

# Interactiunea Proiectarea interactiunii

HCI – curs 4

# Din cursurile anterioare

- Oameni
  - Capacitati, limitari, emotii, erori...
  - Canale de I/O
  - Memorie
  - Procesare
- Calculatoare
  - Capacitati, limitari
  - Canale de I/O
  - Memorie
  - Procesare

**Interactiunea = comunicarea intre cele doua componente**

# Interactiunea

- Studiul modului in care oamenii folosesc calculatoarele pentru a realiza, simplifica sarcini sau ca asistenti in realizarea unor sarcini
- Pentru o simplificare a intelegerii interactiunii – modele
- Modelele vor ajuta la intelegerea interactiunii si vor oferi un cadru de comparatie pentru diferite stiluri de interactiune:

Modelul lui Norman

Interaction framework (Abowd & Beal)

# Modelul lui Norman

- Ciclul executie-evaluare
- Componente: scopuri si actiuni utilizator
- Utilizatorul formeaza un plan de actiune care este realizat impreuna cu interfata
- Cand planul e executat partial sau integral, utilizatorul observa interfata pentru a evalua rezultatul executiei planului si pentru a stabili actiunile viitoare
- 2 mari etape in realizarea unei sarcini:
  - executie
  - evaluare

# Modelul lui Normann

- 7 pasi:

1. Stabilirea unui scop
  2. Formularea intentiei (planul)
  3. Specificarea secventei de actiuni
  4. Executia actiunilor
- } **Executie**
- 
5. Perceperea starii sistemului
  6. Interpretarea starii sistemului
  7. Evaluarea starii sistemului in raport cu scopurile
- } **Evaluare**

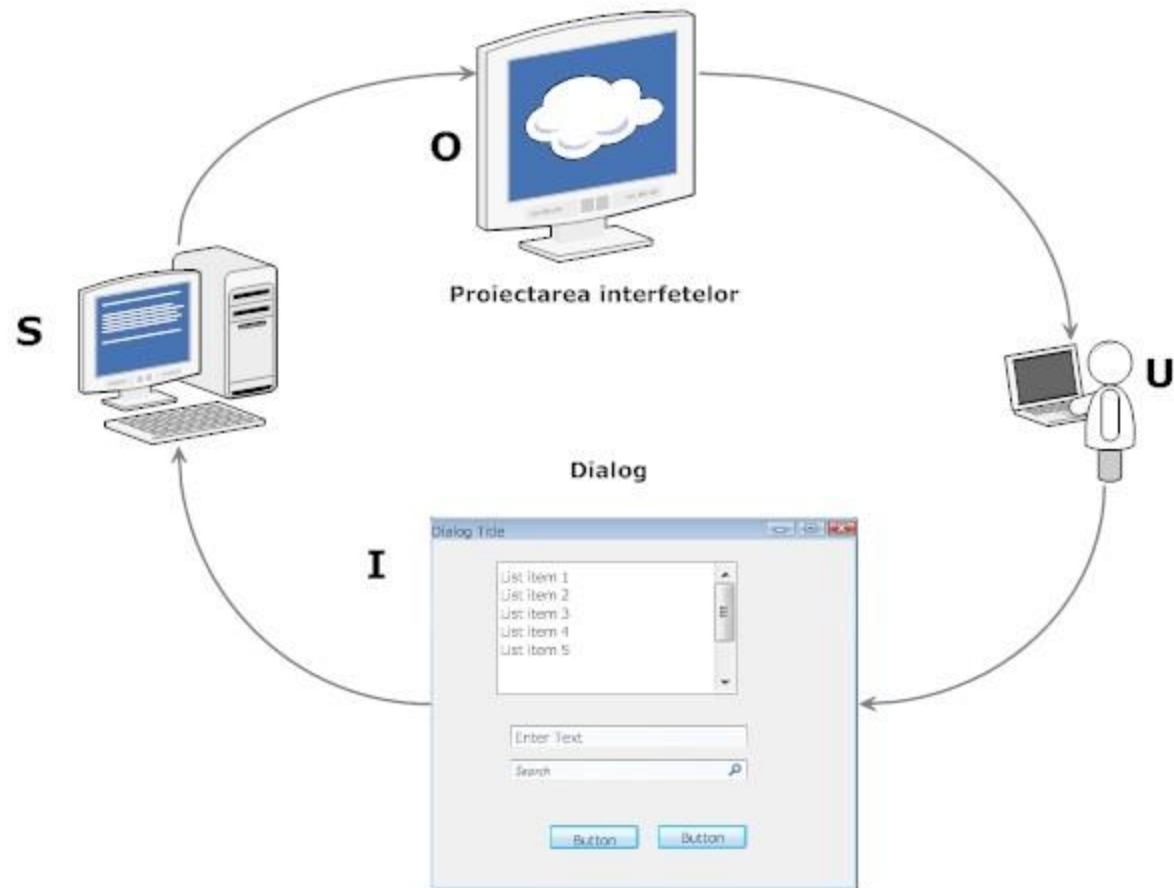
# Exemplu Modelul lui Norman

- Citim, se lasa seara
  1. Avem nevoie de mai multa lumina
  2. Intentia de a aprinde lumina
  3. Actiuni necesare: mergem la intrerupator pentru a aprinde lumina/sau rugam pe cineva sa aprinda lumina
  4. Aprindem lumina
  5. Percepem schimbarea starii (daca nu e ars becul, etc)
  6. Interpretare (e destula lumina?)
  7. Daca scopul e indeplinit – ciclul se considera complet, altfel se creeaza noi scopuri si ciclul reincepe

# Modelul lui Norman

- Poate fi folosit pentru a explica de ce unele interfete provoaca probleme – doua tipuri de probleme:
  - “prapastia in executie” - utilizatorul are planul de executie, dar interfata nu ii permite sa execute actiunile necesare
  - “prapastia in evaluare” – diferenta dintre prezentarea fizica a starii sistemului si asteptarile utilizatorului – cu cat efortul utilizatorului de a interpreta interfata e mai mare, cu atat e mai putin eficienta interactiunea
- Modelul e simplist – se raporteaza doar la interfata utilizator, fara a se raporta si la comunicarea dintre utilizator si sistem
- Abowd & Beal – Interaction framework

# Interaction framework





# Interaction framework

- 4 componente:
  - Utilizatorul (U)
  - Sistemul (S)
  - Intrările (I)
  - Iesirile (O)

**! Fiecare foloseste un alt limbaj**
- Apare necesitatea translatarei (articulării) unei cereri in mai multe limbaje (4 translatari)

# Interaction framework

- Utilizatorul are un scop de realizat – trebuie sa-l formuleze in limbajul interfetei (1)
- Mesajul utilizatorului este reformulat in limbajul aplicatiei (operatii care se executa – proceduri/functii) (2)
- Noua stare a sistemului trebuie comunicata utilizatorului – reformulata in termeni ai iesirilor (3)
- Iesirile sunt observate de catre utilizator si comparate cu scopul propus (4)
- Probleme pot sa apara la fiecare din aceste reformulari...

