

Projektiranje informacijskih sustava

Kolokvij 1 - skripta

1. Projektiranje informacijskih sustava

1.1. Projektiranje informacijskog sustava - što?

- Osmišljavanje i oblikovanje informacijskog sustava koji funkcijom, strukturom, podatkovnim sadržajem i ponašanjem odgovara potrebama organizacijskog sustava za koji se razvija i budućih korisnika.

Projektiranje informacijskog sustava se sastoji od više faza i elemenata, a to su:

- koncipiranje
- dizajn
- svrha i procesi
- građa (arhitektura) - razine - dijelovi i njihove veze (hardware, software, lifeware i orgware)
- baze podataka, dokumenti, zapisi
- nefunkcionalna svojstva, performanse
- misija, vizija, strategija, ciljevi, zahtjevi

1.2. Projektiranje informacijskog sustava - kako?

Postoji više načina na koji možemo projektirati informacijski sustav, a to su:

- Korištenje odgovarajuće metodike projektiranja
- Modeliranje kao osnova projektiranja
- Inženjerski pristup

Da bismo izgradili informacijski sustav, također trebamo razumijeti poslovni sustav koji razvijamo, istražiti ponašanje sustava u radu, poznavati procese sustava kojeg projektiramo na svim razinama, poznavati same tehnologije koje koristimo u izradi informacijskog sustava, moći preustrojiti organizacijske sustave i odabrati optimalnu tehničku potopru za izgrađeni sustav.

No, izgraditi informacijski sustav također znači i projektirati i izvesti sve programe unutar toga IS-a, uskladiti te programe s poslovanjem, funkcionalno i ekonomski optimirati IS, uvesti taj novoizgrađeni IS i obrazovati njegove korisnike o njegovom korištenju, zaštititi taj IS tijekom njegova rada te pratiti uspješnost njegova korištenja.

1.3. Inženjerske discipline

Projektiranje informacijskih sustava je znanost koja je velikim dijelom preuzela svoje spoznaje iz ranijih znanstvenih područja (informacijske znanosti, računarstvo, organizacijske znanosti...), te su na temelju tih spoznaja razvijene *inženjerske discipline i informacijska struka*

Praktičnom primjenom znanstvenih spoznaja u rješavanju problema razvoja, primjene i održavanja informacijskih sustava bave se posebne **inženjerske discipline**:

- Inženjerstvo sustava (System Engineering)
- Informacijsko inženjerstvo (Information Engineering)
- Programsko inženjerstvo (Software Engineering)

Osnovni cilj primjene metodike inženjerskog projektiranja IS-a je određivanje funkcije, strukture i ponašanja informacijskog sustava, primjerenih ciljevima organizacije.

2. Odnos informacijskog sustava i organizacije

2.1. Sustav

Sustav je cjelovita tvorevina, sastavljena od skupa međusobno povezanih dijelova, koja svrhovito djeluje i u međudjelovanju je s okolinom.

Sustave možemo razvrstati prema građi, porijeklu i drugim kriterijima:

- društveni, tehnički, biološki...
- prirodni i umjetni
- složeni (sastavljeni od podsustava) ili jednostavni (dijelovi su im elementi)
- samoregulirajući ili vođeni izvana
- stalne strukture ili promjenjive strukture
- prema namjeni (organizacijski, informacijski te zdravstveni, financijski, upravni...)

2.2. Informacijski sustav

Informacijski sustav je društveno-tehnički sustav koji prikuplja, pohranjuje, čuva, obrađuje ili isporučuje informacije svojim korisnicima, koji mogu biti ljudi, organizacije ili drugi tehnički sustavi.

Informacijski sustav je složen sustav, a njegovu strukturu čine:

- Strojna oprema (hardware)

- Programska oprema (software)
- Komunikacijska i druga infrastruktura
- Podatci (dataware)
- Ljudi (lifeware)
- Društveno-tehničko ustrojstvo, tj. organizacija (orgware)

2.2.1. Procesna komponenta sustava

Sustav ostvaruje svoju svrhu procesom transformacije ulaznih tokova u izlazne.

2.3. Kibernetički model sustava

Kibernetika je znanost o sustavima i upravljanju sustavima.

Kibernetički pogled na sustave pretpostavlja da su sustavi upravljivi. Upravljanje sustavom je njegovo održavanje u željenom stanju, ostvarivanje željenog ponašanja i funkcionalnosti.

Osnovni koncept i mehanizam održavanja sustava u željenom stanju je negativna povratna veza.

2.3.1 Negativna povratna veza

U proces unutar sustava ulaze ulazni tokovi, te ih proces potom transformira u izlazne tokove. Jedan od tih izlaznih tokova se uzima kao mjerni član, preko kojeg se kontrolira ispravnost transformacije unutar procesa. Mjerni član se potom prosljeđuje upravljačkoj jedinici, koja s obzirom na vrijednost mjernog člana nastavlja daljnju uporabu procesa, ili ju zaustavlja.

2.4. Odnos organizacijskog i informacijskog sustava

S obzirom na mjesto i ulogu informacijskog sustava unutar organizacijskog, razlikuju se dva pogleda na informacijske sustave: **preslikavanje organizacije i upravljanje organizacijom**.

