1.INTRODUCERE

1.1 Motivatia lucrarii

Facilitatea de transport reprezinta un element esential in societatea zilelor noastre si uneori lipsa acestei facilitati poate reprezenta un inconvenient. Problema ce urmeaza a fi discutata vizeaza persoanele care nu dispun de o masina personala pentru a se putea deplasa catre anumite puncte de interes ale acestora. Acest tip de persoane ce sunt predispuse in mare parte la acest tip de problema o reprezinta turistii, insa desigur, mai exista si alte cateva categorii mai mici. Astfel, in unele cazuri, acestia doresc o masina timp de cateva zile pentru a nu mai depinde de sistemele de transport public sau privat si desigur si de itinerariul acestora, acest lucru simplifica mult mai mult lucrurile atunci cand vine vorba de parcurgerea unor distante medii sau mari de la punctul A la punctul B. Asa ca, disponibilitatea unei aplicatii web ce permite inchirierea unui mijloc de transport la orice ora din zi si din noapte, face ca lucrurile sa devina benefice atat pentru client cat si pentru compania furnizoare. In zilele noastre, la distanta de cateva click-uri putem obtine multe lucruri pe care in trecut nici nu ne-am fi imaginat ca sunt posibile. De exemplu, deja cunoastem fenomenul de cumparaturi online, servicii bancare online, plus multe altele, acestea devenind lucruri ce simplifica viata de zi cu zi a unui individ.

Similar cu aceste exemple, sistemul de inchiriere de masini este facilitatea de a inchiria masini online, la distanta de doar cateva click-uri. Acest sistem include varietati de masini, in functie de pret si comfort, astfel clientul avand de unde alege in functie de bugetul si pretentiile sale.

Astfel, alegerea domeniului de comert online, reprezinta in mod sigur viitorul principalelor activitati de vanzare de servicii si produse, o buna parte din producatorii si distribuitorii de bunuri si servicii au facut deja pasul catre expunerea in mediul online, iar cei care inca nu au facut-o vor urma sa o faca daca vor dori sa concureze sau sa se mentina intr-o piata deja foarte competitiva. Comertul electronic in momentul de fata, are o foarte multe avantaje, bucurandu-se de un real succes la nivel global, schimband efectiv perceptia oamenilor in privinta procesului de achizitionare, prin oferirea unei game foarte largi si variate de optiuni. Pe langa faptul ca exista un numar imens de produse si servicii din ce in ce mai variate, putem spune ca o alta caracteristica fundamentala a acestui tip de comert ar fi aceea ca se economiseste foarte mult timp, totul fiind mai rapid si mai eficient, fapt pentru care in urmatorii zeci de ani cozile la magazin sau la ghisee vor deveni un mit devenit realitate. Toate aceste aplicatii reprezentand aplicatii ale comertului electronic ce necesita o buna sustinere informationala, cat si sisteme si infrastructuri organizationale.

Pe de alta parte, comertul online a devenit in ultimii ani cea mai rapida metoda de extindere a unei afaceri la nivel mondial. Din punct de vedere economic, acest tip de comert devine de la un la altul un element din ce in ce mai important in economie, deoarece nu are limitari fizice.

1.2 Contextul problemei

Dat fiind faptul ca tehnologia in zilele noastre este intr-o continua crestere si faptul ca comertul electronic devine din ce in ce mai popular la nivel global, voi pune in evidenta trei elemente ce descriu cel mai bine nevoia unei astfel de aplicatii web si ce solutii poate aduce aceasta.

Cele trei elemente sunt urmatoarele:

* Cresterea continua a numarului de date

Dupa cum stim, numarul datelor ce trebuiesc pastrate se mareste pe zi ce trece. Astfel, este foarte dificil sa stocam si sa pastram toate datele manual scrise pe formulare. De aceea, este absolut necesara o baza de date bine structurata, pentru a putea mentine datele intr-un mod sigur si eficient. Iar de asemenea, acele date sa fie usor de actualizat si de cautat.

* Sistemele necomputerizate

Aceasta aplicatie web este dezvoltata pentru a aduce un upgrade sistemului manual. In zilele noastre, multe firme de inchirieri de masini inca folosesc un sistem manual de inregistrare a datelor. In cazul in care acele date nu sunt pastrate in siguranta si gestionate eficient, pot interveni probleme in cadrul companiei. Acele probleme datorate pierderilor de date sau a organizarii deficitare, pot aduce la randul la o scadere a calitatii serviciilor si totodata a unor pierderi de natura financiara.

* Securitatea datelor nu este asigurata

Acest lucru este datorat tot faptului ca inregistrarile datelor se fac manual pe foaie, iar aceste foi pot fi usor deteriorate sau pierdute.

1.3 Aria de interes si obiectivele proiectului

Acest proiect traverseaza multe faze pana a ajunge de la un concept de afaceri la o aplicatie finita si functionala, fiind necesar inainte de toate cateva cercetari pentru a putea fi capabili sa indeplinim toate obiectivele proiectului.

Ariile de interes ce ar trebui acoperite sunt:

* Industria existenta de inchirieri de masini: Acest lucru include studierea modului in care afacerile de inchirieri de masini functioneaza, procesele implicate si oportunitatile ce pot duce la imbunatatiri si optimizari
* Tehnologiile: HTML,CSS,JavaScript,BootStrap,Php,MySQL ce sunt necesare dezvoltarii aplicatiei
* Aplicatia sa poata fi folosita atat de clienti cat si de cei care administreaza pagina
* Obiectivele fundamentale ale proiectului sunt:
* acela de a produce o aplicatie web ce permite utilizatorului sa se inregistreze si sa rezerve masini
* compania posesoare a aplicatiei sa isi administreze propria afacere
* sa se usureze si sa se simplifice munca clientului atunci cand vine vorba de inchirierea unei masini
* Sa se usureze si sa se simplifice munca angajatului in privinta gestionarii procedurilor de inchiriere de masini si a pastrarii datelor clientilor
* Sa se creeze o interfata interactiva si intuitiva, usor de utilizat atat de clienti cat si de angajatii ce gestioneaza aplicatia
* Aducerea unui plus de eficienta si o imbunatatire fata de ceea ce este existent deja pe piata

1.4 Beneficiile unui sistem de inchirieri de masini

Beneficiile aduse de o astfel de aplicatie sunt:

* existenta aplicatiei salveaza foarte mult timp, bani si munca
* compania is poate extinde zonele de activitate pe piata nationala si internationala mult mai usor, cu o investitie mica raportat la posibilele castiguri
* internetul, mediul prin care comertul electronic se realizeaza, este omniprezent, accesibil si de asemenea si ieftin
* nu exista limitari din punct de vedere geografic, atunci cand vine vorba de expunerea serviciilor companiei
* cresterea vitezei de comunicare
* intuitivitatea si usurinta utilizarii
* afacerea in general devine mult mai usor de administrat si include mai putina birocratie
* se pot atrage mai multi posibili noi clienti
* permite persoanelor angajate sa lucreze de acasa in unele cazuri
* duce la servicii de o calitate mult mai buna catre clienti, datorita faptului ca eficienta managerierii creste
* procedurile de administrare a companiei se standardadizeaza
* aplicatia se comporta ca un magazin deschis 24/7
* toate informatiile despre produse si clienti sunt pastrate intr-o baza de date
* comertul online va aduce companiei ce detine aplicatia venituri mai mari

1.5 Notiuni generale despre comertul electronic

Dupa cum a mai fost mentionat anterior de cateva ori, comertul electronic este o activitate de vanzare si achizitionare de bunuri si servicii ce are loc la nivel online. Acest termen de comert online este datorat faptului ca toate aceste actiuni au loc prin intermediul internetului si reprezinta in prezent un element foarte important la nivel global din punct de vedere economic.

Obiectivul fundamental al acestui tip de comert este acela de a raspandi informatii despre bunurile si serviciile oferite de catre o companie, la cat mai multi clienti posibili. Aceasta strategie in zilele noastre, aduce profituri foarte mari companiilor, comparativ cu costul investitiei facute.

