



UNIUNEA EUROPEANĂ



GUVERNUL ROMÂNIEI



Instrumente Structurale
2007-2013



Platformă de e-learning și curriculum e-content pentru învățământul superior tehnic

Managementul Proiectelor Software

Metode de dezvoltare

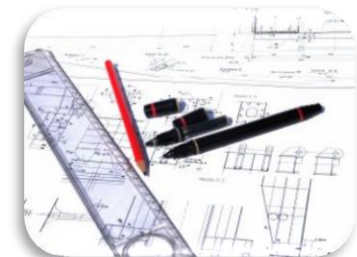
Metode structurate (inclusiv metodele OO)

- O mulțime de pași și reguli care
 - generează prodiagramele de flux, de date, etc.
 - (fiecare asemenea produs este documentat atent)
- De cele mai multe ori sunt mult mai consumatoare de timp decât metodele intuitive, acest lucru ducând și la o creștere a costurilor proiectului
- Avantaje:
 - Sistemul este mult mai puțin sensibil la erori
 - Mult mai ușor de întreținut la sfârșit
- Recomandate în cazul proiectelor mari, care implică mulți dezvoltatori și mulți utilizatori



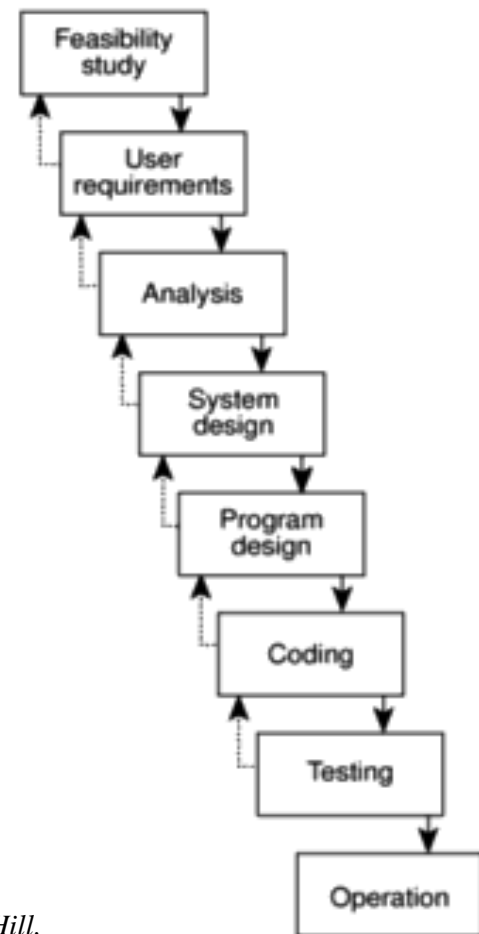
Metode de dezvoltare rapidă

- Workshop-uri de trei-cinci zile în care dezvoltatorii lucrează intensiv împreună cu clienții pentru a identifica și pentru a cădea de acord asupra cerințelor business ale proiectului
- Time-box – întinderea fiecărei etape a proiectului este constrânsă de un deadline predeterminat, foarte scurt și inflexibil
- Cerințele ce nu pot fi satisfăcute într-un anumit time-box, sunt mutate în etapele următoare



Modelul în cascadă

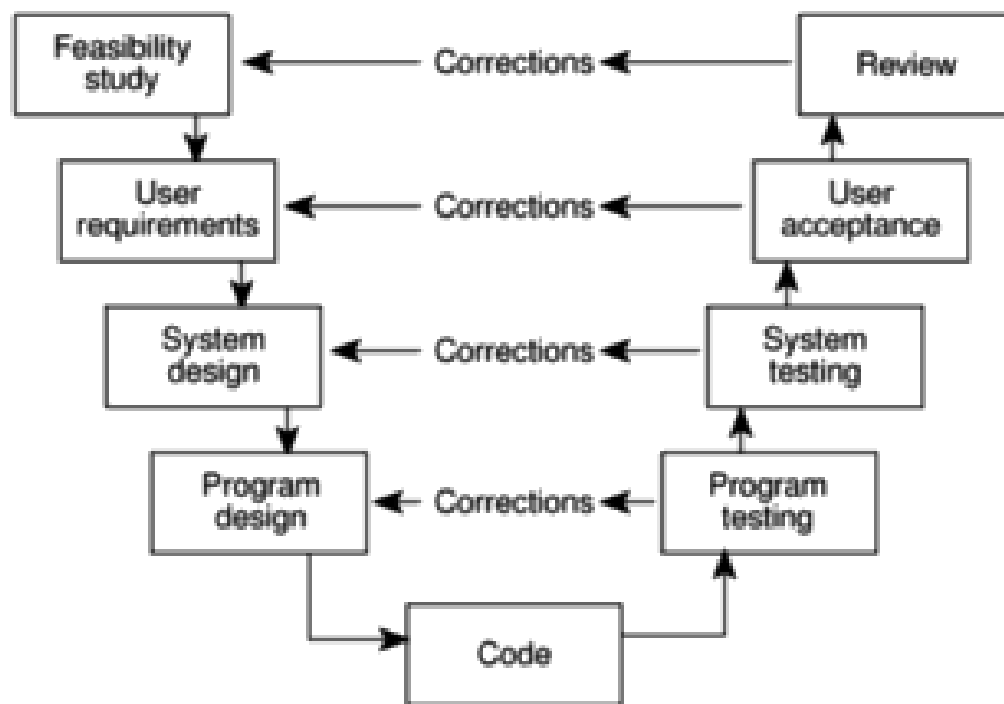
- Considerat metoda “clasică” de dezvoltare a sistemelor
- Permite controlul eficient al proiectelor și estimarea foarte exactă a timpilor de execuție



