

Vanzarea de masini la un dealer auto.

Am creat proiectul numit Vanzarea de masini la un dealer auto.

Dealerul auto oferă informații despre vehicule pentru clienții care își doresc să cumpere mașini.

Sunt câteva roluri numite entități care lucrează pentru a sprijini reprezentanța auto, cum ar fi clienții, angajații și vehicule. Mai mult, există o serie de atribute și relații bazate pe aceste entități, cum ar fi: angajații vând vehicule și clienții cumpără vehicule. Toate aceste entități, atribute și relații formează un sistem de baze de date, iar proiectul meu vizează crearea unui sistem de baze de date care poate gestiona toate datele în mod clar.

În al doilea rând, pentru structurarea bazei de date există 8 entități: dealerships – reprezentanți, employee- angajați, branch office – sucursale, contracts- contracte, vehicles – vehicule, truck – camioane, sedan și SUV. Fiecare dintre entități are atribute atașate. Entitățile camion, sedan și SUV sunt specializările entității vehicul și moștenesc toate atributele din vehicule, având și atribute proprii.

Există multe relații și cardinalități între diferite entități, astfel încât stabilitatea și complexitatea sistemului să poată fi garantată. Am folosit SQL și PHP pentru a dezvolta sistemul de baze de date online.

Am realizat și o interfață de utilizator online care permite angajaților reprezentanței auto să se conecteze.

Informațiile clienților pot fi adăugate la sistem de către angajați, datele lor vor fi stocate și gestionate în sistemul de baze de date.

Sistemul de baze de date poate face unele gestionări și operațiuni la datele stocate în sistem. De exemplu, sistemul va

afișa lista vehiculelor care se leagă de un anumit angajat prin căutarea informațiilor.

Introducere

Proiectul este un sistem online care stochează și gestionează datele de la o reprezentanță auto. Am proiectat un sistem care conține un sistem frontal și un sistem de baze de date. În primul rând, sistemul frontal este o interfață de conectare online pentru angajați. Folosesc Hypertext Preprocessor (PHP), și Hyper Text Markup Language (HTML) pentru a dezvolta sistemul frontal. Angajații trebuie să se conecteze în sistem prin ID-uri de utilizator și parole, astfel încât sistemul să poată obține autentificarea angajaților, managerilor și clienților. Numai utilizatorii autorizați pot utiliza baza de date. În al doilea rând, sistemul de baze de date de fundal este un sistem de baze de date care stochează și gestionează toate datele din reprezentanță.

Baza de date conține date de la reprezentanțe, angajați, sucursale, contracte, vehicule, camion, sedan și SUV. Toate datele sunt înregistrate și actualizate la sistem de către angajați.

Pentru dezvoltarea bazei de date de fundal, am folosit MySQL, adăugarea de tabele, câmpuri și chei străine pentru a genera fișiere SQL.

Am creat, unele date stocate în baza de date, cum ar fi informațiile clientului și informațiile vehiculului, astfel încât sistemul de baze de date de fundal să permită utilizatorilor să caute și să găsească date prin căutarea informațiilor.

De exemplu, managerul poate vedea lista vehiculelor care se leagă de un anumit angajat căutând numele și prenumele angajatului. Un angajat poate găsi vehiculele cumpărate de un client căutând numărul său de telefon mobil.

Secțiunea Proiectare

Administrația (Administration)

Administrația se va putea autentifica folosind informațiile de conectare ale acesteia. Odată ce s-a conectat, va putea adăuga noi angajați și va putea căuta diverse tranzacții. Chiar la autentificare, vor avea o imagine completă a vânzărilor totale și a cantității de vehicule vândute pe lună. Administrația poate introduce un nou angajat utilizând ID-ul angajatului, numărul SIN(Social insurance number) al angajatului, numele angajatului, codul de supraveghere pentru acest angajat, numărul sucursalei la care a fost angajat.