Comertul electronic este cunoscut drept o unealta destinata si construita atat pentru firme cat si pentru consumatori, imbunatatindu-se calitatea produselor si a serviciilor oferite, datorate unei mai bune gestionari a timpului in care se efectueaza o comanda si se primeste.

Drept consecinta, cresterea tranzactiilor online se bucura de o crestere din ce in ce mai sporita de la un an la altul. Nivelul vanzarilor online ajungand in anul 2017 la peste 2.2 triliarde de dolari, lucru ce reprezinta o crestere de peste 20% fata de anul anterior.

Comertul electronic in momentul de fata, se afla intr-o continua schimbare si imbunatatire si reprezinta o alternativa mult mai buna a comertului traditional, datorita faptului ca se afla intr-o continua ascensiune ce ofera un spectru larg de posibilitati de noi afaceri si totodata de o imbunatatire a interactiunii cu utilizatorii.

1.6 Impactul economic al inchirierilor de masini pe platforme online

In prezent, datorita comertului economic ce s-a dezvoltat foarte puternic in ultimii ani pe mai multe ramuri de activitate in care se pot oferi bunuri si servicii a aparut si expansiunea serviciilor de inchiriere de masini in mediul online s-a bucurat de un real succes.

Conform unei statistici din anul 2019 de pe [www.statista.com](http://www.statista.com) care prezinta veniturile din mediul online la nivel global pentru inchirieri de masini din anii anteriori si un pronostic pentru urmatorii 3 ani, arata faptul ca veniturile din acest tip de comert electronic se vor afla pe o traiectorie ascendenta.

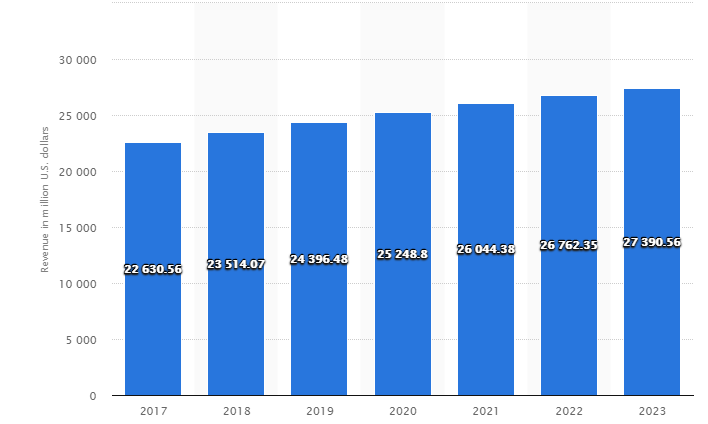


Figura 1-Statistica a veniturilor la nivel global din inchirieri de masini in mediul online

Conform statisticii prezentate, acest segment de servicii contine statistici doar a inchirierilor facute in mediul online, independent de inchirierile de masini facute la reprezentantele companiilor. Astfel, fiind excluse inchirierile efectuate offline sau telefonic.

2.TEHNOLOGIILE UTILIZATE LA REALIZAREA APLICATIE DE INCHIRIERI DE MASINI

In acest capitol vor fi prezentate si descrise tehnologiile ce au facut posibila realizarea acestei aplicatii.

2.1 Prezentare generala a unei aplicatii web

Termenul de aplicatie web este utilizat pentru acel tip de aplicatie ce poate fi accesata de mai multi utilizatori concomitent datorita unei retele de internet. Totodata termenul poate sa se refere la o aplicatie software ce a fost dezvoltata intr-un limbaj de programare ce este suportata de un browser.

Aplicatia de inchirieri de masini se bazeaza pe mai multe tehnologii ce au avut ca scop atat partea de front-end, pentru crearea interfetei,cat si cea de back-end, ce asigura buna functionare a aplicatiei si conexiunea la baza de date.

2.2 MySQL

MySQL este un sistem popular de gestionare a bazelor de date de tip relational. Acesta asigurand un nivel inalt de performanta si stabilitate. Sistemul este utilizat de dezvoltatorii de de aplicatii web pentru construirea a diferite tipuri aplicatii web.

O baza de date permite utilizatorului sa stocheze eficient date, sa caute date, sa le sorteze si sa le extraga daca este cazul. Initial a fost dezvoltat pentru a opera pe date din baze de date ce urmeaza modelul relational. De asemenea, bazele de date sunt reprezentate sub forma de randuri si coloane, randurile reprezentand intrarile si coloanele reprezentand atributele. Datele trebuind bine structurate si schema de memorare a datelor sa fie foarte stricta. Diferite tabele mentinand un fel de conexiune logica ce reprezinta relatia dintre acestea. Aceste tipuri de conexiuni numite relatii sunt create pe baza interactiunii dintre tabele, iar astfel se explica si provenienta termenului de baze de date de tip relational.

Bazele de date de tip relational sunt folosite in cazuri in care exista scheme stricte si are rolul de a oferi o mai mare flexibilitate. MySQL este un sistem cu o relatie de tipul client-server. Astfel, o aplicatie client-server este impartita in doua portiuni distincte. Portiunea server-ului fiind un soft numit server de baze de date ce este responsabil cu manipularea si accesarea datelor, numai server-ul interactionand cu fisierele de date. Toate cererile de accesare de date, adaugari, stergeri sau actualizari de date fiind filtrate prin intermediul soft-ului server, toate aceste cereri venind de la soft-ul client. Partea de client fiind cea de a doua portiune si are scopul de a interactiona cu utilizatorul. Soft-ul client trimitand solicitarea catre soft-ul server-ului. Soft-ul server-ului dupa ce a primit solicitarea, va filtra si va face operatiile necesare, dupa care va trimite rezultatele inapoi catre soft-ul client.

Serverul MySQL controleaza accesul la date si asigura faptul ca mai multi utilizatori pot lucra concomitent, desigur utilizator trebuind sa fie autorizati pentru acces. MySQL utilizeaza limbajul SQL pentru a opera date, iar spre deosebire de alte limbaje, acesta detine putine instructiuni. Acest limbaj fiind proiectat doar pentru a comunica cu bazele de date.

Totodata, SQL este un limbaj neprocedural, iar astfel utilizatorii specifica prin intermediul limbajul de ce informatii au nevoie si nu cum sa le obtina.

Pe de alta parte, din punct de vedere al arhitecturii, arhitectura sistemului MySQL este formata din trei nivele. Primul nivel cuprinde si include gestionarea conexiunii, autentificarea si securitatea. Al doilea nivel putem spune ca este nucleul pentru MySQL si cuprinde examinarea query-urilor, analiza si optimizarea datelor. Cel de al treilea nivel al arhitecturii sistemului MySQL este un mecanism de stocare responsabil pentru accesarea datelor.

Desi, de cele mai multe ori MySQL este utilizat alaturi de PHP, acesta totusi permite construirea de aplicatii in orice limbaj popular de programare, precum C++, Java, Perl sau Phyton. Administrarea bazelor de date se poate face in doua moduri, primul ar fi prin intermediul liniei de comanda, iar cel de al doilea prin intermediul interfetei grafice: MySQL Administrator.

De asemenea, MySQL poate fi rulat pe foarte multe sisteme de operare existente, precum Windows, GNU/Linux, Mac OS X, conferindu-i astfel flexibilitate si o foarte buna pliere.

2.3 PHP(PHP: HYPERTEXT PREPROCESSOR)

PHP este un limbaj de programare destinat paginilor web, ce a fost proiectat cu scopul de a produce pagini web dinamice. Cu acest scop, codul de PHP este incorporat in codul html si interpretat de un server web. Astfel, datorita limbajului PHP, dezvoltatorii de aplicatii web au reusit sa aduca mai multe functionalitati aplicatiei web, decat ar fi putut numai prin intermediul limbajului HTML care este interpretat de browser si nu executat de un server. De exemplu, utilizatorii pot crea aplicatii for complexe ce interactioneaza cu o baza de date si genereaza continut dinamic. Astfel, limbajul PHP usurand faptul ca anumite continuturi ale unei pot fi actualizate, iar pe de alta parte acest limbaj poate crea pagini customizate ce afiseaza un anumit tip de informatii doar unui utilizator sau unui anumit tip de utilizator. In concluzie, PHP se potriveste perfect in codul HTML, acesta fiind invizibil pentru persoanele ce viziteaza aplicatia web, singurul lucru la care vizitatorii au acces fiind codul sursa HTML, astfel dezvoltatorii ce utilzeaza PHP avand o mai multa securitate si totodata flexibilitate.

