SE322-iNŽENJERSTVO ZAHTEVA

**L01**

1. **Objasnite tri nivoa zahteva.**

Zahtevi određuju svrhu i ponašanje sistema. Mogu biti prikazani na tri nivoa:

1**. poslovni zahtev**i - objašnjavaju ZAŠTO organizacija primenjuje sistem, koje poslovne želi da ostvari. Fokus je na poslovnim ciljevima.

2**. zahtevi korisnik**a - opisuju ciljeve ili zadatke koje korisnici moraju da izvrše na sistemu da bi obezbedili nekome neku vrednost.

3. **funkcionalni zahtevi** - određuju ponašanje sistema pod određenim uslovima. Oni opisuju ŠTA inženjer razvoja treba da uradi da bi omogućui korisniku da izvrši svoj zadatak.

1. **Objasnite disciplinu inženjerstva zahteva koja se naziva upravljanje softverskim zahtevima.**

**Upravljanje zahtevima** je proces upravljanja promenama u zahtevima sistema. Zahtevi sistema se uvek menjaju kako bi odslikali promene potreba skateholdera, promene u okruženju u kojem je sistem instaliran, promene u preduzeću koje planira da instalira sistem, promene u zakonskoj regulativi itd.

* Osnovne aktivnosti upravljanja zahtevima su **kontrola promena** i **ocena uticaja promena**.

1. **Objasnite disciplinu inženjerstva zahteva koja se naziva razvoj zahteva.**

Inženjerstvo zahteva se može podeliti na: **razvoj zahteva** i na **upravljanje zahtevima.**

Razvoj zahteva se realizuje izvršenjem sledećih aktivnosti:

• Prikupljanje zahteva • Analiza zahteva • Specifikacija zahteva • Potvrđivanje

1. **Na šta se odnose zahtevi projekta, a na šta zahtevi proizvoda?**

**Zahtevi proizvoda** utiču na svojstva softvera u razvoju. To su zahtevi koji se vezuju za sam softverski proizvod koji se razvija.

**Zahtevi projekta** obuhvataju elemente koji učestvuju u projektovanju i u izradi softvera. Oni uključuju - Fizičke resurse koji su potrebni razvojnom timu, Potrebe obuke članova tima.

1. **Šta su procesni zahtevi? Navedite bar dva primera.**

**Procesni zahtevi** se odnose na proces razvoja softverskog sistema i postoje kada korisnici sistema žele da utiču na njega i uključuju zahteve za korišćenjem standarda, CASE alata i dobijanje različitih izveštaja za menadžment.

**Primer:**

1. Sistem se mora razvijati korišćenjem XYZ paketa CASE alata.
2. Razvojni proces mora biti eksplicitno definisan i mora se uklapati u ISO 9000 standard.
3. **Na šta se odnose eksterni zahtevi? Navedite bar dva primera.**

**Eksterni zahtevi** su zahtevi koji se mogu svrstati i u zahteve proizvoda i u procesne zahteve. Oni proizilaze iz okruženja u kojem se sistem razvija.

**Primer:**

1. Sistem za edukaciju studenata: Format podataka o studentskom zapisu se mora podudarati sa onim koji se koristi u nacionalnom sistemu za evidenciju studenata. 2.
2. Sistem medicinskih podataka: Službenik za zaštitu podataka u organizaciji mora da pre nego što se sistem stavi u rad, izda sertifikat da se svi podaci održavaju prema legislativi za zaštitu podataka.
3. **Koje su posledice primene loših zahteva? Navedite i objasnite bar tri.**

* Glavna posledica pogrešno definisanih zahteva je ponovni rad na utvrđivanju boljih i realnijih zahteva, a to povećava troškove projekta.
* Greške u postavljanju zahteva učestvuju u troškovima ponovnog rada sa sa 70 do 85%.
* Troškovi otklanjnja loših zahteva i definisanja boljih su najmanji ako se te promene izvrše što pre. Ako se te izmene vrše kasnije, troškovi znatno rastu, jer se ponovno moraju obaviti sve aktivnosti

1. **Koji su problemi vezani za zahteve?**

* Najčešći problemi su: softver se isporučuju sa zakašnjenjem i prevazilazi se planirani budžet.
* Ne rade ono što korisnici zaista žele i često se nikada ne iskoriste neke mogućnosti sistema za koje su korisnici platili.

**L02**

1. **Opišite osnovne aktivnosti procesa razvoja zahteva.**

Proces razvoja zahteva:

1. **Prikpuljanje** – Primenjuju se različiti načina za prikupljanje zahteva od korisnika i drugih aktera.

2. **Analiza** - Analiza zahteva poboljšava zahteve, otklanja greške i nedostatke u njima, a omogućava bolje razumevanje zahteva od strane svih aktera

3. **Specifikacija** - Svrha specifikacije zahteva je dokumentovanje zahteva različitog tipa na jedan konzistentan, pristupačan i proverljiv način koji je razumljiv akterima.

4. **Potvrdjivanje -** Potvrđivanje obezbeđuje kontrolu da li su zahtevi korektni, i pokazuju karakteristike kvaliteta i zadovoljenje potreba korisnika.

1. **Koji sve načini mogu biti za prikupljanje zahteva?**

Definiši viziju projekta i njegov okvir, Utvrdi grupe korisnike Sistema, Odaberi "šampiona proizvoda" za svaku klasu korisnika, Organizuj intervjue sa akterima, vodite fokus grupe sa tipičnim korisnicima.

1. **Koje akcije podrazumeva potvrđivanje zahteva?**

Recenzija zahteva, Testiranje zahteva, Definisanje kriterijuma prihvatljivosti, Simulacija zahteva.

1. **Ko su akteri u razvoju softverskog sistema? Navedite i objasnite različite klase aktera.**

Klasa korisnika je podskup šireg skupa korisnika, a skup korisnika je podskup aktera procesa.

1. **Klasa omiljenih korisnika** je ona čije zadovoljenje je blisko povezano sa ostvarenjem poslovnih ciljeva firme koja kupuje softver. Ovo su korisnici zbog koji se prvenstveno i razvija novi softver.
2. **Klasa nenaklonjenih korisnika** su korisnici koji ne koriste softver iz legalnih, bezbednosnih ili sigurnosnih razloga.
3. **Ko je šampion proizvoda?**

**Šampion proizvoda** služi kao primarni interfejs između određene klase korisnika i biznis analitičara. Idealno, oni su i krajnji korisnici, a ne neki posrednici. Šampioni proizvoda prikupljaju zahteve od drugih članova njegove klase korisnika i otklanjaju nekonzistentnosti.

1. **Koje pripremne aktivnosti se mogu sprovesti pre razvoja softvera u određenom domenu primene?**

Obuka analitičara poslovanja, Obuka aktera o zahtevima, Obuka inženjera razvoja o domenu primene softvera, Definisanje procesa inženjerstva zahteva, Kreiranje rečnika termina.

**L03**

1. **Objasnite pojam poslovnih zahteva.**

**Poslovni zahtevi** predstavljaju skup informacija koji, u zbiru, opisuje potrebu koje rezultat projekata razvoja softvera mora da zadovolji sa stanovišta zahteva.