Legătura tranzacției de căutare le permite acestora să vadă ce angajat a vândut ce vehicul pe baza ID-ului tranzacției. Aceasta este

de ajutor pentru a determina care angajat se descurcă bine în vânzări și la a stabili orice acte frauduloase, dacă ar apărea vreunul.

Angajat

Angajatul va avea acces la sistemul în care se va autentifica folosind datele de autentificare ale angajaților.

După conectare, ei vor putea vizualiza un istoric a ceea ce vânduseră. Aceasta fiind modelul vehiculului, culoarea vehiculului și totalul vânzărilor.

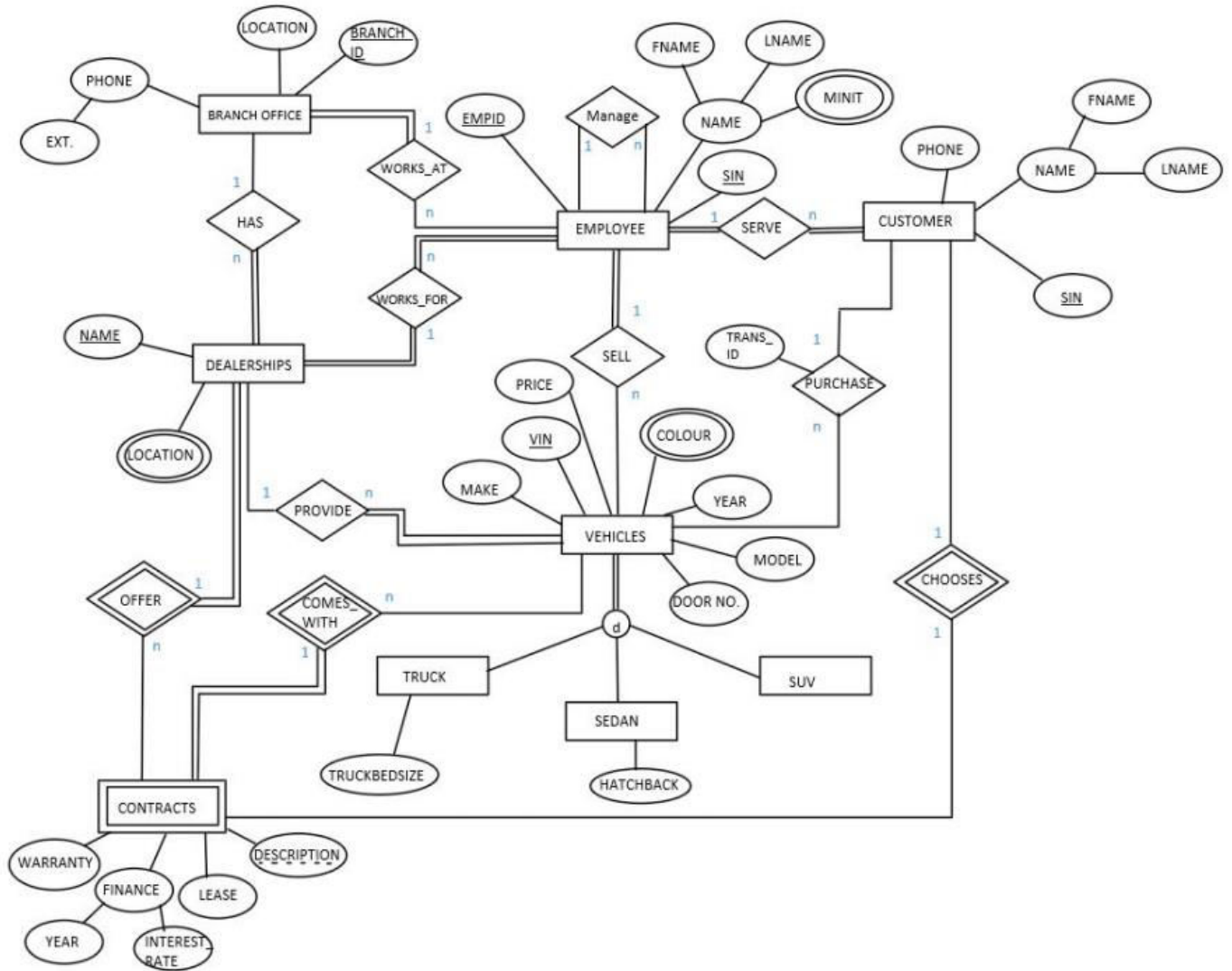
Ei vor putea să-și modifice istoricul unde pot introduce un client nou căruia i-au vândut un anumit vehicul.

