**Navedite barem 4 pravila pisanja dobre dokumentacije.**

Pisati dokumentaciju s gledišta krajnjeg korisnika; Ispravna gramatika; Izbjegavati nepotrebna ponavljanja; Izbjegavati dvosmislenost; Koristiti standarde organizacije; Bilježiti logički

**Navedite i objasnite 4 vrste dokumentacije na nekom projektu.**

1. PROJEKTNA DOKUMENTACIJA (Project documentation)

Planovi, troškovi, rokovi, norme; Projektna dokumentacija kreirana za uspješno upravljanje projektom razvoja PP

2. PROCESNA DOKUMENTACIJA (Process documentation)

Bilješke o procesu razvoja i održavanja PP-a; Relativno kratki životni ciklus; Važna u internom procesu razvoja

3. DOKUMENTACIJA O PP-u (System documentation)

Opisuje PP koji se razvija/održava i način na koji radi ali ne i kako radi; Dokumentacija o sustavu; Primjer: specifikacije zahtjeva, opis arhitekture, komentirani izvorni kod, planovi; Testiranja s uključenim testnim uzorcima, izvještaj o poznatim neispravnostima...

4. KORISNIČKA DOKUMENTACIJA (User documentation)

skup dokumenata koji opisuju PP; postupak instalacije i puštanja u rad, administracija i održavanje; opis PP-a, opis rada i načina korištenja.

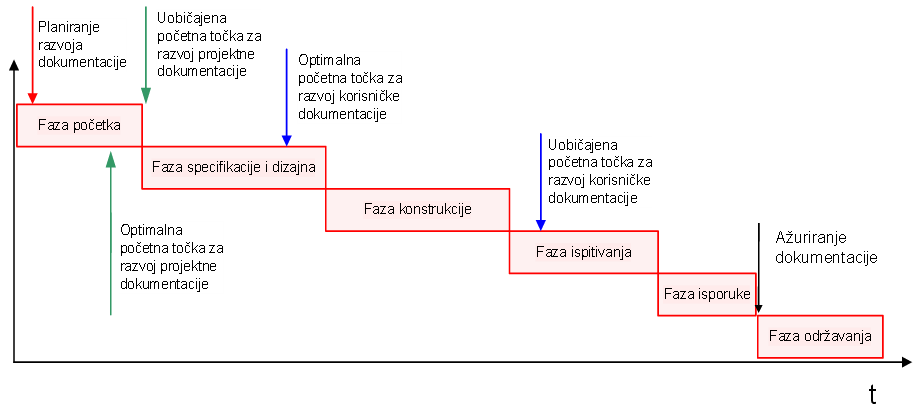
**Objasnite u kakvoj su vezi proces razvoja PP-a i proces kreiranja dokumentacije.**

Proces treba imati :

-dobro definirane spojne točke između razvoja PP-a, testnih aktivnosti i razvoja dokumentacije

-dobro definirane odgovornosti svake pojedine uloge koja sudjeluje u procesu razvoja PP i u procesu razvoja dokumentacije

**Navedite i obrazložite optimalnu i uobičajenu početnu točku za razvoj korisničke dokumentacije.**



Izrada dokumentacije mukotrpan je i težak posao, a većina ljudi ga ne voli raditi. Zato se često radi što kasnije, jer su moguće brojne promjene u skladu s kojima je potrebno mijenjati i dokumentaciju. Zato se često korisnička dokumentacija piše tek nakon što se stabiliziraju korisnički zahtjevi i broj njihovih promjena se značajno smanji.

**Norme dokumentacije:**

-procesna:definirati proces za ravoj kvalitetne dokumentacije  
-norma proizvoda:definirati kojeinfo o PP-u dokumentacija treba sadržavati, kako će izgledati  
-norma za promjene:koji alati će se koristiti za održavanje dokumentacije i osiguravanje praviila formatiranja

**Dobro strukturirana dokumentacija**

-Naslovna stranica:naslov,naziv projekta, razvojni tim, datumi, čitatelji, povjerljivost podataka  
-Podjela na poglavlja i podpoglavlja  
-Sadržaj dokumenta  
-Rječnik  
-Tablice  
-Slike

**Dokumentacija važna za održavanje je:**

Dokumentacija o programskim zahtjevima; Dokumentacija o arhitekturi; Programski kod; Dokumenti o testiranju; Dokumenti o prethodnim zahvatima održavanja

**Kategorije održavanja**

-Korektivno:pronalaženje i ispravljanje pogrešaka pronađenih u programu nakon isporuke(20%)-ono što se nije otkrilo tijekom testiranja  
 ispad:izlazno stanje definirano specifikacijom nije isto kao i stvarni rezultat izvođenja  
-Adaptivno:održavanje inicirano promjenama u programskoj okolini(novo sklopovlje, OS, nova verzija OS-a)(25%)  
-Perfektivno:zadovoljavanje novih i modificiranih potreba korisnika; promjena postojećih ili dodavanje novih funkcionalnosti(55%)-uvijek od strane korisnika; nedostatak iskustva s aplikacijom održavatelja; nestrukturirani razvoj-2.Lehmanov zakon  
-Preventivno:sprječavanje eventualnih problema prije nego se pojave(5%)