# Interaction framework

- Exemplu mapare proasta U-I
  - Intrerupatoarele, interfoane
  - Comportamentul uzual – incercare-eroare repetate
  - Problema: dificultatea de a articula un scop din limbajul natural intr-o linie de intrerupatoare (limbajul interfetei)
- Exemplu articulare I-S
  - E important ca limbajul interfetei sa permita la fel de multe operatii ca si manipularea directa a sistemului - telecomanda unor sisteme audio nu are optiunea “off”
- Stilul de interactiune influenteaza prezenta/absenta acestor probleme

# Interaction framework

- Exemplu articulare S-O
  - Rezultatul executie actiunilor trebuie exprimat in limitele interfetei (dispozitivului de iesire)
- Exemplu articulare O-U
  - Utilizatorul trebuie sa interpreteze stimulii (vizuali, auditivi, etc) transmisi de componenta O si sa evalueze starea sistemului in raport cu scopul
  - E dificil sa evaluam rezultatul unei copieri/mutari de fisiere la linia de comanda sau sa dezvoltam o pagina web fara sa avem un browser unde sa previzualizam rezultatul muncii noastre

# Stiluri de interactiune

- Interfete la linia de comanda (text)
- Meniuri
- Limbaj natural
- Intrebare/raspuns si interogare
- Completare de forme si foi de calcul
- WIMP (Windows, Icons, Menus, Pointers)
- Indica si actioneaza (point and click)
- Interfete tridimensionale

# Interfete la linia de comanda

- Primul stil de interfete interactive
- Permite transmiterea de comenzi folosind taste functionale, caractere, abrevieri sau comenzi explicite
- In anumite sisteme unica modalitate de comunicare este de acest tip (telnet ssh)
- Avantaje:
  - Oferă acces direct la functionalitatile sistemului
  - Permit combinarea comenzilor pentru a gestiona informatiile
  - Sunt flexibile prin intermediul parametrilor
  - Pot fi aplicate asupra unui numar mare de obiecte – utile in cazul sarcinilor repetitive
- Dezavantaje:
  - Dificultati in utilizare si invatare (comenzile trebuie invatate pentru ca nu exista indicii in interfata) > indicate pentru experti
- Solutie:
  - Folosirea de comenzi/abrevieri consistente si semnificative semantic
  - Comenzile ar trebui sa foloseasca comenzi din vocabularul utilizatorului, nu al proiectantului

# Meniurile

- Optiunile valabile sunt disponibile pe ecran si pot fi selectate folosind mouse-ul, taste numerice sau alfabetice
- Nu solicita utilizatorul, bazandu-se pe recunoastere, nu amintire
- Optiunile trebuie grupate logic pentru a putea fi recunoscute
- Meniurile pot fi organizate ierarhic, iar o optiune ar putea sa nu fie vizibila la primul nivel al ierarhiei – gruparea si denumirea optiunilor sunt singurul indiciu pentru utilizator
- Pot fi meniuri pur text sau pot avea o interfata grafica

# Interfete in limbaj natural

- Calculatorul ar trebui sa fie capabil sa raspunda la comenzi scrise sau rostite in limbaj natural
- Problema: ambiguitatea limbajului natural
  - La nivel sintactic
  - La nivel semantic
- Oamenii rezolva astfel de probleme bazandu-se pe context sau pe cunostintele generale despre lume > greu de furnizat calculatorului
- Pot fi construite interfete care sa recunoasca submultimi restranse ale limbajului (pentru un domeniu limitat) – dezambiguarea poate fi realizata de catre calculator



# Intrebare/raspuns, interogari

- Intrebare/raspuns – mecanism simplu de a furniza intrari sistemului intr-un domeniu specific
- Utilizatorului ii sunt adresate o serie de intrebari la care poate raspunde cu da/nu, optiuni multiple sau coduri
- Ex: chestionarele web
- Limbajele de interogare – folosite pentru a extrage informatii din bazele de date
- Folosesc formulari similare limbajului natural, dar cu o sintaxa specifica + cunostinte despre structura bazei de date

# Completare de forme

- Completarea de forme – folosite pentru introducerea de date, dar si pentru aplicatii folosite in regasirea de date
- Utilizatorului ii este prezentata o forma in care introduce valori – posibilitate de modificare a valorilor introduse si de nespecificare a altor valori

The screenshot displays the 'Mersul trenurilor' website interface. At the top, there is a banner for Peugeot with the text 'La PEUGEOT ai acces GARANTAT la finanțare indiferent de nivelul tău de venit.' Below the banner, the website's logo 'Mersul trenurilor' is visible on the left. The main content area features a search form with the following fields and options:

- Ruta** (selected tab)
- Statie** (tab)
- Tren** (tab)
- Plecare:** (text input field)
- Sosire:** (text input field)
- Data:** (dropdown menu showing '17 Martie 2009')
- Cauta ruta** (blue button)
- Via (optional):** (text input field)
- ☐ Doar trenuri cu legaturi
- [Filtre](#)

To the right of the search form, there is a list of destinations organized in three columns:

<a href="#">Aeroport Otopeni</a>	<a href="#">Bacau</a>	<a href="#">Baia Mare</a>
<a href="#">Brasov</a>	<a href="#">Bucuresti Nord</a>	<a href="#">Budapesta</a>
<a href="#">Cluj Napoca</a>	<a href="#">Constanta</a>	<a href="#">Craiova</a>
<a href="#">Deva</a>	<a href="#">Galati</a>	<a href="#">Iasi</a>
<a href="#">Istanbul</a>	<a href="#">Kiev</a>	<a href="#">Moscova</a>
<a href="#">Oradea</a>	<a href="#">Ploiesti</a>	<a href="#">Praga</a>
<a href="#">Satu Mare</a>	<a href="#">Sibiu</a>	<a href="#">Suceava</a>
<a href="#">Timisoara</a>	<a href="#">Venetia</a>	<a href="#">Viena</a>

At the bottom right, there is a link to 'Sugestii MersulTrenurilor.ro'.

# Foi de calcul

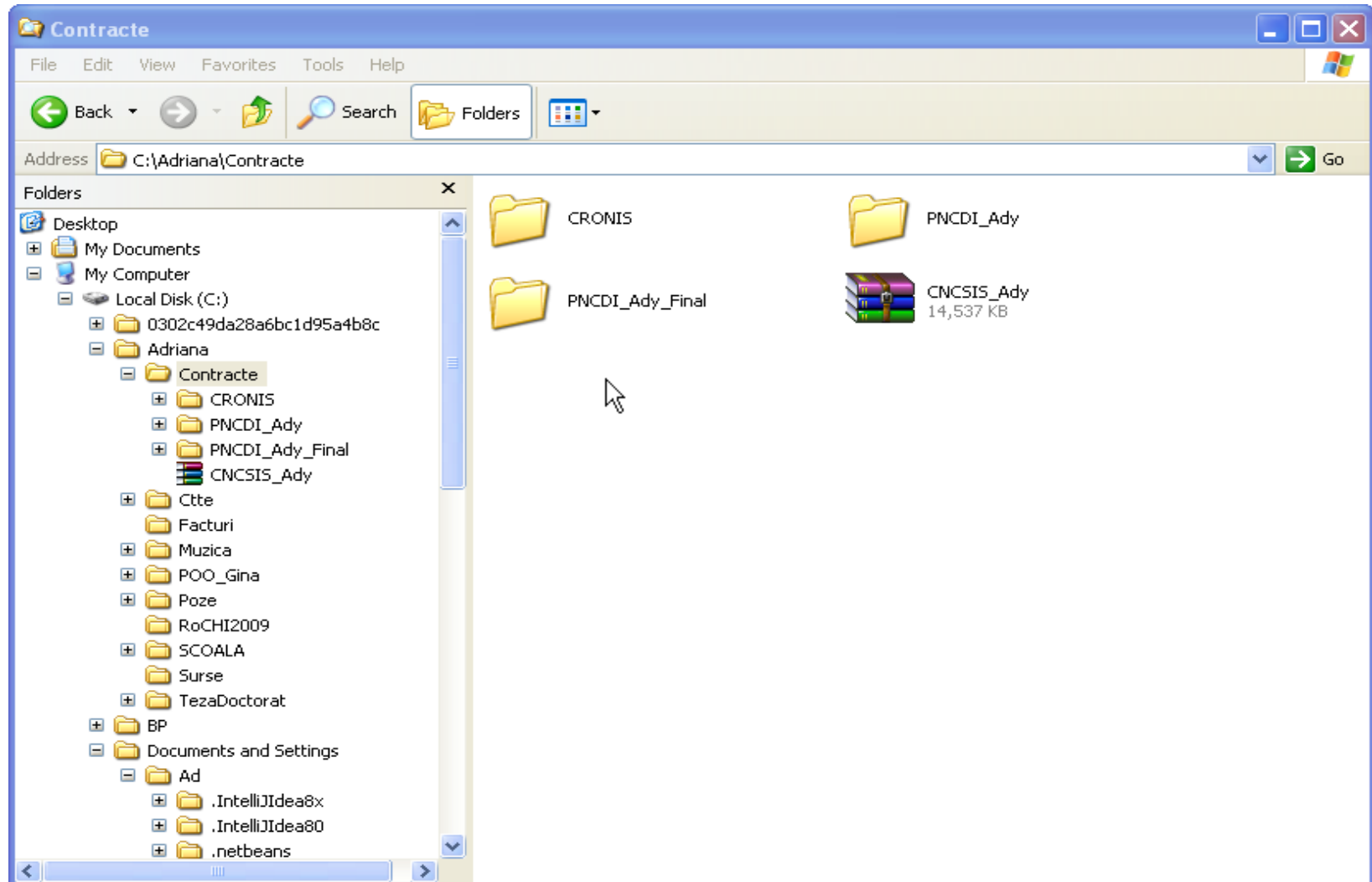
- Variatiune sofisticata a completarii de forme
- Cuprinde un grid de celule care contin valori sau formule
- Utilizatorul are libertatea de a manipula valorile, iar modificarile sunt prezentate instantaneu> interfata flexibila si naturala

