

POLITEHNICA BUCHAREST NATIONAL UNIVERSITY FOR SCIENCE AND TECHNOLOGY

Faculty of Automatic Control and Computers
Computer Science and Engineering Department



Sistem de Gestiune pentru Servicii de Curierat Proiect Baze de Date

VĂDANA IOAN-GRIGORE

2023

Cuprins

1. [Descrierea Cerințelor](#)
2. [Etapă de Proiectare](#)
3. [Descrierea Tabelelor și Relațiilor](#)
4. [Constrângerile de Integritate](#)
5. [Funcționarea Aplicației](#)
6. [Interogările Simple](#)
7. [Interogările Complexe](#)

Descrierea Cerințelor

Sistemul de gestionare a serviciilor de curierat este proiectat pentru a facilita urmărirea eficientă și gestionarea operațiunilor de curierat. Scopul principal este de a oferi utilizatorilor o metodă sistematică de a administra clienți, curieri, pachete, și procesul de livrare. În plus, sistemul oferă funcționalități de generare a facturilor și de urmărire a evaluărilor, îmbunătățind astfel comunicarea cu clienții și serviciul oferit.

Aplicația este concepută pentru a oferi o soluție de gestionare a serviciilor de curierat, permițând utilizatorilor să:

- Aduge, actualizeze și vizualizeze informații despre clienți, curieri, pachete și livrări.
- Urmărească locațiile pachetelor și actualizeze pozițiile acestora.
- Gestioneze detaliile de facturare și generarea facturilor.
- Vizualizeze și gestioneze evaluările clienților și curierilor.
- Definăscă și gestioneze diferite tipuri de servicii de curierat.
- Genereze rapoarte detaliate pentru analiza operațiunilor.

Etapa de Proiectare

Proiectarea include crearea unei baze de date relaționale cu următoarele tabele, fiecare corespunzând unei entități din aplicație:

1. **Customers (Clienți)** - pentru stocarea datelor despre clienți.
2. **Couriers (Curieri)** - pentru înregistrarea informațiilor despre curieri.
3. **Packages (Pachete)** - pentru detaliile legate de pachetele care trebuie livrate.
4. **Deliveries (Livrări)** - pentru urmărirea statusului și detaliilor livrărilor.
5. **Locations (Locații)** - pentru urmărirea istoricului locațiilor pachetelor.
6. **Billing (Facturare)** - pentru gestionarea facturării și plăților.
7. **Ratings (Evaluări)** - pentru înregistrarea evaluărilor date de clienți și curierilor.
8. **ServiceTypes (Tipuri de Servicii)** - pentru a defini diferitele servicii oferite de curierat.

Descrierea Tabelelor și a Relațiilor dintre Tabele

Tabele:

1. **dbo.Customers**

- Stochează informații despre clienți, cum ar fi numele, adresa, contactul, emailul și metoda preferată de contact.

2. **dbo.Packages**

- Include detalii despre pachete, cum ar fi descrierea, greutatea, data estimată a livrării, dimensiunile și instrucțiunile speciale.

3. **dbo.Couriers**

- Conține informații despre curieri, inclusiv numele, contactul, tipul de vehicul și statusul disponibilității.

4. **dbo.Deliveries**

- Urmărește detaliile livrărilor, statusul, data estimată și cea reală a livrării.

5. **dbo.Locations**

- Înregistrează locațiile curente ale pachetelor și timestamp-ul acestora.

6. **dbo.Billing**

- Gestionează informațiile de facturare, inclusiv statusul plății și metoda de plată.

7. **dbo.Ratings**

- Permite clienților și curierilor să lase evaluări și comentarii.

8. **dbo.ServiceTypes**

- Definește diferitele tipuri de servicii de curierat disponibile.

Relații:

- **dbo.Packages -> dbo.Customers:** Relație one-to-many (un client poate avea mai multe pachete).
- **dbo.Deliveries -> dbo.Packages:** Relație one-to-one (un pachet are o singură livrare).
- **dbo.Deliveries -> dbo.Couriers:** Relație many-to-one (un curier poate efectua mai multe livrări).
- **dbo.Locations -> dbo.Packages:** Relație one-to-many (un pachet poate avea mai multe înregistrări de locație).