Cand vine vorba de conexiunea dintre PHP si MySQL, exista cateva comenzi MySQL aflate in codul de PHP ce au rolul de accesa si opera in baza de date. Iar comenzile ar fi urmatoarele:

⦁ Mysql\_connect("hostname","user","pass") ->ce are rolul de a se conecta la serverul MySQL

⦁ Mysql\_select\_db("database name")-> face baza de date selectata in una activa

⦁ Mysql\_query("query")->folosita pentru a trimite orice tip de comanda MySQL catre server

⦁ Mysql\_fetch\_array("result variable from query")->intoarce ca rezultat cateva randuri rezultate in urma comenzii date catre server

⦁ Mysql\_error()->afiseaza un mesaj de eroare returnat direct de la serverul MySQL

In concluzie, limbajul PHP poate obtine acces la fisier, la baze de date si la alte resurse ce sunt inaccesibile limbajului JavaScript, aceste lucruri constituind o sursa bogata de continut dinamic.

2.4 HTML(HyperText Markup Language)

HTML este o abreviere de la HyperText Markup Language si reprezintă în zilele noastre unul din cele mai utilizate instrumente în realizarea unei aplicații web. Acest limbaj nu este un limbaj de programare, ci unul de formatare a textului, utilizand marcaje, ce sunt numite drept tag-uri in limba engleza.

Ideea de HTML aparut in anul 1980, cand un fizician britanic pe nume Tim Berners-Lee a dorit un limbaj pentru distribuirea de documente stintifice, insa ideea a fost dusa pana la capat abia in anul 1991. De asemenea acestea este recunoscut si pentru aparitia HTTP(Hypertext Transfer Protocol) ce a permis transferul de documente HTML intre computerele ce faceau parte din aceeasi retea.

Tag-urile apar intre semnele < si >, iar acestea reprezinta instructiuni ce spun browser-ului cum sa interpreteza tag-urile respective. Astfel un document HTML este impartit in doua sectiuni, una din ele este sectiunea antet ce apare intre <head> si </head> si reprezinta inceputul si sfarsitul unui antet si o sectiune corp cuprinsa intre <body> si </body> care reprezinta inceputul si sfarsitul documentului propriu zis. De asemenea cele doua sectiuni vor fi cuprinse de <html> si </html> ce au rolul de a descrie pagina web.

Navigatorul nu afiseaza marcajele HTML, ci le foloseste pentru a interpreta intr-un mod corect informatia introdusa si modul cum trebuie sa fie afisata si pozitionata in pagina.

Fiecare fisier de tipul HTML va trebui sa inceapa cu o declaratie de tipul <!DOCTYPE>, aceasta declaratie nereprezentand un tag, ci fiind mai degraba o instructiune ce are menirea de a furniza browser-ului datele necesare prin care afla ce versiune de HTML este utilizata.

Limbajul HTML este unul de tipul whitespace-insesitive, ceea ce duce la faptul ca acesta permite spatierea fara a exista eventuale probleme.

Concluzia ar fi ca, pe intregul parcurs al existentei acestui limbaj de programare, acesta a avut parte de o dezvoltare continua, prezentandu-se mai multe versiuni. Ultima versiune a aparut in anul 2014, numita HTML5.

2.5 CSS(CASCADING STYLE SHEETS)

Cascading Syle Sheet, cunoscut sub prescurtare de CSS este un limbaj de stilzare a unei pagini web, rolul acestui fiind imbunatatirea prezentarii elementelor unui document de tip HTML din punct de vedere al designului, cum ar fi paragrafe, liste, tabele si multe alte elemente constituente paginii respective. Astfel, putem observa ca limbajul HTML are propriile sale limitari, scopul sau principal fiind acela de a acorda mai multa atentie catre structura, decat catre design. Limbajul HTML fiind mai mult orientat sa exprime informatii de genul "unde se afla un tabel" sau "unde se afla un paragraf". Din acest motiv, CSS a fost introdus pentru a rezolva problemele de restrictii.

Limbajul CSS ocupandu-se astfel la modul cum arata o pagina web, utilizatorul acestui limbaj avand puterea de a schimba de exemplu culoarea unui text, stilul unui font, dimensiuni, introducere de imagini cu diferite efecte, plus multe alte beneficii.

Facand o comparativa cu limbajul HTML, CSS are un numar mult mai mare de optiuni de formatare a unui text, iar pe langa acest fapt, s-a observat ca incarcarea paginilor web ce utilizau limbajul CSS pentru design, erau mult mai rapide decat cele care utilizau doar limbajul HTML si il utilizau si pentru design.

Documentele CSS sunt atasate documentelor HTML, acestea putand fi incluse din documente externe, dar totodata pot fi scrisa si in cadrul documentului HTML, utilizand marcajele <style> si </style>.

Datorita CSS-ului, pagina web este schimbata dinamic, reusind si in prezent sa pastreze o calitate buna de afisare a paginii.

Sintaxa limbajului CSS spre deosebire de ceea a limbajului HTML, nu se bazeaza pe elemente si pe tag-uri, ci se bazeaza in principal pe reguli ce au parti componente un selector alaturi de un set de declaratii ce sunt despartite prin punct si prin virgula.

Desi exista numeroase avantaje existente, exista de asemenea si anumite limitari, cum ar fi acelea de a se adapta la diferite browsere, dar totusi acest limbaj reprezinta in continuare o modalitate foarte utila de realizare a designului unei pagini web.

2.6 BOOTSTRAP

Bootstrap este un framework pentru frond end, ce este gratuit si open source, ce are scopul dezvoltarii rapide de aplicatii web, utilizand CSS,HTML si JavaScript. Bootstrap este atat de popular din cauza faptului ca este foarte adaptabil. Utilizarea Bootstrap-ului nu este una dificila in cazul in care exista cunostinte de baza anterioare de HTML si CSS.

Inainte de Bootstrap, erau utilizate varietati de librarii pentru dezvoltarea interfetei, ceea ce a dus la o problema foarte mare cand venea vorba de mententa interfetei, existand inconsistente.

Initiatorii proiectului de dezvoltare a Bootstrap-ului au fost Mark Otto si Jacob Thornton din cadrul companiei Twitter, fiind lansat ca un proiect open-source in anul 2011 pe data de 19 august.

Bootstrap ofera numeroase stiluri CSS de baza, cum ar fi butoane, tabele sau imagini, ce pot fi usor de ajustat si care se pot adapta.

Bootstrap vine de asemenea si cu cateva componente de JavaScript sub forma de plugin-uri jQuery. Acestea asigurand elemente grafice aditionale cum ar fi casutele de dialog sau carusele,

Aparitia Bootstrap-ului este datorata si faptului ca in ultimii ani popularitatea utilizarii telefonului mobil a crescut, iar astfel a fost necesara unui astfel de limbaj ce sa asigura adaptabilitatea paginilor web, atat pentru dispozitivele mobile cat si pentru computere.

2.7 SERVERUL XAMPP

XAMPP este cel mai popular mediu de dezvoltare PHP si contine pe langa MySQL si bineinteles PHP si server-ul Apache si Perl.

Provenienta numelui vine de la initialele componentelor ce le contine: X este pentru multi-platforma, deoarece functioneaza pe multe sisteme operative, A vine de la Apache, M de la MySQL, primul P vine de la PHP, iar ultimul de la Perl.

Apache HTTP Server-este platforma de server web cea mai utilizata, acesta fiind un software ce are rolul de a realiza functii de transport ale informatiei.