Poslovne zahteve čine poslovne mogućnosti, poslovni ciljevi, matrica uspeha i iskaz o viziji.

1. **Kako se određuje poslovna korist od projekta i koji je njen značaj za razvoj projekta?**

* Sponzori projekta/rukovodstvo/menadžeri marketinga odredjuju poslovne ciljeve projekta. Obično je to izazov zbog različitih stavova aktera. Zato BA treba da ima pripremljena pitanja i da zna koga treba da pita, da bi mogao da formuliše poslovnu korist koju projekt treba da omogući organizaciji.
* Poslovna korist treba da donese istinsku vrednost sponzorima projekta i korisnicima proizvoda.

1. **Šta predstavljaju vizija proizvoda i okvir proizvoda?**

**Vizija proizvoda** opisuje konačan proizvod koji će ostvariti poslovne ciljeve i ona je stabilna. **Okvir projekta** utvrđuje koji deo vizije proizvoda koji će biti predmet rada na projektu.

1. **Šta se događa ako su poslovni zahtevi suprotstavljeni? Na koji način se može rešiti problem suprotstavljenih poslovnih zahteva?**

* Poslovni interesi aktera mogu biti i suprostravljeni. (Kupac želi sto pre da obavi kupovinu, a prodavac želi da zadrži kupca što duže kako bi još nešto kupio.) Ako njihovo razrešenje dovede do menjanja poslovnih ciljeva, onda treba da zatražite i promenu plana projekta.
* Zahtevajte da se utvrde razlike I izvrše promene u rokovima, budžetu projekta i resursima. Funkcionalni i nefukcionaln zahtevi takođe onda treba da se promene.

1. **Koja je svrha dokumenta o viziji i okviru? Navedite i objasnite neka od najvažnijih poglavlja ovog dokumenta.**

**Dokument o viziji i okviru** sadrži poslovne ciljeve koji su osnova za naredne aktivnosti na projektu. Organizacije koje razvijaju komercijalni softverski proizvod često ovaj dokument zovu"dokument sa zahtevima marketinga".

1. **Poslovni zahtevi** - Poslovni zahtevi opisuju primarne koristi koje novi sistem obezbeđuje sponzorima, kupcima i korisnicima.
2. **Okvir i ograničenja** - Okvir i ograničenja pomažu da projektni akteri imaju realistična očekivanja jer ponekad, kupci zahtevaju svojstva čija realizacija može biti preskupa ili koja su van nameravanog okvira projekta..
3. **Poslovni kontekst** – Razumevanje faktora koji utiču na poslovanje iz različitih perspektiva, uključujući način donošenja odluka, profile aktera, prioritete projekta.
4. **Šta je iskaz o viziji? Kako se formuliše?**

**Iskaz o viziji** sumira dugoročnu svrhu i namenu proizvoda. On treba da odražava uravnotežen pogled koji će da zadovolji očekivanja različitih aktera. Ključne reči: za, koji, naziv proizvoda, je, za razliku od, naš proizvod..

1. **Objasnite dijagram konteksta kao tehniku za određivanje okvira.**

**Dijagram konteksta** vizuelno ilustruje granicu između sistema koji se razvija i svega ostalog u univerzumu. Utvrđuje i spoljne entitete, kao interfejse sistema, podatke, kontrolu i tok.

Ceo sistem je predstavljen jednim krugom a entiteti pravougaonicima.

1. **Objasnite mapu ekosistema kao tehniku za određivanje okvira.**

**Mapa ekosistema** pokazuje sve sisteme povezane sa vašim sistemom koji su u međusobnoj interakciji i prikazuju prirodu ovih interakcija. One pokazuju i druge sisteme koji su povezani sa sistemom koji se razvija, uključujući i sisteme sa kojima on nema direktne interfejse.

1. **Objasnite stablo svojstava kao tehniku za određivanje okvira.**

Stablo svojstava prikazuje logično grupisana svojstva sistema, hijerarhiјski podeljena na detaljnije nivoe prikazivanja.

* Glavna grana na sredini predstavlja proizvod. Svako svojstvo ima svoju granu. Pravougaonici označavaju N1 svojstva. Linije koje polaze od N1 grane su N2 svojstva

**L04**

1. **Kako identifikovati i klasifikovati korisnike softvera?**

Pri identifikaciji korisničkih klasa razmišljajte o zadacima koje će različiti korisnici obavljati sa sistemom. Npr. Finansijska institucija. Pojedinci koji obavljaju odredjene aktivnosti (bilo da su u pitanju zvanja ili jednostavno uloge) imaće slične funkcionalne potrebe za sistem u svim finansijskim institucijama.

1. **Koja je uloga šampiona proizvoda?**

**Šampion proizvoda** služi kao primarni interfejs između članova jedne korisničke klase i poslovnog analitičara projekta i usklađuju neusaglašenosti. Šampioni bi trebali biti efikasni komunikatori kojima je potrebno temeljno razumevanje domene aplikacije i operativnog okruženja rešenja. Može postojati više šampiona proizvoda za jedan projekat.

1. **Koje su zamke šampiona proizvoda koje treba izbegavati?**
2. Šampion proizvoda koji zaboravi da zastupa druge kupce i predstavlja samo svoje zahteve, neće raditi dobro.
3. Menadžeri nadjačavaju odluke koje donosi kvalifikovani i propisno ovlašćeni šampion proizvoda.
4. **Objasnite ulogu vlasnika proizvoda.**

**Vlasnik proizvoda** definiše viziju proizvoda i odgovoran je za razvoj i određivanje prioriteta sadržaja zaostalih proizvoda. Vlasnik proizvoda obuhvata sva tri nivoa zahteva: posao, korisnik i funkcionalna. On u suštini razvlači šampione proizvoda i funkcije poslovnog analitičara, predstavljajući kupca, definišući karakteristike proizvoda, postavljajući im prioritete.

1. **Ko rešava problem sukobljenih zahteva?**

Šampioni i proizvoda ili vlasnik proizvoda to mogu rešiti u mnogim, ali verovatno ne u svim slučajevima. Na početku projekta odredite ko će biti donosioci odluka u vezi sa zahtevima. Ako se to ne uradi odluke će pasti na programere ili analitičare sto nije dobro jer vecina njih nema potrebno znanje.

**L05**

1. **Opišite cikličan način razvoja zahteva.**

Vi prvo vršite utvrđivanje zahteva, analizirate šta ste naučili i onda zapisujete neke zahteve. Tada i utvrđujete šta vam od informacija nedostaje i onda vršite dodatno utvrđivanje zahteva i tako se ciklus utvrđivanje-analizaspecifikacija zahteva ponavlja.

1. **Objasnite intervju kao tehniku za utvrđivanje zahteva.**

**Intervju -** Vode se individualni ili intervjui sa malim grupama korisnika. Jednostavnije ih je organizovati od radionica. Upoznajte sagovornika sa sadržajem razgovora, Držite se okvira, Pripremite pitanja i model sagovornika, Sugerišite ideje, Aktivno slušajte

1. **Objasnite fokus grupu kao tehniku za utvrđivanje zahteva.**

**Fokus grupa** je reprezentativna grupa korisnika koja se saziva radi usmeravane aktivnosti izazivanja zahteva a radi davanja predloga i ideja za funkcionalne zahteve i zahteve kvaliteta. One su interaktivne i očekuje se da svi učesnici govore. Fokus grupe su zgodne za izražavanje stavova korisnika, impresija, preferencija, i potreba.