Nr. Crt.	Lab(20%)	Semin	Practic(15)	Scris(5)	Media	Final
1	5.00	2.00	4.10	a	#VALUE!	
2	8.50	10.00	9.50	6.35	7.80	8(opt)
3	1.00	5.00	0.00		0.95	
4	7.80	10.00	6.10	3.85	5.90	5(cinci)
5	1.00	6.00	0.00		1.10	
6	7.00	10.00	10.00	3.90	6.35	7(Sapte)
7	9.50	7.00	5.63	5.45	6.52	7(Sapte)
8	1.00	0.00	0.00		0.20	
9	9.00	8.00	8.50	6.35	7.45	8(opt)
10	9.00	5.00	9.00	4.95	6.38	7(Sapte)
11	5.00	9.00	6.25	5.70	6.14	6(sase)

# Interfete WIMP

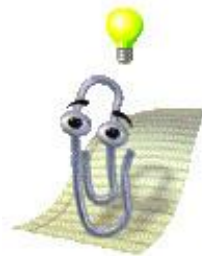
- Windows, icons, menus and pointers/ windows, icons, mice and pull-down menus - ex:Microsoft Windows, MacOS, sisteme X-Windows
- Avantaje:
  - Manipulare directa
  - **Actiunile fizice inlocuiesc folosirea comenzilor cu sintaxa complexa**
  - Actiuni reversibile rapide cu feedback imediat asupra obiectelor de interes
  - Novicii invata repede modul de operare
  - Erorile rare – anxietate scazuta , sentiment al controlului, incredere sporita
- Dezavantaje:
  - Nu toate sarcinile pot fi descrise prin obiecte si nu toate actiunile pot fi realizate direct
  - Unele persoane interpreteaza interactiunea in mod eronat
  - Nu toate sarcinile sunt potrivite pt manipularea directa (spell-checking)

# WIMP



# Interfete bazate pe agenti

- In lumea reala agentii sunt persoane care lucreaza pentru alte persoane (agent imobiliar, agent de vanzari, etc)
- Agentii soft actioneaza in sprijinul utilizatorului
- Ex: agenti pentru filtrarea emailurilor, agenti care cauta informatii pe internet
- Agentii realizeaza sarcini repetitive, monitorizeaza si raspund la evenimente in absenta utilizatorului sau invata din actiunile utilizatorului
- Ex: Office Assistant – inteligenta intr-un domeniu bine definit – are o existenta fizica



# Interfete point and click

- sistemele multimedia si browserele web - aproape toate operatiile necesita un click
- Sunt strans legate de interfetele WIMP
- Interfetele point and click nu impun neaparat utilizarea mouseului, fiind folosite si in interfetele care folosesc touchscreen-uri
- Popular in paginile web