2.4.1. Preslikavanje organizacije

Ovaj pogled pretpostavlja pasivnu ulogu informacijskog sustava. Informacijski sustav sadrži opise **prošlih stanja poslovnih procesa i poslovnih događaja** pohranjene u bazi podataka organizacijskog sustava.

Ovaj pogled je nastao u vrijeme klasičnih sustava obrade podataka, namijenjenih trajnom pohranjivanju podataka o poslovanju, zatim obračunavanju, izvještavanju, kontroli i reviziji poslovanja i sl.

2.4.2. Upravljanje organizacijom

Prema ovom pogledu informacijski sustav bitno utječe na strukturu, funkciju i ponašanje organizacijskog sustava.

Osim “preslikavanja organizacije”, svrha informatičkog sustava je **pružanje potrebnih informacija donositeljima odluka** tijekom samih procesa donošenja odluka.

Informacijski sustav je *dio upravljačkog člana* organizacijskog sustava. Korisnici na temelju informacija donose odluke i poduzimaju aktivnosti, kojima utječu na organizacijski sustav.

2.5. Procesni (funkcionalni) pogled na organizaciju

Procesi reprezentiraju funkcionalnost sustava, a funkcionalnost je preduvjet za ostvarivanja svrhe sustava.

2.5.1. Informatički pogled na procese

Osnovne informacije o objektnom sustavu su proces i podatci. Procesi i podatci su osnovni predmet interesa tijekom razvoja informacijskog sustava.

Proces je skup aktivnosti kojima se skup ulaznih podataka transformira u skup izlaznih podataka.

2.5.2. Funkcionalni pogled na organizaciju

Funkcionalno raščlanjivanje je razlaganje neke funkcionalne komponente na funkcionalne komponente niže razine.

Funkcijsko područje sadrži skup logički povezanih funkcija. Funkcija reprezentira postizanje svrhe sustava. Za razliku od procesa, funkcija nema dinamiku, dok proces ima definirane događaje i uvjete kojima se on pokreće i prekida.

2.6. Model poslovanja, poslovnih procesa i informacijskog sustava

Model poslovanja (Business model - BM) je skup međusobno povezanih modela koji opisuju različite aspekte poslovanja, uključujući strukturu organizacije, poslovne ciljeve i prioritete, poslovne procese i pravila, podataka i sl. Model poslovanja mora predložiti koje se vrijednosti isporučuju kupcima, kako se to čini i s kakvim (financijskim) rezultatima.

3. Metodologija informacijskih sustava

Metoda je planski postupak za postignuće nekog cilja na teoretskom ili praktičnom području.

Metodologija je općenito znanost o metodama i njihovoj primjeni.

Metodologija informacijskih/programskih sustava je znanstvena disciplina o pravilima, pristupima, prpcesima, metodama, tehnikama i sredstvima *razvoja, primjene i održavanja* informacijskih i programskih sustava.

3.1. Metodika

Na temelju metodoloških istraživanja i praktičnih iskustava razvijene su *metodike razvoja, primjene i održavanja informacijskih i programskih sustava*.

Metodika je uređen skup načela, pristupa, pravila, činjenica, obrazaca, metoda i tehnika rješavanja nekog zadatka. Ona je također normativni i preskriptivni obrazac za rješavanje posla te je i formalizirana apstrakcija razvojnog poduhvata.

3.1.1. Struktura metodike - pristupi

Pristup (paradigma) je skup početnih pretpostavki o objektu projektiranja i skup općih načela, koja proizlaze iz pojedinih znanstvenih teorija ili iskustva.

Pristupi određuju svrhu, ulogu, strukturu, ponašanje, način razvoja ili korištenja informacijskog ili programskog sustava, njegov odnos s okolinom i sl.

3.1.2. Struktura metodike - načela

- Kakva je *uloga znanosti i struke u projektiranju*? (inženjerske pristup)
- Kakva je *uloga informacijskog sustava*? (preslikavanje organizacije, upravljanje organizacijom)
- Kakva je *namjena sustava*? (obrada transakcija, podrška u odlučivanju, podrška uredskog rada...)
- Kakva je *usklađenost sa zahtjevima organizacije i korisnika*? (tehnički pristup, društveni pristup, tehničko-društveni pristup)
- Što je *objekt modeliranja*, postojeći sustav, ili vizija budućeg sustava?
- Kakva je *prilagodljivost stanju ili promjenama objektnog sustava*? (strukturni pristup, situacijski/adaptibilni pristup)
- Kako izgleda *razvojni ciklus*? (fazni pristup, inkrementalni pristup)
- Kakav je *odnos prema dijelovima i cjelini*? (analitički pristup, holistički pristup)
- Što je *osnovni model*? (funkcijski pristup, podatkovni pristup, objektni pristup) ...

3.1.3. Struktura metodike - obrasci procesa

Proces razvoja informacijskog sustava je skup međusobno povezanih aktivnosti, koje se izvode tijekom razvojnog ciklusa. Stvarni se razvojni poduhvati izvode u skladu s odabranim obrascem procesa razvoja, koji su propisani ili preporučeni metodikom razvoja.

3.1.4. Struktura metodike - metode i tehnike

Tehnika je skup praktičnih postupaka i vještina primjene zadane metode i obavljanja posla u konkretnoj situaciji.