Va apărea o nouă fereastră în care se cere ID-ul angajatului, numărul de identificare al vehiculului (VIN), numărul de asigurare socială al clientului (SIN), numele clientului, descrierea contractului și dacă e în leasing, finanțare sau garanție. După ce vor indica informațiile dorite, vor da click pe butonul de trimitere, acesta va insera valorile în următoarele entități: client, vânzare și contracte.

Client (Customer)

Clientul este restricționat după autentificare, nu poate modifica alte tabele decât ale lui. Un singur atribut poate fi actualizat de către client și acesta este numărul de telefon al acestuia. Dacă un client își schimbă numărul sau dacă furnizează companiei un număr nevalid, se poate autentifica folosind informațiile de conectare ale clienților și schimba numărul de telefon în consecință. Un client poate vizualiza, de asemenea, istoricul achizițiilor, care este compus din ID-ul tranzacției, numărul de identificare al vehiculului (VIN) și suma plătită. De asemenea, pot vizualiza informațiile de leasing (dacă există) sau informații financiare (suma rămasă de plătit, dacă există).

Diagrama EER



Funcționare:

Am creat entitatea `contracts` împreună cu următoarele relații: un client poate alege un contract în care vin vehiculele și un reprezentant va oferi mai multe contracte.

Entitatea contractuală este un tip de entitate cu o cheie parțială, atributul `DESCRIERE`. `DESCRIPTION` constă dintr-un scurt rezumat al fiecărui contract și

este de tip `char (n)` pentru orice tip de descriere care se potrivește. De exemplu, o descriere pentru un contract va determina data emiterii împreună cu data la care se va încheia.

`GARANȚIA (WARRANTY)` cu tipul de date întreg este în ani. Un client poate alege o garanție de 1, 2 sau 3 ani pentru vehiculul lor. În al doilea rând, `FINANCE` (compunerea dintre `YEAR` și `INTEREST_RATE`) este, de asemenea, de tip întreg date în care un client poate finanța un vehicul bazat pe o anumită rată a dobânzii sau pe baza anului în care trebuie plătit integral.

În final, clientul poate decide să `ÎNCHIRIEZE (LEASE)` un vehicul, acesta încă nu deține acest vehicul deoarece este încă sub reprezentanța (`dealership`) . Leasingul este determinat în ani și este de tip întreg. În plus, am creat atributul `PRICE` entității `VEHICLES` pentru a distinge suma totală făcută de fiecare angajat, un atribut `SIN` entității `EMPLOYEE`, astfel încât administratorul să poată distinge între angajați diferiți și un atribut `PHONE` în care un client își poate actualiza numărul de telefon.