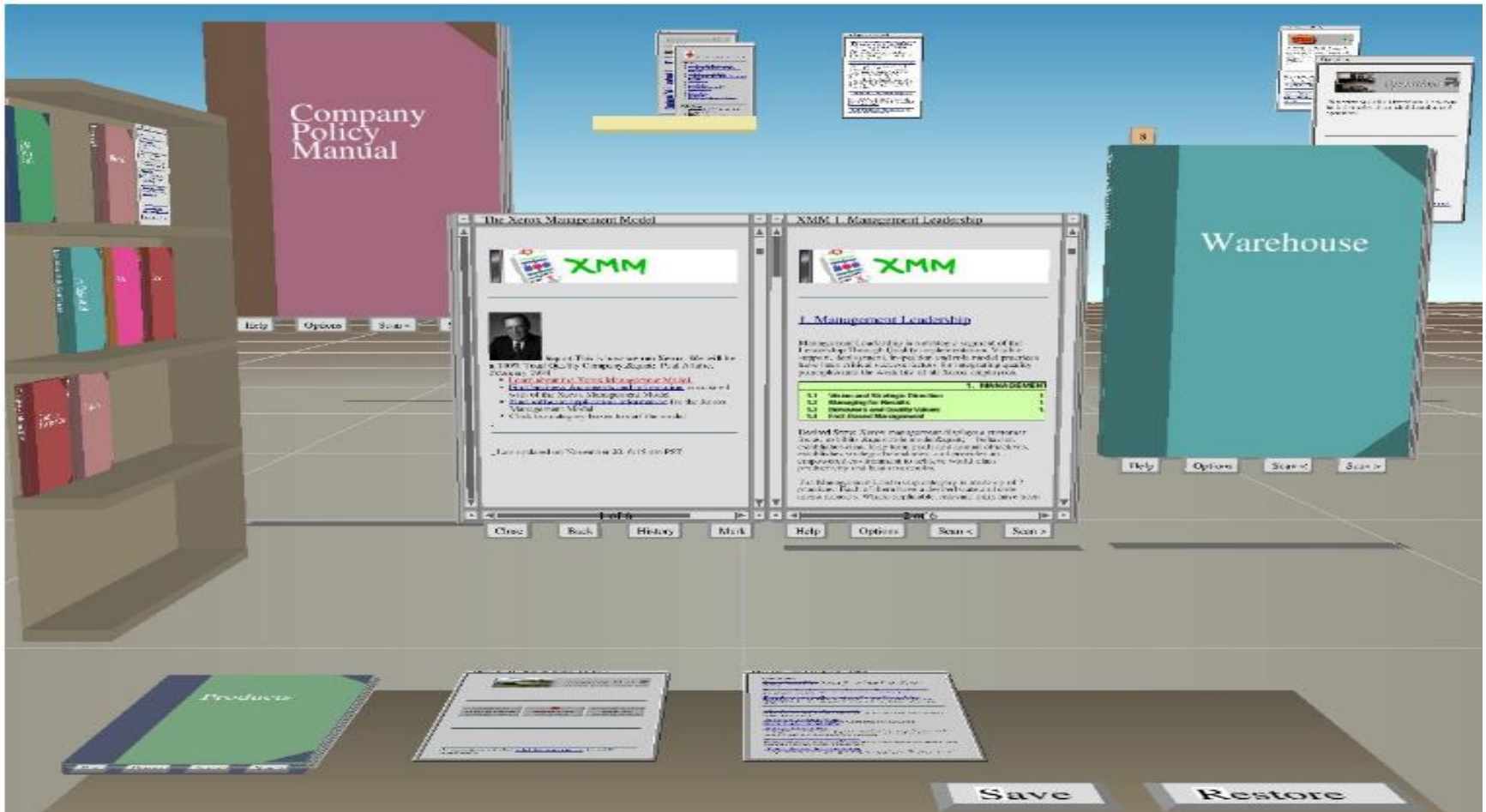
# Interfete tridimensionale



<http://ecodazoo.com/>



# Interfete 3D



<http://www.infovis.net/printMag.php?num=154&lang=2>

# Fundamentele proiectarii interactiunii

- E important sa intelegem ca introducerea unui produs nou (fizic/program) va schimba modul de realizare a activitatilor
- Proiectarea interactiunii nu are legatura doar cu produsul creat ci si cu intelegerea si alegerea modului in care va afecta modul de lucru al oamenilor
- Exemplu: introducerea intr-un birou a unui capsator electric
- Ce este proiectarea?
  - Indepinirea de scopuri respectand constrangeri

# Proiectarea

- Scopuri:
  - Care este scopul produsului cerut?
  - Cui ii este dedicat?
  - De ce il vor?
- Constrangeri:
  - Ce materiale trebuie folosite?
  - Ce standarde trebuie sa respecte?
  - Cat costa?
  - Cat timp avem la dispozitie?
  - Se pun probleme de siguranta/sanatate?
- DAR, mai ales, COMPROMISURI:
  - Alegerea scopurilor esentiale si a constrangerilor care pot fi relaxate pentru indeplinirea altor constrangeri

# Proiectarea

- Regula de aur a proiectarii:  
UNDERSTAND YOUR MATERIALS!
- Pentru noi:
  - Oamenii: aspecte sociale, psihologice, erori
    - “eroare umana” = interfete proaste
  - Calculatoarele: limitari, capacitati, instrumente, platforme
- Aspecte pe care deja le-am studiat...

# Proiectarea

- In constructii se stie ca anumite materiale au anumite proprietati (ex: rezistenta), iar inginerul si arhitectul trebuie sa decida alegerea acestora astfel incat constructia sa reziste ~ proiectarea interactiunii
- In proiectarea de interfete este natural sa ne asteptam la aparitia erorilor, dar:
  - proiectarea trebuie facuta astfel incat sa reducem posibilitatea aparitiei erorilor, iar
  - in cazul aparitiei erorilor consecintele sa fie minime
- De obicei, cand un aspect al interfetei este neclar, se mai adauga un paragraf in manualul de utilizare ☹

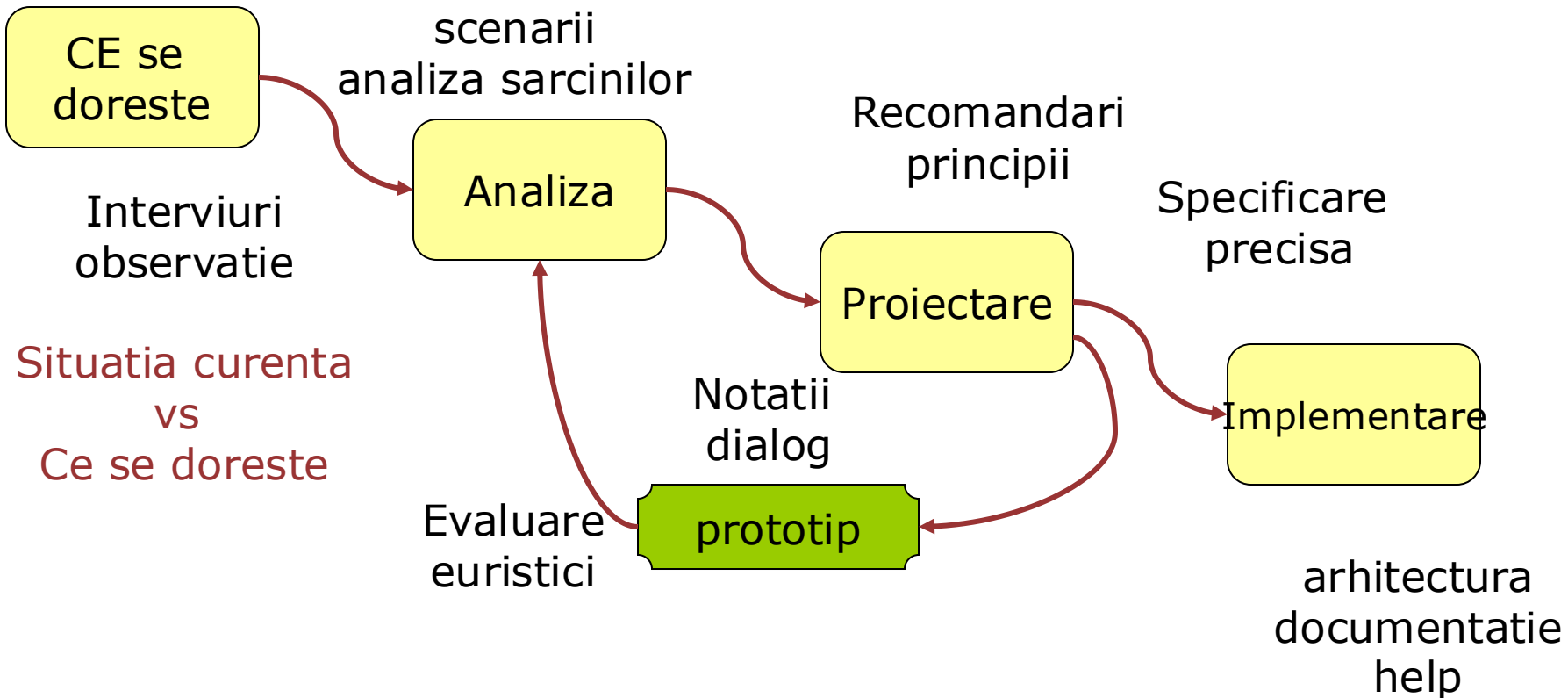
# Proiectarea

- La constructia unei cladiri trebuie intelese aspectele unde ar putea sa apara probleme si efectuate consolidari
- In proiectarea interfetelor trebuie inteles modul in care apar erorile umane, iar interactiunea trebuie proiectata corespunzator
- Ideea centrala in proiectarea interactiunii:

*puneti utilizatorul pe primul loc, pastrati utilizatorul in centrul procesului de proiectare si amintiti-va de utilizator la sfarsitul procesului de proiectare!!!*

- Abordare gresita: s-a proiectat si implementat sistemul si daca se constata ca e neutilizabil, se apeleaza la experti in utilizabilitate
- Utilizabilitatea nu trebuie confundata cu testarea, a verifica faptul ca ceva nu merge si a corecta greselile nu e la fel de recomandat cu a proiecta bine de la inceput

# Ciclul de viata al sistemelor



# Ciclul de viata al sistemelor

- Cerintele:
  - Ce se doreste ...
  - Primul pas – esential – se stabileste cu exactitate ce se doreste
  - Pentru aceasta este nevoie sa stim ce se intampla actualmente (cum realizeaza oamenii sarcinile in mod curent? Ce instrumente folosesc?)
  - Tehnici: interviuri, inregistrari video, studiul documentelor si obiectelor cu care lucreaza, observarea directa



