

# APLICAȚII ÎNTEGRATE PENTRU ÎNTREPRINDERI

## Tema 2

Implementarea modului de aprovizionare și de comunicație cu alți parteneri externi pentru un sistem informatic din cadrul unui lanț de policlinici

| Lansare           | Termen de Predare | Pondere |
|-------------------|-------------------|---------|
| 02.12.2014, 23:55 | 16.12.2014, 23:55 | 9 / 40  |

### Obiective

Obiectivul temelor de casă este reprezentat de implementarea unui sistem informatic performant și scalabil pentru gestiunea resurselor unei întreprinderi de dimensiuni mari. În acest an universitar, organizația pentru care se va proiecta și dezvolta acest pachet de programe este un lanț de policlinici.

Se va realiza implementarea graduală a mai multor module din cadrul aplicației ERP, pe parcursul a patru etape.

În cadrul acestei etape, sistemul informatic va fi capabil să interogheze servicii care rulează în alte mașini virtuale și ale căror funcționalități le cunoaște, acestea fiind accesibile prin intermediul unei rețele de calculatoare.

După rezolvarea temei de casă, studentul va fi capabil să:

- proiecteze o aplicație având o structură distribuită pe componente care poate rula pe mașini diferite;
- modeleze comunicația între componentele distribuite ale aplicației.

Cunoștințele necesare pentru rezolvarea temei de casă sunt:

- programarea în limbajul Java;
- manipularea bazelor de date folosind MySQL;
- folosirea API-ului Java DataBase Connectivity;
- utilizarea, la alegere, a unei dintre tehnologiile RMI, CORBA sau JAX-WS<sup>1</sup> pentru apelarea unor metode aparținând unor obiecte existente într-o altă mașină virtuală, a căror definiție se consideră cunoscută.

### Enunț

Sistemul ERP pentru gestiunea activității dintr-un lanț de policlinici va fi completat spre a comunica cu alte aplicații ce oferă, prin intermediul rețelei Internet, anumite funcționalități al căror mod de accesare este cunoscut.

Astfel, aplicația corespunzătoare lanțului de policlinici va interoga un serviciu care intermediază comunicația cu mai mulți producători de echipamente medicale în scopul achiziționării acestora<sup>2</sup>, într-un anumit interval de timp și la un anumit preț. De asemenea, anumite servicii medicale, oferite de medici aflați în contract cu casa de asigurări de sănătate vor putea fi decontate, în limita unui barem prestabilit, pentru pacienții care pot beneficia de astfel de facilități.

---

<sup>1</sup> NU este permisă folosirea altor tehnologii decât a celor specificate în cadrul enunțului. Definițiile metodelor apelabile la distanță vor fi descrise în pseudocod. Acestea pot fi modificate la nivelul parametrilor sau rezultatelor întoarse, păstrându-se însă funcționalitatea lor.

<sup>2</sup> Această componentă reprezintă modulul SCM (Supply Chain Management), parte integrantă a oricărui sistem ERP.

Programul care gestionează activitatea din cadrul lanțului de policlinici se va conecta la un **agent de intermediere a comunicației cu diferiți furnizori de echipamente medicale** pe care îl poate interoga cu privire la lista producătorilor înregistrați, produsele pe care aceștia le comercializează (împreună cu data de livrare a unei cantități precizate și prețul lor), putând să le rezerve pentru un interval de timp (și să anuleze rezervarea), respectiv să le comande (și să anuleze comanda).

În acest sens, agentul de intermediere a comunicației cu producătorii va gestiona cererile și răspunsurile cu privire la furnizarea anumitor echipamente medicale, stocând local informații cu privire la starea acestor operații și redirectionând solicitările formulate de lanțul de policlinici către componenta corespunzătoare.

Așadar, funcționalitățile accesibile la distanță sunt:

- 1) obținerea listei cu furnizorii înregistrați

```
Manufacturer[] getManufacturers();
```

Un producător este caracterizat prin denumirea sa și o descriere ce conține profilul / gama de produse comercializate.

