

Proiectarea unui sistem automat de cumpărături on-line

Descrierea sistemului software

Sistemul software prezent trebuie să fie capabil să analizeze o fișă cu produse primită la intrare, pe baza căreia să selecteze automat din magazinele online partenere produsele corespunzătoare și să le adauge în coșul de cumpărături. Acesta este menit să ajute marile companii, să realizeze într-un mod eficient cumpărăturile necesare. Metoda de a cumpara angro on-line solicită mult timp, astfel cu acest sistem totul se poate realiza în câteva secunde.

Cerințe funcționale

Fișa încărcată de utilizator poate să fie sub mai multe forme:

- Document .txt
- Fișier .xml
- Încărcată într-o bază de date sub formă de tabel.

Cu toate acestea ulterior trebuie ținut cont de extinderea domeniului de procesare.

Aplicația trebuie să suporte conectarea la cât mai multe servere concomitent. Conectarea unui server nou la aplicație sau a excluderii altuia trebuie să se realizeze cu ușurință.

Există mai multe moduri în care aplicația poate rula. Modul manual, în care o data aduse produsele recomandate de aplicație în coșul de cumpărături, acestea pot fi acceptate sau nu de către utilizator. Modul automat, în care produsele intră direct în coșul de cumpărături, iar clientul trebuie doar să lanseze tranzacția.

De asemenea există posibilitatea ca utilizatorul să își aleaga top 3 site-uri de cumpărături preferate din cele partenere. Clientul inițial deține o croitorie, iar majoritatea site-urilor partenere vor fi destinate cumpărăturilor de materiale textile și auxiliare.

Cerințe non-funcționale

Bazele de date sunt stocate în Azure, astfel se recomandă utilizarea de limbaje .Net pentru a facilita comunicarea cu acestea. Conexiunea la internet este vitală pentru a avea acces la datele din Cloud.

Trebuie să existe modularitate, în sensul în care, aplicația va servi în viitor și altor clienți, iar aceștia vor dori produse din alte sectoare. Conectarea la servere noi cu ușurință poate constitui un avantaj pentru a vinde aplicația cât mai ușor. Trebuie avut în vedere faptul că, cu cât există mai multe servere conectate, aplicația va întâmpina tot mai multe dificultăți, care trebuie eliminate.

Existența aplicației atât pe PC cât și pe telefonul mobil.

Descompunerea în componente

1. Interfața cu clientul

Această parte a programului trebuie să constituie mediul de comunicare dintre logica aplicației și client. Interfața are caracteristica de a fi ușor utilizată de către orice persoană și să fie încadrată într-un mediu plăcut grafic.

Interfața cu clientul va fi o pagină web, care trebuie să permită următoarele acțiuni: încărcare fișă de cumpărături, descărcare fișă compusă de aplicație, selectarea modului de rulare a aplicației (manual/automat), selectarea tipului de fișă încărcat, selectarea firmelor partenere preferate, autentificarea clientului, navigarea printre alte produse similare și a alege o alta alternativă, finalizare comandă.

2. Interpretor – interpretarea fișei

Componenta aceasta va fi împărțită în subcomponente, fiecare interpretând fișa după tipul de document (text, xml, tabele din baza de date) încărcat prin interfața cu clientul. Această componentă va fi în același program cu interfața, legătura dintre ele se face cu ajutorul metodelor. Fiecare subtip de document este activat în funcție de fișierul încărcat prin metode. Aplicația va parcurge rând cu rând înregistrările și va detecta domeniul produsului, numele acestuia și cantitatea (kg, g, m, l, buc).

3. Distribuitor domeniu- de produse

Componentele din interpretarea fișei vor avea ca destinație această parte a aplicației, unde fiecare produs trebuie clasat într-un anumit domeniu pentru a căuta în site-urile cu domenii specifice produselor. Această componentă va fi apelată la terminarea rulării componentei de interpretare a fișei de cumpărături.

4. Distribuitor servere- pe domenii

Pe baza rezultatelor furnizate de componenta anterioară și a opțiunii de selectare a top 3 site-uri preferate de client se compune o listă de prioritate în căutarea de produse. Intern pot exista și alte filtre prin care se realizează alegerea serverelor de furnizori, cum ar fi site-urile cu cele mai mici prețuri. Dacă o componentă din cadrul compunerii de mesaj răspuns, întoarce un produs ca fiind epuizat, se va alege un alt server de căutare.