**Legacy Systems**

naslijeđeni sustavi su stari sustavi koji postaju teški za održavanje; dokumentacija i tehnologija zastarjeli; akumulacija promjena narušava modularnost originalne arhitekture

**Aktivnosti održavanja**

1.Nestrukturirano održavanje:dostupan je jedino kod PPa; dokumentacije ili nema ili je oskudna; analiza koda->konfiguracija, performanse PPa, ograničenje dizajna; teško procijeniti utjecaj promjena i nemoguće regresijsko testiranje(dio testiranja sustava kojim se provjerava je li unošenje promjena prouzročilo neispravnosti unutar programskog proizvoda)  
2.Strukturirano održavanje:PP je razvijen primjenom metodologija programskog inženjerstva; poznata konfiguracija PPa, dostupna dokumentacija; analiza dokumentacije o dizajnu->određivanje značajki PPa; planiranje promjena i primjena regresijskog testiranja

**Prednosti i nedostaci održavanja PP od strane nekog vanjskog odjela/firme**

Prednosti:

veća pažnja posvećuje se stvaranju bolje dokumentacije; uspostavlja se formalna procedura za prebacivanje programskog proizvoda iz faze razvoja u fazu održavanja; osoblje koje vrši održavanje ima mogućnost prepoznavanja prednosti i mana programskog proizvoda

Nedostaci:

prebacivanje sustava iz razvoja u održavanja može biti sporo; pojavljuju se problemi refundiranja troškova održavanja; potrebno je određeno vrijeme da organizacija/odjel za održavanje stekne znanja o novom programskom proizvodu koji je preuzet

**Prednosti i nedostaci održavanja PP od strane odjela za razvoj PP-a**

Prednosti:  
osobe koje su se bavile razvojem nekog programskog proizvoda imaju najbolje znanje o njemu; nema potrebe za detaljnom dokumentacijom; nema potrebe za uspostavom formalnog komunikacijskog sustava između održavanja i razvoja

Nedostaci:  
preveliko dotjerivanje(ono što se nije stiglo pri razvoju)

**Definirati pogodnost za održavanje i koji faktori utječu na nju**

Mjera za definiranje jednostavnosti (lakode) s kojom se neki programski proizvod može razumjeti, ispraviti, prilagoditi ili poboljšati.

****

**Tehnike planiranja proizvoda**

-Roadmapping: usmjerava se na tehničke parametre; elementi planiranja su sustav, proizvod i projekt; Dugoročno planiranje razvoja proizvoda prema njegovim segmentima tako da se osigura stalna dobra pozicija na tržištu; dobro za proizvode koji planiraju biti dugo na tržištu;  
-Metoda različitih scenarija:usmjerava se na poslovne aspekte; najčešće tri scenarija( tradicionalni, umjereni i radikalni); pokretači, potpore, smetnje i zapreke-DEBU metoda-vrednovanje vjerojatnosti  
-Metoda divergentnih inicijativa: ako je budućnost nesigurna; širi se područje djelovanja(više smjerova); potencijalno nove ideje koje kasnije mogu dovesti do većeg profita

**Intelektualno vlasništvo:**

-Industrijsko vlasništvo: patent, žigovi, industrijski dizajn  
-Autorsko pravo i srodna prava:odnosi se na intelektualne tvorevine poput: književno djelo, glazbeno djelo, fotografska djela, prijevodi, koreografska djela…



**Patent**

-isključivo pravo priznato za izum koji nudi novo rješenje nekog tehničkog problema(može se prenositi i ima ga samo vlasnik)  
-svojstva:novost, primjenjivost, inventivnost(90% su unapređenja postojećih)

**Patentni sustav**

-razmjena u kojoj izumitelj pristaje otkriti društvu pojedinosti svoga izuma u zamjenu za pravo isključivog korištenja tijekom određenog razdoblja

**Patentna prijava**

-izričita naznaka da se zahtijeva priznanje patenta

-naziv izuma, podaci o podnositelju prijave, podaci o izumitelju, opis izuma

**Opis izuma**

-područje tehnike:A do H naznaka oznake  
-tehnički problem:ne cilj već samo opis problema  
-stanje tehnike:slična rješenja(opis 3-5)  
-izlaganje biti izuma:novost rješenja u odnosu na ona opisana kod stanja tehnike  
-način ostvarivanja izuma:detaljan opis bar jednog načina   
-primjena izuma:način industrijske ili druge primjene  
-patentni zahtjevi:moraju biti sastavljeni tako da definiraju izum isključivo tehničkim karakteristikama

