

# Seminar 8

Realizarea sistemelor informatice  
pentru management

Diagramele de interacțiune UML

# Rolul diagramelor de interacțiune

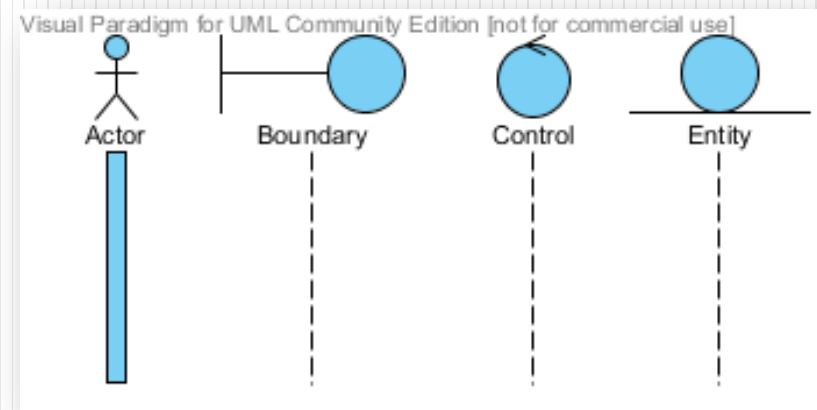
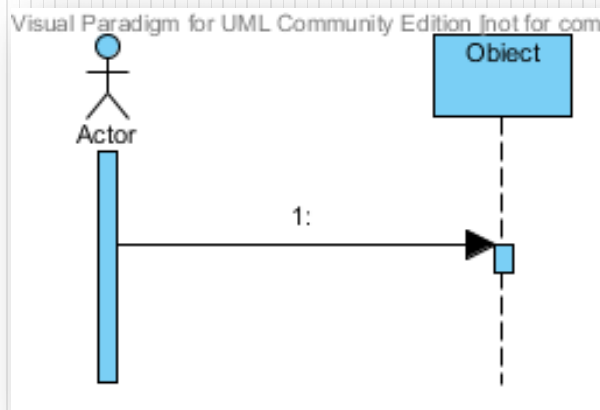
- Modelează aspectele dinamice ale sistemului.
- Sunt alcătuite dintr-un set de obiecte și relațiile dintre ele, incluzând și mesaje pe care obiectele le trimit de la unul la altul.
- Există două tipuri de diagrame de interacțiune: *diagrama de secvență* și *diagrama de comunicare* (în UML 1.4 numită de *colaborare*).
- Cele două diagrame sunt echivalente din punct de vedere semantic și se pot transforma una din alta.

# Diagrama de secvență

- Este o diagramă de interacțiune formată din *obiecte*, *mesajele* care se schimbă între acestea și *dimensiunea temporală* reprezentată progresiv pe verticală.
- Subliniază ordinea mesajelor în funcție de timp.
- **Obiectele** sunt plasate în marginea de sus a diagramei, de-a lungul axei OX, de la stânga la dreapta.
  - Ele sunt aranjate în orice ordine care permite simplificarea diagramei.
  - De obicei, obiectele care încep interacțiunea se așează la stânga iar obiectele care urmează în partea dreaptă.
  - Existența obiectelor este reprezentată prin liniile de viață ale acestora.

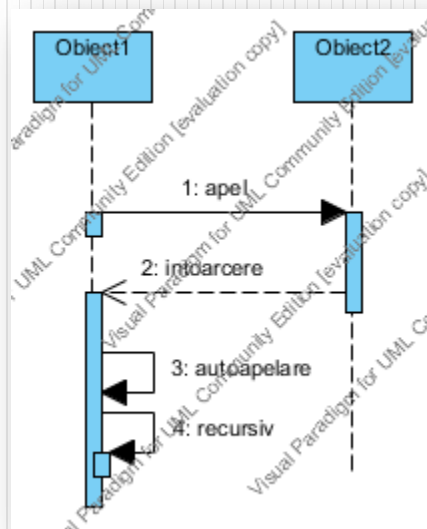
# Diagrama de secvență - obiecte

- *Linia de viață a obiectelor*: linie verticală care reprezintă existența unui obiect de-a lungul unei perioade de timp. Majoritatea obiectelor care apar în diagramă există pe toată durata interacțiunii, având linia de viață trasată de la vârful diagramei până la bază. Alte obiecte pot fi create pe parcursul interacțiunii.
- *Activare (Punct de control)*: un dreptunghi înalt și subțire care indică perioada de timp în care obiectul realizează o acțiune. Capătul de sus al dreptunghiului este aliniat la începutul acțiunii iar capătul de jos la sfârșitul acțiunii.
- Obiectele pot fi reprezentate folosind *stereotipurile* actor, limită, entitate și control.