- **dbo.Billing -> dbo.Customers:** Relație one-to-many (un client poate avea mai multe facturi).
- **dbo.Ratings -> dbo.Customers și dbo.Couriers:** Relații many-to-one (evaluări pot fi lăsate de mulți clienți sau pentru mulți curieri).

Constrângerile de Integritate

Constrângeri pe coloane:

- **PK (Primary Key):** Fiecare tabel are o cheie primară, cum ar fi **CustomerID**, **PackageID**, **CourierID**, care sunt unice și not null.
- **FK (Foreign Key):** Există chei străine pentru a menține integritatea relațională, cum ar fi **CustomerID** în tabelul **Packages** care leagă pachetele de clienții specifici.
- **not null:** Anumite câmpuri, precum **CustomerID** în tabelul **Customers**, sunt obligatorii și nu pot fi lăsate goale.

Constrângeri de relație:

- **Chei străine:** Asigură că orice valoare introdusă într-o coloană cheie străină există deja ca o cheie primară în tabelul la care se referă, asigurând integritatea referențială.
- **Unicitatea datelor:** De exemplu, **PK_Customer** sau **PK_Package** asigură că fiecare înregistrare în tabelele respective este unică.

Constrângeri de tip de date:

- **nvarchar, int, float, date, bit:** Aceste tipuri de date specifică natura datelor care pot fi stocate în fiecare coloană și ajută la prevenirea erorilor de date.
- **money:** Folosit în tabelul **Billing** pentru a asigura că valorile introduse respectă formatul monedei.

Alte constrângeri:

- **Constrângeri de domeniu:** De exemplu, un câmp **Rating** poate fi limitat la valori între 1 și 5.
- **Constrângeri de business:** Reguli de business care nu sunt direct impuse prin chei sau tipuri de date, dar sunt implementate în logica aplicației, cum ar fi "un curier nu poate avea mai multe livrări în același timp".

Fiecare tabel servește un scop specific și relațiile dintre ele permit aplicației să opereze logic și eficient.

Funcționarea Aplicației

Sistemul de gestionare pentru servicii de curierat este o aplicație web care servește mai multe funcții esențiale pentru afacerea de curierat. Utilizatorii interacționează cu aplicația prin intermediul unei interfețe grafice bazate pe web, care comunica cu un server backend Flask.

Backend

- **Serverul Flask:** Inițializează aplicația și așteaptă cereri HTTP de la utilizatori.
- **Rute Flask:** Definite în **app.py**, rutele sunt punctele de intrare ale aplicației. Fiecare rută este asociată cu o funcție care procesează cererile și returnează un răspuns. Funcțiile pot realiza operațiuni CRUD pe baza de date și pot returna template-uri HTML populare cu date.
- **Gestiunea Bazei de Date:** Utilizează **db.py** pentru a se conecta la baza de date. Funcția **get_db_connection** este invocată pentru a iniția conexiunea, iar cursorul este folosit pentru a executa interogările.
- **Închiderea Conexiunii:** După fiecare interogare, conexiunea este închisă pentru a preveni scurgeri de memorie și pentru a menține performanța aplicației.

Frontend

- **Template-uri HTML:** Fiecare pagină web este construită folosind un template corespunzător din folderul **templates**. Flask populează aceste template-uri cu date din baza de date înainte de a le trimite ca răspuns la browserul clientului.
- **CSS:** Stilizează paginile web pentru a asigura o experiență utilizator coezivă și atractivă.
- **JavaScript:** Adaugă funcționalități interactive paginilor web, cum ar fi validarea formularului pe partea clientului sau actualizări asincrone folosind AJAX.