# Ciclul de viata al sistemelor

- Analiza
  - Rezultatele obtinute in faza de cerinte trebuie ordonate, organizate astfel incat sa rezulte problemele cheie care trebuie abordate
  - Tehnici: scenarii, analiza sarcinilor
  - Aceste tehnici pot fi folosite pentru a descrie situatia curenta a sarcinilor, cat si situatia viitoare, dupa introducerea sistemului proiectat

# Ciclul de viata al sistemelor

- Proiectarea
  - Face trecerea de la CE la CUM
  - Se folosesc reguli, recomandari, principii, sabloane
  - E nevoie de inregistrarea deciziilor de proiectare (notatii, metode)
  - Notatii pentru proiectarea navigarii
  - Atentie speciala dispozitiei spatiale a elementelor pe ecran

# Ciclul de viata al sistemelor

- Iteratii si prototipizare
  - De obicei solutia potrivita nu se obtine din prima incercare
  - E nevoie de evaluarea unei proiectari pentru a vedea cum functioneaza si unde pot fi aduse imbunatatiri – tehnici de evaluare (se vor discuta)
  - Unele forme de evaluare se pot realiza pe prototipuri de hartie, dar e greu de obtinut feed-back-ul de la utilizator fara a-l lasa sa interactioneze in mod real – e preferabila dezvoltarea unui prototip

# Ciclul de viata al sistemelor

- Implementarea si livrarea
  - Cand proiectarea e finalizata se poate trece la implementarea acesteia
  - Activitati: scriere de cod, scriere de documentatii si manuale, dezvoltare de hardware
- Cum se pot parcurge toate aceste etape?
- Compromisuri:
  - uneori se accepta o proiectare drept finala chiar daca nu e perfecta (e mai bine sa avem un produs acceptabil, dar livrat la timp si in limita bugetului) decat unul care are interactiunea perfecta dar nu se incadreaza in timp si buget

# Utilizatorii

- Cand se incepe proiectare unui produs cel mai important este sa stim CINE il va folosi

REGULA: CUNOASTE-TI UTILIZATORII !!!!

- chiar daca e un singur utilizator, e posibil ca alte persoane sa fie afectate de utilizarea produsului
- de-a lungul timpului mai multe persoane vor fi afectate direct sau indirect de utilizarea sistemului = *stakeholders*

# Cum sa ne cunoastem utilizatorii...

- Cine sunt?
  - Tineri/varstnici
  - Novici/experimentati in utilizarea calculatoarelor
  - Probleme apar cand se proiecteaza un produs de larg consum (procesor de texte) unde avem multi utilizatori diferiti, cu diverse scopuri, sau pagini web
  - In astfel de situatii incercam sa identificam categorii specifice de utilizatori

# Cum sa ne cunoastem utilizatorii...

- Probabil NU sunt similari noua!
- E mult mai usor sa concepi un produs considerand ca utilizatorul principal esti tu (se cunosc interesele, abilitatile)
- Problema:
  - majoritatea dezvoltatorilor de soft sunt barbati –  
posibilitatea de empatie este redusa fata de femei

# Cum sa ne cunoastem utilizatorii...

- Vorbeste cu utilizatorii !
- Este foarte greu sa aflam cum gandeste cineva – cea mai la indemana metoda este discutia
- Interviuuri despre munca lor, viata lor
- Discutii libere
- Implicarea utilizatorilor in procesul de proiectare – proiectare participativa – asigura o proiectare mai buna dar si acceptarea si utilizarea produsului
- E importanta discutia cu utilizatorii vizati de introducerea sistemului, deoarece reprezentatii organizatiei vor descrie modul in care trebuie sa se realizeze o sarcina, dar utilizatorii vor spune cum se realizeaza in realitate



# Cum sa ne cunoastem utilizatorii...

- Priveste utilizatorii
- Ceea ce spun utilizatorii nu este intotdeauna suficient, deoarece este foarte greu sa descriem in cuvinte tot ceea ce facem
- E foarte important sa privim ceea ce fac utilizatorii, ceea ce spun (notite, inregistrare), obiectele pe care le manipuleaza si cele pe care le creeaza (rezultatele muncii lor)
- Observatia trebuie urmata de discutii pentru a lamuri lucruri pe care au omis sa le spuna, dar pe care le efectueaza
- Observatia ne arata CE fac, iar discutiile ne vor lamuri DE CE fac anumite lucruri

# Cum sa ne cunoastem utilizatorii...

- Folositi-va imaginatia!
- Chiar daca se doreste implicarea unui numar mare de utilizatori in procesul de proiectare, uneori acest fapt este imposibil (cost, acces)
- Solutia: incercarea de a ne imagina experienta lor – periculos!
- Folosirea personajelor/caracterelor (personas)- imagine detaliata a unei persoane imaginare care reprezinta grupul tinta de utilizatori
- O echipa de proiectare va avea mai multe personaje acoperind diferitele tipuri de utilizatori – construite pe baza utilizatorilor tinta si diferitele roluri identificate prin studiul/observatia lor
- La propunerea unei solutii de proiectare echipa se va intreba:”Cum va reactiona X in aceasta situatie?”

# Personas

- Concepte precum “user”, “user-friendly” sunt prea vagi pentru comunicarea in cadrul echipei de dezvoltare
- Persona devine un instrument de comunicare intre dezvoltatori, clienti si alti stakeholderi
- *“A precise description of our user and what he wishes to accomplish.”* (Cooper, 1999)
- *“User models, or personas, are fictional, detailed archetypical characters that represent distinct groupings of behaviours, goals and motivations observed and identified during the research phase.”* (Calde, Goodwin & Reimann (2002) )
- Ne informeaza despre:
  - Modul in care se comporta utilizatorii
  - Cum gandesc
  - Ce scopuri au de indeplinit
  - De ce urmaresc anumite scopuri

# Personas

- Personas nu sunt persoane reale, dar sunt construite pe baza unor comportamente si scopuri reale
- Contin descrieri care includ sabloane de comportament, scopuri, aptitudini, reactii, mediu si cateva detalii fictive care sa faca personajul credibil
- In proiectarea interactiunii e nevoie sa consideram pe langa scopuri de indeplinit si oamenii implicati
- “You cannot have purposes without people [...] That is why the two key elements of our design process are **goals** and **personas**; purposes and people.” (Cooper)

# Personas- exemplu

- **Persona: Rhonda Wilson, Nurse Unit Coordinator**



- Rhonda is a 36-year-old registered nurse who has worked at several skilled nursing facilities. She started out in acute care but moved to long-term care so she could have more autonomy. Rhonda was promoted to Unit Coordinator four years ago because she is very competent and generally well organized. Rhonda is entirely overwhelmed and is drowning in paper, even more so than the average nurse. She often misses eating dinner with her boyfriend because she has to work late, filling out forms and reports.
- Rhonda's goals are to:
- Spend time on patient care and staff supervision, not paperwork.
- Be proactive. Rhonda needs to understand trends in order to solve problems before they happen, instead of just reacting to crises.
- Know that things are being done right. Rhonda supervises the unit because she's good at what she does. If nurses aren't following procedure or documenting things, she wants to know right away.

