

Curs 4

Agile

Daniela Dănciulescu

Problemele waterfall

- ▶ Modelul de management waterfall are originile in industria manufacturiera si de constructie. Acestea sunt medii foarte bine structurate in care modificarile dupa pornirea implementarii sunt foarte costisitoare (daca nu chiar imposibile)
- ▶ Se bazeaza foarte mult pe principiul "Big Design Up Front" (BDUF) in care se considera ca timpul investit la inceput in perfectarea requirement-urilor si a design-ului va economisi timp si efort in timpul dezvoltarii
- ▶ Dar aceste considerente nu se aplica foarte bine in industria software din diferite motive
 - ▶ Este imposibil ca o faza dintr-un proiect (non-trivial) sa fie declarata 100% terminata pentru a putea trece la faza urmatoare
 - ▶ Presupunerea implicita ca orice design este fezabil pentru a deveni un produs este falsa
 - ▶ Modificarile de requirements in ciclul de dezvoltare a unui proiect software sunt foarte frecvente si normale.

Problemele planificarii

- ▶ Cele mai importante cauze care duc la eșecul unei planificări exhaustive sunt:
 - ▶ Planificarea se concentrează mai multe pe finalizarea activităților, decât pe livrarea caracteristicilor.
 - ▶ Se lucrează concomitent la mai multe taskuri
 - ▶ Dezvoltarea descrisă de plan nu este prioritizată după nevoile utilizatorilor și clientului
 - ▶ Ignorarea incertitudinii
 - ▶ Se cofundă estimările cu angajamentul de a finaliza ce s-a estimat

Agile manifesto

- ▶ Persoanele si interactiunile intre oameni sunt mai importante decat procesele sau tool-urile
- ▶ Software-ul functional este mai important decat documentatia cuprinzatoare
- ▶ Colaboarea cu clientii este mai importanta decat negocierea unui contract
- ▶ Posibilitatea de a raspunde la modificari este mai importanta decat urmarirea unui plan

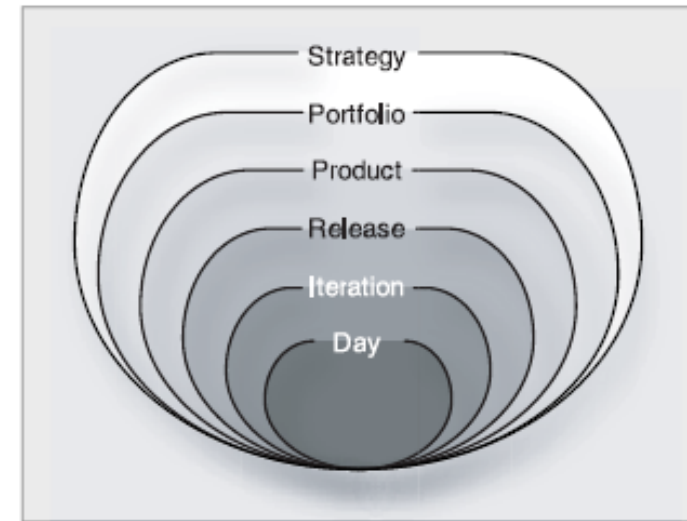
- ▶ <http://agilemanifesto.org/>
- ▶ <http://agilemanifesto.org/principles.html>

O abordare agilă

- ▶ Agile este un mindset
- ▶ Principii agile:
 - ▶ Se lucreaza ca o echipa
 - ▶ Se lucreaza in iteratii scurte
 - ▶ Telul este de a livra ceva functional in fiecare iteratie
 - ▶ Focusul sa fie pe prioritatile de business
 - ▶ Sa inspectam si sa ne adaptam
 - ▶ Un plan reprezintă o viziune asupra viitorului si nu un contract
 - ▶ Experiența și cunoștințele sunt factori importanți care sunt luati in considerare
- ▶ O abordare agilă este ca un joc în care toți jucătorii câștigă sau toți jucătorii pierd la sfârșitul jocului