- 2) obținerea listei cu echipamentele medicale furnizate

```
MedicalSupply[] getMedicalSupplies(Manufacturer manufacturer);
```

Un echipament medical este caracterizat prin furnizorul care îl comercializează, denumirea sa, descrierea ce conține informații despre utilizarea sa, prețul unitar, stocul de care dispune producătorul la momentul respectiv și timpul de execuție, exprimat în număr de zile.

Există posibilitatea de a se filtra lista de echipamente medicale doar pentru un anumit furnizor, care este transmis ca parametru al metodei, în caz contrar întorcându-se toate produsele care pot fi rezervate / comandate prin intermediul agentului de intermediere, de la producătorii înregistrați.

- 3) rezervarea unui echipament medical, într-o anumită cantitate

```
int makeReservation(string manufacturer, string medicalSupply,  
int quantity);
```

Metoda va furniza un număr de înregistrare prin intermediul căruia aplicația corespunzătoare lanțului de farmacii va putea să realizeze și alte operații.

La apelul acestei metode, agentul de intermediere va apela metoda corespunzătoare producătorului indicat, obținând oferta pentru echipamentul medical solicitat.

În cazul în care producătorul sau echipamentul medical nu există, se va întoarce valoarea -1.

- 4) anularea rezervării unui echipament medical

```
boolean cancelReservation(int reservationId);
```

În cazul în care anularea rezervării reușește, metoda va întoarce rezultatul true, altfel (inclusiv în situația când se furnizează un număr de înregistrare inexistent sau care reprezintă o rezervare care a fost procesată deja) rezultatul va fi false.

- 5) obținerea stării unei rezervări pentru un echipament medical

```
Offer getReservationStatus(int reservationId);
```

O ofertă corespunzătoare unei rezervări constă în data calendaristică la care produsele pot fi livrate, respectiv prețul care trebuie plătit pentru cantitatea solicitată.

În situația în care metoda este apelată cu un număr de înregistrare invalid (inexistent, corespunzător unei rezervări procesate), rezultatul va fi null.

- 6) comandarea unui echipament medical, pentru care s-a realizat o rezervare

```
boolean makeOrder(int reservationId);
```

Prin intermediul acestei metode, se plasează o comandă către furnizorul respectiv, agentul de intermediere apelând metoda corespunzătoare producătorului în cauză.

Rezultatul este **true** în situația în care cantitatea disponibilă pe stocul producătorului este suficientă și **false** în caz contrar sau dacă numărul de înregistrare este inexistent / corespunde unei rezervări care a fost procesată (pentru care a fost emisă deja comanda).

7) anularea comenzii unui echipament medical

```
boolean cancelOrder(int reservationId);
```

Această metodă semnalează faptul că lanțul de policlinici nu dorește achiziționarea produsului aferent numărului de înregistrare, anulând atât rezervarea cât și comanda.

Metoda întoarce **true** în cazul în care operația a reușit și **false** în caz contrar (dacă numărul de înregistrare este inexistent sau comanda nu a fost realizată).

8) obținerea stării unei comenzi pentru un echipament medical

```
int getOrderStatus(int reservationId);
```

Rezultatele pe care le poate întoarce această metodă sunt:

- **INVALID\_REGISTRATION\_ID** – pentru situația în care a fost furnizat un număr de înregistrare inexistent sau pentru care nu s-a realizat o comandă;
- **ORDER\_PLACED** – în cazul în care comanda a fost transmisă la furnizor pentru numărul de înregistrare transmis;
- **ORDER\_CANCELED** – dacă comanda aferentă numărului de înregistrare a fost anulată.