5. Interogare servere

Această componentă este divizată în câte servere disponibile sunt, iar componentele vor lucra în paralel. Fiecare dintre acestea trebuie să returneze câte un rezultat pentru fiecare produs asignat în componenta de alegere a serverelor de căutare. Această componentă va executa interogări de tip SQL pe tabele stocate în Azure.

6. Compunerea fișei răspuns

După ce toate produsele au fost încadrate ca fiind disponibile de către componenta 4, se va realiza un apel către această componentă pentru a alătura răspunsul final și al transmite mai departe interfeței de utilizator.

7.Componenta de validare a fișei de cumpărături

Aceasta va avea rolul de a mai face o verificare suplimentară a produselor disponibile, în momentul în care clientul a acceptat tranzacția. Va fi apelată de către partea de interfață a aplicației și va returna acesteia un mesaj pozitiv sau negativ, în funcție ultima interogare a bazelor de date.

Prezentarea sistemului software

1.Perspectivă logică

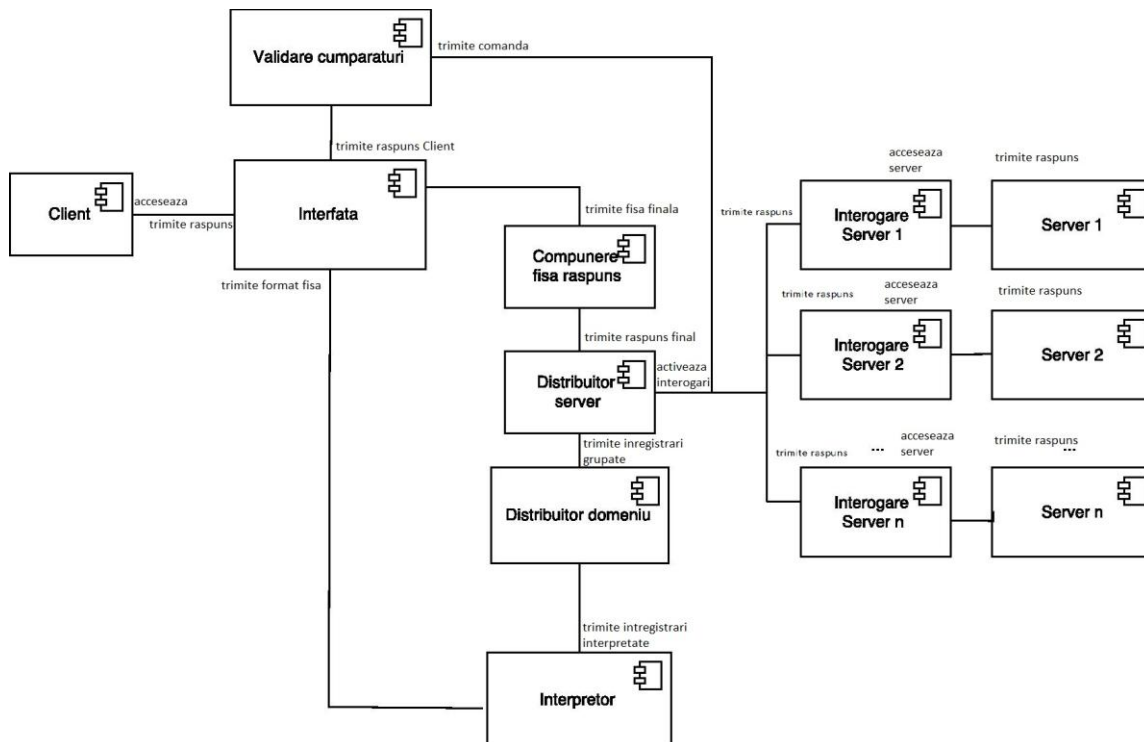


Fig 1 Diagramă de componente

În Fig 1 s-a realizat o diagramă de componente, alcătuită din principalele structuri detaliate la punctul anterior, "Descompunerea în componente". Aceste componente au fost astfel împărțite pentru a modifica (extinde) cu ușurință site-urile partenere, tipul de fișă returnată la final, algoritmi de alegere a serverelor și domeniile de distribuire a produselor. Componentele trebuie astfel construite încât să permită adăugarea unui modul de securitate ulterior după ce se vor încheia contractele cu site-urile partenere. Acest modul trebuie localizat între componenta Distribuitor server și Interogare Server(1 .. n).