Codul sursă furnizat detaliază o aplicație Flask care gestionează adăugarea, actualizarea și ștergerea datelor într-o bază de date pentru un serviciu de curierat. Fiecare secțiune a codului corespunde unei anumite funcționalități ale aplicației, care sunt descrise mai jos:

Rute Flask și Operațiuni CRUD:

1. **Index (/):** Aceasta este pagina de start a aplicației, care returnează **index.html**.
2. **Adăugarea Clientilor (/customers/add):**
 - **GET:** Afișează formularul pentru adăugarea unui nou client.
 - **POST:** Procesează datele trimise de formular și le inserează în tabelul **Customers** al bazei de date.

3. **Actualizarea Clienților (/customer/update/<int:customer_id>):**
 - GET: Afișează formularul pentru actualizarea datelor unui client existent, bazat pe **customer_id**.
 - POST: Actualizează datele clientului în baza de date.
4. **Ștergerea Clienților (/customer/delete/<int:customer_id>):**
 - POST: Șterge un client din baza de date.
5. **Adăugarea Pachetelor (/packages/add):**
 - GET: Afișează formularul pentru adăugarea unui nou pachet.
 - POST: Procesează și inserează detaliile pachetului în tabelul **Packages**.
6. **Actualizarea Pachetelor (/package/update/<int:package_id>):**
 - GET: Afișează formularul pentru actualizarea unui pachet existent.
 - POST: Actualizează detaliile pachetului în baza de date.
7. **Ștergerea Pachetelor (/package/delete/<int:package_id>):**
 - POST: Șterge un pachet din baza de date.
8. **Adăugarea Curierilor (/courier/add):**
 - GET: Afișează formularul pentru adăugarea unui nou curier.
 - POST: Procesează și inserează detaliile curierului în tabelul **Couriers**.
9. **Actualizarea Curierilor (/courier/update/<int:courier_id>):**
 - GET: Afișează formularul pentru actualizarea unui curier existent.
 - POST: Actualizează informațiile curierului în baza de date.
10. **Ștergerea Curierilor (/courier/delete/<int:courier_id>):**
 - POST: Șterge un curier din baza de date.
11. **Adăugarea Livrărilor (/delivery/add):**
 - GET: Afișează formularul pentru adăugarea unei noi livrări.
 - POST: Inserează detaliile livrării în tabelul **Deliveries**.
12. **Actualizarea Livrărilor (/delivery/update/<int:delivery_id>):**
 - GET: Afișează formularul pentru actualizarea unei livrări existente.
 - POST: Actualizează detaliile livrării în baza de date.

13. Ștergerea Livrărilor (/delivery/delete/<int:delivery_id>):

- POST: Șterge o livrare din baza de date.

Interogări Simple: Operațiunile CRUD de mai **sus** sunt exemple de interogări simple, unde se execută acțiuni directe pe un singur tabel, cum ar fi inserarea, actualizarea sau ștergerea unei înregistrări.

Redirecționări:

După fiecare operație POST de succes, utilizatorul este redirecționat la o pagină de vizualizare (de exemplu, **view_customers** sau **view_packages**), ceea ce sugerează că există și alte rute Flask care se ocupă de afișarea datelor.

Interogări simple

Aceste interogări simple utilizează join-uri între tabele pentru a aduce împreună informații din diferite părți ale bazei de date. Ele sunt folosite pentru a popula diferite pagini în aplicația web cu datele necesare. Fiecare rută Flask returnează o pagină HTML, furnizându-i datele obținute din interogări.

descrierea funcționalității fiecărei interogări:

1. **Livrările cu Numele Curierilor și Descrierea Pachetelor:**

- Această interogare selectează ID-ul livrării, numele curierului și descrierea pachetului pentru toate livrările.
- Pagina **deliveries_couriers_packages.html** va afișa aceste informații.

2. **Clienții și Detaliile Pachetelor Lor:**

- Interogarea extrage numele clientului și detaliile pachetelor asociate fiecărui client.
- Rezultatele sunt afișate pe pagina **customers_packages.html**.

3. **Pachetele cu Ultimele Actualizări ale Locației:**

- Selectează ID-ul pachetului, descrierea și ultima locație cunoscută, grupând rezultatele astfel încât să fie afișată ultima actualizare pentru fiecare pachet.
- Aceste date sunt prezentate pe pagina **packages_latest_locations.html**.