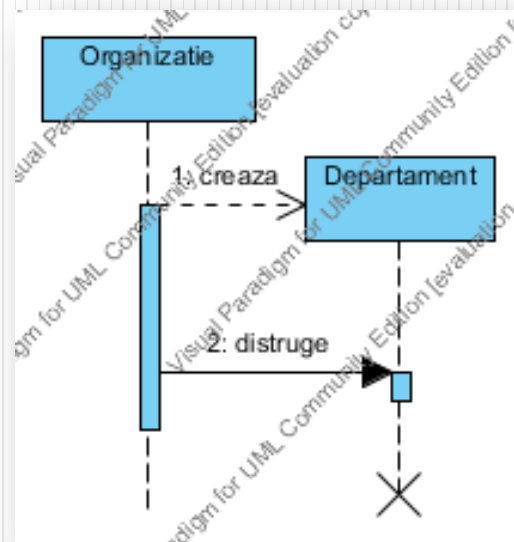
# Diagrama de secvență - mesaje

- *Mesajele* sunt reprezentate în forma unor arcuri. Acestea pornesc de la linia de viață a unui obiect și se opresc la linia de viață a altui obiect. Mesajele pot fi de mai multe tipuri și pot include și condiții, ca în diagramele de stare.
- Un mesaj de tip *apel (call)* reprezintă o cerere a obiectului care trimite mesajul către obiectul care primește mesajul. Cererea implică faptul că receptorul va executa una dintre operațiile sale.
- Emitătorul așteaptă ca receptorul să execute operația și să primească un răspuns de la acesta (*return*).
- Un obiect poate trimite mesaje și către sine - *autoapelare*. Un astfel de mesaj poate semnifica apelul recursiv al unei operații sau o metodă care apelează altă metodă a aceluiași obiect.