2.Perspectivă de proces

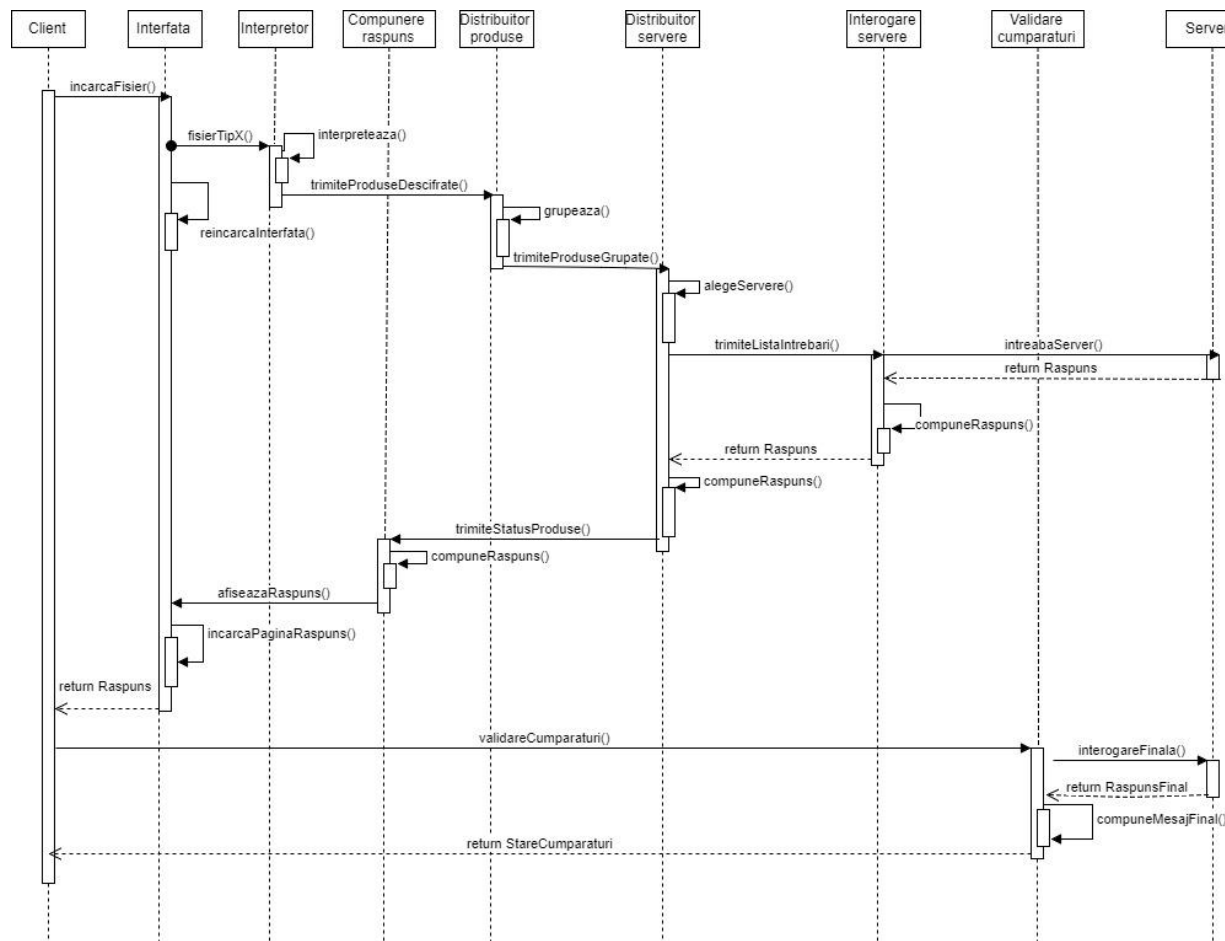


Fig 2. Diagramă de secvență

În Fig 2 sunt ilustrați următorii pași descriși în cuvinte posibili în sistemul software.

