

# **Sistem software:** Sistem de achiziționare a bunurilor în mediul online

Prezentarea cerințelor funcționale și non-funcționale și identificarea acelor cerințe care influențează arhitectura:

Sistemul ales cuprinde trei tipuri de utilizatori: operatorii care gestionează comenzile, angajații din depozit care le trimit și clienții.

## 1. Cerințe funcționale:

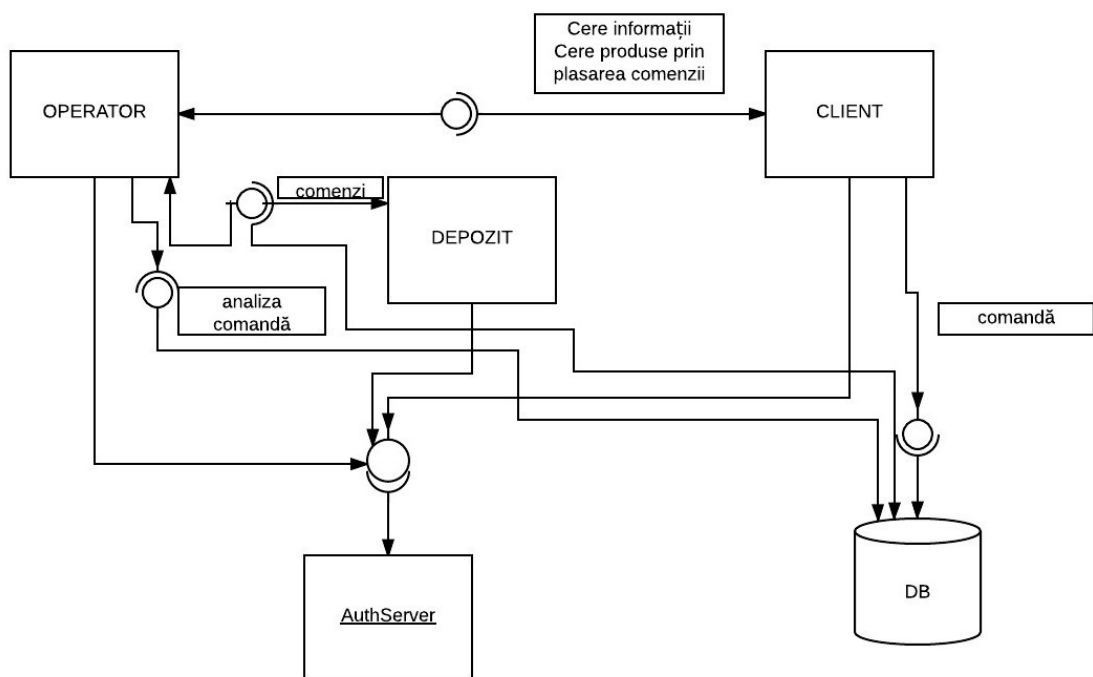
- Operatori:
  - logare
  - vizualizarea comenzilor și a datelor clienților
  - trimiterea datelor comenzilor către depozit
  - transmiterea mesajelor către clienți
  - verificarea stocului
- Angajați din depozit:
  - primirea datelor de la operatori
  - căutarea produselor comandate
  - scanare produs pentru scăderea din stoc
- Client
  - logare
  - cautare produse și vizualizarea informațiilor
  - trimitere mesaje dacă are nevoie de informații suplimentare
  - plasare comandă
  - plata online

## 1. Cerințe non-funcționale

- informațiile legate de comenzi vor fi primite de operatori, analizate, iar apoi trimise la depozit
- la logare se va face diferența între drepturile unui angajat și cele ale unui client

## Descompunerea în componente

În primul rând, apare componenta CLIENT care poate achiziționa bunuri prin intermediul unei aplicații web. Componenta OPERATOR va primi informațiile despre comenzi într-o aplicație web, informații ce vor fi stocate în componenta BAZA DE DATE. DEPOZIT este componenta care pune la dispoziție firmei de curierat bunurile comandate de client pe baza datelor primite de la operatori. Pentru acest lucru va fi folosită o aplicație mobile pentru a le fi mai ușor angajaților să caute produsele. Pentru a fi mai evident rolul fiecărei componente(client, depozit, operator) se va folosi un server de autentificare AuthServer.