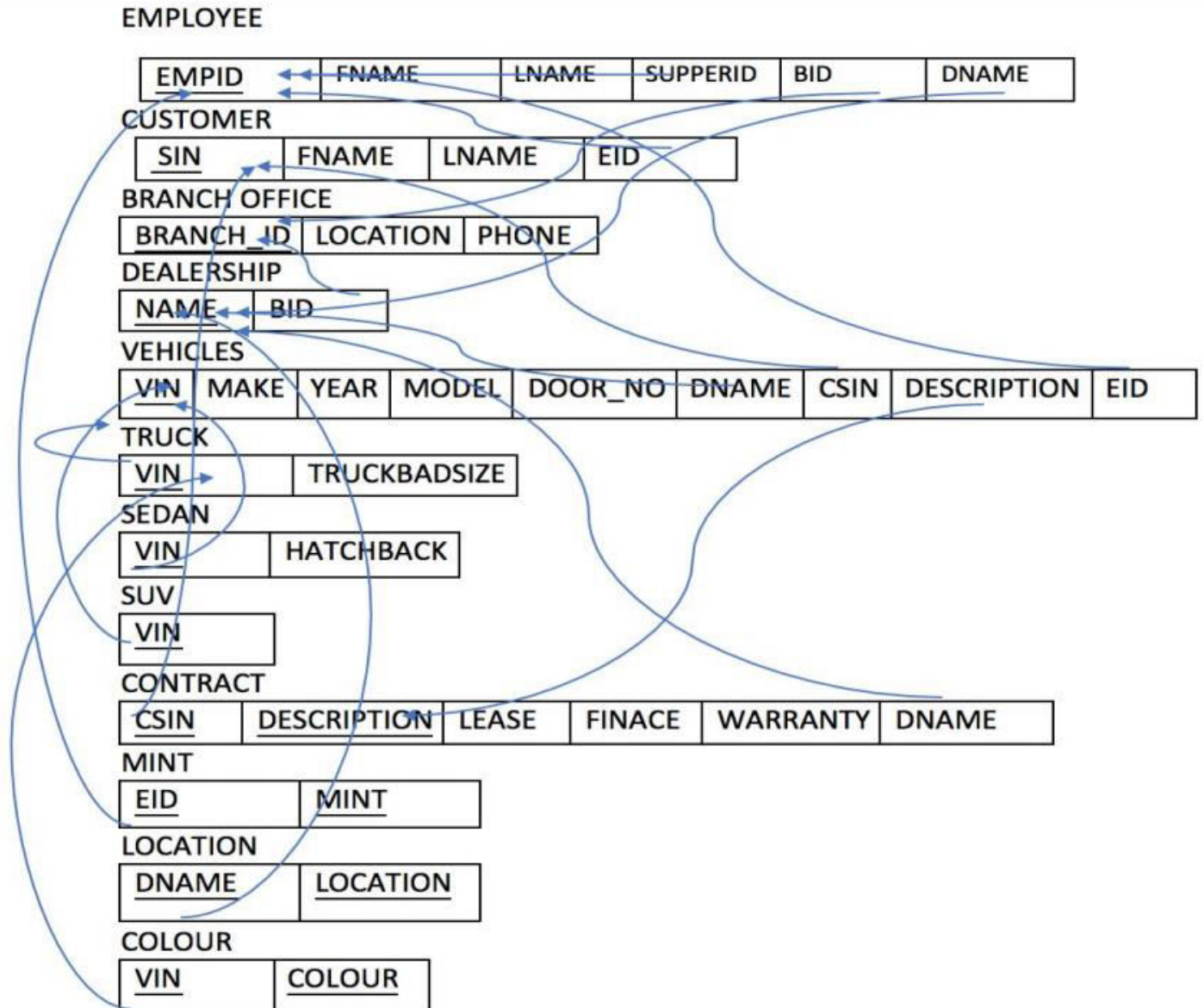
Ipoteze:

Un client poate achiziționa mai multe vehicule: dacă se realizează o acțiune de cumpărare, acesta va primi un `TRANS_ID` pentru acea achiziție specială. Nu fiecare client va cumpăra un vehicul și nu fiecare vehicul va fi vândut unui client.

Fiecare vehicul este mapat la specializări, `TRUCK`, `SEDAN`, și `SUV` și sunt disjuncte deoarece un camion nu poate fi și un sedan, un sedan nu poate fi un SUV și așa mai departe.