**Producătorul unui echipament medical** poate fi interogat de agentul de comunicare, furnizând denumirea și descrierea sa, produsele medicale pe care le comercializează, realizând rezervarea unui produs și anularea rezervării, comanda unui produs și anularea comenzii (pentru o rezervare anterioară).

```
string getName();  
string getDescription();  
MedicalSupply[] getMedicalSupplies();  
Offer makeReservation(int reservationId, string medicalSupplyName,  
                      int medicalSupplyQuantity);  
boolean cancelReservation(int reservationId);  
boolean makeOrder(int reservationId);  
boolean cancelOrder(int reservationId);
```

În momentul în care se realizează rezervarea unui produs, cantitatea solicitată urmează a fi retrasă din stoc, pentru o perioadă de timp standard, interval în care aceasta poate fi anulată, respectiv se poate plasa o comandă.

Dacă stocul corespunzător produsului nu este suficient pentru a satisface solicitarea, oferta generată va conține un termen de livrare care va reflecta timpul de execuție pentru echipamentul medical în cauză. Într-o astfel de situație, nu se va putea transmite o comandă aferentă rezervării până în momentul când întregul lot de produse solicitat nu va fi disponibil, adică până la termenul de livrare transmis.

Anularea explicită a unei rezervări implică marcarea produselor respective ca fiind disponibile, pentru numărul de înregistrare respectiv nemaiputând fi realizate nici un fel de alte operații în continuare.

Dacă produsele corespunzătoare unei rezervări nu sunt comandate în perioada de timp standard, rezervarea respectivă va fi considerată ca fiind anulată implicit.

O comandă se poate plasa doar pentru acele echipamente medicale care au fost rezervate în prealabil și pentru care întreaga cantitate solicitată se găsește în stoc.

O comandă poate fi anulată tot doar în cadrul unei perioade de timp standard de la momentul în care a fost plasată, ulterior această operație fiind imposibilă. Anularea se extinde și asupra rezervării corespunzătoare, nemaiputând fi realizate nici un fel de operații ulterioare.

Nu se va lua în considerare posibilitatea de expirare a dispozitivelor medicale furnizate de producători.

De asemenea, furnizorul de echipamente medicale poate invoca metode oferite de agentul de intermediere pentru înregistrarea sa în contextul acesteia, respectiv pentru deînregistrarea sa. Disponibilitatea metodelor pe care le implementează este condiționată și de starea sa în cadrul componentei de intermediere (înregistrat / neînregistrat), nu doar de accesibilitatea sa prin intermediul rețelei de calculatoare (starea componentei de tip producător este pornit / oprit).

```
int registerManufacturer(string name, string description);  
boolean unregisterManufacturer(int registrationId);
```

Metoda `registerManufacturer` furnizează denumirea și descrierea unui producător și întoarce numărul de înregistrare din cadrul agentului de intermediere. În cazul în care există un producător având aceeași denumire și descriere, se va transmite numărul de înregistrare deținut de acesta, modificând eventual starea sa (din neînregistrat în înregistrat). Componenta de intermediere trebuie să asigure totodată unicitatea numerelor de înregistrare (nu pot exista producători diferiți cu același număr de înregistrare alocat).

Metoda `unregisterManufacturer` marchează indisponibilitatea furnizorului pentru invocarea funcționalității. Aceasta întoarce `true` dacă starea componentei respective a fost modificată corespunzător și `false` în situația în care numărul de înregistrare este inexistent sau dacă starea producătorului nu a putut fi schimbată / nu a fost necesar să fie schimbată.

**Casa de asigurări de sănătate** reține informații cu privire la contractele medicilor (încheiate pe o anumită perioadă de timp), starea pacienților (asigurați sau neasigurați), plafonul pentru decontarea de servicii medicale în cursul unui an calendaristic, investigațiile care fac obiectul decontării (cu specificarea eventuală a unui procent plătit din prețul total), costurile suportate în trecut pentru decontarea anumitor proceduri.