Osnovne metode i tehnike u fazi projektiranja su modeliranje, a u fazi konstrukcije su programiranje.

Metoda - teorija načina primjene teorije *Tehnika* - praksa primjene, vještina

3.1.5. Struktura metodike - koncepti i metamodeli

Modeli sustava oblikuju se i izrađuju pomoću **konceptata**, od kojih je svaki na nekoj apstraktnoj razini reprezentira određeno svojstvo predmeta modeliranja.

Metamodel je skup svih konceptata i načina njihove primjene, u okviru neke metode ili tehnike modeliranja. To je model svih mogućih modela koje možemo izraditi pomoću konceptata neke metode i tehnike.

3.2. Model

Model je eksplicitna interpretacija nečijeg shvaćanja situacije, ili samo predodžbe situacije. Može biti izražen matematički, riječima ili simbolički, ali iznad svega mora biti koristan.

Vrste modela:

- **Slikovni** - slikovna predodžba predmeta, smanjenih, stvarnih ili povećanih dimenzija
- **Analogni** - neka svojstva originala u istom ili drugom fizičkom mediju
- **Matematički**
- **Konceptualni** - opis *kvalitativnih* aspekata područja od interesa ili pojave skupom dogovorenih konceptata (poznata simbolika, sintaksa i semantika)

Svojstva modela:

- **Verbalni modeli** sadrže opis sustava s komentarom na prirodnom jeziku
- **Fizički modeli** su povećane ili smanjene replike fizičkog sustava, izrađeni su u jednakom ili različitom fizičkom mediju u odnosu na original, a oponašaju funkciju i ponašanje sustava

- **Grafički modeli** prikazuju neka svojstva originala u formi slike, crteža ili dijagrama
- **Formalni modeli** su modeli kod kojih su svojstva originala prikazana matematičkih ili logičkim relacijama.

Mjesto modela:

- **Unutarnji (interni)** modeli su uključeni u sustav i dio su njegove strukture
- **Vanjski (eksterni)** modeli su izvan strukture sustava

3.3. Veze metoda, tehnika i modela

Metode mogu biti opće (npr. apstrakcija, generalizacija, klasifikacija i sl.) ili posebne (strukturna analiza sustava, modeliranje ponašanja i sl.). Metode i tehnike projektiranja čvrsto su povezane, tako da se ta dva pojma često koriste kao sinonimi (npr. metoda entiteti-veza i tehnika entiteti-veza).

Tehnike se više vežu uz primjenu metoda. Tako govorimo o **dijagramskim tehnikama** (npr. oblikovanje dijagrama primjenom određene notacije, na temelju znanstvene metode), **tehnikama izrade specifikacija** (primjena specifikacijskog jezika određene sintakse i semantike), **tehnikama grupnog rada** s korisnicima i sl.

Projektiranje informacijskih i programskih sustava je **modeliranje**. Tijekom aktivnosti projektiranja izrađuju se modeli predmeta projektiranja uz pomoć različitih metoda i tehnika modeliranja.

4. Razvojni ciklus informacijskog sustava

4.1. Razvojni ciklus

Razvojni ciklus informacijskog sustava ili podsustava je vremensko razdoblje između donošenja formalne odluke o razvoju i formalne isporuke ili formalnog prekida razvoja.

Rezultat samog razvoja je ciljni proizvod, tj. određeni tip, varijanta ili generacija primjenjivog proizvoda. Rezultat svake faze razvojnog ciklusa je cjelovit, prepoznatljiv i provjerljiv model ili proizvod, koji se primjenjuje u sljedećoj fazi.

Svaki stvarni poduhvat razvoja informacijskog ili programskog sustava odvija se u skladu s neakvim *predloškom* (modelom, obrascem, uzorkom), koji mora biti unaprijed opisan, te usklađen i prihvaćen od strane korisnika i projektanta.

Za neki stvarni poduhvat razvoja, predložak razvojnog ciklusa preuzima se doslovno iz odabrane metodike razvoja ili djelomično prilagođava stvarnim potrebama.

Predložak razvojnog ciklusa uvijek sadrži:

- podjelu razvojnog ciklusa na faze
- opis mogućeg slijeda faza
- uvjete prelaska s jedne faze razvoja na drugu

4.2. Strategijsko planiranje informacijskog sustava

Izrađuje se grubi konceptualni model postojećih procesa i podataka organizacijskog sustava, te model postojećeg informacijskog sustava.

Poslovodstvo aktivno sudjeluje u definiranju modela poslovnog sustava, što uključuje *model ciljeva*, *model ključnih čimbenika uspjeha*, *model kritičnih pretpostavki*, *model problema* i *model potreba za informacijama te drugih zahtjeva prema IS-u...*

Na temelju analize ovih modela određuje se: *gruba struktura informacijskog sustava*, *prioriteti* i *redoslijed realizacije podsustava* i *okvirni troškovi* i *izvodljivost*.

Rezultat je strategijski plan informacijskog sustava, s planskim obzorom 3 do 5 godina.

4.3. Utvrđivanje izvodljivosti poduhvata

Cilj je utvrđivanje granica (opsega) i izvodljivosti planiranog poduhvata razvoja s tehničkih, tehnoloških, organizacijskih, ekonomskih i drugih gledišta.