Clientul trimite o cerere prin interfață.

Din interfață se apelează component Interpretor, în funcție de tipul fișei încărcat.

Interpretorul descifrează fișa cu cumpărături și trimite înregistrările Distribuitorului de domeniu.

Distribuitorul de domeniu clasează produsele pe domenii și le trimite împachetate

Distribuitorului de servere.

Distribuitorul de servere alege serverele cu care va comunica.

Interogatorul de server este activat în funcție de Distribuitorul de servere, care va trimite o cerere către server și va reîntoarce un răspuns formulat de server.

Distribuitorul de server este anunțat de fiecare interogator, dacă produsul a fost localizat și în funcție de aceasta se ia o decizie.

Compunerea fișei răspuns este activată de către Distribuitorul de servere, numai atunci când toate produsele au fost marcate cu un status.

Interfața afișează un răspuns clientului.

Clientul din interfață accepta/rejectează fișa de cumpărături.

Dacă fișa a fost acceptată prin modulul de Validare se lansează comanda.

Indicatori de calitate

1. Performanța

Timpul de răspuns al aplicației crește mult din cauza timpilor de conectare la servere. Dacă aceasta ar fi făcută pe threaduri diferite, timpul final ar fi vizibil redus.

Puterea de procesare trebuie garantată mai ales în sezoanele de vârf a clienților, când se vor face mai multe accesări ale aplicației și astfel trebuie garantat faptul că va funcționa în toate momentele anului.

2. Scalabilitatea

Aplicația este într-adevăr validată, dacă răspunde bine atât unei fișe scurte de cumpărături, cât și unei cerințe vaste sau complicate. Pentru aceasta s-a ales componenta de sortare a produselor de cumpărat pe domenii, astfel printr-o singură conectare la un server se pot bifa mai multe cerințe din lista de cumpărături. Urmând ca acestea să se separe pe threaduri pentru interogarea bazelor de date.

3. Toleranța la modificări

Numărul de site-uri cu produse disponibile este dinamic, iar aplicația curentă va suferi modificări adesea, prin introducerea sau scoaterea unor legături cu bazele de date aferente site-urilor de cumpărături. Astfel, poate cel mai important indicator este cel de a tolera modificări. De asemenea și clienții se pot schimba adesea, iar domeniile de acoperire trebuie mereu să satisfacă toți clienții.

Tehnologii middleware

Cele mai importante tehnologii folosite sunt:

Azure-SQL- pentru conectarea la baza de date și interogarea tabelor

.Net- pentru crearea unei aplicații web care să se compună din algoritmii de alegere a produselor.

Principalele stiluri arhitecturale

1. Broker

Acest stil arhitectural l-am ales pentru procesul ca întreg, deoarece intrările și ieșirile din cadrul aplicației sunt expuse în mai multe tipuri și necesită anumite transformări pentru a scoate informația utilă. Acest pattern este util deoarece separă parte de transformare și rutare a intrărilor respectiv ieșirilor, de parte de procesare (logică a aplicației). Prin această separare aplicația tolerează ușor modificări (adăugare de noi tipuri de intrări și ieșiri din aplicație). Performanța este afectată de ieșiri (servere) deoarece este nevoie de mult timp la conectarea cu acestea, dar pe de altă parte disponibilitatea a mai multor porturi de intrări duce la o utilitate mai mare a aplicației. Acest tip de arhitectură se poate vizualiza în Fig 3.