Astfel, multe servere web actuale utilizeaza aproximativ aceleasi componente ca si ale XAMPP-ului, ceea ce face usoara tranzitia dintre un server local si unul ce nu functioneaza doar la nivel local.

2.8 JAVASCRIPT

JavaScript este un limbaj de scriptare multi-platforma, de nivel inalt, orientat pe obiecte.Alaturi de HTML si CSS, JavaScript este un element ce a stat la baza internetului din zilele noastre, fiind un element esential al aplicatiilor web ce permite ca paginile web sa devina mai interactive. Marea majoritate a site-urilor web utilizandul si mare parte din browsere avand dedicat un motor JavaScript care sa il execute. Initial acesta a fost implementat doar pe partea de client in browserele web,era un limbaj al carui cod sursa era procesat de catre browser. Dar in zilele noastre este incorporat in multe alte tipuri de soft-uri gazda, incluzand si partea de servere din servere web si baze de date.

De asemenea, JavaScript nu este un limbaj ce poate fi compilat, ci este unul de tip interpretabil, codul fiind parcurs si rezultatul obtinut imediat.Browserul nu va executa codul de JavaScript daca nu sunt existente deja elemente de tip HTML si CSS, returnand o eroare in cazul in care nu sunt prezente.

Script-urile pot fi incluse in codul HTML sau pot fi stocate in fisiere externe si legate in interiorul codului de HTML.La fel ca si la limbajul PHP si codul JavaScript poate fi implementat in interiorul codului de HTML, atata timp cat sunt specificate tag-urile <script> si </script>.

Limbajul Javascript este de tipul case-sensitive si permite definirea a mai multe tipuri de date primitive precum:boolean,int,double.

Folosind JavaScript se pot realiza de exemplu: meniuri de tip drop-down, animare a textului, se pot face procesari de baza numerice, plus multe alte beneficii.

Desi sunt existente anumite asemanari intre limbajul JavaScript si Java, putem spune ca sunt similare din anumite puncte de vedere, dar sunt fundamental diferite in altele. Desi Javascript urmeaza majoritatea sintaxei utilizate de Java, Javascript nu face distinctia intre tipuri de obiecte si proprietatiile si metodele pot fi adaugate oricarui obiect in mod dinamic.

In ciuda faptului ca limbajul JavaScript este unul foarte popular si are foarte multe avantaje, are si doua dezavantaje destul de mari, acelea fiind ca nu poate functiona fara existenta unui program gazda si acela ca nu poate garanta o securitate de nivel inalt.

2.9 Apache

Apache HTTP Server este un server web open source, destul ce popular, ce la inceputurile sale era destinat mediului de lucru Unix, dar dupa a fost adaptat si pentru alte sisteme de operare, cum ar fi Windows-ul, Linux sau Mac OS X.

Apache a avut un rol destul de important in dezvoltarea web, fiind utilizat in prezent la mai mult de jumatate din paginile de internet active.

Functionalitatea Apache poate fi extinsa de o multitudine de module care variaza de la programarea pe partea de server, pana la anumite scheme de autentificare dinspre client. Printre limbajele de programare suportate de Apache putem enumera: Perl, Phyton sau PHP.

Apache si-a castigat renumele datorita faptului ca este utilizat de unele din cele mai mari aplicatii web din lume, de exemplu motorul de cautare de Google, utilizeaza un modul adaptat de Apache.

3.STRUCTURA APLICATIEI

3.1 Scop

Specificatiile tehnice sunt considerate cruciale si ofera o fundatie solida in procesul de dezvoltare ale unei aplicatii. Analiza unui sistem trebuind sa fie facuta intr-un mod care descrie si reflecta cat mai bine functionalitatile unui sistem.

Dupa cum am enumerat si descris in capitolul 2 tehnologiile necesare producerii aplicatiei web de inchirieri de masini, continutul si prezentarea paginii este facuta datorita limbajelor HTML, CSS si Bootstrap, tehnologia JavaScript este pentru partea de interactivitate a aplicatiei, limbajul PHP pentru partea de server si conexiune la baza de date, iar MySQL ca si baza de date de tip relational. Astfel, toate aceste componente si tehnologii functioneaza impreuna pe un Server HTTP Apache.

Figura 1 de mai jos este reprezentarea grafica a structurii aplicatiei web.

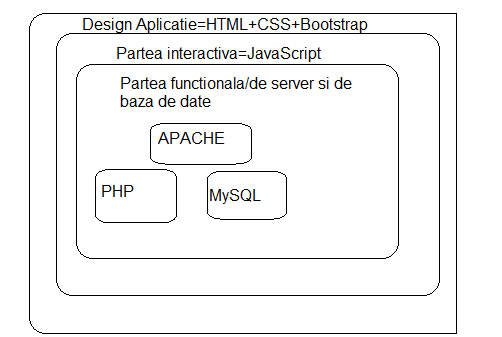


Figura 1. Structura Aplicatiei Web

3.2 Arhitectura generala a elementelor functionale a aplicatiei web de inchirieri de masini

Arhitectura generala a elementelor functionale a aplicatiei web de inchiriere este descrisa in diagrama din figura 5 de mai jos, in care sunt prezenti patru actori ce pot interactiona in moduri diferite cu aplicatia.

Astfel, utilizatorul putand fi de mai multe categorii, vom explica in detaliu pe fiecare dintre acestia. Primul dintre acestia si totodata cel care are cele mai putin privilegii este utilizatorul neinregistrat, care nu beneficiaza de nici o operatiune. Astfel, utilizatorului neinregistrat i se va cere sa se inregistreze pentru a putea avea dreptul sa faca mai multe operatiuni in cadrul aplicatiei. Odata aprobata cererea de inregistrare respectivul utilizator va putea sa se conecteze si astfel va deveni cel de al doilea tip de utilizator, un utilizator inregistrat.

Utilizatorul inregistrat odata ce se va conecta, va fi directionat catre pagina principala, iar dupa aceea va putea sa acceseze diferite pagini web ale aplicatiei in functie de dorintele acestuia si va putea face anumite operatiuni.

Principalele operatiuni permise utilizatorului inregistrat, ce sunt prezentate si in figura 5 vor fi urmatoarele:

* Dupa ce utilizatorul s-a conectat, acesta va putea sa isi vizualizeze profilul, in care sunt prezente informatiile personale si de asemenea va putea sa isi actualizeze datele respective
* Va putea sa se deconecteze de pe contul pe care s-a conectat, daca va dori asta
* Va putea vizualiza o lista de masini disponibile oferite de compania de inchirieri, aceasta lista fiind extrasa din baza de date a aplicatiei
* Va putea sa isi aleaga o masina din acea lista, iar daca este interesat de aceasta, va avea dreptul la o cerere de inchiriere ce va fi inregistrata intr-o baza de date
* Va putea vizualiza cererile de inchiriere aprobate de catre administrator

Cel de al treilea tip de utilizator este utilizatorul de tip administrator care se va ocupa de gestionarea diferitelor cereri primite de la utilizatori. De asemenea se va ocupa si de actualizarea produselor de care dispune compania de inchirieri de masini ce detine aplicatia web. Astfel operatiunile puse la dispozitie utilizatorului de tip administrator in concordanta cu figura 5 sunt urmatoarele:

* Daca conectarea acestuia este permisa, atunci acesta va fi redirectionat catre un panou de comanda
* Odata ajuns la panoul de comanda acesta va avea optiunea sa isi schimbe parola daca va dori si evident ca va avea si optiunea de deconectare de pe contul curent
* Respectivul administrator va putea vizualiza lista curenta de utilizatori inregistrati, ce este extrasa din baza de date de utilizatori
* Va putea confirma sau respinge cererile de inregistrare nou aparute
* Va putea confirma sau respinge cererile de inchiriere nou aparute ce au fost trimise de catre utilizatori, iar acesta va alege una din cele doua optiuni in functie de disponibilitatea masinii in perioada ceruta

Ultimul tip de utilizator si totodata si cel mai important, ce sta la baza bunei functionari a aplicatiei web, este web master-ul care poate oferi suport companiei de inchirieri de masini, ce detine aplicatia respectiva. Acesta poate administra baza de date MySQL si structura acesteia, are acces deplin la toate componentele aplicatiei si se ocupa de mentenanta si dezvoltare a aplicatiei.