# Personas

- Descrierea unei persona trebuie sa cuprinda:
  - Scopuri
  - Atitudini
  - Abilitati
  - Mediul
  - Cateva detalii personale fictive care sa aduca personajul la viata

# Personas

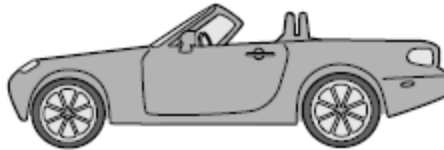
- Pentru a crea un produs care sa satisfaca o paleta larga de utilizatori – un produs cu cat mai multe functionalitati – gresit
- Abordarea corecta – proiectarea pentru tipologii de utilizatori cu nevoi specifice
- Prin adaugarea continua de functionalitati unui produs se determina cresterea incarcarii cognitive si complicarea navigarii

# Personas



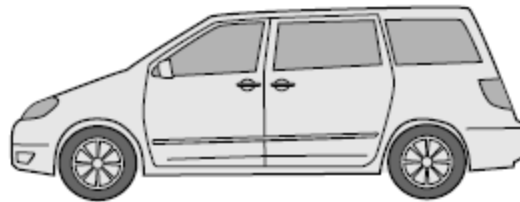
Alesandro's goals

- Go fast
- Have fun



Marge's goals

- Be safe
- Be comfortable



Dale's goals

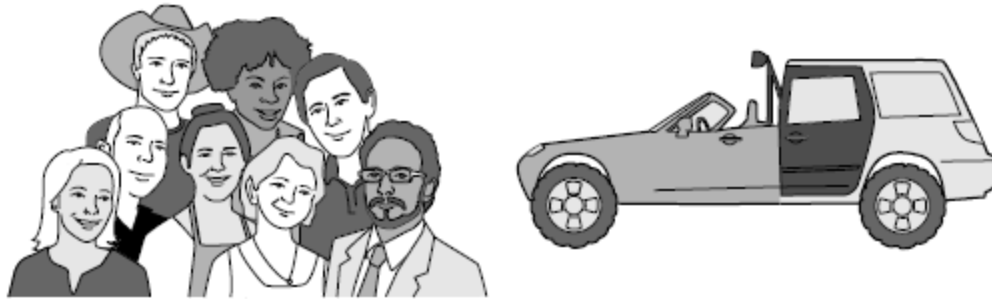
- Haul big loads
- Be reliable





# Personas - exemplu

- Proiectarea unei masini pe placul tuturor



- O masina cu foarte multe functionalitati care nu place nimanui ~ softul

# Avantajele utilizarii personas-urilor

- Descriu ceea **ce** trebuie sa faca un produs si **cum** trebuie sa se comporte
- Faciliteaza comunicarea cu stakeholderii oferind un limbaj comun si pastreaza utilizatorii in centrul procesului de proiectare
- Determina consens si implicare in proiectare pe baza unei intelegeri comune a sarcinilor care trebuie indeplinite
- Masoara eficienta solutiei proiectate – se poate face testarea pe baza personas-urilor (nu inlocuieste insa testarea cu utilizatori reali)
- Contribuie la alte aspecte colaterale ale produsului: marketing, planuri de vanzare, etc.

# Personas



- Alte probleme rezolvate de folosirea personas-uri:
  - “utilizatorul elastic” - fiecare membru al echipei de proiectare isi imagineaza ca utilizatorul este asa cum se asteapta el
  - proiectarea centrata pe sine – “self-referential design”
  - cazurile limita – situatii care e posibil sa apara, dar apar extrem de rar pentru personas-urile tinta (acestea trebuie luate in considerare, dar nu trebuie centrata proiectarea pe ele)
    - Se va pune intrebarea: “Va dori X sa realizeze aceasta operatie foarte des?”/”Va dori X sa realizeze aceasta operatie vreodata?”
- Personas-urile sunt construite in urma observatiei/interviurilor realizate in situatia reala

# Personas

- Actioneaza ca arhetipuri
- Desi sunt descrise individual, ele reprezinta grupuri de utilizatori
- Personas descriu sabloane de interactiune cu un produs interactiv
- Nu se pot reutiliza pentru alte produse, deoarece sunt construite specific unui context
- Personas <> stereotipuri (presupuneri ale proiectantului despre posibillii utilizatori)
- Daca nu sunt tratate cu rigurozitate, personas-urile pot deveni stereotipuri ☹️

# Personas

- Scopul unui personas nu este de a descrie un utilizator “medie” a utilizatorilor, ci de a descrie comportament semnificativ pentru fiecare din grupurile de utilizatori identificate
- E esential a intelege motivatia pentru care personas-urile au un anumit comportament – astfel vor fi intelese scopurile utilizatorilor

# Cum sa ne cunoastem utilizatorii...

- Scenariile – povestioare detaliate destinate proiectarii
- Urmaresc satisfacerea scopurilor personasurilor
- Scenariul – descrierea concisa a modului in care persona foloseste sistemul pentru a indeplini un scop
- Una din cele mai simple reprezentari folosite in proiectare, dar foarte flexibile si eficiente
- Sunt mult mai detaliate decat ar fi necesar, dar detaliile au scopul de a le face credibile
- Pot fi insotite de schite, capturi de ecran, etc – story boards

# Scenariile

- Scenariile pot fi folosite pentru:
  - **A comunica cu ceilalti (proiectanti/clienti/utilizatori)**
    - Discutiile abstracte pot duce la neintelegeri
    - Exemplele concrete de folosire sunt mai usor de impartasit
  - **Pentru a valida modele** – pot fi transpuse in modele de navigare sau modele ale sarcinilor – validate automat
  - **Exprima dinamismul** – scenariile sunt liniare si descriu o singura cale in interactiune-avantaje si dezavantaje – e mai usor de urmarit povestiri liniare (+)/ nu sunt prezentate alternativele (-)
- Scenariile sunt o resursa care poate fi refolosita pe parcursul procesului de proiectare pentru:
  - a ne ajuta sa vedem ce se doreste,
  - a sugera modul in care utilizatorii se comporta cu proiectarea propusa,
  - a verifica faptul ca proiectarea propusa poate functiona
  - a genera cazuri de testare in final

# Exemplu scenariu

- **Scenariu #1: Schimbarea functiei unui angajat**
- A B, specialist in resurse umane, primeste o cerere prin email de schimbare a postului lui X Y din tester in programator.
- Actiunea: A B deschide HRWeb si apasa pe butonul Cautare Angajat. A B foloseste task-bar-ul pentru a se intoarce la email pentru a copia identificatorul lui X Y. Ea foloseste mouse-ul pentru a marca id-ul, il copiaza si se intoarce in HRWeb, copiaza id-ul in campul ID Angajat si actioneaza butonul Cautare. Apoi apasa pe butonul Informatii angajat, dar este intrerupta apoi de un telefon:
- A B:
  - A B de la Resurse Umane.
- Interlocutor:
  - Am gasit o greseala in fluturasul de salariu....
- A B:
  - Spuneti-mi care e problema...

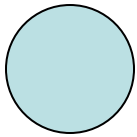


# Interactiunea

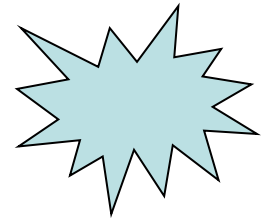
- Comportament in care se cauta modalitati de indeplinire a unor scopuri
- Daca utilizatorul ar avea o idee clara a modului in care functioneaza sistemul pentru orice scop, ar alege mereu cea mai scurta cale
- In realitate intelegerea sistemului este partiala - utilizatorul “bajbaie” prin sistem
- Faptul ca utilizatorul nu alege la fiecare pas cea mai eficienta varianta nu e atat de important ca faptul ca la fiecare pas din interactiune utilizatorul sa isi dea seama daca se apropie de scopul propus
- Pentru a atinge scopul propus, fiecare stare a sistemului sau fiecare fereastră trebuie sa dea suficienta informatie utilizatorului asupra a ceea ce trebuie facut pentru a se apropia de scopul propus