Sursa: Bob Hughes, Mike Cotterell, Software Project Management - Second Edition (Chapter 2), McGraw-Hill, 1999



Modelul procesului în V

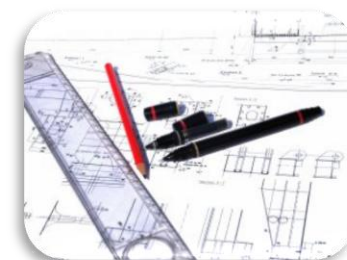


Sursa: Bob Hughes, Mike Cotterell, *Software Project Management - Second Edition (Chapter 2)*, McGraw-Hill, 1999



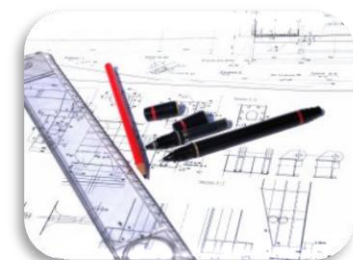
Modelul procesului în V (continuare)

- Extinde activitățile de testare din modelul în cascadă
- Fiecare pas are un proces de validare corespunzător.
- În cazul în care apar defecte, procesul de validare întoarce dezvoltarea la pasul de dezvoltare corespunzător; toți pașii următori trebuie să fie refăcuți.
- Ideal, acest tip de feed-back ar trebui să apară numai în cazul unei discrepanțe mari între specificațiile unei anumite activități și ceea ce a fost de fapt implementat

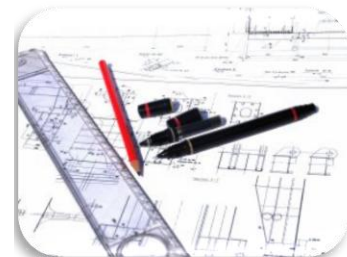
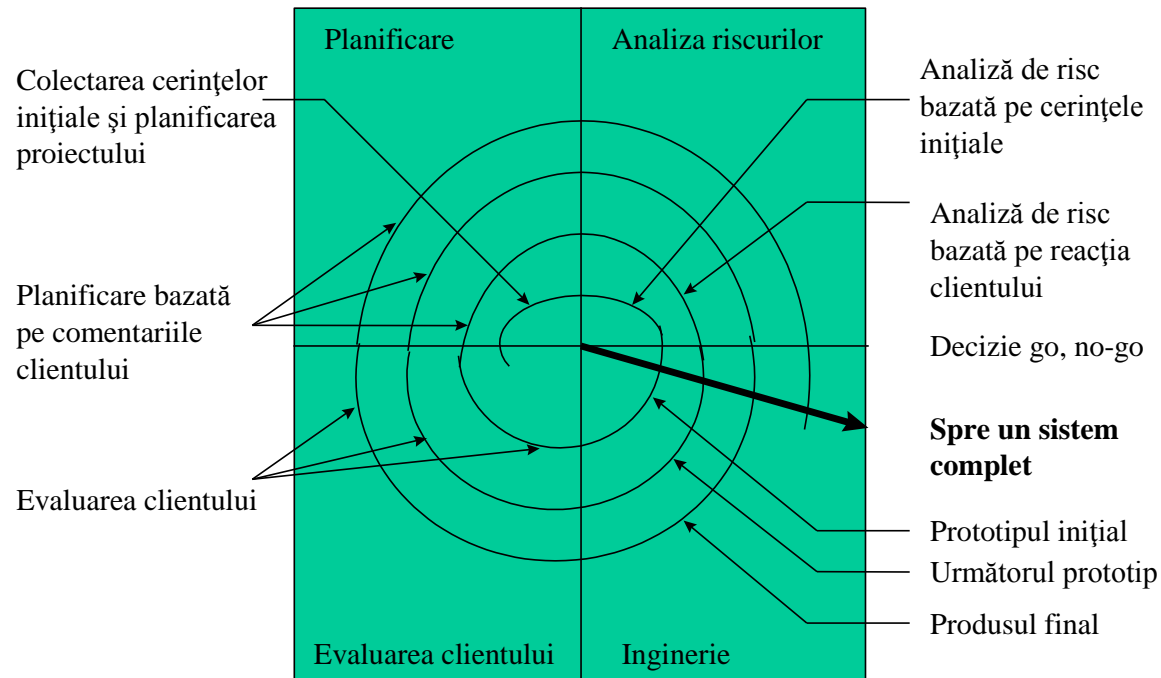