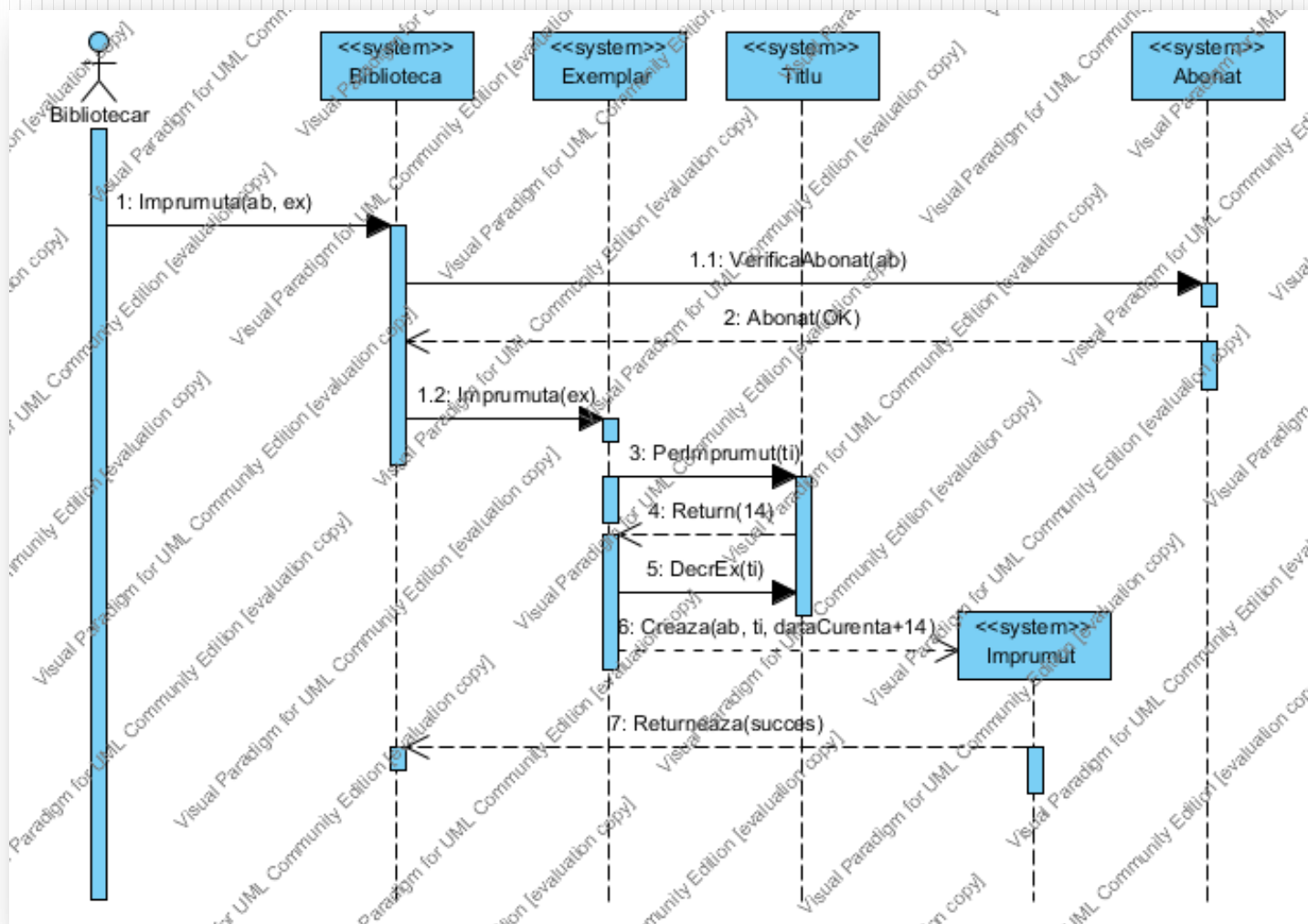


# Diagrama de secvență - mesaje

- Mesajele de *creare* (*create*) și *distrugere* (*destroy*) a unui obiect încep și respectiv încheie linia de viață a unui obiect. Acestea sunt opționale și se folosesc atunci când se dorește specificarea explicită a acestor evenimente.
- Mesajul de *distrugere* poate genera distrugerii ulterioare ale unor obiecte pe care acesta le conține prin compunere. După distrugere, un obiect nu mai poate fi creat din nou pe aceeași linie de viață.