1. **Objasnite upitnik kao tehniku za utvrđivanje zahteva.**

**Upitnici** - su jeftini i dobri u slučaju velikog broja ispitanika. Najvažnije je dobro pripremiti pitanja. Preporuke: (Ponuđeni odgovori ne smeju da budu međusobno isključivi. • Ne postavljajte pitanja na koje se može samo potvrdno odgovoriti. • Ako koristite ocene, onda ih koristite u svim pitanjima)

1. **Objasnite analizu interfejsa sistema kao tehniku za utvrđivanje zahteva.**

**Analiza interfejsa sistema** ispituje sistem sa kojim je vaš sistem povezan. Analiza interfejsa sistema otkriva funkcionalne zahteve u vezi sa podacima i servisima koje sistem razmenjuje sa drugim sistemima. Dijagrami konteksta i mape ekosistema su dobra priprema za ovu analizu.

1. **Objasnite radionice kao tehniku za utvrđivanje zahteva.**

**Radonica** je strukturisan sastanak na kome grupa aktera i eksperata rade zajedno na definisanju, kreiranju, poboljšanju i postizanju saglasnosti o zahtevima korisnika. Radionice treba dobro planirati da se ne bi gubilo vreme, jer mogu da traju i nekoliko dana. (Postavite osnovna pravila, Planirajte agendu, Držite se okvira, Zapisivanje ideja..)

1. **Koje su preporuke za planiranje izazivanja zahteva? Kako se pripremiti za aktivnost izazivanja zahteva?**

Planirajte okvir sesije i agendu. Pripremite resurse. Pripremite pitanja o modelu sagovornika.

Planirajte okvir i dnevni red sednice. Saznajte više o akterima Prepoznajte relevantne aktere za sesiju. Pripremite UML model aktera. Pripremite pitanja.

1. **Šta treba uraditi posle sesije prikupljanja zahteva?**

Posle izvršenja aktivnosti prikupljanja zahteva, ima puno preostalog posla. Treba da organizujete i podelite beleške sa sesija, da dokumentujete otvorena pitanja, da klasifikujete prikupljene informacije.

1. **Kako klasifikovati informacije dobijene od kupca posle aktivnosti prikupljanja zahteva?**

Analitičar mora da klasifikuje pregršt zahteva koje je čuo u više kategorija zahteva, da ih dokumentuje i da ih koristi kako treba. Morate da spojite informacije u jednu jasno prikazanu i dobro organizovanu kolekciju zahteva. Prođite nekoliko puta kroz prikupljene informacije, da bi proverili da li klasifikovani zahtevi imaju karakteristike zahteva visokog kvaliteta.

1. **Kako znate da ste završili sa utvrđivanjem zahteva?**
2. Korisnici ne mogu da iznesu još neki slučaj korišćenja niti priču korisnika.
3. Korisnici predlažu nove scenarije, ali oni ne dovode do novih funkcionalnih zahteva.
4. Korisnici ponavljaju pitanja koja su već ranije prodiskutovana.
5. Inženjeri razvoja i testeri koji pregledavaju zahteve u nekom domenu, postavljaju malo pitanja.
6. **Koji se problemi javljaju uzrokovani pretpostavljenim i podrazumevanim zahtevima? Kako naći zahteve koji nedostaju?**
7. **Pretpostavljeni zahtevi**: Njih korisnici očekuju i bez formalne specifikacije. Ono što je korisniku očigledan zahtev, inženjeru razvoja ne mora da bude.
8. **Podrazumevani zahtevi**: Oni moraju da postoje jer ih zahtevaju drugi zahtevi, ali nisu eksplicitno definisan u projektu. Inženjeri razvoja ne mogu da primene funkcionalnost za koju ne znaju.

* Dekomponujte uopštene zahteve u konkretnije, detaljnije zahteve,
* Pratite zahteve korisnika
* Proverite granične vrednosti zbog nedostajućih zahteva
* Predstavite informacije o zahtevima na više načina
* Model podataka može da otkrije funkcionalnost koja nedostaje

**L06**

1. **Objasnite primenu slučajeva korišćenja i korisničkih priča.**

**Slučaj korišćenja ili upotrebe** opisuje sekvencu interakcija između sistema i spoljnih aktera koja dovodi akterima ostvarenje rezultata od značaja za njih. Nazivi slučajeva korišćenja se uvek pišu u formatu "glagol + objekat".

**Priča korisnika** je kratak opis svojstva ispričan iz perspektive osobe koja želi novu sposobnost, koja je obični ili korisnik ili kupac. Kao <tip korisnika> želim <neki cilj> tako da<neki razlog>

1. **Uporedite slučajeve korišćenja i korisničke priče. Koje su razlike, a koje su zajedničke karakteristike?**

* Slučajevi korišćenja i priče korišćenja služe da se utvrde šta korisnici žele da sistem radi. Korisne su za poslovne aplikacje, razvoj veb sajtova, i interakcije.
* Slučaj korišćenja pomaže analitičaru i korisniku da otkriju sve neophodne informacije vezane za slučaj korišćenja, a kor. priče i otkrivaju dodatne informacije.
* Priče korisnika nude prednost u konciznosti i jednostavnosti, a slučajevi korišćenja daju učesnicima strukturu i kontekst, što pričama korisnika nedostaje.

1. **Šta su scenariji korišćenja i u kakvoj su vezi sa slučajevima korišćenja?**

**Scenario** je opis jednog primera upotrebe sistema. Jedan slučaj korišćenja ima kolekciju scenarija a scenario je jedan primer (instanca) slučaja korišćenja.

1. **Šta se opisuje preduslovima i postuslovima slučajeva korišćenja?**

**Preduslovi** definišu šta mora da bude ispunjeno da bi sistem počeo da izvršava slučaj korišćenja.

**Postuslovi** opisuju stanje sistema posle uspešnog izvršenja slučaja korišćenja.

1. **Šta se opisuje normalnim tokom slučaja korišćenja, šta alternativnim tokom i izuzetkom?**

Jedan scenario predstavlja jedan **normalan tok** (glavni tok) događaja za slučaj korišćenja. Može imati i alternativne tokove (scenarije).

**Alternativni tokovi** isporučuju rezultat kao i normalni tok, ali predstavljaju manje uobičajen rezultat ili neki specifičan rezultat.

**Izuzeci** opisuju uslove pri kojima se javljaju greške prilikom izršenja slučajeva korišćenja.

1. **Objasnite vezu između slučajeva korišćenja i poslovnih pravila.**

Neka poslovna pravila ograničavaju uloge svih ili pojedinih delova nekog slučaja korišćenja. Poslovna pravila mogu da utiču na naredne korake u normalnom scenariju definišući ispravne ulazne vrednosti ili određujući kako će se izvršavati obračuni.

