

8. MODEL PROTOTIPA

Cilj izrade prototipa u razvoju softvera (sistema) je otklanjanje ili bar smanjivanje nesigurnosti vezanih za: procenu isplativosti projekta, funkcije i korisničke zahteve, redosled funkcija, korektan prikaz sistema, izgled korisničkog interfejsa, uticaj dizajna na korištenje softvera.

Razvoj softvera pomoću prototipa se koristi dosta dugo, još **od početka '90.-tih godina 20. veka**, sa ciljem da se smanji vreme razvoja softvera, poveća učešće korisnika u razvoju, te da se povećao adekvatniji konačan proizvod koji više odgovara potrebama.

Prototip softvera je konkretna reprezentacija jednog dela ili celog sistema. Prototip je opipljiv proizvod, a ne apstraktni prikaz kroz modele koji zahtevaju objašnjenja i razmatranje tekstualne ili pseudokodne specifikacije. Svi učesnici uključeni u razvoj prototipa i njegovo korištenje mogu, na osnovu prototipa, steći predstavu o krajnjem izgledu i funkcionalnostima softvera. Najuspešniji prototip se razvija i integriše u finalni proizvod, koji se i nakon masovne proizvodnje može još više unaprediti (nove verzije softvera).

Kad se gleda prototip kao softverski proizvod, neke osnovne karakteristike njegove su: pomaže programerima da generišu i oblikuju ideje, olakšava otkrivanje potreba korisnika, podstiče komunikaciju između dizajnera, programera i korisnika, klijenta. Omogućava ranu procenu proizvoda pošto ga je moguće testirati od strane korisnika.

Jedan od značajnih razloga pojave ove metode jeste činjenica da **korisnici budućeg sistema često ne mogu jasno i precizno iskazati svoje zahteve**, s obzirom da ne mogu predvideti kako će taj sistem uticati na njihove procese rada, radne navike, u kakvoj će interakciji biti sa drugim sistemima, te u kojoj meri može automatizovati radne procese. **Prototip je jednostavan i brzo razvijeni softver, koji oponaša budući sistem koji se lako menja i nadograđuje**, te omogućuje isprobavanje raznih ideja, koncepcija i opcija.

Dizajn programa se odvija kroz izradu prototipa i implementaciju, bez kreiranja specifikacija i modela.

Unutar svake iteracije, osnovne aktivnosti unutar faza se obavljaju simultano i **ne mogu se razdvojiti**.

Neprihvatljivi prototip se odbacuje, a iz prihvatljivog se razvija finalna verzija.

Da bi razvoj softvera pomoću prototipa imao smisla, sam **prototip se mora stvoriti na brz, jeftin i fleksibilan način**.

Prototipski razvoj softvera se **često meša sa evolucionim razvojem**. Razlike između ova dva načina su:

- Cilj evolucionog razvoja je postepeno poboljšavanje softvera u skladu sa novim zahtevima koji se otkrivaju, sve dok se taj radna verzija ne pretvori u konačan sistem. Prioritet kod implementacije zahteva imaju oni koji su najjasniji i korisnicima najpotrebniji.
- Cilj prototipskog razvoja je otkrivanje i validacija zahteva. Prioritet kod implementacije zahteva imaju baš oni koji su nejasni i koje treba istražiti. Nakon utvrđivanja zahteva prototip se odbacuje, a sistem se dalje oblikuje i implementira na konvencionalni način. Pritom treba biti oprezan u izboru delova iz prototipa, budući da su oni često loše strukturirani, loše dokumentovani, imaju slabe performanse, slabu pouzdanost, te ne zadovoljavaju sigurnosne zahteve.

Karakteristike protipskog razvoja su:

- Otkrivaju se nesporazumi između razvojnog tima i korisnika.
- Lakše se identifikuju komplikovane funkcije koje zahtevaju pažnju.
- Menadžmentu se demonstrira izvedivost i upotrebljivost sistema.
- Pokušaji da se smanji potencijalni rizik neuspeha projekta razbijanjem projekta na manje segmente i pružanje više jednostavnih promena tokom procesa razvoja.
- Korisnik je uključen u toku razvojnog procesa, što povećava verovatnoću prihvatanja konačne realizacije.
- Većina prototipova se razvija sa očekivanjem da će biti odbačeni, a moguće je u nekim slučajevima da evoluiraju u finalni sistem.
- Neophodno je osnovno razumevanje poslovnog problema.