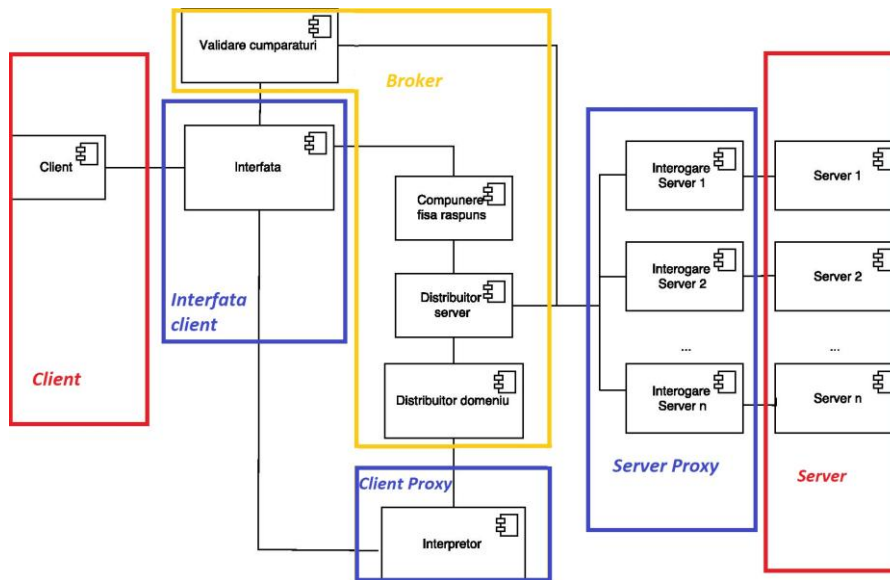


Fig 3 Reprezentarea structurii de tip Broker

2. Client-Server

Acest tip poate fi regăsit o dată între Client și Broker care joacă rolul unui Server, deoarece în final furnizează o informație. În al doilea rând se regăsește între Broker și Servere, deoarece Brokerul trimite cereri către Servere, de unde primește și un răspuns. Aceste două cazuri se pot vedea în Fig 4. S-a ales acest mod deoarece separă foarte bine partea de aplicație vizibilă clientului de partea de logică, respectiv cea de bază de date.

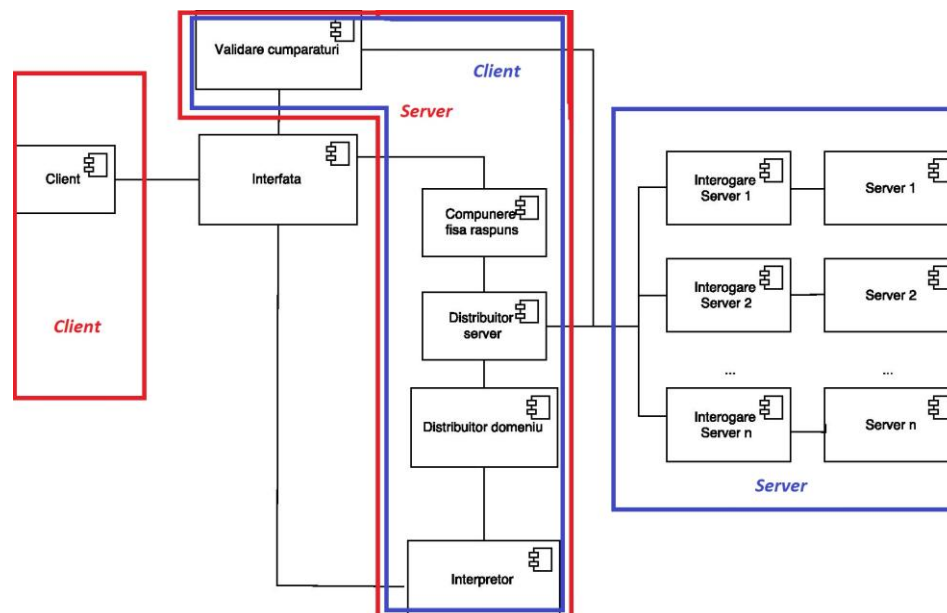


Fig 4 Reprezentarea structurii Client-Server

Validarea arhitecturii

Arhitectura este validată, modul în care datele curg prin aplicație este unul bine determinat. Acesta se poate vizualiza în diagrama de proces(Fig 2).

Oricare ar fi intrările brokerul știe să le interpreteze și să formuleze o cerere într-un limbaj unic specific serverului, acesta întoarce un răspuns în același limbaj, iar brokerul descifrează răspunsul și îl afișează în final clientului într-un mod inteligibil. (Cursul acțiunii se poate vizualiza în Fig 5)



Fig 5 Cursul datelor prin arhitectură