baziranu na logičkom procesu i modelu podataka i onda ih kasnije preispitati u fazi dizajna.
- Matrica također pruža važne informacije za specifikaciju programa zato jer pokazuje točno kako su podaci korišteni i stvarani od strane glavnih procesa sustava.

56. Kako se radi validacija modela podataka? Opišite svaki korak u validaciji.

- Normalizacija je tehnika koja se koristi za validaciju podatkovnih modela, to je proces u kojem se niz pravila primjenjuje na logički model da bi se odredilo koliko je dobro formiran.
- Uobičajeno se koriste tri normalizacijska pravila
- Iz postojećih tablica u bazi podataka nastojimo ukloniti redundantne podatke.

Forme:

- Logički model podataka je u prvoj normalnoj formi (1NF) ako ne sadrži ponavljajuće attribute (atributi koji sadrže višestruke vrijednosti za jednu instancu)

1. Ukloniti višestruko pojavljivanje istog stupca u tablici
2. Kreirati posebne tablice za svaki skup povezanih podataka i svaki red tablice identificirati jedinstvenim ključem

- 2NF zahtijeva da svi entiteti budu u prvoj normalnoj formi (1NF).
- Ako entitet ima spojeni identifikator potražiti attribute koji ovise samo o dijelu identifikatora i odvojiti ih u zasebni entitet.
- 3NF se javlja kad je model i u prvoj (1NF) i u drugoj (2NF) normalnoj formi.
- Trebaju biti zadovoljeni svi uvjeti za prvu i drugu normalnu formu.
- Potražiti attribute koji ovise samo o drugom atributu koji nije identifikator i prebaciti ih u novi entitet tj. iz tablice treba ukloniti kolone koje nisu u potpunosti ovisne o primarnom ključu.