Atributul finance reprezentat în diagramă EER este compus din `YEAR` și `INTEREST_RATE`, în scopul simplificării, implementarea noastră folosește pentru finanțare un atribut ca sold restant care urmează să fie plătit de un client.

Secțiunea de implementare



Imaginea de mai sus prezintă schema relațională a bazei de date.

În acest proces, există trei atribute cu mai multe valori, prenumele angajatului, locația reprezentanței și culoarea vehiculului. Am creat trei relații noi, una pentru fiecare dintre ele. Fiecare relație include atributul de cheie primară (ca cheie străină) a relației care reprezintă tipul de entitate și numele atributului cu mai multe valori.

În aceste relații, putem vedea clar valoarea atributelor cu mai multe valori. Există, de asemenea, trei entități specializate pentru entitatea vehiculului : Truck, SUV și sedan. Pentru fiecare dintre ele, am luat atribute cu mai multe valori, am creat trei entități noi și am folosit numărul vehiculului ca cheie primară pentru toate cele trei entități și adăugându-le atributele unice.

Customer

Acest utilizator are cel mai limitat acces la baza de date a tuturor utilizatorilor. Ei pot folosi baza de date numai pentru a vizualiza informațiile de cumpărare. Următoarele interogări oferă rezultate care vor fi vizibile pentru acest utilizator:

```
$sql = "SELECT p.TRANSID, p.VIN, v.PRICE, v.MAKE, v.MODEL, v.YEAR  
FROM purchase AS p, vehicles AS v, customer AS c  
WHERE v.VIN = p.VIN AND p.CSIN = c.SIN"  
$sql = "SELECT con.LEASE, con.FINANCE  
FROM contracts AS con, customer AS cust  
WHERE con.CSIN = cust.Sin"
```

Actualizarea numărului de telefon:

Clientul își poate actualiza numărul de telefon introducând un nou număr de telefon în formular. Se va actualiza vechiul număr de telefon la noul număr de telefon:

```
$phone = $_POST["phone"];  
$sql = "UPDATE customer  
SET phone = $phone  
WHERE Sin = $sin";
```

Employee

Când angajatul se conectează, poate vedea automat câte mașini s-au vândut și câți bani s-au făcut în vânzări, aceasta este interogarea (ID-ul angajatului a fost codificat în calitate de angajat nr. 8 pentru demonstrație mai ușoară):

```
SELECT COUNT(*) as 'total', SUM(PRICE) AS 'sum' FROM sell, vehicles
WHERE sell.eid = 13
AND sell.VIN = vehicles.VIN
```

Angajatul poate vedea și istoricul vânzărilor într-un tabel. Aceasta este interogarea:

```
SELECT * FROM sell, vehicles, purchase
WHERE sell.eid = 8
AND sell.vin = purchase.VIN
AND sell.vin = vehicles.VIN
```

Acest utilizator poate adăuga noi vânzări la baza de date. Acest lucru se realizează prin crearea unui nou client, unui nou atribut de vânzări, unui nou atribut de achiziții și unul de contracte. Acest lucru se realizează prin următoarele interogări (variabilele \$_POST provin din introducerile utilizatorului pe un formular HTML):

```
$sql = "INSERT INTO sell VALUES (" .
$_POST["eid"] . "," .
$_POST["vin"] . ")";
$sql2 = "INSERT INTO purchase VALUES (" .
$_POST["sin"] . "," .
$_POST["vin"] . "," .
$_POST["transid"] . ")";
$sql3 = "INSERT INTO customer VALUES (" .
$_POST["sin"] . "," .
$_POST["CName"] . "," .
$_POST["eid"] . ")";
$sql4 = "INSERT INTO contracts VALUES (" .
$_POST["DESCRIPTION"] . "," .
$_POST["WARRANTY"] . "," .
$_POST["LEASE"] . "," .
$_POST["FINANCE"] . ")";
```


Administrator

Acest utilizator are cele mai mici restricții dintre toți utilizatorii.