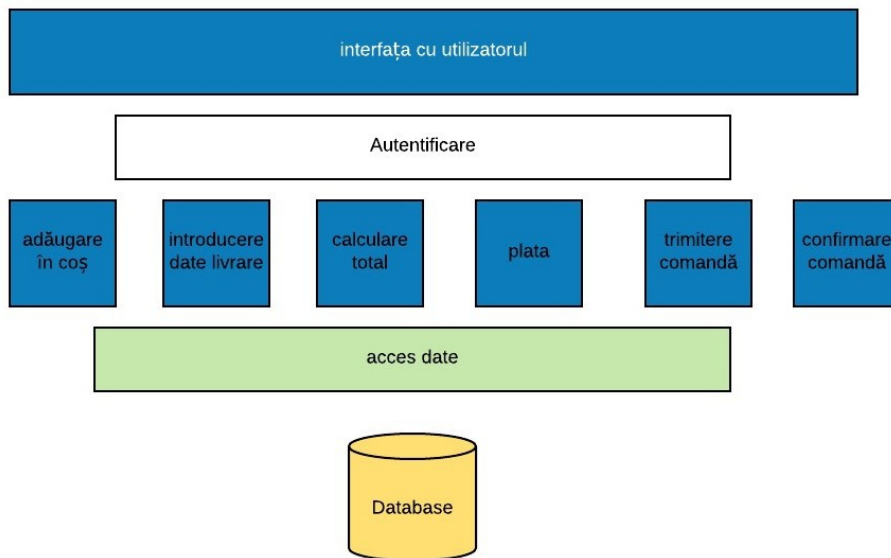


Prin intermediul server-ului de autentificare se va face diferența între principalele componente. Datele înregistrate la crearea contului clientului, dar și detaliile legate de comenzi vor fi stocate în baza de date. În cazul în care clientul are nevoie de informații suplimentare legate de produsele pe care dorește să le achiziționeze, acesta poate contacta persoanele care se ocupă de magazinul online printr-un mesaj pe aplicația web. Dacă posibilul client dorește să cumpere ceva de pe site, va plasa comenzile (adaugă produsele în coșul de cumpărături) completând mai multe câmpuri de la produsul dorit până la adresa de livrare și numărul de telefon.

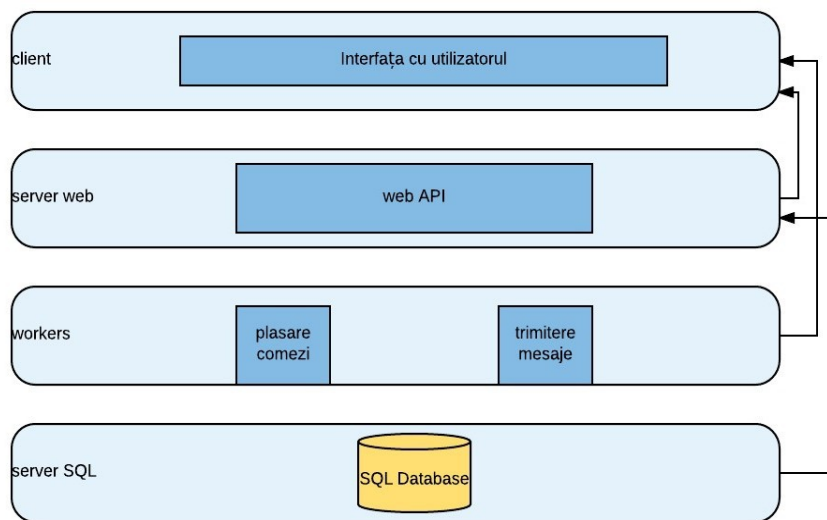
Operatorul preia comenzile din baza de date și în cazul în care nu mai există pe stoc anumite produse va lua legătura cu clientul pentru a stabili detalii legate de comanda respectivă. Clientul are dreptul să știe că va primi numai anumite produse și nu pe toate cele comandate până când vor reveni în depozit cele care nu sunt disponibile. Dacă totul este în regulă, va da mai departe comanda către depozit. Transmiterea comenzilor către depozit este dependentă de baza de date.

Angajații din depozit vor căuta produsele și le vor trimite clienților printr-o firmă de curierat. Nu este reprezentată drept o componentă, deoarece este considerat un serviciu extern sistemului. Tot cei aflați în depozit vor scana produsele pentru scăderea numărului de produse aflate în stoc și un operator va realiza factura.

#### Perspectiva logică



#### Perspectiva de proces



## Indicatori de calitate

### Performanța – puterea de procesare

Pentru sistemul software ales este foarte important volumul de procesări realizat în unitatea de timp, deoarece un magazin online este accesat de foarte multe persoane în același timp, iar fiecare dintre aceste persoane vor realiza mai multe activități în același timp. De exemplu: caută mai multe produse în tab-uri diferite, face comparații, adaugă produse în coș.

### Securitatea – autentificare

Securitatea este foarte importantă pentru că utilizatorii dau date personale la plasare comenzilor și dacă vor să plătească online.

### Disponibilitatea – recuperarea

În cazul în care apar defecțiuni ale server-ului de exemplu, atât angajații cât și clienții trebuie să poată folosi în continuare aplicația. Dacă nu se rezolvă defecțiunile sau au loc foarte des, există riscul ca anumiți clienți să-și îndrepte atenția către alte aplicații similare.

## Tehnologii middleware

MOM(Message Oriented Middleware) – oferă o comunicare asincronă între client și server prin punerea într-o coadă temporară când unul dintre ei nu este functional astfel că nu trebuie să aștepte unul după altul

### Model architectural

Model-View-Controller – deoarece logica de business este izolată de interfața cu utilizatorul, iar modificările unui nivel nu vor afecta și restul

Model- manipulare operațiunilor logice

View – interfața cu utilizatorul

Controller – controlarea accesului la aplicație

Stilul arhitectural bazat pe evenimente consider că este potrivit pentru acest sistem, deoarece timpul de răspuns la acțiunile utilizatorilor va influența semnificativ părerea și modul de utilizare.

## **Scenarii de validare**

Pentru validarea arhitecturii sistemului se va crea un prototip proof-of-concept pentru a verifica dacă arhitectura îndeplinește cerințele.

Vor trebui realizate scenarii pentru fiecare componenta, deoarece toate cele 3 tipuri de utilizatori trebuie să folosească cu succes aplicația.

Pentru acest lucru se pot realiza teste unitare și analizat comportamentul utilizatorilor. De exemplu, când un client plasează comanda, trebuie inspectat dacă datele ajung cum trebuie în baza de date și dacă operatorii sunt notificați. Pe de altă parte, trebuie să se testeze dacă se realizează legătura dintre aplicațiile aflate pentru dispozitivele din depozit(mobile) și cele ale operatorilor(web) pentru ca ambele tipuri de angajați să aibă date legate de produse în timp real.