Postoji više razina studije izvodljivosti i pratećih studija: *studija mogućnosti (opportunity)*, *prethodna studija izvodljivosti (prefeasability)*, *studija izvodljivosti (feasability)* i *studije potpore (support studies)*.

Granice poduhvata se utvrđuju na grubom konceptualnom modelu podataka i procesa.

Rezultat je: **studija izvodljivosti, projektni zadatak budućeg poduhvata razvoja i eventualno natječajna dokumentacija.**

4.4. Analiza i specifikacija zahtjeva

U ovoj fazi se profinjava model podataka i procesa, odnosno objektni model. Detaljno se analiziraju i specificiraju zahtjevi prema budućem sustavu, koji se odnose na: *podatkovne sadržaje*, *funkcionalnost* i *tehnologiju rada*, *sučelje*, *odaziv*, *performanse* i *ostale oblike ponašanja* i *ostale nefunkcionalne zahtjeve*.

4.5. Logičko modeliranje

Ova faza obuhvaća izradu detaljnog logičkog modela budućeg sustava, koji opisuje što on mora biti. U strukturnim tehnikama to znači: *dekompoziciju procesa, dijagrame toka podataka i opise unutarnje logike elementarnih procesa i model entitet-veze, njegovu pretvorbu u relacijski model i relacijsku analizu.*

U objektno orijentiranim tehnikama (npr. UML) to znači:

- specifikacija slučajeva korištenja i opis njihove unutarnje logike dijagramima aktivnosti
- izrada strukturnih modela (dijagrama klasa i podataka)
- opis dinamike objekta (strojevi stanja)
- specificiranje poruka (dijagrami slijeda i komunikacije)

4.6. Fizičko modeliranje i izrada

U ovoj fazi izrađuje se fizički model baze podataka (tablice, ključevi, indeksi, distribucija i replikacija), definiraju se programski moduli i njihova distribucija, konstruiraju se programi, detaljno se razrađuju uloge korisnika i pogledi, detaljno se razrađuje sustav autorizacije korisnika i sigurnosti, detaljne se oblikuje sučelje, testiraju se pojedini moduli, moduli se integriraju i testira se cijelina, model se dokumentira i pojedini dijelovi i sustav u cjelini se validiraju s korisnicima.

4.7. Isporuka i primjena

U ovom procesu se instalira oprema i programi potrebni za IS, osposobljavaju se korisnici (edukacija radnika), obavlja se konverzija i integracija podataka, testira se prihvatljivost sustava tijekom razdoblja probnog rada, po potrebi se obavljaju određene korektivne aktivnosti i daje se ocjena prihvatljivosti.

4.8. Održavanje i poboljšavanje

U ovom procesu se pružaju različiti oblici podrške korisnicima, otklanjaju se uočene greške u funkciji ili ponašanju, sustav se prilagođava novim izdanjima operacijskog sustava, poboljšanjima i proširenjima opreme, novim komunikacijskim mogućnostima i sl., također se sustav prilagođava promjenama poslovnih pravila, poslovnih tehnologija, zakona i sl., poboljšavaju se performanse sustava, u određenoj mjeri se proširuje funkcionalnost sustava, te se sustav primjenjuje novim lokacijama, prihvaćaju se novi korisnici i sl.

4.9. PDCA (Plan-Do-Act-Check)

PDCA je iterativna metoda upravljanja razvojem i kontinuiranim poboljšanjem procesa i proizvoda koja se sastoji od 4 korake (faze): *plan, do, act, check*.

4.10. pristupi s obzirom na razvojni ciklus

Najkorišteniji i najpoznatiji pristupi su: *vodopadni pristup, fazni pristup, evolutivni prototipni pristup, djelomično inkrementalni pristup, inkrementalni pristup, spiralni pristup i agilni pristup*.

Najkorišteniji od ovih u modernom razvoju je agilni pristup.

4.11. Životni ciklus informacijskog sustava

Životni ciklus informacijskog sustava formalno započinje s uspostavom organizacijskog sustava kojem IS pripada, a prestaje s prestankom rada organizacije.

Životni ciklus u užem smislu označava vremensko razdoblje između donošenja formalne odluke o izgradnji sustava bitno različite građe, tehnologije, funkcionalnosti i sl., u odnosu na postojeći sustav, a završava odlukom o zamjeni sustava novim sustavom.

5. Proces razvoja informacijskog sustava

5.1. Proces razvoja

U funkcionalnom smislu, svaki poduhvat razvoja informacijskog sustava je proces (traje i troši resurse).

Proces razvoja informacijskog sustava je skup međusobno povezanih aktivnosti, koje se izvode tijekom razvojnog ciklusa. Aktivnosti se, unutar procesa razvoja, izvode vremenski slijedno (sljedeća aktivnost započinje nakon završetka prethodne), vremenski usporedno ili ciklički (nakon završetka neke od sljedećih aktivnosti, vraća se na prethodnu).

Stvarni se razvojni poduhvati izvode u skladu s odabranim obrascem procesa razvoja, koji je propisan ili preporučeni metodikom razvoja.