# Diagrama de secvență - exemplu

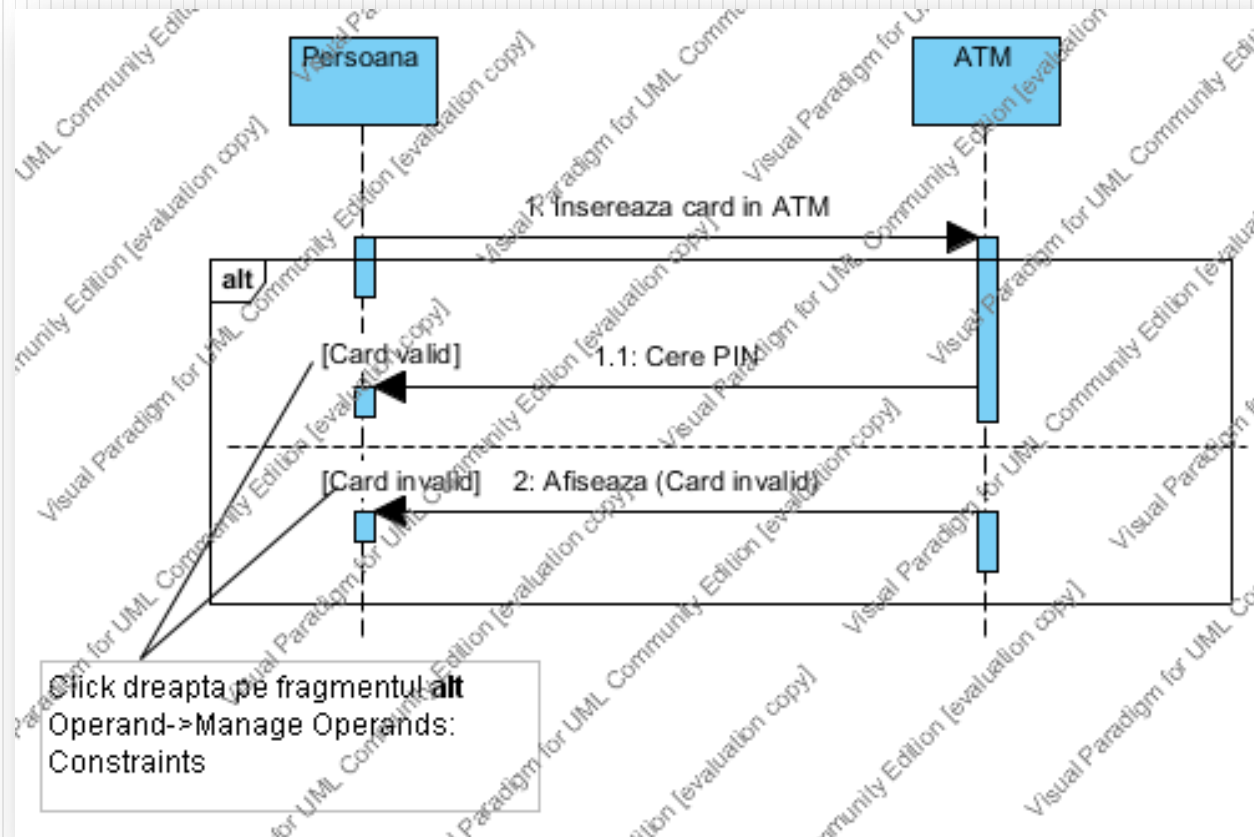


# Fragmente combinate

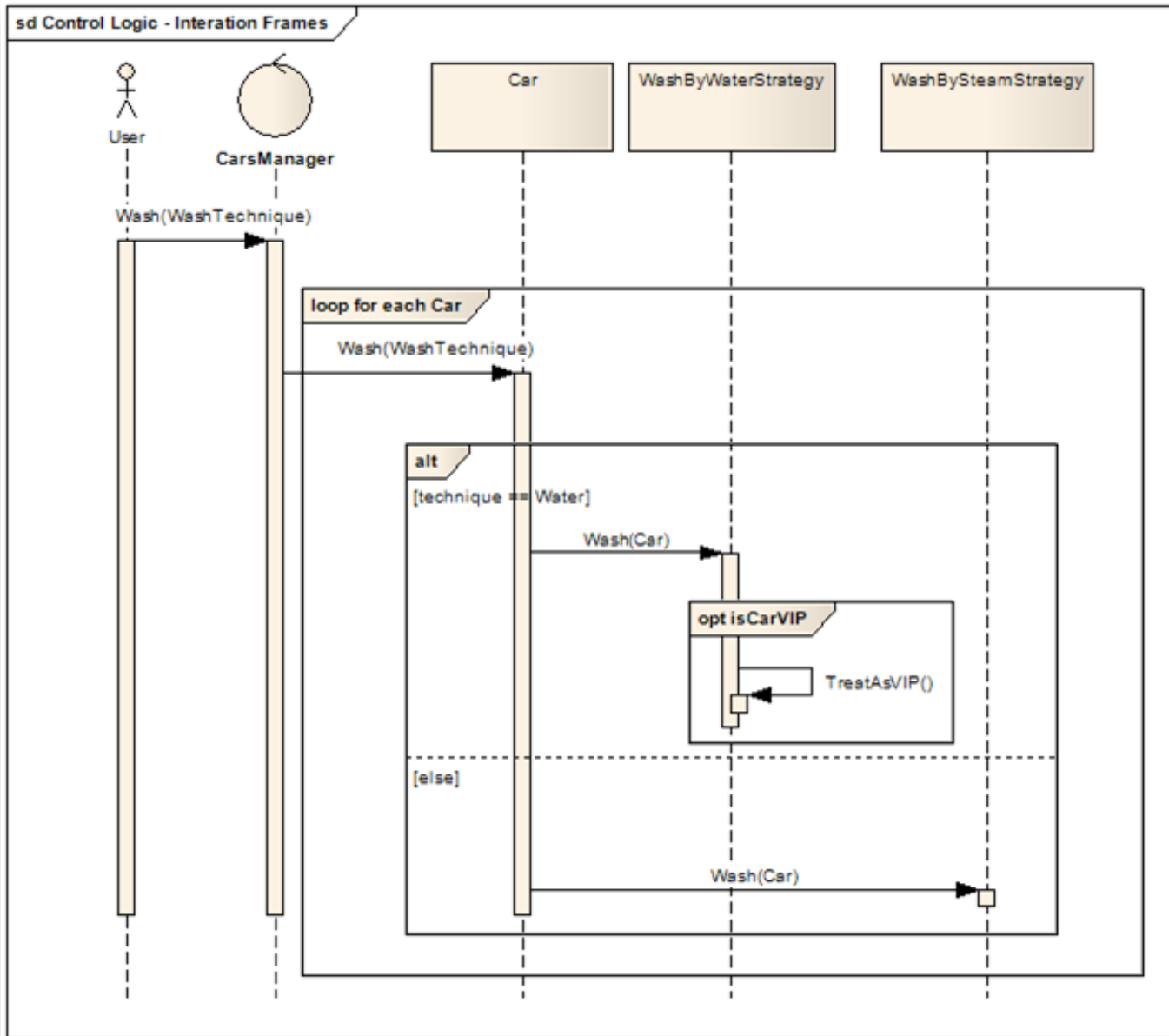
- Diagramele de secvență nu sunt folosite pentru a reprezenta logică procedurală complexă, ci pentru a modela fluxuri de control simple, secvențiale.
- Cu toate acestea, există mecanisme care permit adăugarea unui anumit nivel de logică procedurală în diagrame prin fragmentele combinate.
- Un *fragment combinat* reprezintă una sau mai multe secvențe de procesare incluse într-un cadru și executate în anumite circumstanțe.
- Frecvent utilizate sunt fragmente de tip:
  - Alternative (**Alt**) care modelează construcțiile de tipul if..then..else.
  - Repetitive (**Loop**) care conțin o serie de interacțiuni ce se vor repeta de mai multe ori.
  - Paralele (**Par**) care modelează procesarea concurentă.



