**Metodologii pentru proiectarea sistemelor informatice**

**AGILE**

Agile, după cum îi spune și numele, este o abordare de management care favorizează schimbarea, mai degrabă decât planificarea metodică și funcționează după câteva principii care ne sugerează cum să abordăm proiectele pentru un maxim de eficiență și rapiditate.

În mod obișnuit, proiectele de software development pot fi abordate în două moduri distincte:

* **Cascadă** - se referă la faptul de a planifica totul în avans, pentru ca mai apoi să se construiască conform planului pentru întregul an.
* **Agile -** se referă la faptul de a planifica ce se va construi în următoarele săptămâni, apoi se adaptează de la acel punct.

**Manifestul Agile**

În 2001, 17 software developeri s-au întâlnit în Utah pentru a discuta procesele care erau diferite față de abordarea de management de tip cascadă. Împreună, ei au definit conceptul de Agile software development, inclus în Manifestul Agile. Astăzi, când spunem că metodologia este Agile, înseamnă că ea respectă principiile din Manifestul Agile.

Totuși, nu există o abordare care să fie aplicabilă în orice situație, așa că Manifestul Agile nu e creat pentru a defini cum să se organizeze proiectele, ci mai degrabă, oferă un mod de gândire creat special pentru a coordona aceste proiecte. Iar pentru că această abordare s-a dovedit extrem de eficientă în mediile de lucru dinamice, el a fost preluat și folosit dincolo de sfera IT.

**Valorile și practicile Agile**

Cea mai importantă parte din Manifestul Agile sunt cele 4 valori. Ele sunt „inima” a ceea ce înseamnă Agile. Spre exemplu, una dintre valori este „software-ul care funcționează are întâietate înaintea documentației complexe”. Asta nu înseamnă că documentația nu e importantă – înseamnă doar că dacă este să alegem între a petrece timpul scriind o documentație detaliată sau a repara un bug, ar trebui să alegem a doua variantă. Vom prezenta mai departe aceste valori.

1. **Indivizii și interacțiunile au întâietate asupra proceselor și instrumentelor**

Specialiștii preferă autonomia. Așadar, în domeniul de software development este mai important ca oamenii să rezolve problemele prin colaborare, decât prin a-i forța să adopte o procedură, doar de dragul de a satisface o politică internă.

E de la sine înțeles că fiecare companie are nevoie de procese (mai ales în cazul în care crește la o anumită dimensiune), însă e important să știm de ce o regulă este creată și când o putem încălca. Spre exemplu, când ședințele zilnice nu mai sunt folositoare, nu ar trebui forțate doar pentru că o metodologie spune că ar trebui păstrate.

Momentul în care ne putem da seama că un proces nu funcționează este atunci când oamenii nu mai colaborează eficient. Oamenii sunt motorul din spatele fiecărui proiect, iar dacă ei nu mai pot interacționa din cauza ierarhiei sau a protocoalelor complexe și lungi, atunci ei petrec mai mult timp pe protocoale și procese, decât făcându-și treaba.

1. **Software funcțional înaintea documentației complexe**

În managementul tradițional, fazele se întâmplă în secvențe, iar dacă o parte este greșită (spre exemplu cea de cerințe și documentație), fiecare dintre celelalte faze vor avea de suferit. De aceea, managementul de tip cascadă are nevoie de o documentație bogată. Însă proiectele de tip Agile susțin că lucrurile se pot schimba.

Chiar este important să petrecem o bună parte din timp făcând modificări peste modificări documentației? Ce contează cel mai mult este să avem un produs pe care utilizatorii să îl poată testa. Dacă avem de ales între a repara un bug și a face un raport despre el, a-l repara este cel mai de folos proiectului.

1. **Colaborarea cu clientul deasupra negocierilor contractuale**

Contractele creează cultura în care schimbarea nu este o opțiune. Agile creează cultura unde schimbarea este așteptată. Însă cum controlăm schimbarea? Printr-o colaborare strânsă cu clienții.

Agile presupune că avem acces nelimitat la clienții noștri și că întotdeauna putem discuta cu ei. În contextul în care developerii rezolvă probleme în mod natural, ei au totuși nevoie și de perspectiva clientului pentru a înțelege în detaliu care este problema reală.

Contractele sunt folositoare, însă au un efect neplăcut: oamenii tind să fie preocupați de a livra proiectul în timpul și banii stabiliți, mai mult decât să îndeplinească scopul de business. Mai departe, când echipa este presată de timp, rezultatul este frustrare, panică și o calitate mai mică a produsului.

Mai mult, când semnăm un contract în primele faze ale proiectelor, tindem să aproximăm greșit timpul petrecut pe proiect. Însă tot va fi nevoie să ne încadrăm în termene chiar dacă ele nu au de a face cu nevoile reale. De aceea, Agile favoriezează colaborarea cu clientul și livrarea proiectului în mai multe etape. Asta lasă timp echipei să descopere ceea ce nu știa despre proiect.

1. **Răspunsul la schimbare înaintea urmăririi unui plan strict**

Cu cât alocăm mai mult timp planificării, cu atât ne împotrivim schimbărilor. Iar scopul nu este să livrăm proiectul conform planului (în timpul și bugetul stabilit) – adevăratul scop este să satisfacem un țel de business, și dacă asta înseamnă schimbarea întregului plan, atunci asta ar trebui să facem.

Pentru că, la urma urmei, este mult mai important să construiești ceva ce ai nevoie decât să urmezi cu sfințenie un plan. Developerii pot urî câteodată atunci când codul lor are eroare sau nu funcționează, dar clienții urăsc mai mult atunci când nu primesc produsul de care au nevoie.

De aceea, Agile încurajează echipele să „taie” proiectele în bucăți mai mici, care se pot livra, astfel încât să nu fie nevoite în alte cazuri să refacă părți mari din el.

**Bibliografie**

* <https://www.brandingmagazine.ro/ce-este-agile-si-cum-ne-ajuta-intr-o-organizatie/>
* <https://www.todaysoftmag.ro/article/1140/de-ce-agile>