5.2. Predložak (obrazac) procesa razvoja

Predložak procesa razvoja opisuje: *rašćlanjivanje procesa i aktivnosti na aktivnosti niže razine, veze i slijed izvođenja aktivnosti, uvjete početka i završetka aktivnosti, uvjete početka i završetka aktivnosti, ulazne i izlazne podatke i informacije potrebne za izvođenje svake aktivnosti i metode, tehnike, pomagala i sredstva koja se koriste tijekom izvođenja neke aktivnosti*.

5.3. Vrste aktivnosti

S obzirom na ciljeve koji se njima ostvaruju, aktivnosti se dijele na:

- aktivnosti planiranja i upravljanja
- aktivnosti učinka
- aktivnosti provjere

5.3.1. Aktivnosti planiranja i upravljanja

Svrha **aktivnosti planiranja** je: *zadavanje ciljeva, vodećih veličina i indikatora (vrijeme, resursi, kapaciteti) i određivanje stanja i načinu njihovih postizanja (strategija, taktika i operacije).*

Svrha **aktivnosti upravljanja** je: *upravljanje drugim vrstama aktivnosti (npr. aktivnostima učinka), rezultatima aktivnosti i resursima (financijskim sredstvima, ljudskom snagom...).*

5.3.2. Aktivnosti učinka

Osnovna svrha **aktivnosti učinka** je obavljanje unaprijed određenog posla (postizanja učinka ili stanja). Svrha aktivnosti učinka očituje se u izlaznom rezultatu. U projektiranju informacijskih sustava, izlazni rezultati aktivnosti učinka su razni modeli, specifikacije ili izrađeni dijelovi informacijskih sustava.

Aktivnosti učinka su:

- aktivnost prikupljanja i analize informacija
- aktivnost modeliranja
- aktivnost programiranja
- aktivnost izrade dokumentacije

5.3.3. Aktivnosti provjere

Svrha **aktivnosti provjere** je provjeravanje i vrednovanje rezultata aktivnosti učinka, utroška resursa i vremenskog napredovanja poduhvata.

Dvije osnovne vrste aktivnosti provjere su: *verificiranje* (formalno provjeravanje usklađenosti s ulaznim specifikacijama) i *validiranje* (vrednovanje od strane korisnika).

6. Sustavni pristup razvoju IS i osnovni modeli

6.1. Kibernetički pogled na sustav

Sustav je cjelovita tvorevina koja svrhoviti djeluje u međudjelovanju s okolinom. Prema kibernetičkom pogledu, sustavom se upravlja i odlučuje na temelju podataka i informacija iz procesa.

6.2. Sustavni pristup izgradnji IS-a

IS je model poslovne tehnologije organizacijskog sustava: *programi automatiziraju ili podržavaju realne procese u skladu s poslovnim pravilima, a baze podataka sadrže podatke o stanjima i promjenama stanja sustava i poslovnim transakcijama.*

Temelj razmatranja su **poslovni procesi**, a želimo ih držati stabilnima, kako ne bismo trebali učestalo mijenjati programe.

Podatci su resurs poslovnog sustava i najstabilniji dio IS.

IS se izgrađuje *holističkim pristupom* i integracijom podsustava na temelju *interoperabilnosti*:

- poslovna
- procesna
- podatkovna (semantička)
- tehnička

6.3. Objekt modeliranja

Subjektivistički pristup (TO-BE) zanemaruje važnost modeliranja sadašnjeg sustava. Prema ovom pristupu, razvoj budućeg programskog sustava odmah započinje prikupljanjem, analiziranjem i specificiranjem zahtjeva korisnika prema budućem programskom sustavu, te izradom njegovog modela.

Subjektivistički pristup modeliranju orijetniran je oblikovanju novog sustava (dizajnerski pristup), a objektivistički je orijetiran analizi postojećeg (analitički pristup).

Objektivistički pristup (AS-IS + TO-BE) polazi od objektivne stvarnosti, tj. modela sadašnjeg objektnog sustava (uključujući model sadašnjeg informacijskog i korisničkog sustava).

Modeliranje postojećeg sustava zahtijeva određeni utrošak resursa, ali olakšava:

- određivanje granica poduhvata
- vrednovanje rezultata poduhvata

- upoznavanje problemske domene
- migraciju na novi sustav

6.4. Poslovni model

Poslovni model je skup međusobno povezanih modela koji opisuju različite aspekte poslovanja, uključujući strukturu organizacije, poslovne ciljeve i prioritete, poslovne procese i pravila, podatke i sl. Također predočava koje se vrijednosti isporučuju kupcima, kako se to čini i s kakvim (financijskim) rezultatima i kako se ta vrijednost kontinuirano održava.

6.5. Digitrans metodologija

6.5.1. Faza analize

Analiza postojećeg poslovnog modela poduzeća:

- razvoj mape strateškog krajobraza
- razvoj mape “plavog oceana”
- izrada mape dionika
- analiza kompetencija
- ...

Istraživanja korisničkog potreba:

- izrada platna prijedloga vrijednosti
- intervju
- razvoj profila pojedinačnog kupca (*persona*)
- korisničko putovanje

6.5.1.1. Razvoj mape “plavog oceana”

Cilj je definirati trenutnu poziciju na temelju stanja na tržištu, konkurenciju, kupce a posebno na one koji bi mogli i za koje biste htjeli da vam postanu kupci u budućnosti.