# Fragmente combinate - exemplu



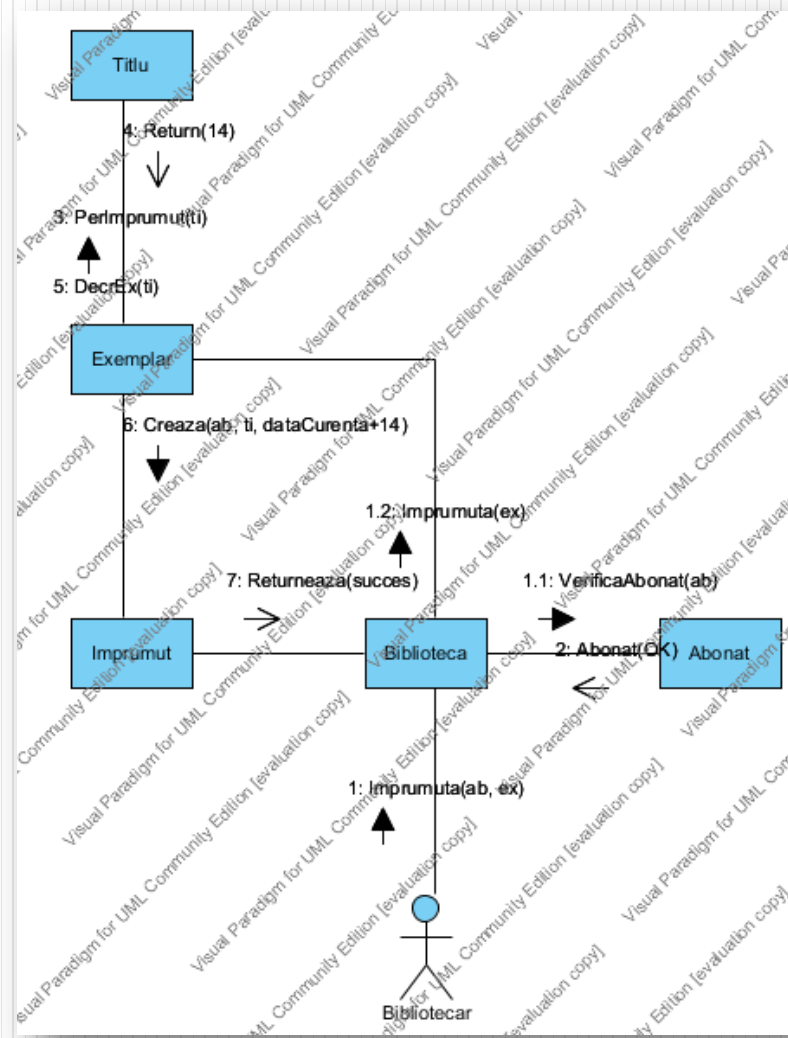
# Fragmente combinate - exemplu



# Diagrama de comunicare

- **Diagrama de comunicare (colaborare - nume în UML 1.4)** este o diagramă de interacțiune care subliniază organizarea structurală a obiectelor care trimit și primesc mesaje.
- Grafic, o diagramă de colaborare este o colecție de vârfuri și arce.
- Reprezintă aceleași informații ca și diagrama de secvență, dar subliniază organizarea obiectelor care participă la interacțiune.
- Obiectele sunt plasate primele, ca vârfuri ale unui graf, se trasează legăturile care conectează obiecte, ca arcuri în acest graf, apoi se adaugă acestor legături mesaje pe care obiectele le primesc sau le trimit.
- Pentru a indica ordinea, mesajul trebuie prefixat cu un număr începând de la 1 și crescând.

# Diagrama de comunicare - exemplu



# Diagrame de interacțiune

- Cele două diagrame de interacțiune sunt echivalente și o diagramă poate fi convertită în cealaltă fără a se pierde informații.
- Pentru a transforma o diagramă în alta, în Visual Paradigm se face click dreapta pe suprafața unei diagrame și se selectează opțiunea *Synctonize to Communication/Sequence diagram*, după caz.
- Diagrama de comunicare arată cum sunt legate obiectele în timp ce diagrama de secvență pune în evidență și mesajele returnate, precum și ordinea temporală a interacțiunilor.

# Diagramele de interacțiune UML



- Care sunt cele două tipuri de diagrame de interacțiune?
- Cum se reprezintă recursivitatea în diagramele de secvență?
- Pentru ce sunt folosite fragmentele combinate?
- Ce reprezintă construcția “alt”?
- Câte dimensiuni are o diagramă de secvență și ce reprezintă acestea?
- Exemplificați rolul diagramelor de comunicare.

## Lucru la seminar



*Să se întocmească diagramele de interacțiune pentru cazul de utilizare “Rezervă camere” din scenariul de mai jos.*

Scopul proiectului este realizarea aplicației informatice pentru gestiunea activității unei unități hoteliere. În vederea cazării, un client poate solicita rezervarea uneia sau mai multor camere prin e-mail sau telefonic. Pentru aceasta furnizează recepționarului informații privind perioada de cazare și tipurile de camere solicitate. Clienții vor beneficia de reduceri dacă rezervă cel puțin 3 camere sau dacă perioada de cazare depășește 5 zile. Recepționarul verifică disponibilitatea camerelor și îl înștiințează pe client de acest lucru precum și de costul estimat al cazării. Dacă nu există camere disponibile conform solicitării, recepționarul poate oferi clientului alternative. De asemenea, clientul poate solicita un discount (suplimentar sau nu), iar recepționarul va decide fezabilitatea discountului, fiind asistat obligatoriu de managerul hotelului. În situația în care clientul este de acord cu prețul propus, se va proceda la realizarea rezervării. Pentru clienții noi, recepționarul solicită datele de identificare, pe care le introduce în aplicație.

Odată ajuns la hotel, și dacă a făcut în prealabil o rezervare, clientul va furniza datele de identificare ale sale și/sau ale rezervării și se face cazarea. Dacă nu există o rezervare, se va verifica disponibilitatea camerelor pentru perioada cerută. Atunci când se găsește o astfel de cameră, se face cazarea. La finalul sejurului, recepționarul întocmește o listă cu toate serviciile solicitate de client și prețul acestora. Lista trebuie validată de client, după care se întocmește factura finală. Factura poate fi plătită parțial sau integral, prin transfer bancar, numerar sau folosind un card bancar. Totodată, înainte de a părăsi hotelul, clientul este rugat să completeze un formular prin care să evalueze serviciile oferite de unitatea hotelieră.