# Atingerea scopurilor

start

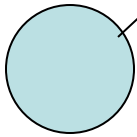


scop

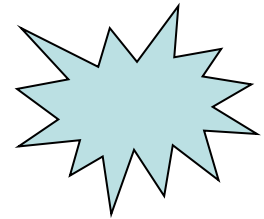


# Atingerea scopurilor

start

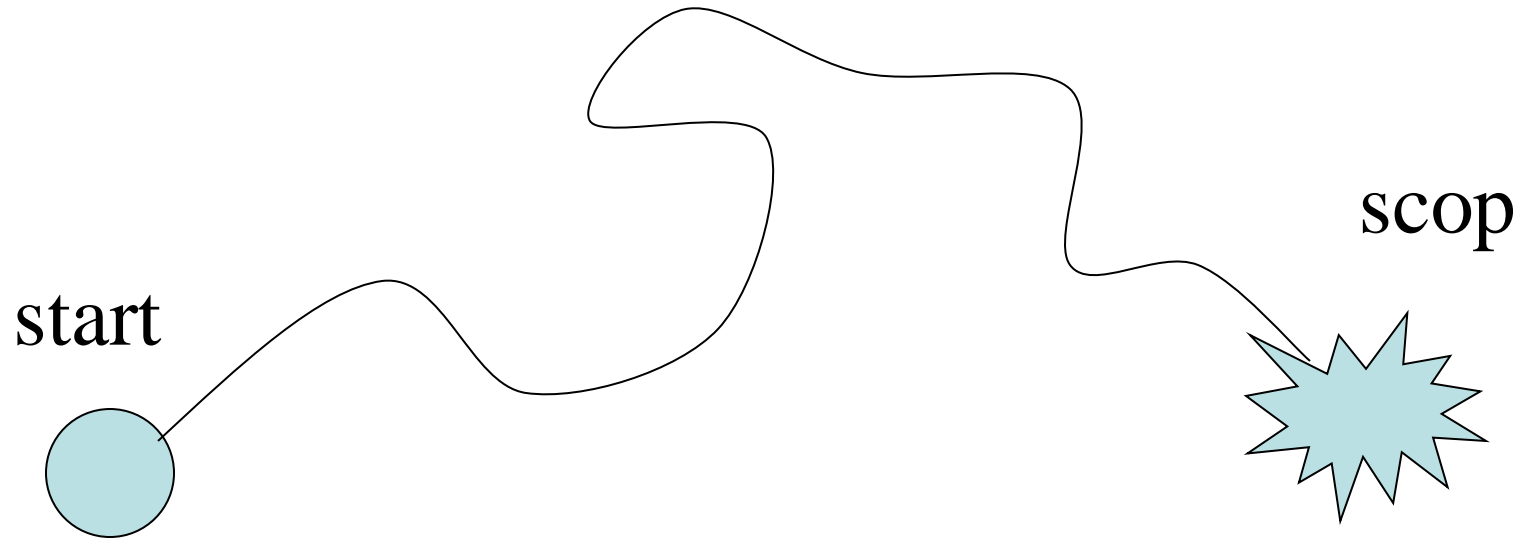


scop

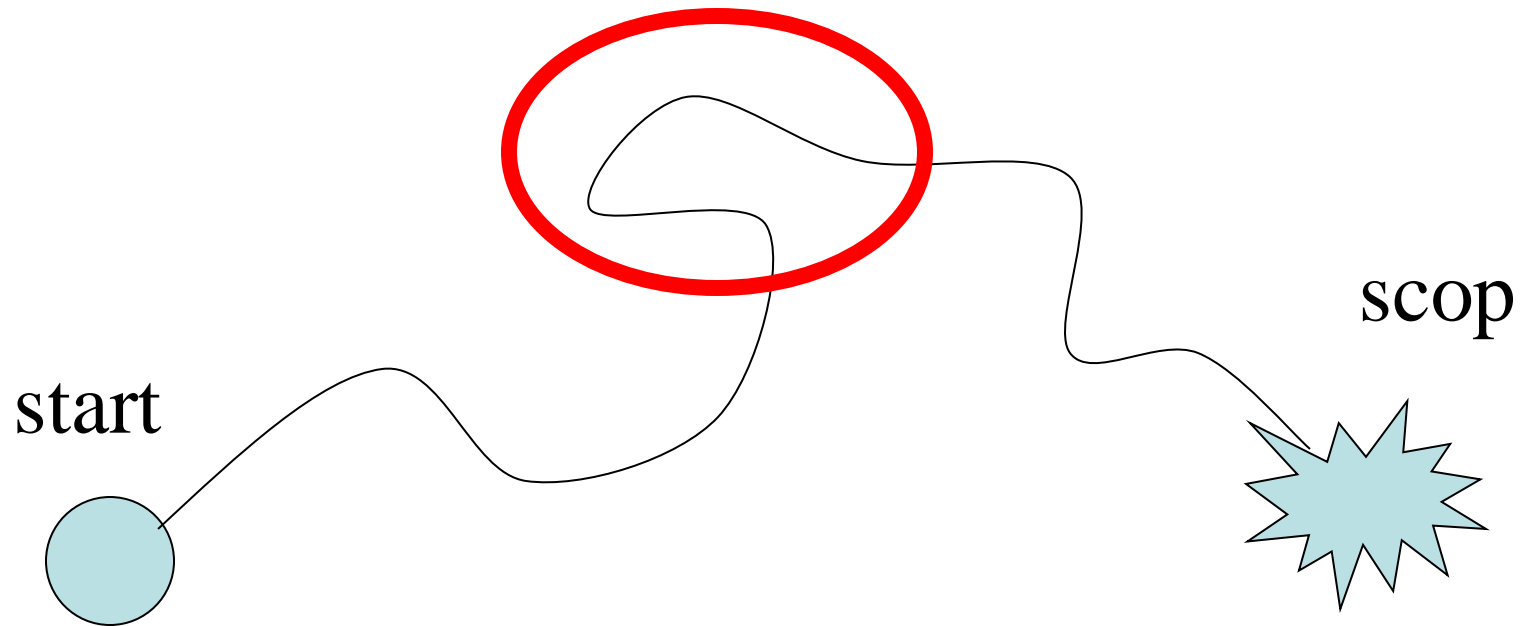


Cunoastere partiala

# Atingerea scopurilor



# Atingerea scopurilor



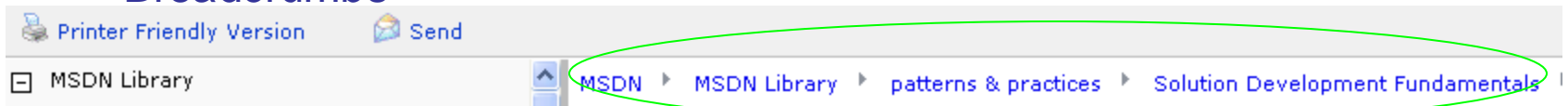
... incercati sa evitati aceste "rataciri"!

# Interactiunea

- Aspecte importante care sprijina interactiunea:

- A sti UNDE ESTI

- Breadcrumbs



- A sti CE POTI SA FACI

- Linkuri vizibile, butoane active/inactive

- A sti UNDE POTI SA MERGI (ce poti sa faci)

- In paginile web se poate explora si reveni cu Back, dar in unele aplicatii consecintele “explorarii” ar putea fi nedorite

- A sti DE UNDE VII (ce ai facut)

- Feed-back-ul sistemului ajuta la constientizarea consecintelor actiunilor noastre

# Proiectarea navigarii

- Interactiunea se realizeaza in mod concret prin:
- Obiecte grafice (widgeturi) – alegerea corespunzatoare a widgeturilor si a denumirilor meniurilor, optiunilor, butoanelor vor usura selectia/actionarea actiunii potrivite
- Ecrane si ferestre – obiectele grafice trebuie regasite usor pe ecran, e necesara intelegerea logicii gruparii elementelor pe ecran
- Navigarea in cadrul aplicatiei – trebuie sa putem intelege ce se intampla la apasarea unui buton, unde suntem in interactiune

# Principii de baza in proiectare

## Intreaba

Ce face utilizatorul?