6.5.1.2. Izrada platna prijedloga vrijednosti

Vrijednosno platno (*value proposition canvas*) nam pomaže u shvaćanju, prepoznavanju i opisivanju potencijalnih ciljanih skupina kupaca. Izradom platna lakše se definiraju aktivnosti kupca, dobrobite ili dobitke i problemi koje kupac ima, a vaš proizvod može riješiti.

6.5.1.3. Razvoj profila pojedinačnog kupca

Profil pojedinačnog kupca predstavlja idealnog kupca. Temelji se na istraživanju korisnika, npr. putem intervjua i uključuje potrebe, ciljeve i obrasce promatranog ponašanja ciljane publike.

Stvara se empatija prema ciljanim skupinama/kupcima kojima će novo razvijeni digitalni poslovni model riješiti određeni problem ili zadovoljiti potrebe.

6.5.2. Faza dizajna

Kreiranje ideja i odabir rješenja:

- brainstorming
- brainwriting
- ...
- odabir najbolje ideje

Prototipiranje i testiranje:

- prototipiranje
- SAP scene
- ...
- izrada platna poslovnog modela

6.5.2.1. Prototipiranje i testiranje

Svrha prototipa jest vizualizacija ideje koja se razvila na temelju istraživanja korisnika i dionika kako bi se stvorila zajednička osnova razumijevanja i rasprave.

Pomaže u stvaranju ideja koje su opipljive i povratnih informacija prezentiranjem ideje uključenima u proces.

6.6. Osnovni model tijekom razvoja IS

Osnovne informacije bitne za razvoj informacijskog sustava su podatci i procesi objektnog sustava. Zbog toga su *modeli podataka i modeli procesa* osnovni modeli koji se izrađuju tijekom projektiranja.

Modeli procesa su formalizirani opisi hijerarhijskih struktura procesa, unutarnje logike procesa, međusobnih odnosa procesa, događaja u objektnim sustavima, te odnosa procesa i okoline.

Osnovna svrha pojedinih procesa informacijskog sustava je prikupljanje, pohranjivanje, pretraživanje, obrada i prikazivanje podataka. Da bi se ova svrha ostvarila, nužno je poznavanje i razumijevanje strukture podataka objektnog i informacijskog sustava.

Modeli podataka prikazuju strukturu podataka, odnose među skupovima podataka i njihova ograničenja te mehanizma promjene podataka.

Modeli objekata u semantičkom smislu objedinjavaju modele podataka i modele procesa, njihovu statiku i dinamiku.

6.7. Izrada plana razvoja i utvrđivanje izvodljivosti

1. Određivanje prioriteta i redoslijeda realizacije podsustava (na temelju PK matrice)
2. Definiranje pristupa s obzirom na vlastiti razvoj i korištenje vanjskih usluga
3. Planiranje projekta
4. Utvrđivanje izvodljivosti planiranog poduhvata razvoja s tehničkog, tehnološkog, ekonomskog, organizacijskog i drugih gledišta.
5. Definiranje plana nabave, projektnog zadatka i natječajne dokumentacije za proizvode i usluge koje se kupuju.

6.8. Pristup s obzirom na odnos vlastitog razvoja i gotovih komponenta

Outsourcing je način razvoja informacijskog sustava gdje organizacija naručuje od informatičke tvrtke provedbu dijela ili cijelog postupka razvoja IS-a. Prednost ovog načina je da će sustav prilagođen organizaciji i njezinom poslovanju, a nedostatak su cijena i dugo trajanje izrade.

Insourcing je vlastiti razvoj IS-a, odnosno postojanje dovoljnog broja informatičara koji su sposobni izraditi IS za organizaciju. Prednost ovog pristupa je prilagodljivost sustava, no nedostatak je moguća smanjena učinkovitost u redovnom poslovanju.

Kupovanje gotovog rješenja je opcija u kojoj organizacija kupi jedno od mnogobrojnih rješenja koja su dostupna na tržištu. Prednost gotovog rješenja je da se učinkovitost IS-a skoro odmah vidi, no nedostatak su cijena i potreba za prilagodbom procesa samom IS-u.

Program kao usluga (SaaS) je pristup gdje je organizacija vlasnik podataka, a sam sustav (programi, platforma i infrastruktura) su unajmljeni kao usluga. Opsluživanje, potpora i održavanje se također unajmljuju vanjskom dobavljaču.

7. Slučajevi korištenja

Slučajevi korištenja koriste se u sistematskom i programskom inženjerstvu. Opisuju odnos sustava (aplikacije) i okoline (korisnika i drugih sustava). Slučajevi korištenja opisuju osnovnu specifikaciju funkcioniranja sustava i

odnosa s okolinom, koju razumiju i korisnici. Srodni su sa korisničkim pričama i dijagramima konteksta.

7.1. Model slučajeva korištenja

Model slučajeva korištenja se sastoji od *specifikacije slučajeva korištenja* i *dijagrama slučaja korištenja*. Slučajevi korištenja su **izvorno tekstualni opisi**, a ne dijagrami. Dijagrami se u slučajevima korištenja koriste da daju kontekst tekstualnom opisu.

Postoje tri razine složenosti:

- **Skraćena**(pregledna) - samo glavni scenarij
- **Puna** - više scenarija i specifikacije svakog scenarija
- **Detaljna** - svaki scenarij s detaljnom specifikacijom unutarnje logike, početnim i završnim stanjima, zahtjevima prema sučelju, specifikacijom obrade grešaka i iznimki...