1. **Dajte preporuke za utvrđivanje slučajeva korišćenja.**

Prvo utvrdite aktere , onda postavite poslovne procese koje sistem treba da podrži, i onda definišite slučajeve korišćenja za aktivnosti u kojima raspodelu dolazi do interakcija sistema i aktera.

1. **Objasnite vezu između slučajeva korišćenja i funkcionalnih zahteva.**

Slučajevi upotrebe opisuju perspektivu korisnika, sagledavajući spoljno vidljivo ponašanje sistema. Oni ne sadrže sve informacije potrebne programeru da bi napisao softver. slučajevi upotrebe često ne pružaju sve potrebne informacije programeru da zna šta da gradi. BA mora izvući te nedostajuće zahteve i dostaviti ih programerima i ispitivačima.

1. **Koje su zamke pri radu sa slučajevima korišćenja?**

Koristite isuviše mnogo slučajeva korišćenja, Upotreba vrlo složenih slučajeva korišćenja, Uključivanje projektovanje u slučajeve korišćenja, Uključivanje definisanje podataka u slučajeve korišćenja, Slučajevi korišćenja koje korisnici ne razumeju.

1. **Prednosti korišćenja zahteva koncentrisanih na korisnika?**

Slučajevi korisnika pomažu da se razjasne razne nejasnoće i protivurečnosti u ranim fazama razvoja sistema, a i mogu da generišu testove na osnovu slučajeva korišćenja.

**L07**

1. **Šta su poslovna pravila? Kakva je primena poslovnih pravila u praksi?**

* **Poslovna pravila** definišu kontrolne principe poslovanja neke organizacije, kao što su zakoni, strategije i industrijski standardi.
* U svakom sektoru poslovanja, postoji regulativa koja se mora poštovati, i ona se unosi u sistem preko poslovnih pravila.
* **Poslovna pravila** utiču na poslovni proces postavljanjem određenih termina, unoseći određena ograničenja, pokreću aktivnosti, u određuju kao se obračuni moraju da izvrše. Jedno pravilo može da se primeni u više poslovnih procesa.

1. **Objasnite klasifikaciju poslovnih pravila.**

Postoji pet tipova pravila: **činjenice, ograničenja, omogučivači akcije, zaključci i računanja**.

1. **Činjenice** su iskazi poslovanju koji su istiniti i određenom vremenskom trenutku.
2. **Ograničenje** je izjava koja ograničava radnje koje sistem ili njegovi korisnici mogu da obavljaju. Ograničenja su vezana za: pravila organizacije, zakonske regulacije, industrijske standard

**Matrica uloga i ovlašćenja** definiše ovlašćenja koje imaju pojedine uloge korisnika sistema.

1. **Omogućivači akcija** su pravila koja pokreću neku akciju ako su ispunjeni određeni uslovi. Uslovi se mogu definisati složenim logičkim izrazima, koji daju vrednost "istinito" ili ""pogrešno". If <neki uslov je zadovoljen > then < nešto se desi >
2. **Zaključak** stvara novu činjenicu iz drugih činjenica.
3. **Računanja** transformišu postojeće podatke u nove podatke koristeći posebne matematičke formule ili algoritme.
4. **Koji su načini za dokumentovanje poslovnih pravila?**

* U manjim organizacijama se koriste katalozi za dokumentovanje poslovnih pravila. Ako se koristi softver za upravljanje zahtevima, onda se ona unose kao kao poseban tip zahteva.
* Možete koristite i struktuirane uzorke- formulare za definisanje poslovnih pravila. U slučaju korišćenja složenih pravila, ona se mogu dokumentovati primenu stabala odlučivanja i tabela odlučivanja. kao i matricama uloga i ovlašćenja.

1. **Kako se otkrivaju poslovna pravila? Koja pitanja postaviti korisnicima?**

* Da bi utvrdili potrebna poslovna pravila, potrebno je tražite različite izvore u organizaviji u kojima se mogu nači potrebne informacije.
* Poslovna pravila se otkrivaju postavljanjem pitanja korisnicima iz različitih perspektiva.(Zasto to radite tako? Sta je korisniku sledece dozvoljeno?Kako su ovi podaci povezani?)

1. **Kako dobiti funkcionalne zahteve iz poslovnih pravila? Koja je veza između poslovnih pravila i zahteva?**

* Poslovna pravila su spoljni iskaz o politici i pravilima koja morate sprovesti i koja usmeravaju funkcionalnost softvera, a funkcionalni zahtevi da ih primene.
* Ako koristite alat za upravljanje zahtevima, kreirajte atribut zahteva koji se zove „Poreklo“ i označite pravila kao poreklo izvedenih funkcionalnih zahteva.
* Definišite veze sledljivosti između funkcionalnih zahteva i povezanih poslovnih pravila u matrici sledljivosti zahteva ili matrici mapiranja zahteva
* Ako su poslovna pravila i zahtevi smešteni u datotekama za obradu teksta ili proračunske tabele, definišite hiperlinkove od referenci ID-a poslovnih pravila u zahtevima nazad do opisa poslovnih pravila sačuvanih na drugom mestu.

**L08**

1. **Koji su mogući načini za obeležavanje zahteva?**

Svaki zahtev treba da ima jedinstveni identifikator:

1. Broj redosleda (REQ-1)
2. Hijerarhijsko dodeljivanje brojeva (REQ 3.2, REQ 3.2.1)
3. Hijerarhijski tekstualni tagovi (Product.Cart.Discount.Error.Shipping)
4. **Koje preporuke za obeležavanje zahteva možete izdvojiti?**
5. Svaki zahtev treba da ima jedinstveni identifikator
6. Obeležite i uredite konsistentno sekcije, podsekcije, i pojedinačne zahteve.
7. Koristite vizuelna sredstva (boldiran, italik, podvučena i obojena sl
8. Dajte brojeve svim tabelama
9. Koristite linove ka pojedinim delovima SRS dokumenta
10. **Šta sve treba da sadrži SRS dokument?**

Specifikacija softverskih zahteva (SRS -Software Requirement Specification) sadrži funkcije i sposobnosti koje softverski sistem mora da obezbedi, njegove karakteristike, i ograničenja koja mora da poštuje.

**Delovi SRS-a**: Uvod, Opšti opis, Funkcije Sistema, Zahtevi za podatke, Zahtevi za spoljne interfejse, Atributi kvaliteta, Uslovi internacionalizacije I lokalizacije, Ostali uslovi

1. **Kako se sve mogu predstaviti i opisati funkcije sistema?**

Ovaj obrazac ilustruje organizovanje funkcionalnih zahteva za proizvod prema karakteristikama sistema, glavnim uslugama koje proizvod pruža. Ovaj odjeljak možete radije organizovati prema slučaju, načinu rada, korisničkoj klasi, klasi predmeta, funkcionalnoj hijerarhiji..