## Gandeste

Care este informatia de care are nevoie?

Ce comparatii are nevoie sa faca?

In ce ordine sunt necesare informatiile?

## Proiecteaza

Forma trebuie sa urmeze functionalitatea – lasa interactiunea sa determine dispozitia spatiala



# Recomandari

- Grupare si structurare
- Daca obiectele apartin aceleiasi categorii, atunci trebuie dispuse fizic impreuna
- Uneori e nevoie de mai multe niveluri de structurare
- Pentru a stabili ordinea elementelor pe ecran trebuie sa ne gandim la ordinea naturala pentru utilizator
- Pentru formele de introducere a datelor sau cutii de dialog trebuie stabilita ordinea tab-urilor

# Logic impreuna → fizic impreuna

**Furnizor:**

Nume:...

Adresa: ...

Cod fiscal: ...

**Client:**

Nume: ...

Adresa: ...

---

**Produse:****Cantitate Cost/unitate Cost**

Produs 1

7

3.71

25.97

.....

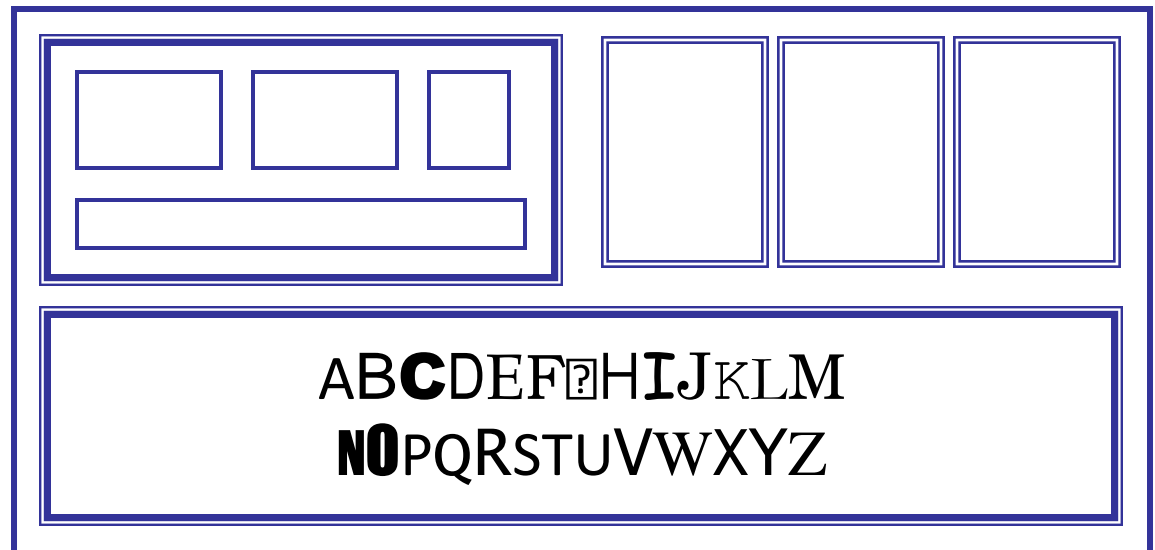
...

...

...

# Decoratiuni

- Folosite pentru a grupa logic obiectele
- Fonturile pot fi folosite pentru a sublinia/evidentia ceva
- Folosite cu cumpatare!



# Alinierea textului

- Citirea se face de la stanga la dreapta – alinierea se va face la stanga

Ion  
Mara  
Amintiri din copilarie  
La moara cu noroc



Usor de scanat




Ion  
Mara  
Amintiri din copilarie  
La moara cu noroc

# Alinierea numelor



Calinici Mihai
Florescu Lucian
Moldovan Ioana
Rus Vasile



Calinici	Mihai
Florescu	Lucian
Moldovan	Ioana
Rus	Vasile



Mihai, Calinici
Lucian, Florescu
Ioana, Moldovan
Vasile, Rus

# Alinierea numerelor

- Care este mai mare?
- Aliniati dupa punct numerele reale
- Aliniati la dreapta numerele intregi
- Forma coloanei da o indicatie asupra ordinului de marime



532.56
179.3
256.317
15
73.948
1035
3.142
497.6256



627.8650
2382.5830
2502.5600
432.9350
2.0175
652.8700
56.3400

# Coloane multiple



<b>Introducere</b>	1
<b>Capitolul 1 Introducere în șabloane de proiectare</b>	4
Introducere	4
Probleme rezolvate de șabloanele de proiectare	6
<b>Capitolul 2 Șablonul de proiectare MVC</b>	9
Introducere	9
Abordările Model 1 și Model 2	12
Rolurile componentelor MVC	15



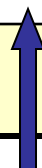
<b>Introducere.....</b>	<b>1</b>
<b>Capitolul 1 Introducere în șabloane de proiectare.....</b>	<b>4</b>
Introducere.....	4
Probleme rezolvate de șabloanele de proiectare.....	6
<b>Capitolul 2 Șablonul de proiectare MVC... ..</b>	<b>9</b>
Introducere.....	9
Abordările Model 1 și Model 2.....	12
Rolurile componentelor MVC.....	15

# Coloane multiple



Ciocolata Milka	37
Ciocolata Poiana	42
Ciocolata Africana cu lapte	12
Ciocolata Kandia	56

Ciocolata Milka	37
Ciocolata Poiana	42
Ciocolata Africana cu lapte	12
Ciocolata Kandia	56



Daca scanarea se face mai des dupa numere!



# Folosirea spatiilor – pentru a separa



Spatii intre  
paragrafe sau  
sectiunile unor  
rapoarte

# Folosirea spatiilor – pentru a structura



# Folosirea spatiilor – pentru a evidentia



# Griduri



System Properties

Computer name Hardware Advanced System protection Remote

Windows uses the following information to identify your computer on the network:

Computer description:

For example: "Kitchen Computer" or "Mary's Computer".

Full computer name: Jonathan.Fabrikam.com

Domain: Fabrikam.com

To use a wizard to join a domain or workgroup, click Network ID.

Network ID...

To rename this computer or change its domain or workgroup, click Change.

Change...

OK Cancel Apply



System Properties

Computer name Hardware Advanced System protection Remote

Windows uses the following information to identify your computer on the network:

Computer description:

For example: "Kitchen Computer" or "Mary's Computer".

Full computer name: Jonathan.Fabrikam.com

Domain: Fabrikam.com

To use a wizard to join a domain or workgroup, click Network ID.

Network ID...


To rename this computer or change its domain or workgroup, click Change.

Change...


OK Cancel Apply

# Introducerea informatiilor

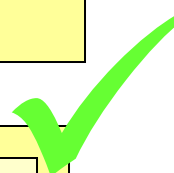
- Ferestre si cutii de dialog
- Complexitate: date care se afiseaza+date care se introduc
- Numele campurilor sunt de dimensiuni diferite...cum aliniam?
- folosirea unui font mai mic si situarea etichetelor in stanga sus a campului text
- Etichetele aliniate la dreapta



<b>Nume:</b>	Popescu Dan
<b>Adresa:</b>	Brasov



<b>Nume:</b>	Popescu Dan
<b>Adresa:</b>	Brasov



<b>Nume:</b>	Popescu Dan
<b>Adresa:</b>	Brasov

# Elemente active/pasive

- Elemente pasive – ofera informatii
- Elemente active: asteapta sa fie completate, sau sa le actionam/selectam
- Widgeturile ar trebui folosite corespunzator (nu vom asocia functionalitate actionarii unei etichete!)
- Obiectele active care nu ar trebui actionate intr-un context specific trebuie facute inactive!