Dijagram slučajeva korištenja opisuje *što* sustav radi, s gledišta vanjskog promatrača. Ne prikazuje kako sustav funkcionira iznutra, jer to i nije bitno za vanjskog promatrača.

7.2. Osnovni koncepti

Slučaj korištenja (Use Case) je skup akcija koje može izvesti neki sustav u međudjelovanju sa učesnicima. To je priča koja opisuje kako učesnici koriste sustav da bi postigli određene ciljeve ili da bi obavili neki zadatak. On predstavlja apstraktni zadatak (sadrži skup aktivnosti) kojeg izvode neki učesnici.

Veza (Association) je sudjelovanje učesnika u slučaju korištenja, odnosno komunikacija između instanca korisnika i slučaja korištenja.

Učesnik (Actor) je skup podudarajućih uloga koje korisnici imaju u interakciji sa slučajem korištenja. Učesnik je neko tko spada izvan promatranog sustava i s tim sustavom je u međudjelovanju, ali nijeg dio samog sustava. Učesnik može biti živo biće ili pak može biti drugi sustav.

7.3. Generalizacija učesnika

Koristi se kada su različiti učesnici u jednakom međudjelovanju sa slučajevima korištenja i imaju zajedničko ponašanje. Djeca nasljeđuju svojstva i ponašanje od roditelja, a nasljeđuju i veze sa slučajevima korištenja od roditelja (slično nasljeđivanju klasa).

7.4. include i extend

Uključi (include) je veza od *osnovnog slučaja korištenja* prema *uključenom slučaju korištenja*, koja označava da osnovni slučaj korištenja sadrži ponašanje uključenog slučaja korištenja. Na taj se način funkcionalnost i ponašanje koje se često koristi izdvaja u slučaj korištenja kojeg možemo uključivati u druge slučajeve korištenja prema potrebi. Osnovni slučaj korištenja **nije kompletan** bez uključenja.

Proširi (extend) je veza od slučaja korištenja koji je *proširenje prema osnovnom slučaju korištenja*, koja označava da se ponašanje osnovnog slučaja korištenja proširuje ponašanjem proširenja. Na primjer, *plati karticom* proširuje slučaj korištenja *plati račun*, ali nije njegov dio, jer ipak možemo plaćati račune i bez kartice. *Plati račun* funkcionira i bez *plati karticom*.

7.5. Granice sustava

Granica sustava predstavlja granicu između fizičkog sustava i učesnika koji su u interakciji s fizičkim sustavom.

8. Logički model podataka

8.1. Osnovni model tijekom razvoja IS-a

Razlikuju se četiri glavna pristupa:

- Funkcijski pristup, tj. pristup orijentiran procesima
- Podatkovni pristup, tj. pristup orijentiran podacima
- Funkcijsko-podatkovni pristup
- Objektni pristup, tj. pristup orijentiran objektima

Osnovne informacije o objektnom sustavu su podatci i procesi objektnog sustava, pa su zbog toga modeli procesa i modeli podataka osnovni modeli koji se izrađuju tijekom projektiranja. Modeli podataka i modeli procesa mogu biti potpuno odvojeni, ili mogu biti međusobno povezani.

U semantičkom smislu, modeli objekata objedinjavaju modele podataka i modele procesa.

8.2. Podatkovni pristup

Podatkovni pristup pretpostavlja da je model podataka osnovni model, koji se oblikuje tijekom razvoja informacijskog i programskog sustava.

Model podataka je stabilniji od modela procesa, struktura procesa i njihova unutarnja logika je više izložena promjenama tijekom projektiranja, dok struktura podatka nije. Prikupljanje, pohranjivanje, obrada i prikaz podataka osnovna je svrha velikog broja funkcija informacijskog sustava, te da bi se ova svrha ostvarila, potrebno je dobro poznavati strukturu podataka s kojom radimo. Također, izrada formalnih modela funkcije i ponašanja nije moguća bez prethodne izrade modela podataka.

8.3. Definicija i osnovni koncepti

Model podataka je apstraktni prikaz skupova podataka, njihovih međusobnih veza i ograničenja, te načina manipulacije podatcima.

Osnovni koncepti modela podataka su:

- skup koncepata za **opis strukture podataka** (podatci koji opisuju objekte i pojave mogu se razvrstati, tako da svaki elementarni podatak pripada nekom tipu atributa nekog tipa podatkovnog objekta)
- skup koncepata za **očuvanje integriteta podataka** (skup pravila koja opisuju dozvoljena stanja sustava i dozvoljeni prijelaz iz stanja u stanje)
- skup operatora kojima je moguće opisati **promjenu stanja podataka** sustava (mehanizmi promjene vrijednosti podataka)

8.4. Statički i dinamički model

Cjeloviti model sustava prikazuje statička i dinamička svojstva sustava.