Modelul în spirală

- Poate fi considerat ca o alta vedere a modelului în cascadă
- Un mai mare grad de detaliu este necesar la fiecare etapă a proiectului, acest fapt justificând și un mai mare grad de încredere în probabilitate de succes a proiectului
- Acest model poate fi văzut ca o spirala în care sistemul dezvoltat este văzut din ce în ce mai în detaliu la fiecare rotație
- Un proces de evaluare a etapei precedente are loc înaintea începerii unei noi iterații



Modelul în spirală (continuare)

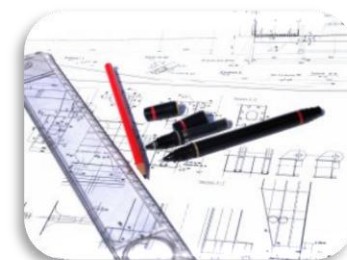


Modelul în spirală (continuare)

- Dezvoltare iterativă
 - Bazată pe ideea de ciclu de producție
 - Procesul de dezvoltare cuprinde mai multe cicluri de producție
- Dezvoltare incrementală
 - Fiecare ciclu are o complexitate (un nivel de detaliere) mai mare decât precedentul
- Modelul spirală (Boehm, 1986)

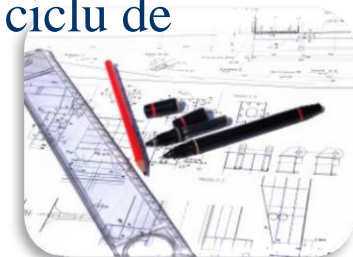
B. W. Boehm, A spiral model of software development and enhancement, ACM Sigsoft, Software Engineering Notes, 11(1986), No. 4, 14-23.

 - Combină trăsăturile
 - Ciclului clasic de viață
 - Prototipizării
 - Element nou: ANALIZA RISCURILOR



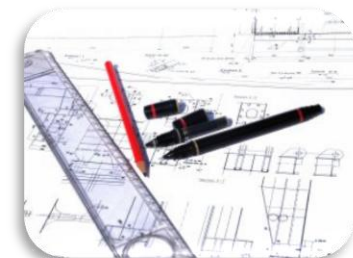
Modelul în spirală (continuare)

- Activitățile unui ciclu de producție
 - (1) Planificare
 - Stabilirea obiectivelor, alternativelor de rezolvare și a restricțiilor pentru ciclul curent
 - (2) Analiza riscurilor
 - Analizează alternativele de rezolvare și restricțiile din (1)
 - Identifică factorii de risc
 - Decizia GO/NO GO (continuă/renunță)
 - Dacă toate cerințele clientului sunt îndeplinite, dezvoltarea este încheiată
 - Dacă riscurile sunt prea mari se oprește dezvoltarea
 - Dacă riscurile se pot ține sub control, se începe un nou ciclu de producție



Modelul în spirală (continuare)

- (3) Inginerie - începutul unui ciclu nou
 - Dezvoltarea produsului pe următorul nivel de detaliere
 - Se pot folosi
 - Modelul clasic
 - Prototipizarea - pentru clarificarea unor cerințe
- (4) Evaluarea clientului



Modelul în spirală (continuare)

- Avantaje
 - Abordare evoluționistă
 - Ajută la înțelegerea riscurilor și la identificarea modalităților de ținere sub control a acestora
 - Prototipizarea este folosită ca mecanism de reducere a riscurilor
 - Ciclul clasic de viață este încorporat într-un cadru iterativ, care reflectă mai bine lumea reală
- Dezavantaje
 - Analiza riscurilor este o activitatea critică
 - Atenție acordată riscurilor tehnice în toate etapele proiectului
 - Dacă un risc major nu este descoperit