4. **Clienții și Suma Totală Facturată:**

- Calculează suma totală facturată pentru fiecare client prin sumarea tuturor facturilor.
- Informațiile sunt disponibile pe pagina **customers_total_billing.html**.

5. **Curierii și Numărul de Livrări Finalizate:**

- Numără câte livrări a completat fiecare curier, dar numai pentru cele care au statusul 'Delivered'.
- Datele sunt afișate pe pagina **couriers_deliveries_count.html**.

6. **Livrările Inclusiv Numele Clienților și Statusul Livrării:**

- Extrage ID-ul livrării, numele clientului și statusul livrării pentru toate livrările.
- Rezultatele sunt vizualizate pe pagina **deliveries_customers_status.html**.

7. **Pachete cu Numele și Detaliile de Contact ale Clienților:**

- Această interogare nou adăugată selectează ID-ul pachetului, descrierea acestuia, numele clientului și detaliile de contact, oferind o vedere de ansamblu a pachetelor și a clienților corespondenți.
- Informațiile sunt listate pe pagina **packages_with_customers.html**.

Fiecare funcție Flask se conectează la baza de date, execută interogarea SQL și închide conexiunea după finalizare, apoi trimite rezultatele interogării către un template HTML pentru afișare.

Interogările complexe

Interogările complexe efectuează operațiuni mai avansate, cum ar fi subinterogări, funcții de agregare și clauze **HAVING**. Explicarea detaliată a fiecărei interogări:

1. **Clienți cu Facturare Peste Media:**

- Află clienții ai căror sume facturate sunt peste media tuturor facturilor.
- Folosește o subinterogare pentru a calcula media (**AVG**) facturilor și apoi filtrează înregistrările care au o valoare mai mare decât această medie.
- Rezultatele sunt afișate prin intermediul paginii **customers_billing_above_average.html**.

2. Curieri cu Cel Mai Mare Număr de Livrări în Luna Curentă:

- Identifică curierii care au efectuat cele mai multe livrări în luna curentă.
- Utilizează funcții de data **MONTH** și **YEAR** pentru a restricționa interogarea la perioada curentă.
- Interogarea subordonată calculează numărul maxim de livrări efectuate de un curier în luna curentă.
- Clauza **HAVING** este folosită pentru a selecta doar curierii care ating acest număr maxim de livrări.
- Rezultatele sunt prezentate pe pagina **top_couriers_this_month.html**.

3. Clienți cu Livrări în Mai Mult de un Oraș:

- Obține clienți care au pachete livrate în mai mult de un oraș.
- Utilizează **COUNT(DISTINCT)** pentru a număra orașele unice unde au fost livrate pachetele fiecărui client.
- Filtrarea se face folosind **HAVING** pentru a avea doar clienții cu livrări în mai mult de un oraș.
- Rezultatele sunt afișate pe pagina **customers_multi_city_deliveries.html**.

4. Clienți Care Au Trimis un Pachet cu Greutate Peste Media:

- Identifică clienții care au trimis cel puțin un pachet cu o greutate mai mare decât media greutăților tuturor pachetelor.
- Se folosește o subinterogare pentru a determina media greutăților pachetelor și apoi selectează pachetele care depășesc această medie.
- Rezultatele sunt listate pe pagina **customers_heavy_packages.html**.

5. Clienți Care Au Trimis Pachete de Peste 5 kg în 2024:

- Află clienții care au trimis pachete ce cântăresc mai mult de 5 kg în anul 2024.
- Interogarea folosește **MAX** pentru a găsi greutatea maximă a pachetelor trimise de fiecare client.
- Condiția este restricționată pentru anul 2024 și pachetele peste 5 kg.
- Datele sunt prezentate prin intermediul paginii **heavy_packages_2024.html**.

Aceste interogări demonstrează capacitățile SQL de a realiza analize complexe și de a furniza informații valoroase pentru deciziile de afaceri. În aplicația Flask, aceste interogări sunt executate la accesarea rutei corespunzătoare și rezultatele sunt trimise către template-urile HTML pentru a fi afișate utilizatorilor finali.