Statički modeli prikazuju *strukturu i stanje podataka sustava*:

- podatci koji opisuju stvarne ili zamišljene objekte i pojave objektnog sustava mogu se razvrstati u *podatkovne tipove* (klase, entitete), tako da svaki elementarni podatak opisuje neko svojstvo (atribut) nekog podatkovnog objekta
- podatkovni objekti su međusobno povezani **vezama**
- grupiranje pojava u tipove obavlja se na temelju zajedničkih tipova atributa i/ili na temelju sličnih veza s drugim podatkovnim tipovima
- svaki podatkovni objekt u jednom trenutku ima jedno stabilno, prepoznatljivo i konzistentno **stanje**, opisano skupom vrijednosti njegovih atributa

Dinamički modeli opisuju *promjene stanja sustava*, odnosno njegovih podatkovnih objekata i veza među njima:

- stanje sustava mijenja se vanjskim djelovanjem (događaj), na način da se promijene vrijednosti podataka koji opisuju stanje sustava

- nakon ove promjene nastaje novo stabilno, prepoznatljivo i konzistentno stanje sustava

8.5. Model entiteti-veze-atributi

Model entiteti-veze-atributi (ERA model) prikazuje podatke sustava u obliku tipova entiteta, koji su opisani tipovima atributa i povezani tipovima veza.

ERA model je osnovni i najčešće primjenjiva vrsta konceptualnog i logičkog modela podataka. Prema logičkom modelu se određuje buduća struktura baze podataka unutar sustava. Samo logičko modeliranje je neovisno o software-u i hardware-u, uključujući i sustav za upravljanje bazom podataka.

Najbitnije verzije ovog modela su *ERA model Chena i Martina*. Notaciju koju koristimo pri kreiranju ERA modela na vježbama i projektu je Martinova.

8.6. Entitet

Entitet je nedvosmisleno prepoznatljiv koncept, predmet, biće ili događaj o kojem se u informacijskom sustavu prikupljaju i pohranjuju podatci. Entiteti mogu biti materijalni ili apstraktni (nematerijalni), stvarni ili zamišljeni.

Pojave entiteta se mogu grupirati u *tipove entiteta*, odnosno skupove pojava s jednakim tipovima atributa.

8.7. Veza

Veza povezuje pojave dva tipa entiteta ili različite pojave istog tipa entiteta. *Tip veze* se predočava spojnom crtom, a *red veze* određuje koliko tipova entiteta sudjeluje u vezi (unarna, binarna, n-arna).

Unarna (rekurzivna, refleksivna) veza povezuje dvije različite pojave istog tipa entiteta. Najčešće pokazuje strukturu, odnosno da pojava entiteta sadrži pojave istog tipa, ali niže razine složenosti.

8.8. Ograničenja

Osnovna ograničenja u ERA modelu su:

- ograničenja domene atributa
- **ograničenja kardinalnosti veza entiteta**
- ograničenja kardinalnosti veza entiteta i njihovih atributa

Ograničenja kardinalnosti veza određuju broj pojava jednog tipa koji može ili mora biti povezan s jednom pojavom drugog tipa. Kardinalnost može biti 0, 1, M (više) ili neki određeni broj (npr. 4).

8.9. Asocijativni tip entiteta

Asocijativni tip entiteta nastaje od:

- *veze kardinalnosti više:više ($M:N$)* - svaka veza tipa $M:N$ se može pretvoriti u dvije veze tipa $1:M$ i $1:N$
- veze koje sadrže atribut
- veze tri ili više tipova entiteta (n-arna veza)

8.10. Atribut

Atribut je podatak koji opisuje entitet ili omogućava njegovo prepoznavanje. Atributi entiteta razvrstavaju se na:

- *identifikacijske* (omogućavaju prepoznavanje pojave tipa entiteta)
- *opisne* (opisuju stanje, položaj i kvalitetu tipa entiteta)
- *izvedene* (vrijednosti im se svode na vrijednosti drugih tipova atributa)

U jednom vremenskom trenutku, jedna pojava tipa entiteta može imati samo jednu vrijednost atributa za svaki tip atributa, što omogućuje praćenje stanja pojava entiteta.

9. Relacijski model i pretvorbe

9.1. Relacijski model

Relacijski model zasnovan je na teoriji skupova. U ovom modelu, svaki iskaz se može napisati u obliku relacijske sheme, npr.:

- “Kupci u našem poduzeću naručuju određene količine različitih proizvoda.”

se može zapisati kao: `NARUDŽBA(Kupac, Proizvod, Količina)`.

Relaciju čini skup zapisa sa značenjima, kao zapis iznad, koji se mogu prikazati u tabličnom obliku.

9.2. Relacijska shema

Relacijska shema je definicija relacije u obliku naziva relacije i popisa atributa (obilježja) koji su u sastavu te relacije: $R(a_1, a_2, a_3, \dots, a_n)$. To je shema po kojoj se podatci zapisuju u bazu, ona ne predstavlja same podatke. Sama shema je nepromjenjiva, dok podatci nisu.

9.3. Ograničenja relacijske sheme

Relacijska shema ne smije sadržavati dva ista jednaka atributa. Sam poredak atributa u tablici (shemi) je proizvoljan i bira ga sam projektant. Različite relacije smiju sadržavati jednake attribute, ali u takvom slučaju se dodaje prefiks na ime atributa kako bi se označilo kojoj relaciji pripada.

Posljedice ovih ograničenja su:

- tablica *ne smije* sadržavati dva ista stupca
- redoslijed stupaca u tablici je proizvoljan
- promjena stupca ne mijenja sadržaj ni značenje