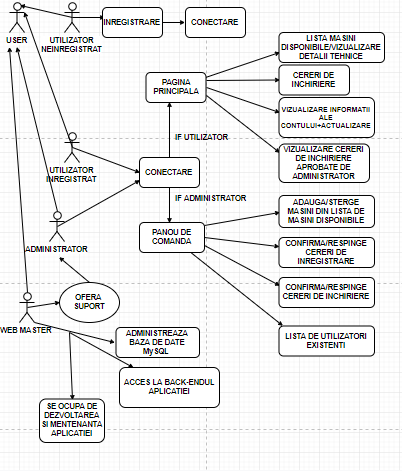


Figura 5-Arhitectura generala a aplicatiei de inchirieri de masini

3.3 Principalele functionalitati ale aplicatiei

Pentru a putea reprezenta grafic decursul procedurilor si actiunilor ce au loc, sunt necesare cateva diagrame de activitati. Diagramele de activitati sunt o reprezentare grafica a activitatilor si actiunilor ce au loc si arata utilizarea a diferiti actori ce pot interveni in sistem. Use case-urile diagramelor de activitati sunt prezentare pentru partea logare a utilizatorului, cea de logare a administratorului si activitatea de inchirieri de masini.

3.3.1 Back-end-ul conectarii administratorului la panoul de comanda

Dupa cum se poate observa si in figura 2 de mai jos, in momentul in care administratorul va accesa link-ul catre pagina de conectare la panoul de comanda, acesta va trebui sa isi introduca datele de acces, mai exact, username-ul si parola.

Username-ul si parola administratorului sunt deja predefinite in baza de date, fapt pentru care dezvoltatorul va trebui sa comunice cumparatorului aplicatiei aceste date. Odata ce i-au fost comunicate aceste date, acesta se va putea conecta si va avea optiunea de a isi schimba parola.

Trecand peste faptul ca username-ul si parola trebuie comunicate cumparatorului aplicatiei, in momentul in care sunt introduse datele respective si solicitata cererea de conectare, aceste date se vor procesa si se vor trimite prin intermediul unui query catre baza de date. In baza de date se vor cauta datele respective, baza de date va intoarce un rezultat, care la randul lui va fi de asemenea procesat.

In cazul in care procesul valideaza datele, se va permite accesul catre panoul de comanda, iar in caz contrar se va returna o alerta care va atentiona utilizatorul ca username-ul sau parola sunt incorecte.

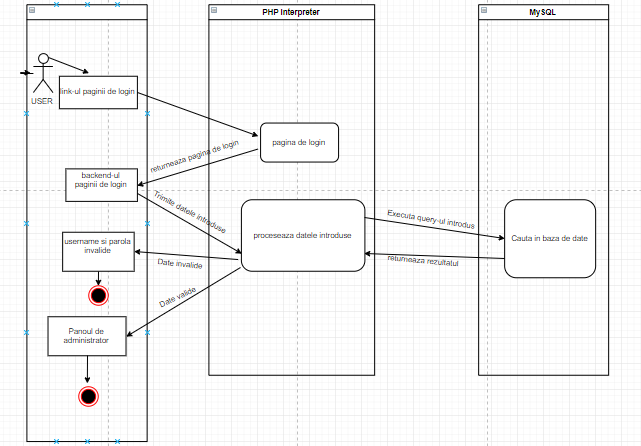


Figura 2: Conectarea administratorului la panoul de comanda

3.3.2 Backend-ul conectarii clientului

Dupa cum s-a observat si in figura 2 de la conectarea administratorului la panoul de comanda, procedeul este aproximativ acelasi si la conectarea utilizatorului. Singura diferenta dintre administrator si utilizator fiind aceea ca el nu are drepturile de acces la panoul de comanda si va ramane la interfata pregatita pentru utilizator. In momentul cand clientul va incearca sa se conecteze, datele introduse i se vor cauta dintr-o alta baza de date, fata de cea pentru administrator. Daca conexiunea va fi reusita, acesta va ramane de pagina de acasa unde ii va aparea iconita cu numele sau. Acesta va avea drepturi de care un utilizator neconectat nu poate beneficia. Drepturile principale de care beneficiaza utilizatorul conectat fiind acelea de a isi updata datele profilul, sa isi schimbe parola si aceea de a face cereri de inchiriere de masini.

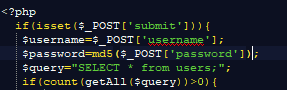
Comparativ cu un sistem simplu de conectare a unui utilizator care poate avea anumite probleme, sistemul de conectare al acestei aplicatii web are urmatoarele caracteristici:

* Conectarea utilizand username
* Criptarea parolei utilizatorului
* Conectarea utilizand sesiuni php
* Doar membrii activi/aprobati se pot conecta

Primul pas pentru a face un sistem de login este acela de a conecta pagina respectiva la baza de date, iar dupa ce a fost conectata vom selecta baza de date.

Vom incepe prin verificarea sesiunii, astfel, in cazul in care utilizatorul este deja conectat acesta va fi redirectionat catre pagina principala a aplicatiei. Daca valoarea post-ului butonului de conectare este setata, vom atribui unor variabile numele si parola introduse de catre utilizator.

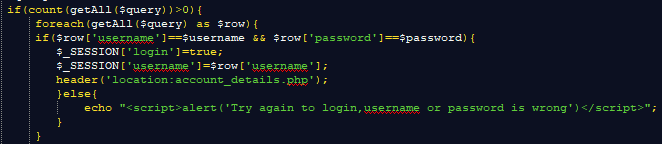
Codul acestei operatiuni este urmatorul:



Folosind md5 in acest cod, vom cripta parola, pentru a nu exista probleme de securitate si pentru a nu se cunoaste in baza de date, parola respectivului utilizator, conturile acestora devenind mai securizate.

Dupa aceasta operatiune, vom scrie un query SQL pentru a verifica detaliile utilizatorului in tabelul de utilizatori din baza de date. Iar query pentru aceasta operatiune este de forma: $query="SELECT \* from users;";

Executam acel query si numaram totodata si numarul de linii din setul rezultat, iar daca numarul de linii este egal cu 1 vom seta sesiunea de login ca fiind adevarata si sesiunea de username careia i se va atribui username-ul contului respectiv. In cazul in care username-ul si parola din tabelul de utilizatori din baza de date nu este in concordanta cu datele introduse, atunci se va afisa un mesaj de eroare in care utilizatorul este atentionat ca parola sau username-ul sunt incorecte. Daca totusi datele sunt corecte si sesiunea de login este setata pe true, atunci vom fi redirectati catre o noua pagina.