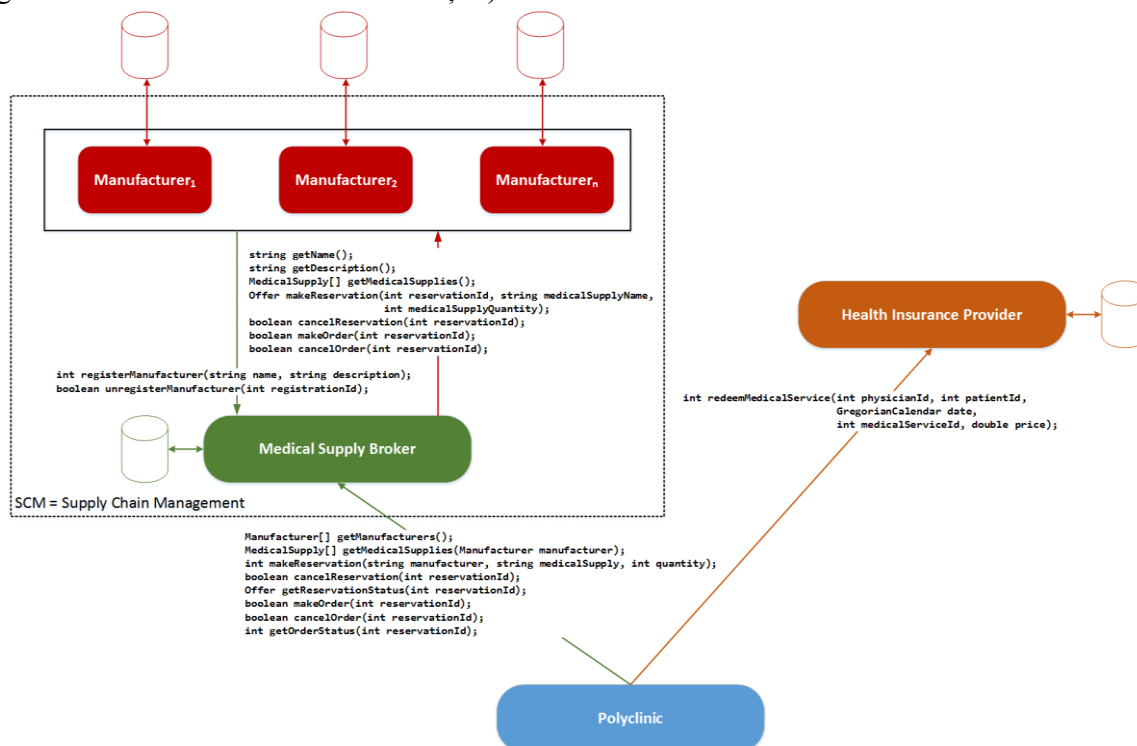
Pe baza lor, lanțul de policlinici poate solicita plata din fondurile casei de asigurări, pentru un serviciu medical realizat de un medic, pentru un anumit pacient, la o anumită dată, având un anumit cost. Datele sunt prelucrate, urmând ca cererea de decontare să fie aprobată sau nu.

```
int redeemMedicalService(int physicianId, int patientId,  
                        GregorianCalendar date,  
                        int medicalServiceId, double price);
```

Rezultatul întors de această metodă conține și justificarea motivului pentru care solicitarea nu a fost aprobată:

- **PHYSICIAN\_NOT\_IN\_CONTRACTUAL\_RELATIONSHIP** – dacă medicul care a furnizat serviciul medical nu se află în relații contractuale cu casa de asigurări de sănătate (verificările se realizează raportat la data calendaristică la care a fost efectuat respectivul serviciu medical);
- **PATIENT\_NOT\_INSURED** – dacă pacientul care a beneficiat de serviciul medical nu deține statutul de asigurat;
- **UNSUPPORTED\_MEDICAL\_SERVICE** – serviciul medical nu face obiectul decontării de casa de asigurări de sănătate;
- **FOUND\_EXCEEDED** – plafonul stabilit de casa de asigurări ar fi depășit prin decontarea serviciului medical respectiv;
- **INFORMATION\_NOT\_AVAILABLE** – nu pot fi obținute informații la nivelul componentei corespunzătoare casei de asigurări sau s-a produs un alt tip de eroare (medic / pacient inexistenți, dată calendaristică invalidă – perioada care a trecut de la furnizarea serviciului medical până în prezent este prea mare sau data calendaristică se găsește în viitor, serviciu medical nerecunoscut de casa de asigurări de sănătate, preț negativ).