1. **Objasnite zahteve za podatke i zahteve za spoljne interfejse.**

* Ovaj odeljak opisuje različite aspekte podataka koje će sistem koristiti kao ulaze, obrađivati na neki način ili kreirati kao izlaze. (Logicki model, recnik podataka, izveštaji, prikupljanje podataka integritet zadrzavanje I odlaganje)
* Ovaj odeljak sadrži informacije kojima se osigurava da će sistem pravilno komunicirati sa korisnicima i sa spoljnim hardverskim ili softverskim elementima. (Zahtevi za spoljne interfejse, Korisnički i Softverski interfejsi)

1. **Objasnite atribute kvaliteta i uslove internacionalizacije.**

* Atributi kvaliteta - ovde se definišu nefukcionalni zahtevi
* Zahtevi internacionalizacije i lokalizacije osiguravaju da će proizvod biti pogodan za upotrebu u nacijama, kulturama i geografskim lokacijama koje nisu one u kojima je stvoren. Takvi zahtevi mogu da reše razlike u: valuti; formatiranje datuma, brojeva, adresa i telefonskih brojeva; jezik

1. **Objasnite aktivnost specifikacije zahteva pri agilnom razvoju softvera.**

Umesto funkcionalnih zahteva, najčešće se razvoj upravlja prema postavljenim testovima prihvatanja nastalim iz korisničkih priča. Svaka priča je jedan iskaz o potrebama korisnika ili o funkciji koju celi sistem treba da ima. Kod svake iteracije akteri projekta diskutuju i definišu nove detalje priča korisnika.

1. **Koje su prepoznate karakteristike odličnih zahteva?**

Kompletnost, tačnost, ostvarljivost, potreba, usklađenost sa pioritetima, nedvosmislenost, proverljivost - to su karakteristike dobro definisanih zahteva.

1. **Koje preporuke za pisanje zahteva možete izdvojiti?**
2. Razmenjujte specifikacije zahteva sa vašim kolegama, analitičarima.
3. Pišite zahteve na konzistentan način
4. [ opcioni uslov] [opcioni događaj pokretanja] [očekivan odgovor sistema] sistem treba [odgovor sistema ]
5. Pišite jednostavne, posebne i pojedinačne zahteve.
6. Pišite aktivnim rečenicama tako da bude jasno šta entitet treba da uradi
7. **Kako se pišu zahtevi iz perspektive korisnika, a kako iz perspektive sistema?**

Pri pisanju zahteva, možete ga pisati tako da kažete šta sistem treba d a uradui (perspektiva sistema) ili šta korisnik treba da uradi (perspektiva korisnika). "Sistem treba..." ili "Korisnik treba... . Specificirajte akciju pokretača i uslove za to.

[klasa korisnika ili ime aktera ] treba da [uradi nešto ] [nekom objektu ] [uslovi prihvatanja, vreme odgovora, ili iskaz kvaliteta ]

**L09**

1. **Koji se sve modeli zahteva koriste za opisivanje zahteva? Koji se od navedenih modela koriste i u drugim fazama inženjerstva zahteva i s kojim ciljem?**

* Dijagrami toka podataka, Dijagrami toka procesa, UML dijagrami stanja, Mape dijaloga, Stabla odlučivanja i tabele odlučivanja, Tabele odgovora na događaje, Stablo svojstava, UML dijagrami slučlajeva korišćenja, UML dijagrami aktivnosti, Dijagrami relacija entiteta UML dijagrami klasa, UML dijagrami interakcije
* UML sekvencijalni dijagrami, UML dijagrami klasa, UML dijagrami stanja, UML dijagrami aktivnosti. Njihova primena olakšava iterativan razvoj zahteva, kao i otkrivanje grešaka.

1. **Kako izabrati pravi model za opisivanje zahteva?**

Često razvojni tim mora da koristi sve tipove modela analize. Najčešće izvrši izbor onih koji su odgovarajući za modeliranje najsloženijih i najrazičnijih delova sistema, kao i onih delova u kojima se najčešće javljaju dvosmislenosti i neizvesnosti.

1. **Objasnite dijagram toka podataka i dijagram konteksta.**

**Dijagram toka podataka** pokazuje kretanje podataka kroz sistem i procese njihove transformacije. Elementi dijagrama toka podataka su: eksterni entiteti, tokovi podataka, skladišta podataka i procesi.

**Dijagram konteksta** utvrđuje tzv. spoljne entitete van granica sistema koji su interfejssa sistemom.

1. **Objasnite UML sekvencijalni dijagram.**

**UML sekvencijalni dijagram** prikazuje aktere i entitete sistema, poruke koji oni razmenjuju, duž vertikalnih vremenskih osa na kojima su pravougaonicima obeležena trajanja aktivnosti aktera i entiteta.

1. **Objasnite dijagram aktivnosti i dijagram stanja.**

**UML dijagram aktivnosti** je sličan trakastom dijagramu, samo su aktivnosti obeležene pravougaonicima sa zaobljenim uglovima, postoje posebni elementi za spajanje i račvanje grana, a postoje i elementi za osnačavanje početka i kraja procesa.

**UML dijagram stanja** predstavlja koncizan, kompletan, i nedvosmislen opis stanja nekog objekta u sistemu, kao i uslova za promenu tih stanja

1. **Objasnite stablo odlučivanja i tabelu odlučivanja.**

**Tabela odlučivanja** opisuje šta sistem treba da radi i koje odluke da donese u određenim složenim uslovima.

**Stablo odlučivanja** dokumentuje grafičkim putem poslovna pravila koje treba ugraditi u softver.

**L10**

1. **Koji se modeli mogu koristiti za modeliranje podataka? Objasnite ih ukratko.**

ERD i UML dijagram klasa. (opis 3 i 4)

1. **Šta je rečnik podataka?**

**Rečnik podataka** je kolekcija detaljnih informacija o entitetima podataka koji se upotrebljavaju u nekoj aplikaciji. To su informacije o kompoziciji entiteta, tipovima podataka, dozvoljenim vrednostima, podaci korisni sa potvrđivanje ( validaciju) podataka, koji omogućavaju programerima da napišu tačan kod.

1. **Objasnite UML model klasa.**

UML dijagram klasa na visokom nivou apstrakcije se koristi pri utvrđivanju zahteva podataka.

Klase se reprezentuju pravougaonicima, podeljenim u više delova:

* U gornjem delu se pise naslov tj ime klase
* U sredinise pišu atributi klase
* U donjem delu se pišu operacije (metodi) klase

1. **Objasnite model entiteta i relacija?**

**Model entiteta i relacija (ERD)** povezuje entitete sa relacijama koje na krajevima daju kardinalnost, tj. broj primeraka entiteta koji se može povezati relacijom.

Pomoću ERD vi u fazi projektovanja sistema kreirate logičku ili fizičku (primenjenu) strukturu podataka baze podataka sistema.

1. **Opišite kako se sprovodi prikupljanje zahteva za izveštajima.**

Mnoge aplikacije generišu izveštaje iz baza podataka, datoteka, i drugih informacionih resursa. Izveštaji mogu da budu formi tabela, grafikona, različitih dijagrama..

Analitičar mora da utvrdi koje izveštaje korisnici očekuju od novog sistema.

* Koji je naziv izveštaja?
* Koja je svrha izveštaja ili poslovna namera izveštaja?
* Koja je tipična a koja maksimalna veličina izveštaja?