Administratorul (care ocupă o poziție înaltă în companie) va folosi baza de date pentru a adăuga noi angajați, pentru a genera statistici privind vânzările companiei și pentru a cauta tranzacții specifice.

La conectare, administratorul este întâmpinat cu numărul total de mașini vândute până în prezent în companiei, precum și vânzările curente ale companiei până în prezent. Acest lucru se face prin următoarea interogare:

```
SELECT COUNT(*) as 'total', SUM(PRICE) AS 'sum'
FROM sell, vehicles
WHERE sell.VIN = vehicles.VIN
```

Managerul folosește submission form pentru a adăuga un nou angajat la baza de date.

Aceasta este realizată prin următoarea interogare (variabilele \$_POST provin din introducerile utilizatorului pe un formular HTML):

```
$sql = "INSERT INTO employee VALUES (" .
$_POST["id"] . " " .
$_POST["sin"] . " " .
$_POST["name"] . " " .
$_POST["super_id"] . " " .
$_POST["branch_no"] . " " .
$_POST["dealership"] . " " . ")";
```

Căutare tranzacții:

Administratorul poate localiza o tranzacție căutând ID-ul tranzacției. Acest lucru se face prin următoarea tranzacție:

```
SELECT * FROM sell, vehicles, employee, customer, purchase
WHERE purchase.TRANSID = " . $_POST["trans_id"] . "
AND purchase.VIN = sell.vin
AND customer.Sin = purchase.CSIN
AND sell.vin = vehicles.VIN
AND sell.eid = employee.EMPID
```

User

Site-ul are o interfață de utilizator foarte minimalistă. Acest lucru se datorează faptului că este utilizat în principal pentru introducerea simplă a datelor și interogărilor.

Am păstrat text negru pe fundaluri albe, iar câmpurile de formular, butoanele și linkurile sunt autodescriptive pentru a preveni orice confuzie din partea utilizatorilor.

Interogari complexe:

- 1) Numele cumpărătorilor de vehicule din anul 2017.

```
SELECT CName as 'name'
FROM vehicles
INNER JOIN sell ON sell.vin = vehicles.VIN
INNER JOIN customer ON sell.eid= customer.eid
WHERE PRICE IN (
SELECT PRICE FROM vehicles
WHERE year = 2017
ORDER BY PRICE ASC )
```

- 2) Transaction-ID-ul pentru contractele de leasing realizate pe o perioada mai scurta de 3 luni.

```
SELECT p.TRANSID as 'tid' FROM purchase p
INNER JOIN contracts c
ON p.CSIN = c.CSIN
WHERE p.CSIN IN (
SELECT CSIN FROM contracts
WHERE 3 >= LEASE)
```

- 3) Selectati numele directorului cu cei mai multi angajati si numarul de angajati ai acestuia.

```
SELECT  dname,COUNT(*) as num
FROM employee
WHERE dname in
(SELECT dname FROM employee
GROUP BY dname
HAVING COUNT(*)>1)
GROUP BY dname
LIMIT 1
```

- 4) -- Afisati cel mai scump vehicul in afara de truck-uri , care au fost tranzactionate in ultimii 5 ani

```
SELECT Model as 'mod' FROM vehicles
WHERE VIN NOT IN
(SELECT TVIN FROM Truck)
AND VIN IN
(SELECT VIN FROM SUV)
AND YEAR IN (
SELECT YEAR From VEHICLES WHERE YEAR BETWEEN CURDATE() - INTERVAL 5 YEAR AND CURDATE()
)
ORDER BY PRICE DESC
LIMIT 1
```

Pagina login.php ce contine meniul de login pentru administrator/angajat/comparator.