**Kuharica za predviđanje trendova**

1.Učenje na temelju povijesti:stanje,parametri,metoda;iskoristiti ono što je bilo uspješno  
2.Identifikacija parametara koji bi mogli utjecati na trendove:tehnologija i ljudski faktor  
3.Predviđanje na temelju faktora:tehnologije obično prate S-liniju;pretvoriti trendove tržišta u poslovnu strategiju;DEBU  
4.Periodička analiza predviđenih i stvarnih trendova i korekcije:male kompanije i startup tvrtke; prilagođavanje promjenama kod velikih tvrtki jako teško

**Dvije komponente predviđanja u ICTu**

-TRŽIŠTE:potencijalni kupci,što koriste?, ankete, istraživanja tržišta  
-TEHNOLOGIJE:u kojem se smjeru kreće tehnologija; što rade velike tvrtke a što startupi; što se sada ne može napraviti,a moglo bi se

**Nabrojati Lehmannove zakone**

1. Lehmannov zakon - KONTINUIRANA PROMJENA

Programski sustav koji se koristi u realnoj okolini mora se nužno mijenjati, inače postaje progresivno sve manje koristan u toj okolini.

2. Lehmannov zakon - POVEDANJE SLOŽENOSTI

Uvođenjem promjena u neki programski sustav, njegova struktura ima tendenciju degradacije.

čim se nešto mijenja povećava se entropija i složenost

3. Lehmannov zakon - EVOLUCIJA VELIKIH PROGRAMSKIH SUSTAVA

Veliki sustavi imaju vlastitu dinamiku evolucije koja se uspostavlja u ranim fazama razvoja (održavanjem se ne može postidi svaka željena promjena)

nije moguće implementirati svaku promjenu

**Objasni razliku između re-inženjerstva i reverznog inženjerstva**

Reverzno inženjerstvo (Reverse Engineering )

Proces analize nekog programskog sustava kako bi se identificirale komponente sustava i njihovi međuodnosi s ciljem kreiranja prikaza sustava na višoj razini apstrakcije.

Ova tehnika ne mijenja sustav i ne kreira novi sustav. Primjer je obnavljanje dokumentacije i dizajna. Iz izvršnog koda se izvlači logika funkcioniranja sustava

Programsko reinženjerstvo (Re-engineering )

Kombinacija reverznog inženjerstva i restrukturiranja programskih podataka, arhitekture i logike

razvije se dio sustava koji radi isto što i stari dio, ali ima poboljšane performanse   
Cilj: razvoj programskog proizvoda iste funkcionalnosti i veće kvalitete   
Ponovna implementacija cijelog ili dijela naslijeđenog sustava kako bi postao pogodan za održavanje; Pogodno kad neki dijelovi sustava zahtijevaju česte zahvate održavanja;Uz sustav treba se ažurirati i dokumentacija

**Tehnologija**

-koristiti tehnologiju s kojom se ima najviše iskustva  
-dva tipa rješenja:  
1.open source:besplatan, naplaćuje se podrška, velika zajednica razvijatelja  
2.vlasnička:skupo na početku(godišnje licence), dobra podrška, obično stabilnija opcija  
a)razvoj vlastitog rješenja:dugotrajno, zna se točno kako sve funkcionira,dobro za manje projekte  
b)korištenje radnih okvira:drastično ubrzava razvoj, problem ako dođe do greške

**SCRUM**

Sprint: vremenski proizvod(<mjesec dana) u kojem se isporučuje završeni Inkrement

Artefakti: posao ili vrijednosti korisni za pružanje transparentnosti i prilika za kontrolu i prilagodbu

Product Backlog: sortirana lista svega što će biti potrebno za proizvod-jedini izvor zahtjeva, funkcionalnosti i unapređenja(objašnjavanje timu vizije, ciljeva i stavaka-Vlasnik proizvoda)

Vlasnik proizvoda: odgovoran za maksimizaciju vrijednsoti proizvoda i rada razvojnog tima;jedini odgovoran za upravljanje Product Backlogom

Razvojni tim: radi konkretan posao; samoorganizirajući-nitko mu ne govori kako da Backlog pretoče u Inkrement koji samo oni mogu stvoriti(7-9 ljudi)

Scrum Master: odgovoran da se Scrum razumije i koristi; Brine se o ponašanju tima kako bi se lakše maksimizirale njihove vrijednosti

Sastanak planiranja Sprinta:posao koji će se obavljati tijekom Sprinta

Dnevni Scrum: sastanak(15min) koji služi da Razvojni tim uskladi aktivnosti

Revizija Sprinta: na kraju svakog Sprinta radi kontrole Inkrementa i prilagodbe Backloga

Retrospektiva Sprinta: na temelju prošlog Sprinta članovi tima imaju priliku samokontrole i unapređenja za sljedeći Sprint