1. **U čemu se ogleda primena alata za razvoj zahteva?**

Alat za upravljanje (menadžment) zahtevima skladišti informacije u višekorisničku bazu podataka i daje robustno rešenje problema koji se javljaju pri skladištenju zahteva u dokumentima.Obavlja: Upravljanje verzijama i promenama, Skladištenje atributa zahteva, Olakšavanje analize uticaja

1. **Koje su tipične funkcionalnosti alata za upravljanje zahtevima?**
2. Prepoznavanje nedostajućih i venserijskih zahteva
3. Status zahteva za praćenje
4. Alat za kontrolu pristupa
5. Komuniciranje sa zainteresovanim stranama
6. Generisanje prilagođenih podskupova
7. **Koje su prednosti korišćenja alata u razvoju i upravljanju zahtevima?**

* Upravljanje verzijama i promenama
* Komuniciranje sa akterima projekta
* RM olakšava analizu uticaja zahteva, prepoznavanje nedostajajućih zahteva, praćenje statusa zahteva, kontrolu pristupa zahtevima
* RM vam omogućava da definišete dozvole pristupa pojedincima ili grupama korisnika

**L11**

1. **Objasnite klasifikaciju kvaliteta na spoljni i unutrašnji.**

Jedan od načina klasifikacije kvaliteta razlikuje one karakteristike koje su vidljive izvršenjem softvera (spoljni kvalitet) od onih koji nisu (unutrašnji kvalitet)

1. **Kako doći do dobro definisanih atributa kvaliteta?**

* Zahtevi za kvalitetom moraju biti merljivi kako bi se uspostavio precizan dogovor o očekivanjima između BA, kupaca i razvojnog tima.
* Umesto da izmislite svoj način dokumentovanja nepoznatih zahteva, potražite postojeći obrazac zahteva koji treba slediti.
* Možete uključiti kriterijum podobnosti

1. **Šta su atributi kvaliteta? Kako se klasifikuju?**

**Atribut kvaliteta** jekarakteristika funkcije. On ne opisuje samu funkciju, već jedan aspekt načina na koji se funkcija ponaša i koji se uslov ispunjava u tom pogledu.

Dele se na spolašnje i unutrašnje.

1. **Objasnite atribute spoljnih zahteva kvaliteta. Navedite bar pet atributa kvaliteta ovog tipa.**

**Spoljni atributi kvaliteta** opisuju karakteristike koje se primećuju prilikom izvršavanja softvera. Oni duboko utiču na korisničko iskustvo i korisnikovo opažanje kvaliteta sistema.

Dostupnost, Lakoća instaliranja, Integritet, Performanse, Pouzdanost, Robusnot, Bezbednost, Sigurnost, Upotrebljivost.

1. **Objasnite atribute unutrašnjih zahteva kvaliteta. Navedite bar pet atributa kvaliteta ovog tipa.**

**Unutrašnji kvaliteti** važniji za osoblje za razvoj i održavanje. Nisu vidljivi izvršenjem softvera.

Efikasnost, Promenljivost, Prenosivost, Ponovna upotrebljivost, Skalabilnost, Proverljivost

1. **Na šta se odnosi integritet, kao atribut kvaliteta? Dajte jedan primer ovog zahteva.**

Integritet se bavi sprečavanjem gubitka informacija i očuvanjem ispravnosti podataka.

* Sistem se štiti od neovlašćenog dodavanja, brisanja ili modifikacije podataka.

1. **Na šta se odnosi robusnost, kao atribut kvaliteta? Dajte jedan primer ovog zahteva.**

Robusnost je stepen do koga sistem i dalje pravilno funkcioniše kada se suoči sa nevažećim ulazima i oštećenjima.

* Ako uređivač teksta padne pre nego što korisnik sačuva datoteku, on će oporaviti sadržaj datoteke koja se uređuje najviše od jedne minute pre pada, a drugi put kada isti korisnik pokrene aplikaciju.

1. **Na šta se odnosi sigurnost, kao atribut kvaliteta? Dajte jedan primer ovog zahteva.**

Sigurnost definiše koliko dobro sistem štiti od neovlašćenog pristupa aplikaciji I njenim podacima.

* Sistem će zaključati korisnički račun nakon četiri uzastopna neuspešna pokušaja prijave u roku od 5 minuta.

1. **Na šta se odnosi upotrebljivost, kao atribut kvaliteta? Dajte jedan primer ovog zahteva.**

Upotrebljivost je prilagođenost korisnicima, jednostavnost upotrebe i ljudski inženjering

* Ovaj sistem je dizajniran tako da će moći da ga koriste korisnici Android/iOS-a.

1. **Na šta se odnosi efikasnost, kao atribut kvaliteta? Dajte jedan primer ovog zahteva.**

Efikasnost je merilo koliko sistem koristi kapacitet procesora, prostor na disku, memoriju ili opseg komunikacije.

* Sistem će operateru pružiti poruku upozorenja kada opterećenje upotrebe prelazi 80 posto maksimalno planiranog kapaciteta.

1. **Na šta se odnosi proverljivost, kao atribut kvaliteta? Dajte jedan primer ovog zahteva.**

**Proverljivost** se odnosi na to koliko dobro softverske komponente ili integrisani proizvod mogu da se procene kako bi se pokazalo da li sistem funkcioniše onako kako se očekuje.

* Tester mora biti u mogućnosti da konfiguriše koji se rezultati izvođenja beleže tokom testiranja.

**L12**

1. **Zašto su nam prototipovi potrebni?**

**Prototip softvera** je delimična, moguća ili preliminarna primena predloženog novog proizvoda.

Svrha prototipova:

* Koristan za otkrivanje i rešavanje nejasnoća i nepotpunosti u zahtevima, za potvrdjivanje
* Istraživanje alternativnog dizajna korisničkog interfejsa

1. **Kakvi su prototipovi za bacanje?**

Prototip koji se baca podržava brzu implementaciju i modifikaciju, na račun robusnosti, pouzdanosti, performansi i dugoročne održivosti. On se razvija da bismo odgovorili na pitanja, rešili nesigurnosti i poboljšali kvalitet zahteva.

1. **Kakvi su evolucijski prototipovi?**

Evolucijski prototip mora biti projektovan za lak rast i učestala unapređenja, tako da programeri moraju voditi računa o arhitekturi softvera i principima projektovanja.

Evolucijski prototip mora se od samog početka graditi sa robusnim kvalitetom. Zbog toga je za izradu evolucijskog prototipa potrebno duže vreme

1. **Objasnite elektronske prototipove i mogućnosti njihove primene.**

Elektronski prototipovi će vam omogućiti da lako implementirate i menjate komponente korisničkog interfejsa, bez obzira koliko je neefikasan privremeni kod koji stoji iza interfejsa.

Oni se kreću od jednostavnih alata za crtanje kao što je Microsoft PowerPoint do komercijalnih alata za izradu prototipa i stvaranja grafičkih korisničkih interfejsa.

1. **Koji su rizici pri primeni prototipova?**

Rizici: direktna primena prototipa, smetnje sa detaljima, nerealno očekivanje performansi, ulaganje preteranog rada na prototipovima.