Arhitectura sistemului informatic va fi prin urmare formată din patru componente, dintre care două au comportament de server (furnizor de echipamente medicale, casa de asigurări de sănătate), una are comportament de client (lanțul de policlinici) și una are comportament mixt, atât de client cât și de server, în funcție de entitatea la care se raportează (agentul de intermediere al comunicației).



Arhitectura de nivel înalt a sistemului informatic

### Precizări suplimentare

În cazul agentului de intermediere a comunicației cu furnizorii de produse medicale, care expune funcționalități de tip server atât către producători cât și către lanțul de policlinici, trebuie să se asigure un acces corespunzător la servicii, astfel încât fiecare componentă de tip client să nu poată invoca decât metodele care le privesc, fără să cunoască informații despre celelalte funcții pe care le implementează.

Persistența informațiilor la nivelul componentelor de tip server va fi asigurată prin stocarea acestora în baze de date dedicate. În cadrul interacțiunii cu acestea, operațiile trebuie să se realizeze atomic (în cadrul unor tranzacții).

Definițiile metodelor din cadrul enunțului au fost descrise în pseudocod. Acestea pot fi modificate la nivelul parametrilor și valorilor întoarse, cu menținerea funcționalității. În cazul în care se optează pentru o anumită codificare a semnificației parametrilor / valorilor întoarse ale metodelor, acestea trebuie codificate.

În cazul metodelor `makeReservation`, `cancelReservation`, `makeOrder`, `cancelOrder`, implementarea în cadrul agentului de intermediere a comunicației cu furnizorii implică apelarea metodelor omonime din cadrul producătorului corespunzător echipamentului medical pentru care s-a realizat rezervarea / comanda.

Pentru realizarea cerințelor, poate fi necesară adaptarea bazei de date proiectată în cadrul etapei anterioare.

Anumite date care controlează modul de funcționare al componentelor (perioadele de menținere a unei rezervări sau în care poate fi anulată o comandă, plafonul de decontare a serviciilor medicale) pot fi definite în codul sursă drept constante, pot fi stocate în bazele de date corespunzătoare componentelor respective sau pot fi transmise în linia de comandă.

Orice specificație care nu este menționată mai sus reprezintă decizie de implementare. Puteți considera orice simplificare în condițiile în care enunțul nu precizează altfel.

**Pentru ca tema să fie punctată, trebuie incluse scripturi de lansare în execuție a componentelor în care să fie implicate minim două componente de tip producător și câte o componentă din celelalte tipuri.**

**Barem de corectare și notare**

| Punctaj | Criterii de acordare  |
|---------|---|
| 1 p     | proiectarea metodelor corespunzătoare componentelor de tip server <ul style="list-style-type: none"><li>• componenta producător: 25%</li><li>• componenta agent de intermediere a comunicației cu furnizorii: 65%</li><li>• componenta casa de asigurări de sănătate: 10%</li></ul>   |
| 1 p     | reproiectarea bazei de date astfel încât să corespundă noilor funcționalități <ul style="list-style-type: none"><li>• componenta producător: 40%</li><li>• componenta agent de intermediere a comunicației cu furnizorii: 40%</li><li>• componenta casa de asigurări de sănătate: 20%</li></ul>   |
| 2 p     | implementarea metodelor corespunzătoare componenteii producător <ul style="list-style-type: none"><li>• getName: 2,50%</li><li>• getDescription: 2,50%</li><li>• getMedicalSupplies: 15%</li><li>• makeReservation: 35%</li><li>• cancelReservation: 15%</li><li>• makeOrder: 15%</li><li>• cancelOrder: 15%</li></ul>  |
| 2 p     | implementarea metodelor din agentul de intermediere a comunicației cu furnizorii <ul style="list-style-type: none"><li>• getManufacturers: 10%</li><li>• getMedicalSupplies: 15%</li><li>• makeReservation: 30%</li><li>• cancelReservation: 10%</li><li>• getReservationStatus: 10%</li><li>• makeOrder: 10%</li><li>• cancelOrder: 10%</li><li>• getOrderStatus: 5%</li></ul> |
| 1 p     | implementarea funcționalității casei de asigurări de sănătate (redeemMedicalService)  |
| 1 p     | apelarea metodelor disponibile la distanță din cadrul componentelor de tip client <ul style="list-style-type: none"><li>• componenta lanț de policlinici: 55%</li><li>• componenta agent de intermediere a comunicației cu furnizorii: 30%</li><li>• componenta producător: 15%</li></ul>   |
| 1 p     | modularizare <ul style="list-style-type: none"><li>• structura aplicației: 50%</li><li>• lizibilitatea codului: 20%</li><li>• comentarii, README: 30%</li></ul>   |