Car Dealership

Login

Enter your user :

Enter your password :

Pagina de administrator numita administration.php ce contine caseta de 'New Employee' permite adaugarea unui nou angajat , statistici privind numarul total de vehicule vandute , suma totala obtinuta in urma vanzarilor , transaction id-urile pentru contractele de leasing facute pe o perioada mai scurta de 3 luni , numele directorului cu cei mai multi angajati si numarul angajatilor lui.

Dupa care mai este posibila optiunea de transaction search ce ofera informatii despre tranzactie.



Car Dealership Management

Welcome, Administrator! | [Sign Out](#)

New Employee

Total vehicles sold: 11

Total company sales: \$292,270

Transaction-ID for leasing contracts made for a period shorter than 3 months:

12342

12346

12347

12350

The name of the director with the most employees and the number of its employees.

Harrison

7

Transaction Search

Cand selectez New Employee , se deschide fisierul newemployee.php in care trebuie sa introducem campurile cu informatile despre un nou angajat.



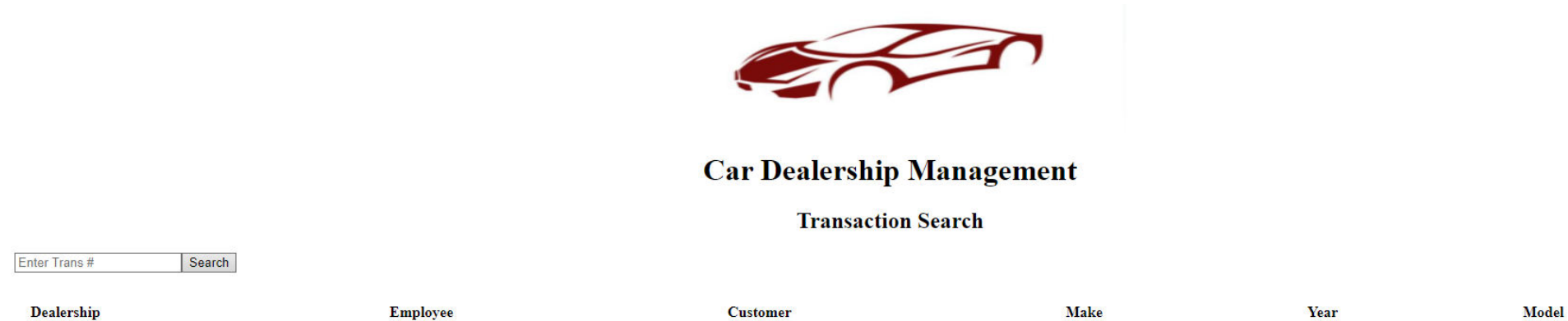
Car Dealership Management

New employee file

Enter new hire information:

<input type="text" value="ID"/>	<input type="text" value="SIN"/>	<input type="text" value="Name"/>	<input type="text" value="Supervisor ID"/>	<input type="text" value="Branch No."/>	<input type="text" value="Dealership"/>	<input type="button" value="Submit"/>
---------------------------------	----------------------------------	-----------------------------------	--	---	---	---------------------------------------