1. **Na koji način se može obezbediti uspešna primena prototipa?**

Faktori uspeha: planirajte zadatke, dajte im svrhu, planirajte više prototipova, ne proveravajte ulazne podatke, bez prototopiva za zahteve koje razumete, koristite verodostojne podatke.

1. **Zbog čega se definišu prioriteti zahteva?**

Prioritizacija pomaže u otkrivanju konkurentskih ciljeva, rešavanju sukoba, planiranju postupnih ili inkrementalnih isporuka, kontroliranju kretanja obima i donošenju neophodnih kompromisnih odluka.

1. **Koje tehnike prioretizacije zahteva se koriste. Opišite ih ukratko.**

Tehnike prioritizacije:

* **Ulazi Ili izlazi**: grupa aktera sastavi listu zahteva i donese odluku: da li je u prioritetu ili nije. Kada je u toku implementacija tog izdanja, možete se vratiti na prethodno „out“ zahteve i ponovo proći kroz postupak za sledeće izdanje.
* **Poređenje, upoređivanje i redosled** - ređanje svih zahteva po prioritetima
* **Trostepena skala** (Zahtevi visokog prioriteta (važni), sa srednjim (važni ali nisu hitni) I niskim prioritetom (nisu ni važni))
* **MoSCoW** Must (mora) - Uslov mora biti ispunjen, Should (treba) - Zahtev je važan i treba ga uključiti u rešenje, Could (može) - To je poželjna sposobnost, ali ona koja se može odložiti, Wont (neće) - Zahtev koji se trenutno neće implementirat
* **100$** – timu se da na raspolaganje daje imaginarnih 100$ a oni ih sami raspodeljuju na zahteve na osnovu prioriteta. Više para, veći prioritet.

**L13**

1. **Opišite V model razvoja softvera.**

* Ovaj model ukazuje da su testovi prihvatanja izvedeni iz potreba korisnika, testovi sistema zasnovani su na funkcionalnim zahtevima, a testovi integracije zasnovani su na arhitekturi sistema.
* V model razvoja softvera pokazuje ispitne aktivnosti koje započinju paralelno sa odgovarajućim razvojnim aktivnostima

1. **Objasnite aktivnosti validacije i verifikacije zahteva.**

**Verifikacija** (potvrđivanje) određuje da li ste pravilno napisali zahteve. Verifikacija određuje da li proizvod neke razvojne aktivnosti ispunjava njegove zahteve.

**Validacija** (provera) zahteva procenjuje da li ste napisali ispravne zahteve. Validacijom se procjenjuje da li proizvod zadovoljava potrebe kupaca

1. **Čemu služe formalne recenzije zahteva? Zbog čega se ističe inspekcijski proces, kao jedna vrsta formalne recenzije?**

* Formalni pregled zahteva, tj. recenzija daje izveštaj koji identifikuje ispitivani materijal, recenzente i ocenu tima za pregled da li su zahtevi prihvatljivi. Glavni rezultat je rezime utvrđenih nedostataka i problema koji su se pojavili tokom pregleda.
* Nekoliko kompanija je izbeglo čak 10 sati rada za svaki sat koji su uložene u uvid u dokumentaciju o zahtevima. To uključuje mali tim učesnika koji pažljivo ispituju proizvod rada radi nedostataka i mogućnosti poboljšanja. Inspekcije služe kao kapija kvaliteta kroz koju moraju proći rezultati projekta prije nego što se primene.

1. **Ko su učesnici procesa inspekcije i koje su njihove uloge?**

U procesu inspekcije, učesniku obavljaju sledeće uloge:

* **Autor**: Autor je kreirao ili održava proizvod pod inspekcijom. Autor dokumenta sa zahtevima je obično poslovni analitičar koji je iskazao potrebe klijenta i napisao zahteve.
* **Moderator**: Moderator planira inspekciju s autorom, koordinira aktivnosti i olakšava inspekcijski sastanak. Moderator deli materijale koje treba pregledati,
* **Čitalac**: Jednom inspektoru dodeljena je uloga čitača. Tokom inspekcijskog sastanka, čitalac parafrazira zahteve i elemente modela koji se ispituju jedan po jedan.
* **Snimač**: Snimač, tj. zapisničar, koristi standardne obrasce za dokumentovanje postavljenih problema i nedostataka koji su pronađeni tokom sastanka.

1. **Kako se planira proces inspekcije? Kako se sprovodi proces inspekcije?**

* Autor i moderator planiraju inspekciju zajedno. Oni određuju ko bi trebalo da učestvuje, koje materijale inspektori treba da dobiju pre inspekcijskog sastanka, ukupno vreme sastanka potrebno za pokrivanje materijala i kada treba da bude zakazana inspekcija.
* Tokom inspekcijskog sastanka, čitač vodi druge inspektore kroz dokument, opisujući jedan uslov po jedan sopstvenim rečima. Dok inspektori postavljaju moguće nedostatke i druga pitanja, snimač ih zapisuje u spisak radnji za autora zahteve.
* Nakon što pregleda sav materijal, tim odlučuje da li će prihvatiti dokument sa zahtevima kakav je ili sa revizijom. (manjom ili većom)

1. **Šta je kontrolna lista neispravnosti i čemu služi?**

Da biste pomogli recenzentima da potraže tipične tipove grešaka u proizvodima koje pregledavaju, razvijte kontrolnu listu neispravnosti za svaku vrstu dokumenta koji zahtevaju vaši projekti.

Kontrolne liste skreću pažnju recenzenata na istorijski česte probleme sa zahtevima.

1. **Objasniti testiranje zahteva.**

Testovi bi trebalo da obuhvate i normalan protok svakog slučaja upotrebe, alternativne tokove i izuzetke, koje ste prepoznali tokom ispitivanja i analize.

Testovi bi trebalo da obuhvate korake poslovnog procesa i sve moguće puteve donošenja odluka.

U idealnom slučaju, BA će napisati funkcionalne zahteve, a tester će napisati testove sa zajedničke početne tačke: korisničkih zahteva.

**L14**

1. **Koje su glavne aktivnosti procesa upravljanja zahtevima. Objasnite ih ukratko.**

* Kontrola verzija – Definisanje identifikacione šeme verzije, praćenje individualnih verzija zahteva
* Kontrola promena – Predlaganje promena, donošenje odluka, merenje nestabilnosti zahteva
* Praćenje statusa zahteva – definisanje mogućih statusa zahteva i zapisivanje statusa svakog zahteva
* Traganje za zahtevima – definisanje linkova ka drugim zahtevima

1. **Šta predstavlja osnovna lista zahteva? Kako se formira i za šta se koristi?**

**Osnovna lista zahteva** je skup zahteva sa kojima su se akteri složili, često definišući sadržaj određenog planiranog izdanja ili iteracije razvoja. Osnovna lista zahteva može se sastojati od nekih ili svih zahteva u određenom SRS-u (bilo za ceo proizvod ili za jedno izdanje), ili određenog skupa zahteva koji se čuvaju u RM alatu ili dogovorenog skupa korisničkih priča za jednu iteraciju na agilnom projektu.

