

11. Agilni razvoj

(Izvor: Saša Malkov, Razvoj softvera, Matematički fakultet, Beograd)

Agilne metodologije su nastale krajem devedesetih godina XX veka, kada je grupa softverskih inženjera zaključila da **dotadašnji pristupi i metodologije razvoja softvera nisu pogodne u uslovima savremenog dinamičnog IT okruženja** i da nije moguće vezivati se i postići čvrste rokove isporuke softverskih rešenja, uz zadovoljstvo klijenta. Oni su se sastali i kroz međusobnu razmenu mišljenja došli do osnovnih principa agilnih metodologija, koje su zapisali u tzv. **Agilni manifest**:

“Otkrivamo bolje načine razvoja softvera razvijajući ga i pomažući drugima u tome.

Kroz taj rad smo naučili da više vrednujemo:

Ljude i odnose među njima – od procesa i alata.

Softver koji radi – od iscrpne dokumentacije

Saradnju sa klijentima – od pregovaranja oko ugovora

Reagovanje na promene – od pridržavanja plana

Odnosno, iako cenimo vrednosti na desnoj strani, smatramo za vrednije one koje su na levoj.”

To je početak stvaranja agilnih metodologija koje bi trebalo da uvažavaju specifičnosti razvoja softvera i proces razvoja softvera učine bržim, fleksibilnijim i efikasnijim. Naziv je oblikovan 2001. godine, prilikom osnivanja Agilnog saveza (engl. Agile Alliance - AA) i formulisanja Manifesta agilnog razvoja softvera. Agilni razvoj softvera ima mnogo sličnosti sa objektno-orijentisanim (OO) metodologijama, ali ima značajno drugačiji pristup planiranju, u odnosu na klasične OO metodologije. Pri posmatranju odnosa agilnih i klasičnih OO metodologija moramo imati u vidu da većina agilnih metodologija pretpostavlja upotrebu osnovnih tehnika OO projektovanja i programiranja, ali da većina elemenata agilnih metodologija može da se primenjuje i na razvojne projekte koji nisu striktno objektno-orijentisani.

Agilni razvoj softvera nije:

- odsustvo sistematičnosti i strukturnog pristupa;
- anarhično ili svojevoljno ponašanje članova tima;
- potpuno odsustvo dokumentacije;
- selektivna primena samo nekih od principa agilnog razvoja softvera;
- najlakši metod razvoja softvera;
- univerzalno rešenje za sve probleme;
- pravi način rada za neiskusne ili nedovoljno stručne programere i
- još mnogo toga što bi neki voleli da bude.

Pretpostavka da je **softver koji radi važniji od iscrpne dokumentacije** podseća da je cilj razvoja softvera, a ne izrada dokumentacija. Mnogo je projekata imalo izvrsnu dokumentaciju, a da softver nije uspešno realizovan. Bila bi velika greška ako bi neko ovu pretpostavku razumeo kao da ne mora da pravi dokumentaciju. **Softver koji nije praćen odgovarajućom dokumentacijom je potpuna propast – kako za korisnike i vlasnike, tako i za one koji ga budu održavali.** Kada bi, kojim slučajem, programski kod bio lak za čitanje i mogao da predstavlja isključivo sredstvo komunikacije, onda niko ne bi ni izmišljao ni pravio druge vidove dokumentacije. Ali stvari ne stoje tako i zato je dobra i razumljiva dokumentacija neophodna. **Suština je u tome da se odredi prava mera količine i preciznosti dokumentacije, kao i vreme njenog pisanja.** Moramo da imamo u vidu da prerano napisana dokumentacija obično nije stabilna i da kasnije često mora da se temeljno prerađuje. Potrebno je da se pažljivo odmeri obim dokumentacije, zato što se obimna dokumentacija veoma teško sinhronizuje sa naknadnim izmenama programa.

Agilni razvoj softvera počiva na dvanaest principa, koji su posledica pretpostavki izrečenih u manifestu. Taj skup principa je zajednički za sve agilne metodologije. Svaka konkretna agilna metodologija može da uvodi nove pretpostavke i dodaje nove specifične principe. Pored toga, svaka konkretna metodologija propisuje metode i tehnike koje služe za praktično ostvarivanje tih principa.

Principi agilnog razvoja softvera su:

- Najviši prioritet je **zadovoljiti klijenta** kroz brzo i neprekidno isporučivanje softvera.
- Uvek **otvoreno prihvatati promene**, čak i u kasnim fazama razvoja.
- **Isporučivati softver koji radi**, što češće, sa intervalom od par nedelja do par meseci.
- **Poslovni ljudi - korisnici i razvijaoči softvera moraju svakodnevno da sarađuju** na projektu.
- **Zasnivati projekat na motivisanim pojedincima**. Pružiti im okruženje i potrebnu podršku i imati poverenja u njih da će obaviti posao.
- Najefikasniji način za prenošenje informacija timu i unutar tima je **razgovor licem u lice**.
- **Softver koji radi** je osnovno merilo napretka.
- Agilni razvoj softvera promovise **uzdržani razvoj**. Sponzori, razvijaoči i korisnici bi trebalo da budu u stanju da neograničeno dugo održavaju ujednačen ritam.
- **Neprekidno posvećivanje pažnje tehničkoj doteranosti i dobrom dizajnu** podiže agilnost.
- **Jednostavnost**, kao umetnost maksimiziranja količine posla koji se ne obavlja, je od suštinskog značaja.
- **Najbolje arhitekture, zahtevi i projekti potiču iz samoorganizovanih timova**.
- U redovnim intervalima **tim mora da sagledava svoj rad i mogućnosti unapređivanja** svog ponašanja i efikasnosti.

Neke od poznatijih metodologija za agilni razvoj softvera su:

- Ekstremno programiranje (engl. XP – Extreme Programming);
- Skram (engl. Scrum);
- Agilan objedinjen proces (engl. AUP – Agile Unified Process);
- Crystal Clear;
- “Suvi” razvoj (engl. Lean Development)
- Adaptivni razvoj;