3.3.3 Backend-ul cererilor de inregistrare ce urmeaza a fi aprobate/respinse de catre administrator

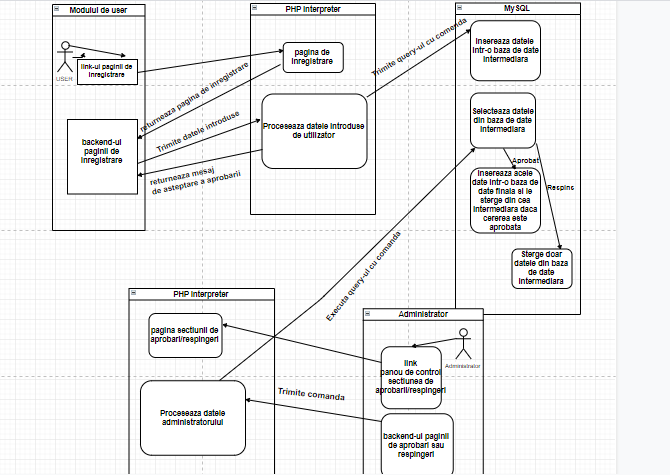


Figura 3-Cererea de inregistrare a unui utilizator ce asteapta aprobarea administratorului

Dupa cum se observa si in diagrama de mai sus din figura 3 ce reprezinta cererile de aprobare trimise de utilizator spre aprobare de catre administrator, putem observa ca odata ce utilizatorul va accesa link-ul spre inregistrare, acesta va trebui sa completeze un formular. Acest formular pe langa username si parola, va trebui sa introduca date de contact precum nume, prenume, oras, tara si numar de telefon. In momentul in care toate aceste campuri au fost completate, utilizatorul va trimite cererea de inregistrare, moment in care va fi primi o alerta prin care este anuntat ca cererea a fost inregistrata si va trebui sa astepte aprobarea acesteia. In acest moment datele sunt procesate si trimise printr-un query catre o baza de date intermediara. In momentul in care administratorul aplicatiei se va conecta si va verifica lista de cereri primite, acesta va avea optiunea de a aproba sau de a respinge cererea.

Daca acesta va aproba cererea, comanda va fi procesata si va fi trimis un query in acest sens care sa selecteze datele respectivei cereri din baza de date intermediara. Odata ce au fost selectate, va interveni un alt query care va prelua aceste date si le va insera intr-o baza de date finala. Totodata, odata preluate si inserate aceste date, datele respective din baza de date intermediara vor fi sterse, pentru a nu fi afisata la nesfarsit acea cerere de inregistrare.

Mai exista si cazul in care administratorul respinge cererea, in acest caz aceasta comanda va fi procesata si va fi trimis un query in acest sens care sa selecteze datele cererii respective si sa stearga cererea.

In concluzie, in cazul aprobarii, odata ce aceste date au ajuns in baza de date finala, utilizatorul va avea dreptul de a se conecta la noul sau cont creat.

Pentru sistemul de inregistrare al utilizatorilor este necesar in prima faza creerea un formular in care utilizatorul poate sa isi introduca datele personale, dupa aceasta este necesara creerea a doua tabele in baza de date, primul tabel va fi unul de cereri de inregistrare, iar cel de al doilea tabel va fi unul final, in care se vor salva cererile aprobate de administrator.

Dupa aceea, este necesara conectarea paginii de inregistrare la baza de date, odata ce a fost conectata vom selecta tabelul de cereri de inregistrare in care vom dori sa se stocam temporar datele.

Daca toate datele au fost introduse si daca valoarea post-ului butonului de inregistrare este setata, atunci vom seta valorile introduse de utilizator in niste variabile. Codul respectiv este urmatorul:

if(isset($\_POST['submit\_reg'])){

$first\_name=$\_POST['firstname'];

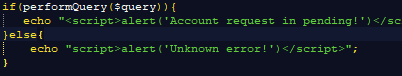
.....

$message="$first\_name $last\_name requested for account approval";

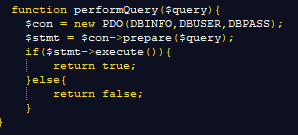
In cazul parolei in momentul cand aceasta va fi stocata, aceasta va fi criptata utilizand md5(), pentru a nu se cunoaste parola utilizatorului de catre administrator sau de catre altcineva.

Dupa aceasta operatiune de stocare a datelor introduse in variabile, vom scrie un query SQL pentru a insera detaliile completate de utilizator in baza de date in tabelul de cereri de inregistrare. Iar query-ul pentru aceasta operatiune este de forma: $query = "INSERT INTO `registration\_requests` (`id`, `firstname`, .....,`message`) VALUES (NULL, '$first\_name',.....,'$message')";

In cazul in care datele si destinatia sunt valide, query-ul va fi procesat si utilizatorul va primi o instiintare prin care este anuntat ca cererea de inregistrare urmeaza sa fie aprobata de catre administrator. In cazul in care exista o problema si query-ul nu este procesat, utilizatorul va fi anuntat ca exista o problema in procesul acestuia de inregistrare.



Metoda performQuery() este o metoda ce preia un query ca si parametru, iar ca si structura este in felul urmator:



Parametrului $con in cazul de fata i se atribuie conexiunea la baza de date. PDO fiind o instanta ce reprezinta aceasta conexiune la baza de date, si care are ca parametri informatiile despre baza de date si despre utilizator.

Parametrului $stmt i se atribuie parametrul $con care pregateste un query SQL ce va returna o stare ce va fi utilizata pentru operatiile urmatoare din instructiune. Astfel $stmt->execute() va executa query-ul pregatit anterior, iar in cazul in care va fi executat se va returna true, iar in caz contrar se va returna fals. Fapt pentru care si la procesul de inregistrare se verifica daca valoarea returnata este true sau false.

Astfel, dupa ce a fost explicat intregul proces ce are loc pentru o cerere de inregistrare de catre utilizator, urmeaza sa descriem amanuntit si intregul proces al acceptarii/respingerii cererilor de inregistrare.

Trecand peste intregul procedeu al conectarii administratorului la panoul de comanda si ajungand direct la sectiunea din panoul acestuia unde poate aproba sau respinge noile cereri de inregistrare aparute, vom descrie putin mai pe larg partea de back-end a intregului proces si ce implica acesta.

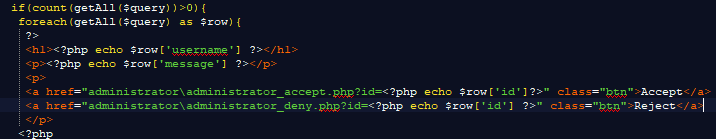
Primul lucru ce se va face pe pagina este acela de a se apela functia session\_start() si trebuie declarat la inceputul paginii inainte care orice alt output sa fie generat. Rolul lui session\_start() in cazul nostru este acela de a recrea o sesiune apelandu-l iar acesta va a trimite cheia corespunzatoare tabloului asociat lui $\_SESSION.

De asemenea, pagina respectiva trebuie conectata la baza de date si selectata baza de date ce contine tabelul cu informatiile ce ne intereseaza, in cazul nostru tabelul in care au fost salvate cererile de inregistrare, impreuna cu toate informatiile ce il contine.

Astfel, in momentul in care administratorul se conecteaza, ($\_SESSION[‘login\_adminstrator’]=true;) sesiunea de logare a administratorului este setata pe true. Datorita acestui fapt pe pagina de cereri se va verifica daca sesiunea respectiva transmisa prin session\_start() este adevarata, iar in cazul in care nu este adevarata, inseamna ca administratorul nu este conectat si va fi redirectionat spre pagina de logare si nu i se va permite sa acceseze pagina de confirmari/respingeri de cereri de inregistrare. Insa in cazul in care utilizatorul de tip administrator este totusi conectat la cont, inseamna ca sesiunea este adevarata si va putea accesa pagina fara niciun fel de probleme.

In momentul in care o va accesa, va fi existent un query, de tipul $query="SELECT \* FROM registration\_requests;"; ce va urma sa fie executat.

Acel query se va executa atata timp cat numarul de linii din tabelul de cereri de inregistrare este mai mare decat 0 si se vor afisa numele utilizatorului, un mesaj predefinit si un button de confirmare si unul de respingere. Fiecare cerere de acest tip are asociate 2 butoane cu id-ul de identificare a fiecarei cereri, pentru a putea sa se putea sti ce se accepta si ce se respinge mai exact. Codul acestei operatiuni fiind acesta.



Daca se va apasa pe butonul de accept, se va accesa fisierul php ce va face intreaga operatiune de preluare a tuturor datelor din tabelul de cereri de rezervare si vor fi introduse in tabelul final de utilizatori.

In primul rand se va folosi un parametru $id care sa preia id-ul cererii respective, operatia fiind de forma $id=$\_GET[‘id’];, iar dupa aceea va fi folosit un query de tipul $query = "SELECT \* FROM `registration\_requests` where `id` = '$id'; ce va avea rolul de a identifica ce informatii sa fie selectate din tabelul de cereri si pe urma inserate in tabelul de utilizatori, atata timp cat numarul de linii selectate din tabel este mai mare decat 0.