1. **Kako se sprovode naknadne promene zahteva?**

* Određivanjem zahteva sa nižim prioritetom za kasnije iteracije ili njihovim smanjenjem u potpunosti
* Pribavljanjem dodatnog osoblja ili podučavanjem nekog dela
* Proširenjem rasporeda isporuke ili dodavanjem iteracija agilnom projektu
* Žrtvovanjem kvaliteta za isporuku do prvobitnog datuma

1. **Kako se sprovodi praćenje statusa zahteva tokom vremena?**

* Broj zahteva se povećava kako se opseg izdanja dodaje i smanjuje kada se funkcionalnost ukloni sa osnovne linije. Svaki zahtev ima vrednost statusa na kraju svakog meseca.
* Krivulje prikazuju kako se projekat približava svom cilju potpune verifikacije.
* Posao je obavljen kada svi zahtevi koji su mu dodeljeni imaju status Verifikovani, izbrisani ili Odloženi.

1. **Kada i zašto se primenjuje praćenje i merenje uloženog rada na zahtevima?**

* Članovi tima stiču dragocen uvid iz saznanja kako su zapravo proveli svoje vreme, u poređenju sa tim kako su mislili da provode vreme, u poređenju sa načinom na koji su trebali da provode vreme. Praćenje rada takođe pokazuje da li tim obavlja planirane aktivnosti vezane za zahteve.
* Vreme provodeno na ove aktivnosti povezane sa zahtevima je ulaganje u uspeh projekta, a ne trošak.

1. **Kako se sprovodi upravljanje zahtevima u projektima agilnog razvoja softvera?**

* Agilni projekti se prilagođavaju promenama tako što će proizvoditi proizvod kroz seriju razvojnih iteracija i upravljati dinamičkim zaostatkom proizvoda (backlog).
* Neki agilni timovi, posebno veliki ili raspoređeni timovi, koriste agilni alat za upravljanje projektima da bi pratili status ponavljanja i priče dodeljene njemu.
* Agilni projekti obično prate njihov napredak grafikom ponavljanja .

1. **Zašto upravljati promenama u zahtevima?**

* Nekontrolisana promena zahteva je uobičajen izvor haosa projekta, promene u rasporedu, problema sa kvalitetom i stresnih situacija.
* Ako akteri na projektu ne upravljaju promenama tokom razvoja, oni zaista neće znati šta će im se dostaviti, što na kraju dovodi do nedostatka očekivanja.

1. **Objasnite problem ’’puzećih’’ promena okvira projekta.**

Problem nije u tome što se zahtevi menjaju, već što kasne promene mogu imati veliki uticaj na već obavljeni posao.

Ako se svaka predložena promena odobri, akterima se može činiti da softver nikada neće biti isporučen - i zaista, možda i neće.

1. **Kako upravljati promenama u zahtevima? Koje atribute treba da ima zahteva za promenu?**

Atributi zahteva za promenu: izvor promene, tip promene, datum podnošenja, datum promene, opis, prioritet primene, modifikator, originator,prioritet originatora, planirano izdanje. projekat, odgovor, status, naslov, verifikator.

**L15**

1. **Šta je sledljivost zahteva?**

Sledljivost zahteva omogućava utvđivanje veza sa potrebama korisnika i sa funkcijama proizvoda. Da bi zahtevi bili sledljivi, svaki od njih mora biti jedinstveno I trajno obeležen tako da možete da se na njega nedvosmisleno pozivate tokom celog projekta.

1. **Objasnite upotrebu matrice sledljivosti.**

**Matrica sledljivosti** (tabela sledljivosti) prikazuje kako je svaki funkcionalni zahtev povezan unazad za određeni slučaj upotrebe i prosleđuje se jednom ili više elemenata, projektnog rešenja, koda i ispitivanja.

Svaka matrice ćelija na preseku dve povezane komponente sadrži simbol koji označava vezu koje mogu da definišu odnose tipa jedan-na- jedan, jedan-na-mnogо mnogo-na-mnogo.

Tabela koristi strelicu da naznači da se iz određenog slučaja upotrebe izvodi određeni funkcionalni zahtev.

1. **Navedite i objasnite korake procedure za praćenje zahteva.**
2. Obučite tim i vaše rukovodstvo o značaju praćenja zahteva, vašim ciljevima ove aktivnosti..
3. Izaberite vezu odnosa koju želite da definišete iz mogućnosti.
4. Odaberite vrstu matrice sledljivosti koju želite koristiti. Izaberite mehanizam za čuvanje podataka.
5. Utvrdite delove proizvoda za koje želite da zadržite informacije o sledljivosti.
6. Odredite pojedince koji će dostaviti svaku vrstu informacija o vezama i osobu koji će koordinirati aktivnosti praćenja i upravljati podacima.
7. Izmenite svoje razvojne procedure tako da podsetite programere da ažuriraju veze nakon primene zahteva ili odobrene promene.
8. Definišite konvencije o označavanju koje ćete koristiti da svakom elementu sistema date jedinstveni identifikator kako bi se elementi mogli povezati.
9. Kako razvoj traje, svaki učesnik treba da obezbedi tražene informacije u tragovima dok završavaju sitnice.
10. Povremeno proverite informacije o tragovima. Ako je zahtev prijavljen kao implementiran i verifikovan, a podaci o njegovim tragovima su nepotpuni ili netačni, postupak praćenja zahteva ne funkcioniše kako treba.
11. **Da li je praćenje zahteva sprovodljivo? Da li je potrebno? Argumentujte odgovore.**

* Samo vaš tim može odlučiti da li praćenje zahteva - bilo da se radi samo o zahtevima za testovima ili nečem složenijim - dodaje vrednosti vašem projektu iznad njegove cene.
* Odlučite da koristite bilo kakvu poboljšanu inženjersku praksu zasnovanu kako na troškovima primene tehnike, tako i na riziku njenog korišćenja.

1. **Kako se mogu poboljšati procesi razvoja i upravljanja zahtevima?**
2. Planiranje projekta: Zahtevi služe kao temelj procesa planiranja projekata.
3. Praćenje i kontrola projekata: Praćenje projekta uključuje nadgledanje statusa projekta tako da rukovodilac projekta može da vidi da li konstruisanje softvera i verifikacija se realizuju po planu.
4. Kontrola promene: Nakon uspostavljanja niza zahteva, sve naredne promene i dopune treba da se izvrše kroz definisan proces kontrole promene.
5. **Kako drugi akteri u razvoju zahteva mogu doprineti razvoju zahteva?**

1. **Prihvatanje i testiranje sistema**. Zahtevi korisnika i funkcionalni zahtevi su bitan ulaz u testove prihvatanja.

2. **Konstruisanje**: Zahtevi su osnova za projektovanje i konstruisanje (kodiranje) softvera, te su u bliskoj vezi sa poslovima konstruisanja.

3. **Korisnička dokumentacija**: Zahtevi proizvoda su ulaz za pripremu korisničke dokumentacije. Ako su zahtevi slabo napisani ili ako kasne, to će se odraziti na kvalitet korisničke dokumentacije.