**BONUS.** Se pot obține punctaje suplimentare, astfel:

- 0,25p – predarea și prezentarea temei până la 11.12.2014, ora 23:55;
- 0,35p – reproiectarea metodelor la distanță makeReservation, cancelReservation, makeOrder, cancelOrder astfel încât să permită rezervarea / anularea rezervării, comanda / anularea comenzii pentru mai multe comenzi concomitent, generând câte un număr de înregistrare pentru fiecare produs distinct, astfel încât să se poată realiza ulterior mai rapid comanda acelor echipamente medicale care sunt disponibile imediat;
- 0,40p – definirea mai multor pachete de către casa de asigurări de sănătate, de care pot beneficia pacienții, fiecare conținând un anumit număr de consultații și plafoane pentru analize medicale și investigații paraclinice în funcție de care să se ia decizia de acceptare sau respingere a solicitării de decontare a unui serviciu oferit.

### Condiții de realizare și predare

Tema va fi realizată individual și va fi prezentată în cadrul laboratorului, până la sfârșitul semestrului. Neprezentarea temei de casă atrage după sine nepunctarea acesteia. În cadrul prezentării veți specifica succint funcționalitățile pe care le-ați dezvoltat și veți răspunde la întrebări cu privire la diferite soluții adoptate în rezolvarea problemelor întâlnite.

Tema va trebui încărcată pe site-ul [cs.curs.pub.ro](http://cs.curs.pub.ro) sub forma unei arhive de tip .zip (având denumirea **Grupa34XCX\_NumePrenume\_Tema2.zip**) care să conțină script-ul pentru crearea și popularea tabelor din bazele de date (numele bazelor de date fiind prefixate de **Grupa34XCX\_NumePrenume**), sursele componentelor cu scenarii de utilizare (scripturi ce conțin lansări în execuție ale componentelor de tip server și client cu diferite operații, specificând argumentele necesare<sup>3</sup>) și un fișier README în care să argumentați tehnologia pe care ați ales-o și deciziile de implementare. Prezentarea se poate realiza numai după ce tema a fost încărcată pe site.

Încărcarea pe site nu este redundantă, temele vor fi comparate prin aplicații specializate pentru a se depista eventualele fraude. În această situație, întreg punctajul pe parcursul semestrului va fi anulat, studenții implicați (atât originalul, cât și copia / copiile) fiind obligați să repete disciplina – cu taxă – în anul universitar următor.

---

<sup>3</sup> În cazul tehnologiei RMI vor trebui indicate proprietățile Java `java.rmi.server.codebase`, `java.rmi.server.hostname`, `java.security.policy`, incluzându-se și fișierele ce conțin politica de securitate. Pentru CORBA, se vor preciza parametrii liniei de comandă `ORBInitialHost` / `ORBInitialPort`. Dacă optați pentru JAX-WS, vor trebui indicate locațiile la care vor fi publicate serviciile web.