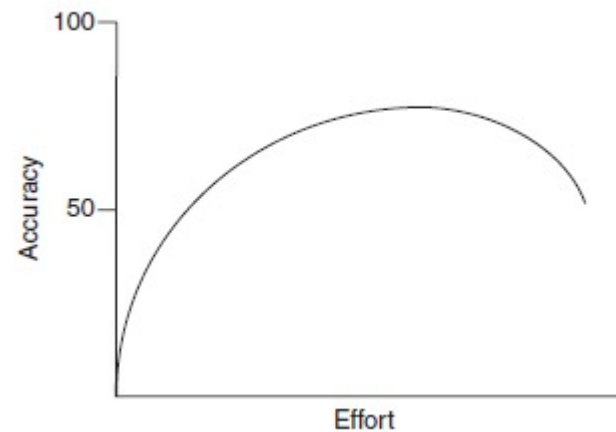
O abordare agilă asupra planificării

- ▶ Este necesară o abordare progresivă a planului
- ▶ În definirea unui plan agil, se observă că:
 - ▶ Suntem focusați mai mult pe planificare, decât pe plan
 - ▶ Se încurajează schimbările
 - ▶ Rezultatele din plan sunt ușor de modificat
 - ▶ Planul agil este răspândit pe toată durata proiectului
- ▶ O echipă agile realizează planificări pe trei orizonturi:
 - ▶ Release
 - ▶ Iterație
 - ▶ Ziua curentă



O abordare agila asupra estimarilor

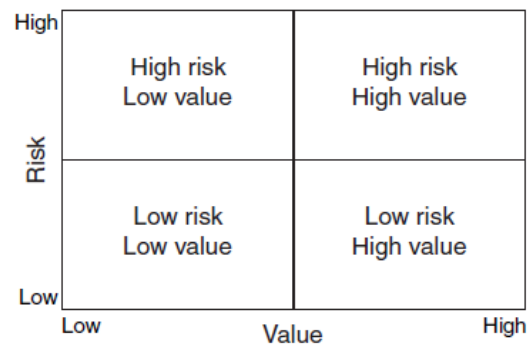
- ▶ Relația dintre acuratețea estimării și timpul petrecut pentru realizarea estimării este prezentată în figura următoare.



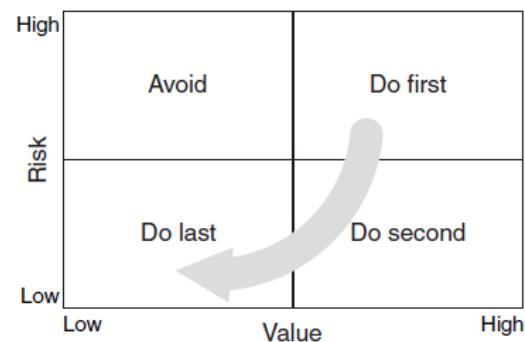
- ▶ Efortul aditional de imbunatatire a unei estimari aduce foarte putina valoare odata ce s-a trecut de un punct critic

O abordare agila asupra prioritizarii

- Relația dintre risc și valoare este prezentată în diagramele următoare:



Cele 4 cadrane ale relației risc - valoare



Combinarea risc - valoare în prioritizarea funcționalităților

O abordare agila asupra prioritizarii

- ▶ Modelul prin care se realizează prioritizarea ținând cont de satisfacția clientului, se numește Modelul Kano
 - ▶ Must-be Quality. Requirement-uri pe care clienții le așteaptă și care sunt considerate general obligatorii (taken for granted)
 - ▶ One-dimensional Quality. Aceste atribute satisfac clientul când sunt îndeplinite, și scad satisfacția când nu sunt îndeplinite
 - ▶ Attractive Quality. Aceste atribute asigură satisfacție când sunt îndeplinite, dar nu produc insatisfacție dacă nu sunt realizate
 - ▶ Indifferent Quality. Aceste atribute nu au nici un impact asupra clienților (pot ține de procesul tehnologic)
 - ▶ Reverse Quality. Aceste atribute pot avea un grad mare de realizare tehnică, dar care produc insatisfacție pentru un anumit segment de clienți

Waterfall vs agile

Waterfall



Agile