Query-ul utilizat pentru inserarea informatiilor este de forma:

$query = "INSERT INTO `users` (`id`, `firstname`,....,`country`) VALUES (NULL, '$first\_name',....,'$country')";

Dupa ce aceasta operatiune de inserare a cererii de inregistrare ce devine o inregistrare finalizata este terminata, se vor sterge informatiile cu respectivul id de identificare pentru a nu mai aparea incontinuu in panoul de cereri al administratorului. Operatiunea de stergere este de asemenea tot prin intermediul unui query.

In cazul in care se va apasa pe butonul de reject, se va accesa respectivul fisier php si pur si simplu se va accesa baza de date si se vor sterge datele din tabelul de cereri de inregistrare pe baza id-ul pe care il detine cererea cu informatiile sale.

3.3.4 Back end-ul cererilor de inchiriere ce urmeaza a fi aprobate/respinse de catre administrator

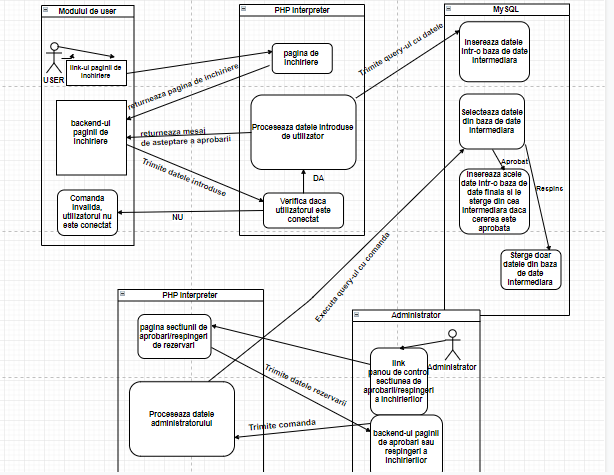


Figura 4-Cererea de inchiriere a unui utilizator ce asteapta aprobarea din partea utilizatorului

Dupa cum se observa si in diagrama de mai sus din figura 4 ce reprezinta cerera de inchiriere a unui utilizator trimisa spre aprobare, putem observa ca odata ce utilizatorul va accesa link-ul de inchirieri de masini acesta va fi nevoit sa completeze date precum data de inceput a inchirierii si data de finalizare. In momentul in care utilizatorul de tip client va trimite cererea de inchiriere, se va verifica in prima faza daca acesta este conectat la contul sau. In caz contrar, cererea nu va merge mai departe si utilizatorul aplicatiei va fi redirectionat catre sectiunea de conectare la cont. Totusi, daca utilizatorul este conectat la contul sau, cererea de inchiriere va fi inregistrata si va primi o alerta prin care este informat ca trebuie sa astepte aprobarea din partea administratorului. Cererea sa de inchiriere in acest moment va fi procesata si trimise datele introduse printr-un query catre o baza de date intermediara. In momentul in care administratorul se va conecta la panoul de comanda si va verifica cererile de inchiriere, acestuia i se vor afisa toate cererile, iar respectivul va avea optiunea de a aproba sau de a respinge cererea in functie de disponibilitatea masinii cerute.

Daca totusi acesta va aproba cererea de inchiriere, comanda va fi procesata si de asemenea va fi trimisa prin intermediul unui query ce va selecta din baza de date intermediara, cererea respectiva. Din momentul in care a fost selectata, va interveni un alt query cu o comanda care va introduce respectivele date, intr-o baza de date finala. De asemenea, ca si la cererea de inregistrare, pentru a nu exista problema afisarii repetitive a aceleasi cereri, se va sterge din baza de date intermediara cererea. In cazul respingerii acestei cereri, comanda se procesa si va fi trimis un query care sa selecteze datele cererii si sa le stearga din baza de date intermediara.

In concluzie, odata ce cererea de inchiriere a fost aprobata si introdusa in baza de data finala, utilizatorul va avea optiunea de a isi vedea datele inchirierii efectuate.

3.3.5 Back end-ul afisarii masinilor, a specificatiilor acestora si a cautarii unui anumit de tip de masina utilizand filtre

* Listarea masinilor

In momentul in care utilizatorul va accesa pagina de listare a masinilor, backend-ul paginii va interveni si va incepe sa performeze instructiunile necesare afisarii.

Astfel, se va utiliza un query ce are rolul de a selecta doar anumite date din tabelul de masini al bazei de date a aplicatiei. Atata timp cat nu s-a ajuns la finalul tabelului, se va executa query-ul respectiv prin intermediul unei functii care returneaza un tablou cu toate informatiile ce s-au dorit a fi selectate din tabelul respectiv si vor fi afisate pe pagina. In cazul de fata se vor extrage din baza de date, numele masinii, anul de productie, numele pozei masinii, tipul de combustibil, numarul de locuri si pretul pe zi al acesteia. Pentru afisarea pozei masinii, calea catre locatia pozei a fost scrisa in codul html al paginii, singurul element extras din baza de date fiind numele pozei.

In urma unei descrieri generaliste a ceea ce se intampla pentru a se efectua listarea masinilor, acum vom discuta mai pe larg si pe baza codului existent, ceea ce se intampla in acest proces de listare al masinilor.

La fel ca si la celelalte functionalitati, este in totalitatea necesara apelarea functiei session\_start() pentru a se incepe o sesiune noua, cat si efectuarea conectarii la baza de date si selectarea acesteia. Dupa ce aceste lucruri au fost indeplinite, se va folosi tot un query ce va avea rolul de a selecta anumite elemente ce le dorim afisate din tabelul de vehicule al bazei de date la care suntem conectati.

De asemenea, citirea din tabel se va face atata timp cat numarul de linii al query-ul folosit pentru selectie este mai mare decat 0 si atata timp cat avem un rezultat al query-ului ca si $row, vom afisa fiecare element al liniei ce dorim sa fie afisat. Faptul ca este necesar ca numarul liniilor sa fie mai mare decat 0 inseamna ca operatiunea se va face pana cand se ajunge la finalul citirii tabelului.

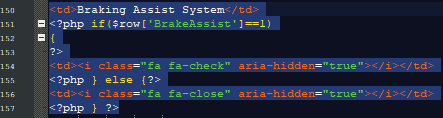
Pentru a vedea o afisare mai detaliata a unei anumite masini din lista extrasa si afisata din baza de date, se va apasa pe butonul de ‘View Details’ care are atasat id-ul din tabelul de masini a masinii respective. Odata ce acest buton a fost apasat utilizatorul va fi redirectionat spre o noua pagina a carei functionalitati o vom discuta in subcapitolul ce urmeaza.

* Afisarea detaliata a specificatiilor si optiunilor unei masini selectate de utilizator.

Pentru afisarea unor informatii mai detaliate a specificatiilor unei masini se va accesa butonul de vizualizare detaliata iar utilizatorul va fi directionat catre link-ul paginii de specificatii ale masinii, in care exista de asemenea si optiunea de rezervare a masinii, insa despre acest mecanism am discutat anterior.

Astfel, la fel ca si la listarea masinilor si aceste informatii sunt afisate dintr-o baza de date. Singura diferenta fiind aceea ca optionalele de care dispune o masina au fost marcate cu 0 sau cu 1 in baza de date, fapt pentru care modul de afisare a specificatiilor va diferi de la o masina la alta in functie de valoarea pe care o anumita specificatie o are in baza de date. Pentru a ne lamuri pana la capat, valoarea 0 asociata unei specificatii semnifica faptul ca masina nu dispune de aceasta optiune, iar in cazul in care valoarea este 1, inseamna ca masina dispune de optiunea respectiva.

O descriere mai detaliata si mai elaborata a acestei functionalitati ar fi aceea ca pentru buna functionare a acesteia, sunt necesare in primul rand functia session\_start() pentru inceperea unei noi sesiuni si aceea de conexiune la baza de date. Dupa ce cele doua cerinte au fost indeplinite, pagina noastra va primi id-ul asociat butonului prin intermediul operatiunii de atribuire a lui $id=$\_GET[‘id’]. Parametrul $id va fi astfel utilizat in query-ul nostru pentru a putea selecta si afisa informatiile corespunzatoare cu ceea ce a dorit utilizatorul sa vada. Parcurgerea query-ul se va face atata timp cat numarul de linii este mai mare decat 0 si atata timp cat rezultatul query-ul este primit si atribuit lui $row. Totusi, anumite specificatii extrase din tabelul de masini al baze de date vor fi identificate cu valori de tip int, mai exact cu 0 sau cu 1. In functie de ce valoare vor avea modul de afisare va fi diferit. Pentru a arata constructia si logica din spate a acestei operatii, voi afisa o mica secventa de cod din proiect.



Astfel, conform codului afisat, in cazul in care optiunea de asistare la franare este in baza de date marcata cu valoarea 1, afisarea se va face intr-un tabel la aceasta sectiune cum ca masina dispune de aceasta optiune, iar in caz contrar se va afisa ca aceasta optiune nu este disponibila pe masina respectiva.

* Cautarea unei masini utilizand filtre

Pentru cautarea unei masini va exista un selector ce va avea toate marcile de masini prezente in baza de date a aplicatiei, iar in momentul in care utilizatorul va selecta o marca de masina de care ar fi interesat si va apasa butonul de search, vor fi selectate din baza de date doar acele masini asociate marcii cautate de catre utilizatorul respectiv.

Aceasta interfata de cautarea de masini se afla in cadrul paginii de listare de masini, insa fiind o cu totul alta functionalitate, am considerat ca merita un subcapitol separat care sa ii descrie modul de functionare.

3.3.6 Back end-ul afisarii informatiilor utilizatorului si a afisarii cererilor de inchiriere aprobate

Utilizatorul pe langa optiunea de inchiriere si de vizualizare a masinilor disponibile, are optiunea de a isi vizualiza informatiile personale introduse la inregistrare, iar acestea reprezinta profilul contului respectiv. De asemenea utilizatorul de tip client daca va dori sa vizualizeze daca o anumita cerere de inchiriere a fost aprobata de catre administrator, va putea sa beneficieze si de o astfel de optiune.

In continuare voi descrie mai pe larg intregul mecanism dupa care functioneaza respectivele.

* Afisarea informatiilor utilizatorului

In primul rand, utilizatorul a carei cereri de inregistrare a fost aprobata de catre administrator, ii vor fi salvate informatiile personale introduse in formularul de inregistrare intr-o baza de date in tabelul de utilizatori.

Odata ce s-a inregistrat, utilizatorul va avea un id de identificare in baza de date care va fi cunoscut in momentul in care va fi dori sa i se afiseze informatiile. Astfel, intreaga operatiune de cautare in baza de date in tabelul de utilizatori se va face dupa id-ul respectivului utilizator. In cazul in care nu ar fi existat un element de identificare a contului, in cazul nostru id-ul contului curent, nu s-ar fi putut stie care informatii din tabelul de utilizatori este nevoie sa fie afisate pentru contul respectiv si astfel nu s-ar fi stiut ce sa se caute mai exact.

Trecand peste lucrurile generale discutate pana acum despre functionalitatea operatiei de afisare a informatiilor utilizatorului din cadrul aplicatiei web de inchirieri de masini, va urma o descriere ceva mai detaliata a intregului proces si a rolului bazei de date in cazul de fata.

Dupa cum se stie, vom avea nevoie in primul rand de functia de session\_start() si de conectarea la baza de date a paginii. Pentru a putea beneficia de aceasta optiune, utilizatorul trebuie in primul rand sa fie conectat, deoarece in momentul cand acesta se conecteaza($\_SESSION[‘login’]=true; ) sesiunea de conectare a utilizatorului va fi setata pe valoarea booleana true. Astfel, datorita acestei operatii, pe pagina unde vor fi afisate informatiile contului se va verifica daca sesiunea transmisa de session\_start() este sau nu setata pe valoarea booleana true. In cazul in care valoarea sesiunii este setata pe true, inseamna ca utilizatorul este conectat si i se va permite sa acceseze pagina de vizualizare a informatiilor contului. In caz contrar insa, va exista o problema, iar utilizatorul va fi redirectionat catre pagina de conectare. Acest lucru, in cele din urma ajutand la securitatea si buna functionare a aplicatiei, nepermitand unui utilizator neconectat sa efectueze operatiuni ce nu ar avea dreptul sa le efectueze.

Revenind insa la afisarea informatiilor contului, aceste informatii vor fi extrase utilizand un query de forma $query="Select \* from users where `id` = '$id'; "; cu care vom efectua o cautare in tabelul de utilizatori confirmati si inregistrati din baza de date. Contul curent de pe care s-a facut aceasta operatiune de afisare a detaliilor detine un id in baza de date, dupa care este identificat, iar astfel dupa acel id se va cunoaste ce informatii sunt dorite pentru afisare si de catre cine. Ideea esentiala a intregii operatiuni este aceea ca doar informatiile asociate contului prin id vor fi preluate din baza de date, nu si a altor utilizatori.

* Afisarea cererilor de inchiriere aprobate de administrator

La fel ca si la afisarea informatiilor contului unui utilizator, se utilizeaza acelasi mecanism si la cel de cereri de inchiriere aprobate. Singura diferenta fiind aceea ca datele sunt preluate dintr-un alt tabel din baza de date si ca poate exista posibilitatea ca nici o informatie legata de inchirieri aprobate sa nu existe in baza de date, datorita faptului ca nu a fost aprobata inca niciuna. Insa de asemenea pot exista si mai multe cereri ce au fost aprobate si asociate contului.

3.4 Implementarea front-end-ului si importanta sa

Dezvoltarea partii de front-end reprezinta o provocare si o etapa extrem de importanta in realizarea unei aplicatii, fiind totodata un element crucial ce are scopul de a atrage si mentine interesul posibililor clienti.

Intr-o lume in care numarul aplicatiilor web sunt in crestere, o pagina web a unei companii reprezinta o prima impresie a clientilor si totodata o prima interactiune cu acestia. Dezvoltatorul aplicatiei trebuie sa ia in considerare un plan bine definit inainte de a se apuca de constructia felului in care va arata pagina.

Clientul care va intalni pagini web ce au probleme de functionare, grafica si design de o calitate inferioara, plus alte posibile probleme, va avea sanse foarte mari sa nu se mai intoarca si a doua oara pe aceasta.

Astfel, constructia aplicatiei si organizarea acesteia trebuie sa fie foarte buna. De asemenea trebuie pastrata totodata o interfata simplista si intuitiva pentru ca utilizatorul sa aiba o experienta cat mai buna de-a lungul timpului petrecut pe pagina.

In realizarea aplicatiei s-a optat pentru partea de front-end implementarea unui template existent. Acest template a fost adaptat pentru aplicatia mea de inchirieri de masini si se afla in concordanta cu specificatiile acesteia.

Alegerea unui astfel de template, consider ca reprezinta o metoda simplificata, mult mai rapida si eficienta in privinta realizarii aplicatiei dorite, astfel beneficiind de un front end profesionist.

Toate fisiere initial au fost de tipul html, dar datorita nevoii de a construi si partea de back-end in interiorul acestora, acestea au devenit php. De asemenea continutul acestor pagini ale template-ului au fost alterate, astfel unele elemente ajungand sa fie scoase, altele introduse, iar la altele modificate dimensiunile, in functie de nevoile si specificatiile aplicatiei.

Datorita avantajelor oferite de utilizarea unui template deja existent, in locul construirii unui front end de la zero, s-a putut oferi o calitate mai ridicata a operatiunilor efectuate pe aplicatie si de asemenea si o varietate a acestora mai crescuta.

Concluzia, ar fi aceea ca aplicatia poate oferi o experienta mult mai placuta a utilizatorului in acest fel.