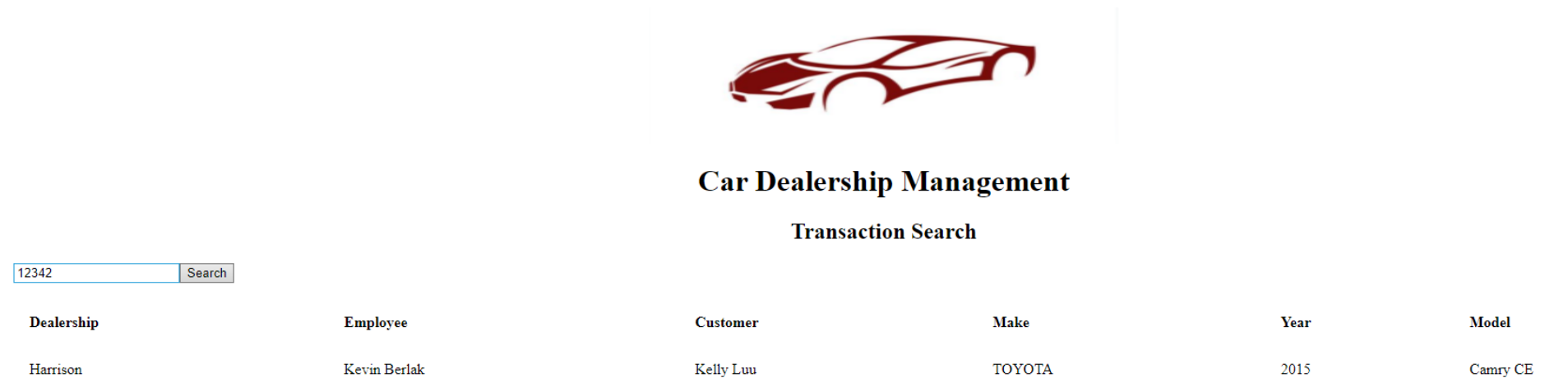
Cealalta optiune din pagina de administrator este Transaction Search, la apasare , apare la fel ca in imaginea de mai jos , introducand numarul tranzactie putem afla informatii despre tranzactie intr-un mod foarte simplu



The image shows a web application interface for "Car Dealership Management". At the top, there is a red car icon. Below it, the title "Car Dealership Management" is displayed in a large, bold, black serif font, followed by "Transaction Search" in a smaller, bold, black serif font. Below the title, there is a search form with a text input field labeled "Enter Trans #" and a "Search" button. Below the search form, there is a table with six columns: "Dealership", "Employee", "Customer", "Make", "Year", and "Model". The table is currently empty.

Dealership	Employee	Customer	Make	Year	Model
------------	----------	----------	------	------	-------

De exemplu ,am introdus numarul tranzactiei 12342 si va aparea astfel :



The image shows the same web application interface as before, but with the search results displayed in the table. The "Enter Trans #" field now contains the value "12342". The table has one row of data.

Dealership	Employee	Customer	Make	Year	Model
Harrison	Kevin Berlak	Kelly Luu	TOYOTA	2015	Camry CE

Pagina employee.php ce contine optiunea de new sale , pentru a adauga o noua vanzare pe care a efectuat-o angajatul.

In plus , gasim aici si numele cumparatorilor pentru anul 2016. Numarul de vehicule vandute , suma facuta in urma vanzarilor, cel mai scump vehicul care nu e truck ce a fost negociat in ultimii 5 ani.

In final niste detalii despre produsul negociat.



Vehicles sold so far: 1
Total sales: \$26,780

Name of customer/customers for the 2016 year:
Allen Zhang

Display the most expensive vehicle apart from trucks, which have been traded in the last 5 years
Land Cruiser

Trans No.	Make	Model	Year	Price
12347	TOYOTA	Camry LE	2017	26780

Daca selectam New Sale , se deschide fisierul newpurchase.php, in care putem introduce informatiile pentru o noua vanzare.



myToyota | New Sale

Enter new purchase information:

Employee ID	VIN	Customer SIN	Customer Name	Contract Description	Warranty	Lease	Finance	Transaction #	Submit
-------------	-----	--------------	---------------	----------------------	----------	-------	---------	---------------	--------

[Back to Employee Page](#)

Ultima pagina este cea a cumparatorului in care isi poate vedea istoricul cumparaturilor si alte detalii , in plus isi poate updata numarul de telefon.



Toyota Customer

Welcome, Customer! | [Sign Out](#)

Customer Purchase History

TRANSID	VIN	Price	Make	Model	Year
12343	10004	27680	TOYOTA	Camry CE	2016

You are on a 7 year lease
Your finance amount dropped to \$17000

Want to update your phone number?
Fill out your new number below:

