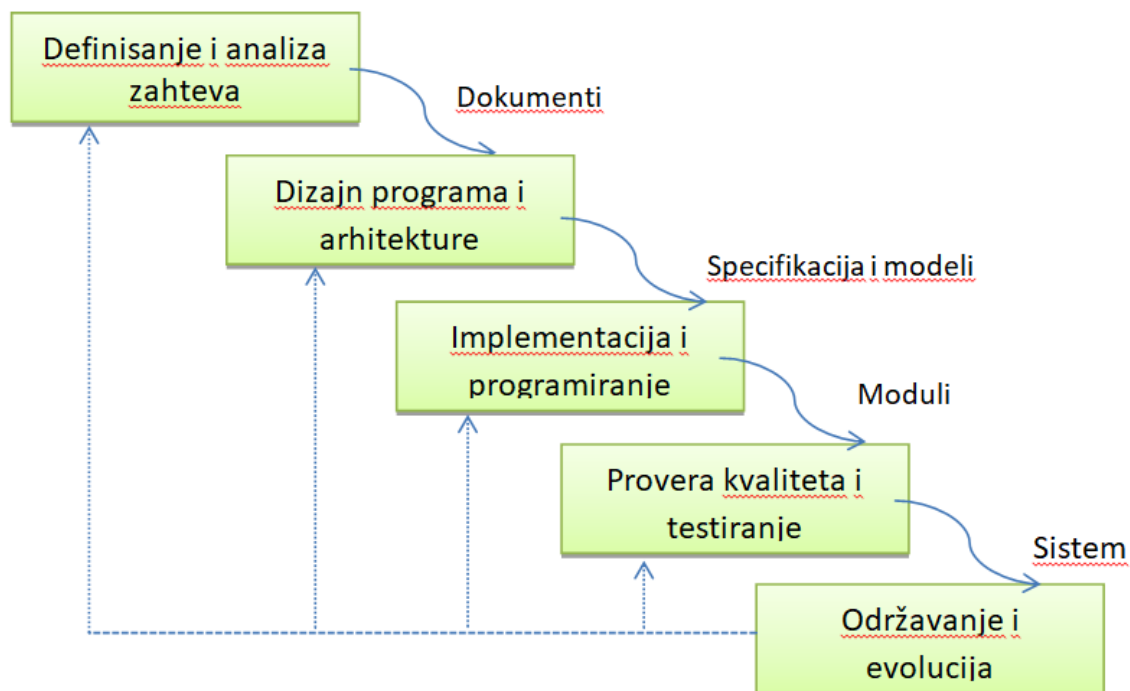


7. MODEL VODOPADA

Model vodopada (Engl. *Waterfall*) je nastao u ranim '70-tim godinama 20. veka, kao neposredna analogija s procesima iz drugih inženjerskih i tehničkih struka kako bi se omogućio efikasniji timski rad u procesu razvoja softvera.

Karakteristike: Najstariji model koji je najšire i najviše primenjivan do danas. Softverski proces je građen kao **niz vremenski odvojenih aktivnosti koje se odvijaju kroz faze sekvencijalno**, jedna iza druge. Svaka faza daje neki rezultat koji predstavlja polaznu tačku i osnovu rada u sledećoj fazi. Svaka aktivnost ima strogo definisane ulaze i izlaze. Povratak u prethodnu fazu je nepoželjan. Vreme realizacije: 6–24 meseca.



Prednosti:

- Omogućeno je detaljno planiranje celog softverskog procesa
- Podela poslova na veliki broj izvršilaca.
- Relativno laka zamena izvršilaca aktivnosti u procesu razvoja.
- Lako se može utvrditi trenutno stanje razvoja.
- Dobro je prihvaćen od strane menadžera.
- Kvalitetna dokumentacija.

Mane:

- Nema povratne sprege između faza.
- Faze je u praksi teško razdvojiti.
- Kasno otkrivanje grešaka što dovodi do nepoželjnog vraćanja na prethodne faze.
- Nemogućnost obavljanja iteracija tokom realizacije razvoja jer iste izazivaju ozbiljne probleme i konfuziju u primeni modela.
- Zbog tendencije da se zbog poštovanja rokova u određenom trenutku „zamrzne“ pojedina faza, sistem se često nastavlja razvijati u nezadovoljavajućem obliku.
- Proces je spor, pa se može dogoditi da u trenutku puštanja u rad sistem već bude neažuran.

- Visoki razvojni troškovi.
- Primenjiv najbolje samo na velike sisteme gde postoje jasni zahtevi, ali gde vreme razvoja nije bitno.

INKREMENTALNI MODEL (ITERATIVNI) u osnovi uključuje model vodopada, ali tako da svaki inkrement predstavlja pojedinačno primenjen model vodopada kojim se razvija potpuno podskup softvera, a zatim u sukcesivnim koracima kao nadgradnja izvode stalno novije i komplikovanije verzije. Svaki inkrement zadovoljava samo manji podskup korisničkih zahteva. Osnovna ideja ove metode jeste da se razvije sistem kroz ponovljene cikluse (inkremente) u manjim vremenskim ciklusima što omogućava programerima da iskoriste urađeno u prethodnim iteracijama. Stvaraju se verzije sistema sa mogućnošću redizajna delova sistema. Implementacija je modularna. Analiza podrazumeva povratne informacije od korisnika. Cilj metodologije jeste pružanje mogućnosti korisniku da reaguje na proces izrade sistema.

Karakteristike: Obezbeđuje transparentan razvoj proizvoda, sa stalno vidljivim rezultatima. Uvek raspoloživ funkcionalno upotrebljiv proizvod, koji zadovoljava određeni podskup korisničkih zahteva. Model podrazumeva testiranje novorazvijenih inkremenata proizvoda jer oni samo dodaju nove funkcionalnosti postojećem upotrebljavanom softveru i na taj način premošćava traumatske efekte uvođenja kompletno novog proizvoda odjednom.

Prednosti:

- Postojanje povratne sprege i permanentne mogućnosti ugradnje bogatog korisničkog iskustva u redefinisani proizvod na manje skup način, putem novih inkremenata odnosno novih funkcionalnosti proizvoda.
- Umanjeni rizik od neuspeha razvoja celine, jer se problemi uglavnom uočavaju u pojedinim inkrementima.
- Skromniji obim kapitalnih ulaganja u razvoj proizvoda i brži povrat investicija, λ Manji broj angažovanih osoba u procesu razvoja.

Nedostaci modela:

- Dekompozicija proizvoda na inkremente, da bi se oni mogli integrirati, nije trivijalan zadatak, kao ni sam proces integracije, a da se pri tome ne ugrozi kvalitet već postojećeg proizvoda.
- Specifikacija detaljnih korisničkih zahteva se kod svakog inkrementa izrađuje neposredno pre nego što se on razvija.
- Integracija može uvek doneti neočekivane probleme i potrebe za reorganizacijom, koja može imati posledice po efikasnost i održavanje.
- Korisnici imaju stalnu želju da menjaju svoje zahteve.

Šematski prikaz inkrementalnog, tj. iterativnog modela:

