

UNIVERSITATEA „ALEXANDRU IOAN CUZA” IAȘI  
FACULTATEA DE INFORMATICĂ



LUCRARE DE LICENȚĂ

# **MesPla – Measurements and Planning, aplicație WEB pentru servicii cadastrale**

propusă de

***Ionuț-Iulian Preda***

**Sesiunea:** *Iulie 2017*

Coordonator științific

**Conf. Dr. Anca Vitcu**

UNIVERSITATEA „ALEXANDRU IOAN CUZA” IAȘI  
FACULTATEA DE INFORMATICĂ

# **MesPla – Measurements and Planning, aplicație WEB pentru servicii cadastrale**

***Ionuț-Iulian Preda***

**Sesiunea:** *Iulie 2017*

Coordonator științific

**Conf. Dr. Vitcu Anca**

# DECLARAȚIE PRIVIND ORIGINALITATE ȘI RESPECTAREA DREPTURILOR DE AUTOR

Prin prezenta declar că Lucrarea de licență cu titlul „*Mespla – Measurements and Planning, aplicație WEB pentru servicii cadastrale*” este scrisă de mine și nu a mai fost prezentată niciodată la o altă facultate sau instituție de învățământ superior din țară sau străinătate. De asemenea, declar că toate sursele utilizate, inclusiv cele preluate de pe Internet, sunt indicate în lucrare, cu respectarea regulilor de evitare a plagiatului:

- toate fragmentele de text reproduse exact, chiar și în traducere proprie din altă limbă, sunt scrise între ghilimele și dețin referința precisă a sursei;
- reformularea în cuvinte proprii a textelor scrise de către alți autori deține referința precisă;
- codul sursă, imaginile etc. preluate din proiecte *open-source* sau alte surse sunt utilizate cu respectarea drepturilor de autor și dețin referințe precise;
- rezumarea ideilor altor autori precizează referința precisă la textul original.

Iași,

Absolvent *Ionuț-Iulian Preda*

---

## DECLARAȚIE DE CONSIMȚĂMÂNT

Prin prezenta declar că sunt de acord ca Lucrarea de licență cu titlul „ *Mespla – Measurements and Planning, aplicație WEB pentru servicii cadastrale*”, codul sursă al programelor și celelalte conținuturi (grafice, multimedia, date de test etc.) care însoțesc această lucrare să fie utilizate în cadrul Facultății de Informatică.

De asemenea, sunt de acord ca Facultatea de Informatică de la Universitatea „Alexandru Ioan Cuza” Iași să utilizeze, modifice, reproducă și să distribuie în scopuri necomerciale programele-calculator, format executabil și sursă, realizate de mine în cadrul prezentei lucrări de licență.

Iași,

Absolvent *Ionuț-Iulian Preda*

---

# Cuprins

Introducere .....	1
Contribuții.....	2
1. Descriere aplicație .....	3
1.1 Baza de date .....	3
1.2 Creare cont.....	3
1.3 Autentificare .....	5
1.4 Bara de navigare .....	6
2. Funcționalități ale aplicației.....	8
2.1 Funcționalități utilizator.....	10
2.1.1 Pagina Request .....	10
2.1.2 Pagina Home .....	17
2.1.3 Pagina de profil .....	18
2.1.4 Pagina de informații .....	21
2.1.5 Pagina About .....	23
2.1.6 Pagina contact .....	24
2.2 Funcționalități administrator.....	26
2.2.1 Pagina Home .....	26
2.2.2 Pagina de istoric a cererilor .....	29
2.2.3 Pagina de informații .....	31
2.2.4 Pagina de profil .....	33
2.2.5 Pagina de management.....	35
2.2.6 Pagina de mesaje .....	40
3. Dezvoltări ale aplicației .....	42
Aplicații similare .....	43
Concluzii.....	44
Bibliografie .....	45

## Introducere

De-a lungul timpului, industria construcțiilor (civile, de drumuri, etc.) în România, s-a dezvoltat din ce în ce mai mult, cunoscând o creștere procentuală de 3% de la an la an. Având în vedere acest aspect putem deduce cu ușurință că și numărul cererilor pentru servicii care au ca scop legalizarea acestor construcții este de asemenea într-o continuă ascensiune. Cu cât numărul cererilor este mai mare, cu atât numărul orelor pierdute de o persoană, pentru a programa și solicita un anumit serviciu de la o societate cadastrală acreditată, crește. Cu toate că revoluția digitală se înfiripă din ce în ce în mai multe domenii (digitalizarea industriilor), în industria construcțiilor nu avem, cel puțin în România, prea multe instrumente care să faciliteze nevoile oamenilor.

**MesPla – Measurements and Planning**, reprezintă o aplicație WEB care oferă utilizatorilor înregistrați, printr-o interfață prietenoasă, posibilitatea de a solicita de la o societate cadastrală acreditată un anumit serviciu, prezentat la o anumită dată (solicitată de către utilizator). Această aplicație are ca scop reducerea timpului de așteptare al persoanelor care au nevoie de servicii cadastrale tocmai prin prisma efectuării cererilor online.

Un alt plus al prezentei aplicații, în materie de inovație, îl reprezintă faptul că nu se adresează doar cetățenilor din municipiul Iași ci și celor care locuiesc în acest județ. Ba mai mult, se adresează tuturor județelor din partea nord-estică a țării. Toți utilizatorii beneficiază de aceleași tipuri de servicii și aceleași modalități de procesare a întocmirii documentelor, însă prețul suferă modificări în funcție de locul în care echipa de ingineri trebuie să se deplaseze.

De asemenea, în cadrul aplicației este implementat și un sistem care simulează efectuarea de plăți online în vederea obținerii serviciilor dorite de către utilizator. Orice utilizator poate alege oricare combinație de forma "serviciu - timpul de procesare al documentelor - județ", atât timp cât suma de bani ce corespunde acestei combinații este mai mică sau egală cu suma de bani ce se află, la momentul efectuării cererii, depusă în contul utilizatorului.

## Contribuții

Pentru început, pot evidenția ca o primă contribuție în vederea conceperii prezentei aplicații, etapa de cercetare pentru a descoperi o soluție în contextul timpului de așteptare mult prea mare în cazul în care o persoană are nevoie de anumite servicii cadastrale.

Am ajuns la concluzia că, o aplicație WEB poate diminua considerabil acești timpi sărind practic peste etapa în care o persoană așteaptă la cozi interminabile doar pentru o simplă programare. Nevoia oamenilor de anumite servicii este satisfăcută de *Mespla* care aduc aceste progrămări în mediul online.

În momentul în care toate aceste idei au prins rădăcini, a trebuit să mă hotărâsc în privința tehnologiilor ce vor fi folosite pentru a realiza această aplicație. Având în vedere că de-a lungul timpului, de cele mai multe ori am cochetat în partea de backend cu tehnologia PHP, am ales ca aceasta să stea și la baza prezentei lucrări<sup>[1]</sup>. Pe lângă logica aplicației, partea vizuală este redată prin combinarea următoarelor limbaje: HTML5, CSS3<sup>[2]</sup>, JAVASCRIPT<sup>[3]</sup>, JQUERY UI. Lăsând la o parte cele de mai sus, inserarea, stocarea și salvarea datelor este asigurată de utilizarea unei baze de date MySQL<sup>[4]</sup> (gestionată prin interfața PHP myAdmin). Ca nivel de securitate, doar persoanele ce au un cont valid se pot conecta în aplicație, URL-urile fiind disponibile doar dacă variabilele de sesiune au fost setate în prealabil. Totodată, toate insert-urile, update-urile sau interogările în baza de date, ce se realizează cu date preluate din câmpuri de tip input, sunt realizate doar cu ajutorul "prepared statement-urilor", evitând astfel unui atac bazat pe SQL injection<sup>[5]</sup>. Astfel, persistența datelor este asigurată și în cazul utilizatorilor "rău voitori" care pot distruge întreg sistemul sau pot sustrage informații și date sensibile.

Așadar, această aplicație poate fi privită ca un punct de pornire pentru digitalizarea acestei industrii, luând în considerare și anumite aspecte importante pentru eventualii clienți (în acest caz, timpul irosit).

# 1. Descriere aplicație

## 1.1 Baza de date

Stocarea și manipularea datelor se realizează cu ajutorul sistemului de gestiune *phpMyAdmin* prin intermediul unui browser web destinat exclusiv bazelor de date MySQL. Aplicația *MesPla* cumulează un total de opt tabele (*accounts*, *county*, *date\_resource*, *date\_resource*, *messages*, *processing*, *requests*, *services*, *users* – figura 3), în care se stochează informații cu privire la utilizatori, cereri de servicii cadastrale, sume de bani și conturi, județe în care firma se deplasează, tipul de procesare a documentelor, etc.

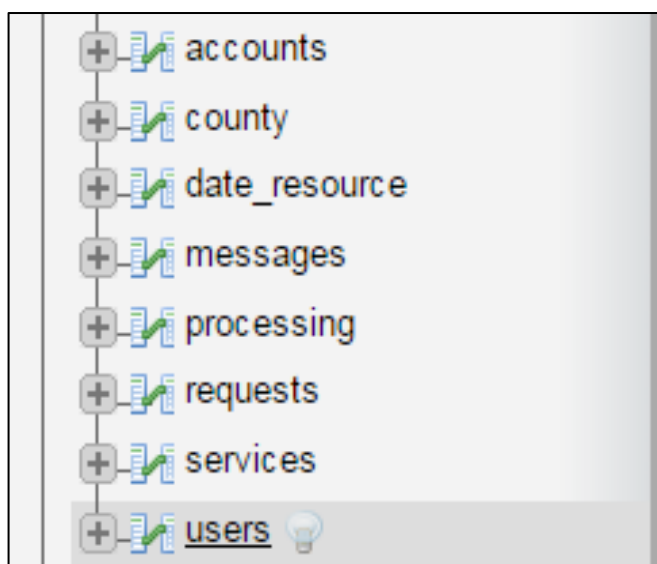


Figura 3: Tabelele utilizate de aplicație

## 1.2 Creare cont

Pentru a dispune de întreaga suită de funcționalități, utilizatorii aplicației trebuie mai întâi să își creeze un cont. Formularul din figura 4 deservește acestui scop și conține doar câmpuri obligatorii fără de care înregistrarea nu se poate efectua cu succes.



The image shows a registration form titled "REGISTER". It contains the following fields and elements:

- username
- firstname
- surname
- password
- phone number
- email
- age
- Sex:
  - ☒ Male
  - ☐ Female
- Register

Figura 4: Formular de înregistrare

Butonul "Register" colecționează datele introduse în formular, doar dacă toate câmpurile au fost completate. Suplimentar, câmpurile "username, phone number și email" sunt supuse și unor validări în partea de back-end. Un nume de utilizator nu mai poate fi folosit dacă deja există în sistem, numărul de telefon trebuie să fie de forma "07xxxxxxx" iar emailul obligatoriu să respecte următorul șablon: [def@xyz.com](mailto:def@xyz.com), cum se poate de altfel observa în figura 5.

```

if(isset($_POST["submitBtn"])) {
    include 'Connection.php';
    $usernameExists = false;
    $phoneCheck = true;
    $result=$conn->query("SELECT * FROM users WHERE Username= '" . $_POST["username"] . "'");
    if($result->num_rows > 0) {
        $usernameExists = true;
    } else {
        $phn = $_POST["phone"];
        if (strcmp(substr($phn,0,2),"07") != 0 || strlen($phn) != 10) {
            $phoneCheck = false;
        }
    }
    if(!$usernameExists && $phoneCheck == true) {

```

Figura 5: Validări asupra câmpurilor de înregistrare

### 1.3 Autentificare

Dacă înregistrarea s-a realizat cu succes utilizatorul este redirecționat către pagina de autentificare care conține două câmpuri: *username* și *password* (figura 6).

here'."/>

Figura 6: Formular de autentificare

Autentificarea se realizează doar în cazul în care, atât numele de utilizator cât și parola corespund unor date deja existente în baza de date. În caz contrar, un mesaj stilizat va fi afișat prin care utilizatorul este informat că datele de conectare sunt greșite. De asemenea, această pagină conține un alt aspect cheie și anume, variabilele de sesiune. Cu ajutorul acestora se va

putea face distincția dintre un utilizator obișnuit și administrator, acesta din urmă beneficiind de cu totul alte privilegii. Se vor reține următoarele aspecte:

- Rolul (administrator sau client)
- Logarea cu success
- Id-ul utilizatorului
- Username-ul
- Email-ul utilizatorului

Cele de mai sus se pot observa cu ușurință în figura 7:

```
if($rowNum > 0) {  
    $row = $result->fetch_assoc();  
    $credentials = true;  
    $_SESSION["isLoggedIn"] = true;  
    $_SESSION["id"] = $row["Id"];  
    $_SESSION["username"] = $row["Username"];  
    $_SESSION["email"] = $row["Email"];  
    if ($row["Role"] == "ADM")  
    {  
        $_SESSION["role"] = "admin";  
    } else  
    {  
        $_SESSION["role"] = "client";  
    }  
    $newURL = "http://localhost/deton/index.php";  
    header('Location: '.$newURL);  
} else {  
    $credentials = false;  
}
```

Figura 7: Setarea variabilelor de sesiune

## 1.4 Bara de navigare

Odată realizată distincția dintre client și administrator, funcționalitățile aplicației vor fi disponibile cu ajutorul unui menu personalizat pentru fiecare în parte. Foile de stil au fost adaptate<sup>[6]</sup> astfel încât, fiecare trecere a cursorului peste orice element din menu, va determina apariția unei linii subțiri deasupra acestuia, iar totodată culoarea textului va deveni neagră (figura 8).

```
nav a:hover {  
    color: black;  
}  
  
nav a::before {  
    content: '';  
    display: block;  
    height: 5px;  
    width: 0%;  
    background-color: black;  
    position: absolute;  
    top: 7%;  
    transition: all ease-in-out 350ms;  
}  
  
nav a:hover::before {  
    width: 100%;  
}
```

Figura 8: Stilizarea navigării

Această linie subțire are rolul de a evidenția utilizatorului faptul că acesta se poate îndrepta către pagina dorită de acesta. Practic linia există de la încărcarea paginii dar cu o lățime de 0 % px. Odată ajuns cursorul peste un element din listă, va determina o tranziție de 350 milisecunde în care lățimea va crește de la 0 la 100 % px.

## 2. Funcționalități ale aplicației

Funcționalitățile aplicației sunt puse la dispoziție doar utilizatorilor care au un cont valid creat înainte de logarea propriu zisă. Orice încercare a unui utilizator de a accesa orice pagină din aplicație, fără ca în prealabil să-și fi introdus credențialele, se va solda cu o redirecționare către pagina de "Login".

În cadrul aplicației se disting două tipuri principale de utilizatori:

- Utilizator obișnuit
- Administrator

Pentru prima situație avem puse la dispoziție următoarele funcții:

- Înregistrare și logare
- Editarea informațiilor personale
- Posibilitatea cererii unui serviciu cadastral
- Posibilitatea revocării unei cereri deja efectuate
- Depunerea pe site a unei sume de bani
- Posibilitatea trimiterii de mesaje administratorului
- Informații cu privire la suma de bani depusă în cont

În cazul administratorului, următoarele scenarii sunt posibile:

- Logarea în aplicație
- Aprobarea sau respingerea cererilor
- Scurt istoric al tuturor cererilor efectuate
- Obținerea unui raport al numărului de cereri efectuate
- Editarea informațiilor personale
- Informații cu privire la suma de bani depusă în cont
- Adăugarea de noi resurse la o dată aleasă
- Posibilitatea de a exclude anumiți utilizatori
- Retragera unui anumite sume de bani din cont
- Vizualizarea mesajelor primite de la utilizatori

Pentru o privire de ansamblu cât mai concisă, următoarele două figuri exemplifică cazurile de utilizare ale aplicației pentru un utilizator, respectiv administrator (figura 1 și figura 2):

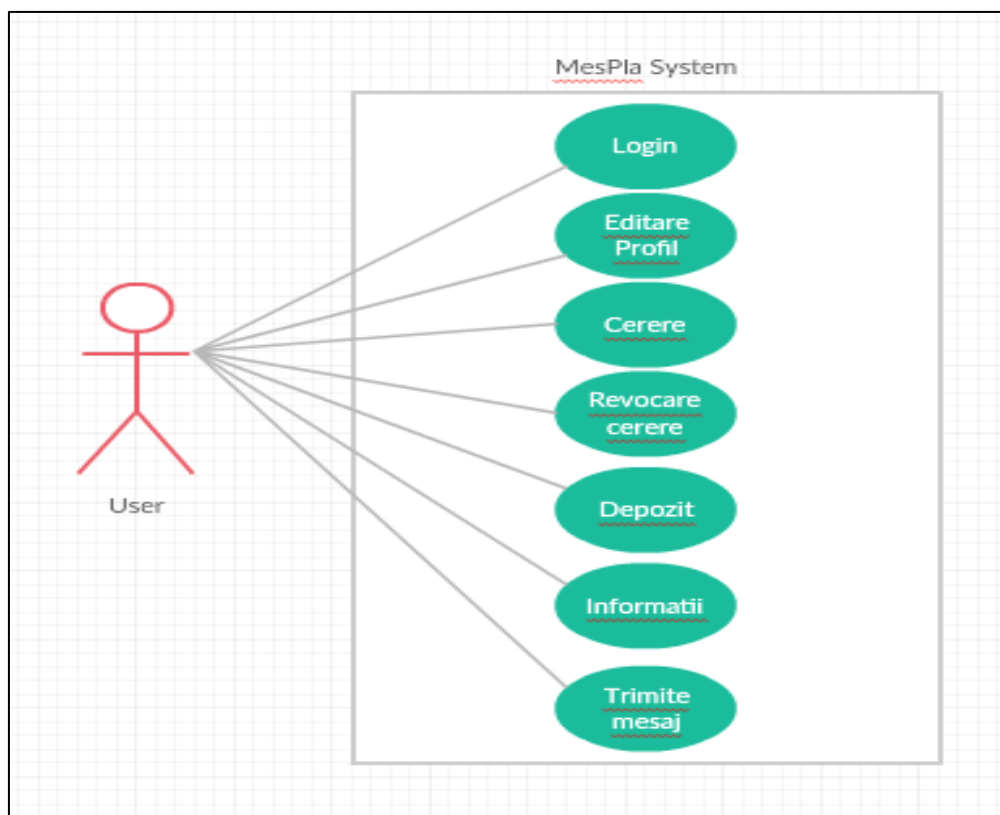


Figura 1: Diagrama cazurilor de utilizare, utilizator

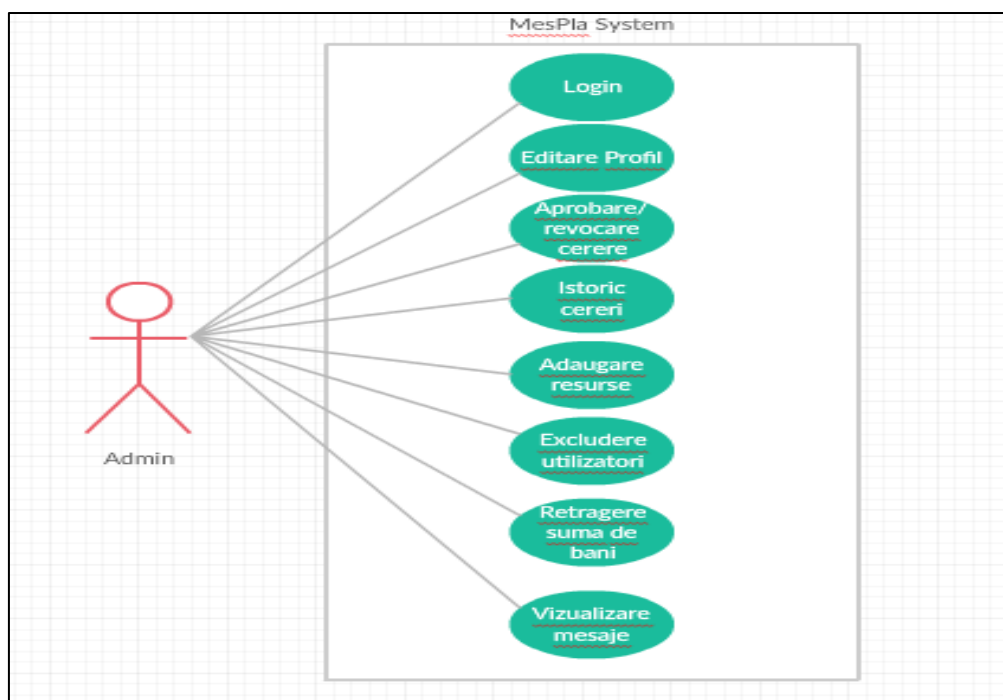


Figura 2: Diagrama cazurilor de utilizare, administrator

## 2.1 Funcționalități utilizator

În continuare vor fi descrise, rând pe rând, funcționalitățile puse la dispoziție utilizatorilor obișnuiți începând cu cea mai importantă dintre ele, și anume posibilitatea cererii unui anumit serviciu cadastral pentru o anumită dată.

### 2.1.1 Pagina Request

După cum se poate observa în figura 9 această pagină conține următoarele câmpuri:

- Date
- Service
- Processing
- County
- City
- Town
- Address
- Other
- Total

Dintre toate enumerate mai sus primele 7 câmpuri sunt obligatorii, câmpul *Other* putând nefiind completat în timp ce câmpul *Total* este dezactivat pentru utilizator, în acesta calculându-se costul total al cererii utilizatorului.

The screenshot shows the 'MesPla' website interface. The header includes the logo 'MesPla' and a power icon, followed by navigation links: HOME, REQUEST, INFO, PROFILE, ABOUT, and CONTACT. The main content area is a light blue box containing a form with the following fields and labels:

Choose date:	zz.ll.aaaa
Choose service:	Please select ▼
Paper processing:	Normal ▼
County	Iasi ▼
City/Town:	
Address	
Other information	
Total (RON)	0

Below the form is a dark grey button labeled 'Schedule'.

Figura 9: Pagina Request, disponibilă utilizatorului

Cu siguranță această pagină reprezintă nucleul aplicației iar acest lucru presupune o multitudine de validări și totodată scenarii de luat în calcul înaintea trimiterii unei cereri administratorului. În cele ce urmează voi surprinde sistematic cele mai importante dintre acestea, fără de care o cerere nu ar putea fi trimisă și prin urmare, concretizată.

**Câmpul Date:** Acest câmp este de tipul date și reprezintă data la care utilizatorul dorește să beneficieze de serviciile firmei. Utilizatorul nu poate alege o data mai mică sau egală cu cea curentă intrucât în primul caz ar fi un lucru absurd iar în al doilea, ar fi posibil ca firma să nu poată onora cererea (vezi figura 10).

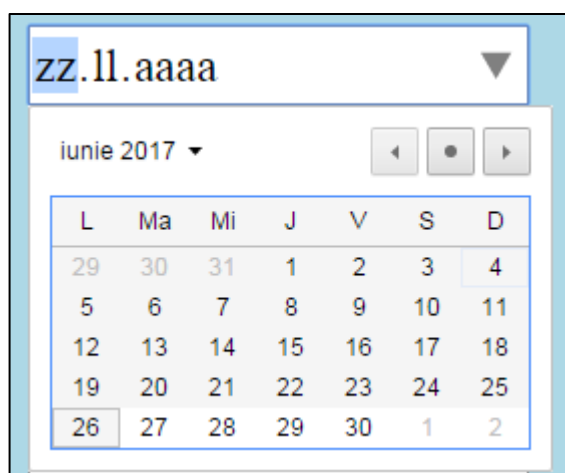


Figura 10: Selecția datei

Pe lângă această validare, în partea de back-end există o logică care verifică dacă firma dispune de resurse, la data aleasă de utilizator, pentru a onora cererea acestuia. Fiecărei cereri i se va asocia un număr de resurse consumabile în funcție de combinația "serviciu - procesare - județ" aleasă. Lista tuturor combinațiilor poate fi ușor sintetizată prin vizualizarea figurilor 11, 12 și 13 figuri ce surprind tipul serviciilor prestate, modul de procesare al documentelor și nu în ultimul rând județul în care firma ar urma să se deplaseze.

Fiecare selecție a unui tip de serviciu consumă un număr de resurse după cum urmează:

- Perosnal measurement – 0
- House measuremeant – 0
- Land inside – 0
- Terrain outside the city – 0
- Marking private property limit – 0
- House measurement (planning construction) - 2
- Road measurement (small) (planning construction) – 4



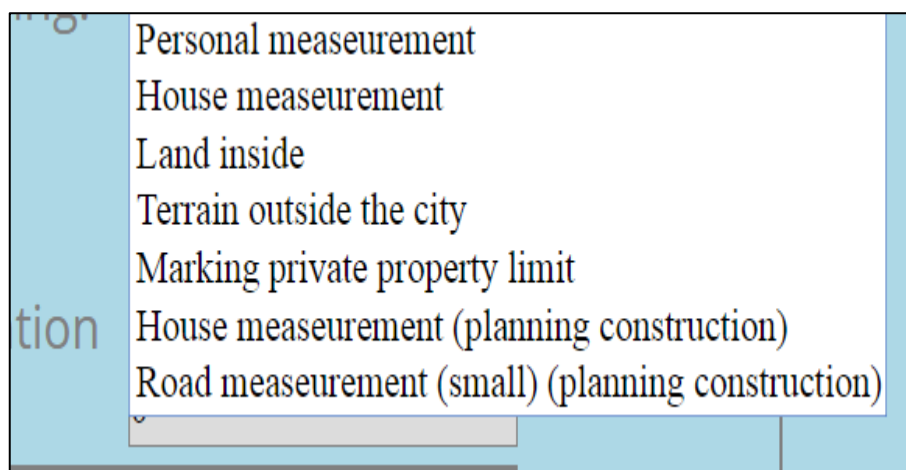


Figura 11: Servicii oferite

Pentru fiecare modalitate de procesare a documentelor numărul de resurse consumabile este 0:

- Normal – 0
- Urgent – 0
- Quick – 0

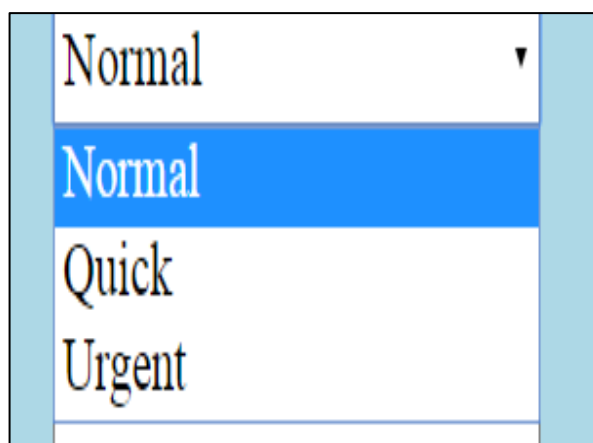


Figura 12: Procesarea documentelor

În ceea ce privește selecția județului în care echipa trebuie să se deplaseze, numărul resurselor consumabile diferă, după cum urmează:

- |                |                |
|----------------|----------------|
| ○ Iași – 1     | ○ Suceava – 3  |
| ○ Neamț – 2    | ○ Harghita – 4 |
| ○ Botoșani – 2 | ○ Covasna – 4  |
| ○ Vaslui – 2   | ○ Vrancea – 4  |
| ○ Bacău – 3    | ○ Galați – 4   |

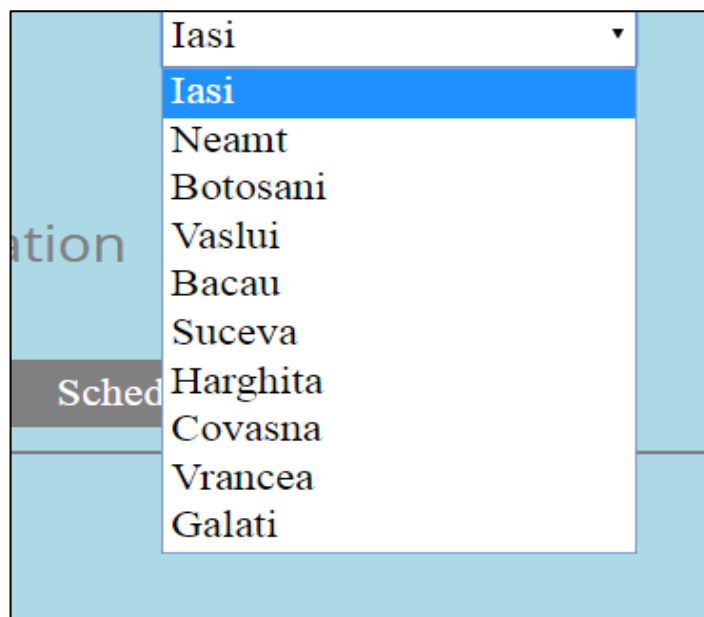


Figura 13: Selecția județului

Pentru fiecare dată, există o tabelă, numită *date\_resource*, în care se stochează informații despre resursele disponibile ale firmei – figura 14.

Schedule_date	Total_resource
2017-06-17	8
2017-06-18	8
2017-06-19	8
2017-06-20	8
2017-06-21	7
2017-06-22	4
2017-06-23	5
2017-06-24	8
2017-06-25	11

Figura 14: Tabela de resurse disponibile

Așadar, în acest moment putem exemplifica cum funcționează de fapt logica aceasta. Să presupunem că un anumit utilizator dorește pe data de 22-06-2017 un serviciu de tipul *house measeurement (planning costruction)*, cu viteza de procesare a documentelor setată pe *normal*, însă în județul *Suceava*. Printr-un simplu calcul, urmărind datele de mai sus, putem deduce că

numărul resurselor de care este nevoie pentru a îndeplini toate cerințele utilizatorului este 5 ( $2 + 0 + 3$ ) dar pentru acea dată nu mai sunt disponibile, în cadrul firmei decât un număr de 4 resurse. Prin urmare cererea nu poate fi înregistrată, un mesaj fiind afișat utilizatorului precum cel din figura 15<sup>[11]</sup>.

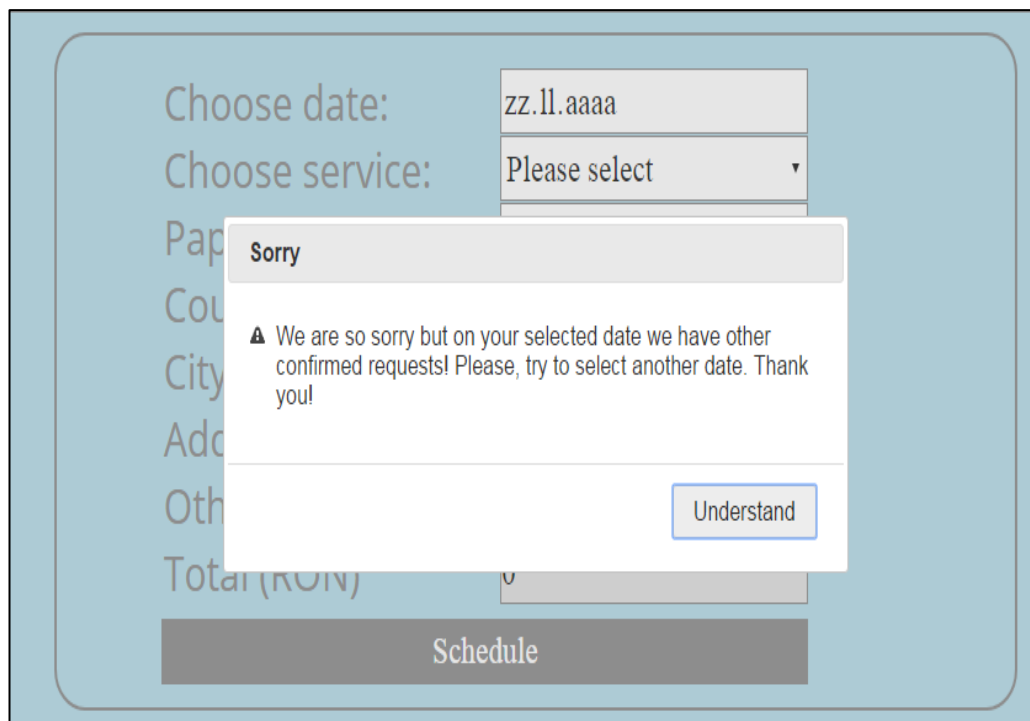


Figura 15: Mesaj de eroare în cazul în care nu mai sunt disponibile resurse

**Câmpul Total:** Este un câmp de tip text dar care este read-only și are ca scop afișarea prețului total al selecțiilor utilizatorului în moneda locală RON. La fel cum fiecărei selecții îi corespunde un număr de resurse așa îi corespunde și un cost iar combinația "serviciu – procesare - județ" le însumează, oferind prețul ce trebuie suportat de către utilizator pentru a beneficia de alegirile făcute. Tipurile de servicii oferite au următoarele costuri:

- |  |   |
|--|---|
| ➤ House measurement – 600              | ➤ House measurement (planning construction) – 2000        |
| ➤ Personal measurement – 350           | ➤ Road measurement (small) (planning construction) – 5000 |
| ➤ Land inside – 450                    |   |
| ➤ Terrain outside the city – 950       |   |
| ➤ Marking private property limit – 800 |   |

Costurile referitoare la tipul de procesare al documentelor sunt următoarele:

- |               |                |
|---------------|----------------|
| • Normal – 0  | • Urgent – 250 |
| • Quick – 100 |                |

În ceea ce privește județul ales avem următoarele costuri:

- |                  |                  |
|------------------|------------------|
| ○ Iași – 0       | ○ Suceava – 250  |
| ○ Neamț – 200    | ○ Harghita – 350 |
| ○ Botoșani – 200 | ○ Covasna – 350  |
| ○ Vaslui – 200   | ○ Vrancea – 350  |
| ○ Bacău – 250    | ○ Galați – 350   |

Întrucât aplicația dispune și de un sistem ce simulează efectuarea plăților putem reconstrui analogia deja prezentată pentru cazul în care resursele nu erau suficiente pentru a programa o vizită. Să presupunem că un utilizator dorește un serviciu de tip *terrain outside the city*, cu viteza de procesare a documentelor setată pe *quick*, în județul *Botoșani*. Ținând cont de faptul că utilizatorul nu dispune în acest moment decât de o sumă de bani de 600 de lei, depusă în cont, iar prețul total al cererii se ridică la suma de 1250 de lei un nou mesaj de eroare va fi afișat, prin care se comunică utilizatorului că suma de bani nu este suficientă pentru a înainta prezenta cerere (figura 16), urmând a fi redirecționat către pagina de profil, de unde se poate depună o sumă de bani (această funcționalitate va fi prezentată ulterior).

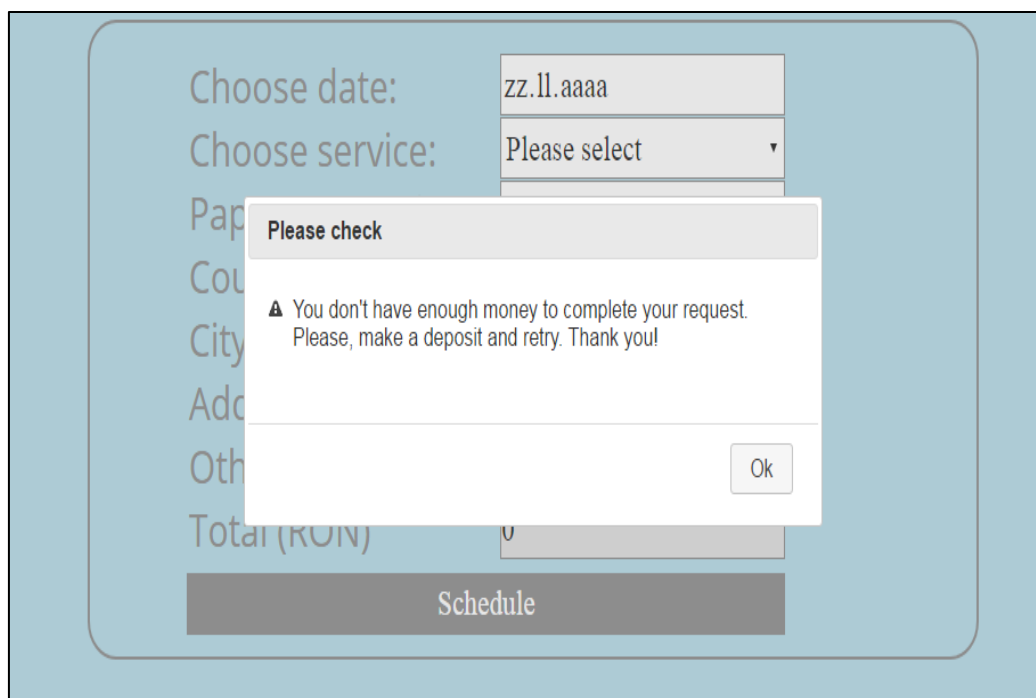


Figura 16: Mesaj de eroare în cazul în care suma de bani din cont nu acoperă prețul total

Pasajul de cod ce umează exemplifică întocmai validările descrise anterior (figura 17), în timp ce primul mesaj de eroare este vizibil în cadrul figurii 18<sup>[12]</sup>.

```

if(isset($_POST["submitBtn"])) {
    $d = $_POST["scheduleDate"];
    $serv = $_COOKIE["nameS"];
    $proc = $_COOKIE["nameP"];
    $co = $_COOKIE["nameC"];
    $ci = $_POST["city"];
    $add = $_POST["address"];
    $ot = $_POST["other"];
    $tot = $_POST["total"];
    $resourceCheckerU = true;
    $moneyCheckerU = true;
    $sqlTotalResource = "Select * from date_resource where Schedule_date='" . $d . "'";
    $sqlServiceResource = "Select * from services where Service_name='" . $serv . "'";
    $sqlCountyResource = "Select Resource from county where Name='" . $co . "'";
    $sqlMoney = "Select * from accounts where Username_id = '" . $_SESSION["id"] . "'";
    $resultTotalResource = $conn->query($sqlTotalResource);
    $resultServiceResource = $conn->query($sqlServiceResource);
    $resultCountyResource = $conn->query($sqlCountyResource);
    $resultMoney = $conn->query($sqlMoney);
    $rowT = $resultTotalResource->fetch_assoc();
    $rowS = $resultServiceResource->fetch_assoc();
    $rowC = $resultCountyResource->fetch_assoc();
    $rowM = $resultMoney->fetch_assoc();
    $varResTotal = $rowS["Resource"] + $rowC["Resource"];
    if ( $varResTotal > $rowT["Total_resource"] ) {
        $resourceCheckerU = false;
    }
    if ($rowM["Money"] < $tot) {
        $moneyCheckerU = false;
    }
    if ($resourceCheckerU == true && $moneyCheckerU == true) {
        $conn = $conn->close();
    }
}

```

Figura 17: Condiții ca o cerere să fie înaintată

```

<script>
$( function() {
    $( "#dialog-message-checker" ).dialog({
        modal: true,
        width: 500,
        dialogClass: "no-close",
        buttons: {
            Understand: function() {
                $( this ).dialog( "close" );
            }
        }
    });
} );
</script>

```

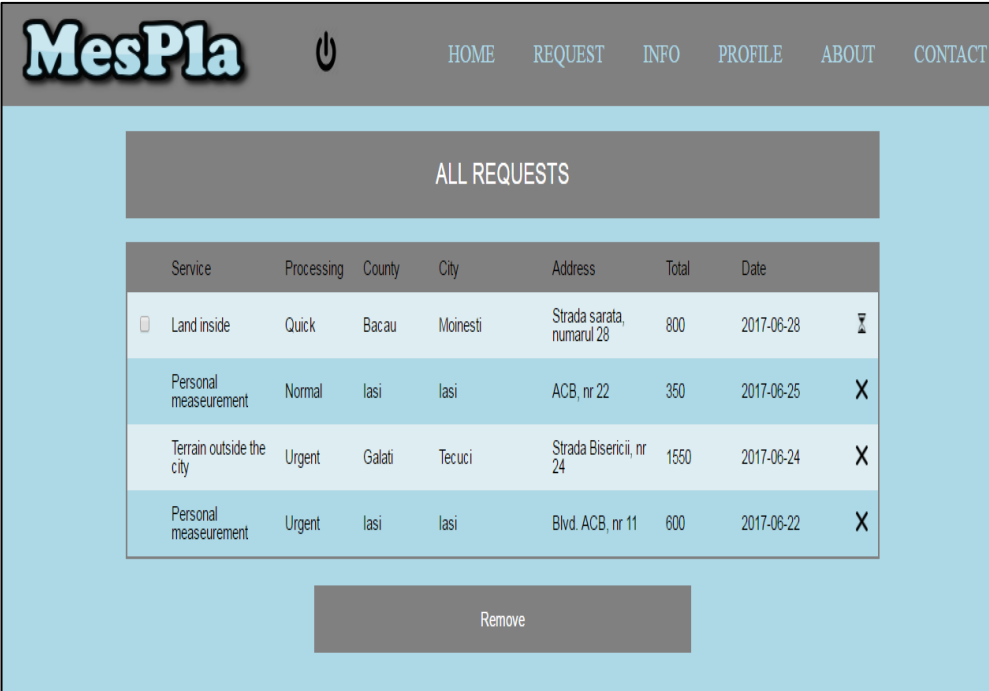
Figura 18: Afîșarea unui mesaj personalizat, număr de resurse prea mic

Lăsând la o parte toate aceste aspecte ce țin de validare, o cerere înaintată cu succes va conduce la următoarele trei acțiuni în cadrul aplicației:

- ✓ Trimiterea unui email administratorului prin care se specifică faptul că o nouă cerere a ajuns în sistem<sup>[13]</sup>
- ✓ Afișarea cererii pe pagina de Home a utilizatorului cu statusul *pending*
- ✓ Afișarea cererii pe pagina de Home a administratorului, fiind disponibilă prelucrarea acesteia

### 2.1.2 Pagina Home

În cadrul acestei pagini, utilizatorii logați își pot urmări toate cererile efectuate până acum indiferent de statusul pe care îl au: *în așteptare*, *respinse* sau *aprobate*<sup>[14]</sup>. Ultimile două statusuri sunt finale, ele nefiind alterabile. O cerere aflată în așteptare poate fi modificată de către utilizator prin bifarea ei și utilizarea butonului Remove. Trăind într-o lume în care nu doar nevoile clientului contează ci și cele ale firmei, o asemenea acțiune din partea utilizatorului necesită o taxă de 150 de lei întrucât alte persoane ar fi putut fi influențate de cererea lor. În același timp se evidențiază și profesionalismul firmei prin impunerea acestei puternice paradigme. Figura 19 surprinde însăși pagina Home, în timp ce figura numărul 20 surprinde modul în care taxa de 150 de lei este reținută din contul utilizatorului în clipa în care acesta se răzgândește și decide să anuleze cererea înaintată.



The screenshot shows the MesPla application interface. At the top is a navigation bar with the MesPla logo, a power icon, and links for HOME, REQUEST, INFO, PROFILE, ABOUT, and CONTACT. Below this is a section titled 'ALL REQUESTS' containing a table with four rows of request data. Each row has a checkbox in the first column, followed by columns for Service, Processing, County, City, Address, Total, and Date. The last column contains an icon: a clock for the first row and an 'X' for the others. Below the table is a 'Remove' button.

	Service	Processing	County	City	Address	Total	Date	
<input type="checkbox"/>	Land inside	Quick	Bacau	Moinesti	Strada sarata, numarul 28	800	2017-06-28	
<input type="checkbox"/>	Personal measeurement	Normal	Iasi	Iasi	ACB, nr 22	350	2017-06-25	X
<input type="checkbox"/>	Terrain outside the city	Urgent	Galati	Tecuci	Strada Bisericii, nr 24	1550	2017-06-24	X
<input type="checkbox"/>	Personal measeurement	Urgent	Iasi	Iasi	Blvd. ACB, nr 11	600	2017-06-22	X

Remove

Figura 19: Pagina Home pentru utilizator

```

if(isset($_POST["ren"])) {
    $box=$_POST["num"];
    while(list($key,$val) = @each ($box)) {
        $sql="UPDATE requests SET Status = 'N' WHERE Request_id = '" . $val . "'";
        if ($conn->query($sql) === TRUE) {
            $sqlRequest = "SELECT * FROM requests Where Request_id = '" . $val . "'";
            $resultRequest = $conn->query($sqlRequest);
            $rowRequest = $resultRequest->fetch_assoc();
            $sqlAccounts = "SELECT * FROM accounts WHERE Username_id = '" . $rowRequest["Username_id"] . "'";
            $resultAccounts = $conn->query($sqlAccounts);
            $rowAccounts = $resultAccounts->fetch_assoc();
            $currentBalance = $rowAccounts["Money"];
            $total = $currentBalance + $rowRequest["Sum_total"] - 150;
            $sqlNewTotal = "UPDATE accounts SET Money = '" . $total . "' WHERE Username_id = '" . $rowRequest["Username_id"] . "'";
            if ($conn->query($sqlNewTotal) === TRUE) {
                $cancelRequest = true;
            } else {
                $cancelRequest = false;
            }
        } else {
            echo "Error deleting record: " . $conn->error;
        }
    }
}
}

```

Figura 20: Anularea unei cereri de către utilizator

### 2.1.3 Pagina de profil

Un alt rol important în cadrul aplicației MesPla îl reprezintă funcționalitățile oferite de pagina Profile. Această pagină se divide în două părți:

- Partea superioară, în care este afișat username-ul, acesta nefiind editabil
- Partea mediană, în care se află informații despre utilizator și de asemenea oferă posibilitatea de a depune o sumă de bani ce poate fi utilizată în cadrul aplicației

Câmpurile editabile sunt distribuite astfel:

- Prenume, este deja populat cu prenumele curent
- Nume, este deja populat cu numele curent
- Telefon, este deja populat cu numărul de telefon curent
- Cont, inițial este format din 16 cifre – 0000000000000000
- Bani, suma ce va fi depusă din contul setat în câmpul Cont
- Parolă, câmp ce nu este populat dar pe care există o validare referitoare la lungimea parolei ce trebuie setată – lungimea trebuie să aibă minim cinci caractere
- Imagine de profil, inițial va fi afișată o imagine default, aceasta putând fi oricând schimbată

Fiecare fotografie de profil se va încărca într-un fișier (profile) pe server, inserându-se numele acesteia în coloana Image din tabela users. Pe baza informațiilor precizate și a modului de identificare a unui utilizator (id-ul) fotografia de profil va fi afișată într-un chenar și va avea dimensiunile: lățime, 212 pixeli; 272 pixeli lungime. Orice modificare cu succes a datelor deja existente va genera un mesaj sugestiv de informare a utilizatorului în clipa în care tabela users a fost alterată cu succes.

Ținând cont că aplicația MesPla doar simulează eventualele plăți online, validări suplimentare nu sunt realizate pentru moment. Condiția principală ca un utilizator să poată depune o sumă de bani este una bazată pe numărul contului din care se face depunerea. Acesta trebuie să aibă 16 caractere și în același timp să fie diferit de numărul contului default și anume: 0000000000000000. Așadar contul trebuie modificat în prealabil.

O imagine de ansamblu a paginii de profil se poate observa în cadrul figurii 21.

**MesPla**  HOME REQUEST INFO PROFILE ABOUT CONTACT

*new123*



Name:

Surname:

Phone:

Account:

Money:

Password:

Nu ai ales niciun fișier

Figura 21: Pagina de profil

Întreg "flow-ul" prezentat mai sus, precum și alte validări prezente se pot observa în figurile ce urmează (figura 22, figura 23).



```

if (isset($_POST["submit"])) {
    move_uploaded_file($_FILES['fileToUpload']['tmp_name'], "profile/".$_FILES['fileToUpload']['name']);
    $sqlPicture = "UPDATE users SET Image = '" . $_FILES['fileToUpload']['name'] . "' WHERE Username = '" . $_SESSION["username"] . "'";
    $resultPicture = $conn->query($sqlPicture);
    $change = false;
    $deposit = false;
    $first = $_POST["firstname"];
    $sur = $_POST["surname"];
    $phn = $_POST["phone"];
    $pass = $_POST["password"];
    $user = $_SESSION["id"];
    $acc = $_POST["account"];
    if ($_SESSION["role"] == $_roleClient) {
        $mon = $_POST["money"];
    }
    if (strlen($pass) > 4) {
        $passCheck = true;
    } else {
        $passCheck = false;
    }
    if ($acc != $_defaultAccount && strlen($acc) == 16) {
        $acc = $_POST["account"];
        $change = true;
        $deposit = true;
    }
    $stmt = $conn->prepare("UPDATE users SET Firstname = ?, Surname = ?, Phone = ? WHERE Id = ?");
    $stmt->bind_param("ssss", $firstname, $surname, $phone, $username);
    $firstname = $first;
    $surname = $sur;
    $phone = $phn;
    $username = $user;
    if ($stmt->execute() == true) {
        $updateBasic = true;
    } else {

```

Figura 22: Validări client, parolă și cont - 1

```

    } else {
        $updateBasic = false;
    }
    if ($passCheck == true)
    {
        $stmt = $conn->prepare("UPDATE users SET Password = ? WHERE Id = ?");
        $stmt->bind_param("ss", $password, $id);
        $password = $pass;
        $id = $user;
        if ($stmt->execute() == true) {
            $updatePass = true;
        } else {
            $updatePass = false;
        }
    }
    if ($change == true && $deposit == true)
    {
        $stmt = $conn->prepare("UPDATE accounts SET Number = ? WHERE Username_id = ?");
        $stmt->bind_param("ss", $number, $id);
        $number = $acc;
        $id = $user;
        if ($stmt->execute() == true) {
            $updateAccount = true;
        } else {
            $updateAccount = false;
        }
        if (isset($mon) && $mon >= 10)
        {
            $total = $sumAccount + $mon;
            $stmt = $conn->prepare("UPDATE accounts SET Money = ? WHERE Username_id = ?");
            $stmt->bind_param("ss", $sum, $id);
            $sum = $total;
            $id = $user;
            if ($stmt->execute() == true) {
                $updateBalance = true;
            } else {
                $updateBalance = false;
            }
        }
    }
}

```

Figura 23: Validări client, parolă și cont - 2

### 2.1.4 Pagina de informații

În cadrul acestei pagini fiecare utilizator are posibilitatea de a vizualiza patru aspecte personale ce țin de:

- Depozitul actual
- Numărul de zile până la cea mai apropiată cerere aprobată
- Numărul total de cereri
- Numărul de mesaje trimise

Pagina în cauză oferă doar o vizualizare atractivă a datelor, modificarea acestora nefiind posibilă. Aspectul vizual primează, informațiile plasându-se în interiorul a patru elemente HTML principale și anume *<div>*. Fiind foarte multe aspecte de care ar fi trebuit să se țină cont, a fost folosită o foaie de stiluri particulară pentru afișarea acestor detalii, aceasta fiind

denumită "stats.css"<sup>[10]</sup>. Figura numărul 24 oferă un imagine de ansamblu a modului în care vizualizarea datelor se realizează la nivel de client.



Figura 24: Imagine de ansamblu a paginii de informații

Fiecare element din cele patru conține un titlu expresiv care vine în ajutorul utilizatorului oferindu-i astfel informații despre numărul ce îl conține. Dacă utilizatorului nu îi este clar totuși ce reprezintă respectivul număr acesta are posibilitatea de a trece cu cursorul peste fiecare declanșând astfel trei evenimente simultan:

- Numărul din interiorul elementului HTML își va schimba culoarea curentă, din gri în negru
- Titlul elementului peste care a fost trecut cursorul își va schimba culoarea în albastru
- Suplimentar, va apărea un text în care se vor specifica informații mai amănunțite legate de ceea ce reprezintă numărul afișat

Întreg procesul acesta este descris în fragmentul de cod prezentat în cadrul figurii numărul 25:

```

.statBubbleContainer h3 {
  font-family:'segoe ui bold',arial;
  font-size:14px;
  color:#828282;
  -webkit-transition:all 0.3s ease-in-out;
  -ms-transition:all 0.3s ease-in-out;
  -moz-transition:all 0.3s ease-in-out;
  -o-transition:all 0.3s ease-in-out;
  -apple-transition:all 0.3s ease-in-out;
  transition:all 0.3s ease-in-out;
}

.statBubbleContainer:hover > h3 {
  color:#0e76bc;
}

.statBubbleContainer:hover > .statBubble {
  border:6px solid #0e76bc;
}

.statBubbleContainer:hover .statNum {
  color:#2b2b2b;
}

.statBubble {
  margin:auto;
  width:150px;
  height:150px;
  border-radius:100%;
  border:6px solid #ddd;
  position:relative;
  text-align:center;
  -webkit-transition:all 0.3s ease-in-out;
  -ms-transition:all 0.3s ease-in-out;
  -moz-transition:all 0.3s ease-in-out;
  -o-transition:all 0.3s ease-in-out;
  -apple-transition:all 0.3s ease-in-out;
  transition:all 0.3s ease-in-out;
}

```

Figura 25: Descrierea efectului hover pentru pagina de informații

### 2.1.5 Pagina About

Precum orice altă companie din lume și aceasta trebuie să ofere utilizatorilor câteva informații ce au rolul de întări nota de profesionalism ce trebuie dovedită. Astfel, sunt afișate într-o manieră plăcută următoarele detalii ce privesc atât activitatea cât și personalul ce desfășoară această activitate:

- ✓ Perioada de când activează pe piață
- ✓ Numărul de solicitări până în prezent
- ✓ Numărul de clienți satisfăcuți
- ✓ Compoaneța echipei

- ✓ Tipologia membrilor echipei
- ✓ Modul de lucru
- ✓ Experiența membrilor

Toate aceste caracteristici se pot distinge și în captura de ecran, prezentată în figura numărul 26:

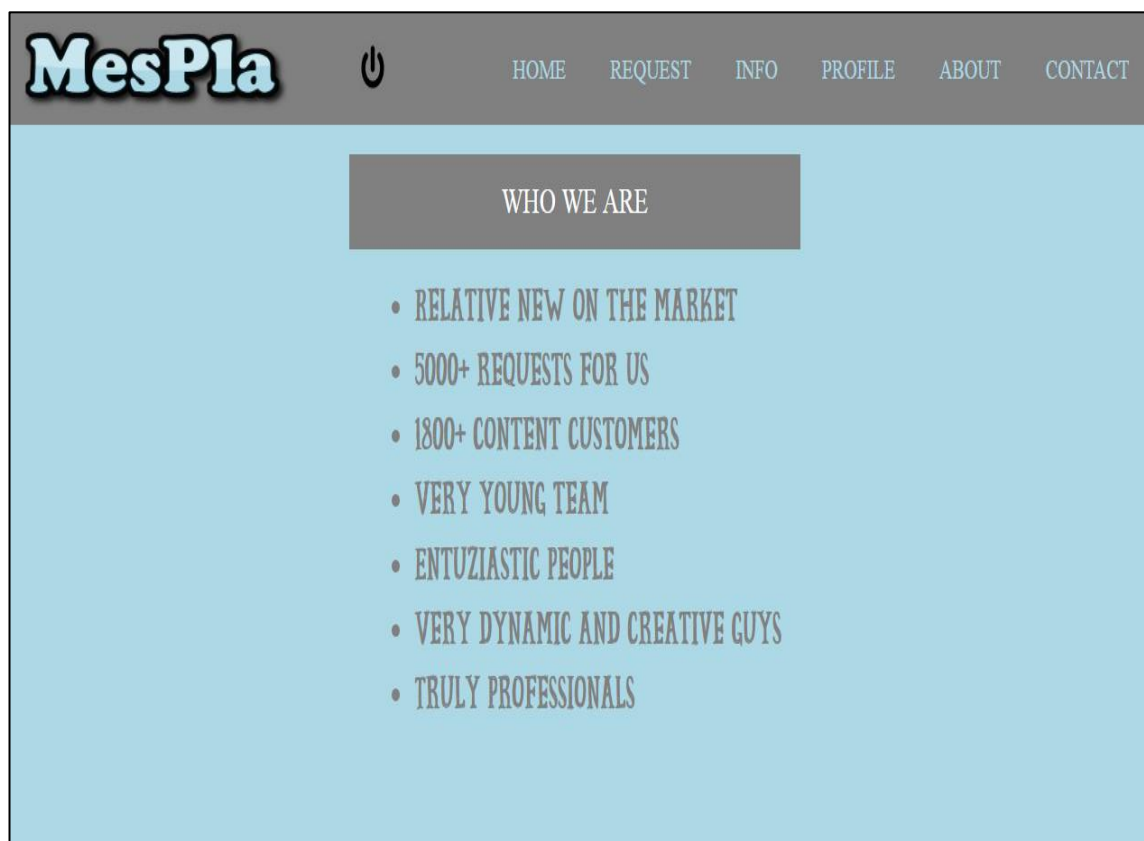


Figura 26: Captură de ecran a paginii About

### 2.1.6 Pagina contact

Ultima funcționalitate pusă la dispoziție de aplicația *MesPla*, pentru utilizatori, o reprezintă cea de contact. Aceasta este practic constituită dintr-un mini-formular pe care utilizatorul are posibilitatea să-l trimită administratorului. Odată ajuns pe această pagină, un utilizator trebuie să introducă un subiect și un mesaj ce dorește să fie primit de administrator. Atât timp cât niciun caracter nu este inserat în câmpul subiectului acesta va avea în background un text "Subject" care va ajuta clientul în momentul creării mesajului să înțeleagă ce ar trebui

să completeze. Oricare mesaj, trimis de utilizator va fi inserat într-o tabelă specială, *messages*, ce va conține următoarele coloane:

- Message\_id
- Subject
- Content
- Username
- Date\_reception

Cheia primară este dată de coloana *message\_id* care se autoincrementează la fiecare inserare iar coloana *subject* are ca valoare default "no\_subject" (figura 27).

#	Nume	Tip	Interclasare	Proprietăți	Nul	Implicit	Comentarii	Extra
1	Message_id	int(11)			Nu	None		AUTO_INCREMENT
2	Subject	varchar(45)	latin1_swedish_ci		Nu	no_subject		
3	Content	text	latin1_swedish_ci		Nu	None		
4	Username	varchar(25)	latin1_swedish_ci		Nu	None		
5	Date_reception	date			Nu	None		

Figura 27: Tabela Messages

Pe lângă cele două câmpuri *subject* și *content* în această tabelă vor fi inserate de asemenea și numele utilizatorului care a trimis mesajul, respectiv data în care acesta a fost înaintat (figura 28).

```

session_start();
if ($_SESSION["isLogged"] == true) {
    if (isset($_POST["subject"]) && isset($_POST["contactText"])) {
        include 'Connection.php';
        if (!empty($_POST["subject"]) && !empty($_POST["contactText"])) {
            $nowFormat = date("Y.m.d");
            $sql = "INSERT INTO messages (Subject, Content, Username, Date_reception) VALUES
            ('" . $_POST["subject"] . "', '" . $_POST["contactText"] . "', '" . $_SESSION["username"] . "', '" . $nowFormat . "')";
            if ($conn->query($sql) === TRUE) {
                $newRecord = TRUE;
            } else {
                echo "Error: " . $sql . "<br>" . $conn->error;
            }
        }
    }
} else {
    $newURL = "http://localhost/deton/login.php";
    header('Location: ' . $newURL);
}

```

Figura 28: Inserarea unui mesaj în tabela Messages

## 2.2 Funcționalități administrator

Funcționalitățile puse la dispoziție administratorului sunt diferite față de cele puse la dispoziția utilizatorului obișnuit tocmai prin prisma faptului că acesta este și deținătorul firmei. Așadar el deține controlul, putând hotărî dacă o cere înaintată va fi sau nu aprobată în cele din urmă. În continuare vor fi prezentate totalitatea acțiunilor ce pot fi efectuate de către acesta.

### 2.2.1 Pagina Home

Pagina curentă se află în corelație cu pagina Request disponibilă pentru utilizator. Dacă de acolo utilizatorul sau clientul poate înainta o cerere, aici administratorul poate aproba sau revoca cererea acestuia. Am observat anterior că o cerere se efectuează pentru o anumită dată calendaristică, iar dacă toate datele introduse de utilizator sunt corecte, aceasta va intra într-un status de așteptare până când va fi prelucrată de administrator.

În continuare vom presupune că avem următorul scenariu: un client înaintează o cerere cu succes pentru data de 29-06.2017. Din motive personale administratorul nu reușește să se conecteze decât după o dată ulterioară celei de 29 iunie. Cu ajutorul unui script ce rulează atât la conectarea pe pagina Home a administratorului cât și pe pagina Home a clientului, cererea va fi procesată în background (automat respinsă) dacă data curentă este mai mare sau coincide cu cea din sistem în momentul conectării. Figura numărul 29 prezintă procesul de cerere – răspuns din punct de vedere funcțional pe când în figura 30 putem distinge scriptul ce anulează automat cererile care nu mai pot fi onorate datorită datei curente.

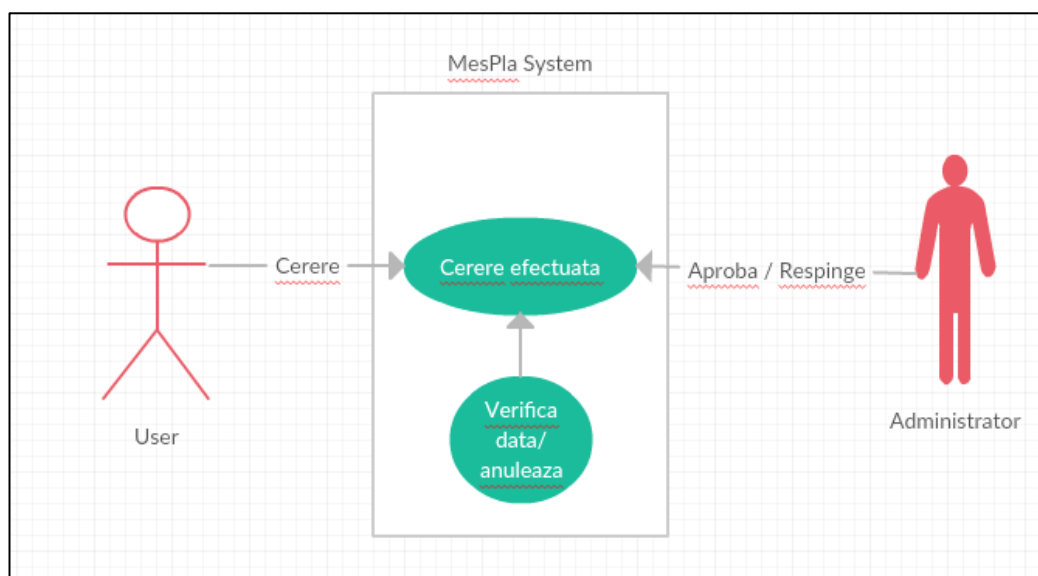


Figura 29: Procesul de cerere – răspuns

```

if ($resultCheckReq && $resultCheckReq->num_rows > 0) {
    while($rowCheckReq = $resultCheckReq->fetch_assoc()) {
        if ($rowCheckReq["Date"] <= $nowFormat) {
            $sql="UPDATE requests SET Status = 'N' WHERE Request_id = '" . $rowCheckReq["Request_id"] . "'";
            if ($conn->query($sql) === TRUE) {
                $sqlAccounts = "SELECT * FROM accounts WHERE Username_id = '" . $rowCheckReq["Username_id"] . "'";
                $resultAccounts = $conn->query($sqlAccounts);
                $rowAccounts = $resultAccounts->fetch_assoc();
                $currentBalance = $rowAccounts["Money"];
                $total = $currentBalance + $rowCheckReq["Sum_total"];
                $sqlNewTotal = "UPDATE accounts SET Money = '" . $total . "' WHERE Username_id = '" . $rowCheckReq["Username_id"] . "'";
                if ($conn->query($sqlNewTotal) === TRUE) {
                    $ok = true;
                } else {
                    $ok = false;
                }
            } else {
                echo "Error deleting record: " . $conn->error;
            }
        }
    }
}
}

```

Figura 30: Scriptul de anulare automată a cererilor

Așa cum pagina Request reprezintă nucleul funcționalităților pentru utilizator așa și această pagină reprezintă nucleul funcționalităților pentru administrator. Pagina este structurată astfel:

- Nu există cereri în așteptare
  - Un mesaj personalizat care apare printr-un efect de tip "bounce" în partea centrală
- Există cereri în așteptare
  - Cererile vor fi afișate în partea centrală într-un tabel personalizat al cărui foaie de stil este "table.css" în partea centrală
  - Tabelul conține următoarele informații:
    - Nume utilizator
    - Serviciul cerut
    - Viteza de procesare a documentelor
    - Județul în care trebuie efectuată deplasarea
    - Orașul
    - Adresa
    - Costul pachetului



- Data la care este solicitată firma
- Numărul de resurse pe care le necesită cererea
- În partea de jos sunt disponibile două butoane:
  - Approve – se pot aproba toate cererile bifate
  - Reject – se pot respinge toate cererile bifate

Cele două situații descrise mai sus sunt exemplificate în capturile de ecran următoare (figura 31 și figura 32):

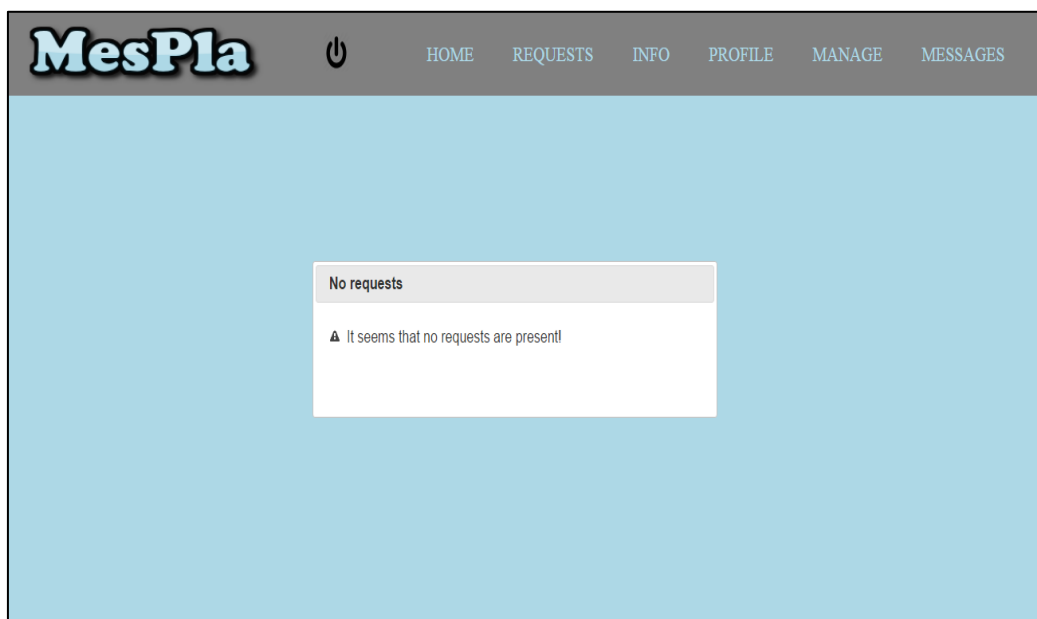


Figura 31: Pagina Home fără cereri prezente

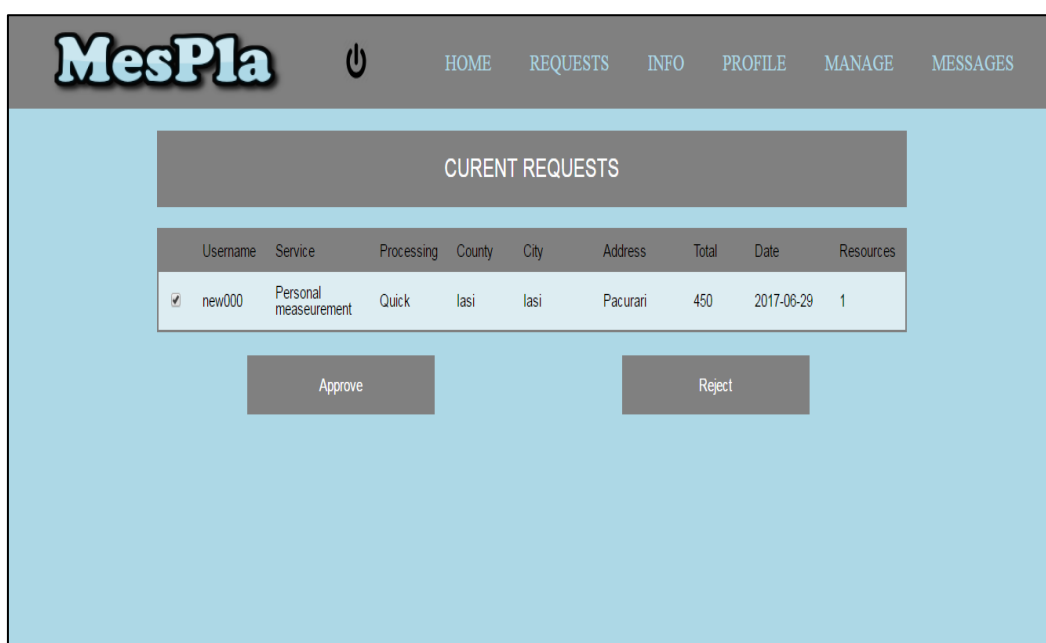


Figura 32: Pagina Home, cerere în așteptare

După cum se poate observa fiecare cerere beneficiază în partea din stânga de un câmp de tipul checkbox. Deci, se pot procesa mai multe cereri simultan însă toate cu același rezultat: toate acceptate sau toate refuzate. Cererile bifate, vor fi procesate, rând pe rând iar de fiecare dată se va verifica dacă mai există resurse disponibile la data selectată pentru a schimba statusul din *în așteptare* în *aprobată*. Procesul de prelucrare multiplă se poate observa în cadrul figurii 33.

```

) if(isset($_POST["app"])) {
    $box=$_POST["num"];
    while(list($key,$val) = each ($box)) {
        $sqlRequest = "SELECT * FROM requests Where Request_id = '" . $val . "'";
        $resultRequest = $conn->query($sqlRequest);
        $rowRequest = $resultRequest->fetch_assoc();
        $sqlResource = "SELECT * FROM date_resource WHERE Schedule_date = '" . $rowRequest["Date"] . "'";
        $resultResource = $conn->query($sqlResource);
        $rowResource = $resultResource->fetch_assoc();
        if ($rowRequest["Resource"] <= $rowResource["Total_resource"]) {
            $sql="UPDATE requests SET Status = 'Y' WHERE Request_id = '" . $val . "'";
            if ($conn->query($sql) === TRUE) {
                $newResource = $rowResource["Total_resource"] - $rowRequest["Resource"];
                $sqlUpdateResource = "UPDATE date_resource SET Total_resource = '" . $newResource . "' WHERE Schedule_date = '" . $rowRequest["Date"] . "'";
                if ($conn->query($sqlUpdateResource) === TRUE) {
                    $sqlAccounts = "SELECT * FROM accounts WHERE Username_id = '" . $_SESSION["id"] . "'";
                    $resultAccounts = $conn->query($sqlAccounts);
                    $rowAccounts = $resultAccounts->fetch_assoc();
                    $currentBalanceAdm = $rowAccounts["Money"];
                    $newBalanceAdm = $currentBalanceAdm + $rowRequest["Sum_total"];
                    $sqlNewTotalAdm = "UPDATE accounts SET Money = '" . $newBalanceAdm . "' WHERE Username_id = '" . $_SESSION["id"] . "'";
                    $conn->query($sqlNewTotalAdm);
                    $acceptRequest = true;
                }
            } else {
                $acceptRequest = false;
                echo "Error deleting record: " . $conn->error;
            }
        }
    }
} ?>

```

Figura 33: Aprobarea mai multor cereri simultan

## 2.2.2 Pagina de istoric a cererilor

Până în acest moment am putut observa cum sunt procesate cererile utilizatorilor dar nu am văzut unde sunt afișate cele care deja sunt într-un status final. Toate cererile, din faza lor de aprobare și până în punctul în care se concretizează se stochează în cadrul tabelii *requests*. Aceasta este formată din 13 coloane fiecare memorând informații cheie cum ar fi:

- Numărul cererii (cheie primară)
- Id-ul utilizatorului care a realizat cererea

- Serviciul pentru care a optat
- Timpul de procesare a documentelor
- Județul în care echipa trebuie să se deplaseze
- Localitatea
- Adresa concretă
- Un câmp care poate conține alte informații
- Numele de utilizator al celui care a realizat cererea
- Data pentru care este solicitată cererea
- Suma totală
- Statusul în care se află cererea (acesta poate avea următoarele valori: P – cererea se află în așteptare, N – cererea este respinsă, Y – cererea este aprobată)
- Numărul de resurse pe care îl necesită cererea

Structura tabelului poate fi consultată în cadrul figurii numărul 34:

#	Nume	Tip	Interclasare	Proprietăți	Nul	Implicit	Comentarii	Extra
1	Request_id	int(7)			Nu	None		AUTO_INCREMENT
2	Username_id	int(10)			Nu	None		
3	Service	varchar(75)	latin1_swedish_ci		Nu	None		
4	Processing	varchar(20)	latin1_swedish_ci		Nu	None		
5	County	varchar(45)	latin1_swedish_ci		Nu	None		
6	CityTown	varchar(45)	latin1_swedish_ci		Nu	None		
7	Address	varchar(100)	latin1_swedish_ci		Nu	None		
8	Other	varchar(100)	latin1_swedish_ci		Da	NULL		
9	Username	varchar(50)	latin1_swedish_ci		Nu	None		
10	Date	date			Nu	None		
11	Sum_total	int(7)			Nu	None		
12	Status	varchar(1)	latin1_swedish_ci		Nu	P		
13	Resource	int(2)			Nu	None		

Figura 34: Structura tabelului requests

Așadar, în cadrul acestei pagini (care este destinată exclusiv administratorului) se poate vizualiza istoricul tuturor cererilor deja procesate, indiferent că sunt aprobate sau respinse. Acestea vor fi afișate în ordinea descrescătoare a datelor în care cererile au fost programate.

Pentru fiecare cerere se va afișa o imagine reprezentativă care ilustrează statusul în care se află cererea (figura 35).

MesPla



HOME

REQUESTS

INFO

PROFILE

MANAGE

MESSAGES

REQUESTS HISTORY

Username	Service	Processing	County	City	Address	Total	Date	
gigi	Terrain outside the city	Quick	Suceva	Iacobeni	Str noua, 1, aa	1300	2017-09-06	X
new000	House measeurement	Normal	Iasi	dsda	asd	600	2017-06-30	✓
new000	Personal measeurement	Quick	Iasi	Iasi	Pacurari	450	2017-06-29	X
new123	Land inside	Quick	Bacau	Moinesti	Strada sarata, numarul 28	800	2017-06-28	X
new000	Personal measeurement	Quick	Neamt	Piatra Neamt	Strada veche, nr 45	650	2017-06-27	X
ionutPreda	House measeurement (planning construction)	Urgent	Galati	Pechea	Soseaua Principala, 678	2600	2017-06-27	X
new000	House measeurement	Normal	Iasi	Iasi	Dacia, nr 21	600	2017-06-27	X
new000	Personal measeurement	Normal	Iasi	dddd	dddd	350	2017-06-25	✓

Figura 35: Istoric al cererilor

### 2.2.3 Pagina de informații

Pagina de informații a administratorului oferă funcționalități asemănătoare cu cele oferite de pagina de informații a utilizatorului. Așadar pagina oferă următoarele posibilități pentru administrator:

- Vizualizarea balanței actuale
- Numărul de utilizatori înscrși

- Numărul de mesaje primite
- Numărul total de cereri
  - Dacă administratorul dorește să vizualizeze un raport detaliat legat de numărul de cereri trimise
  - Acest raport este disponibil dacă administratorul apasă oriunde în acest cadran

După cum se poate observa și în figura 36 modul de afișare este identic cu cel folosit pentru utilizatorii obișnuiți întrucât folosesc aceeași foaie de stil (style.css)<sup>[7]</sup>.

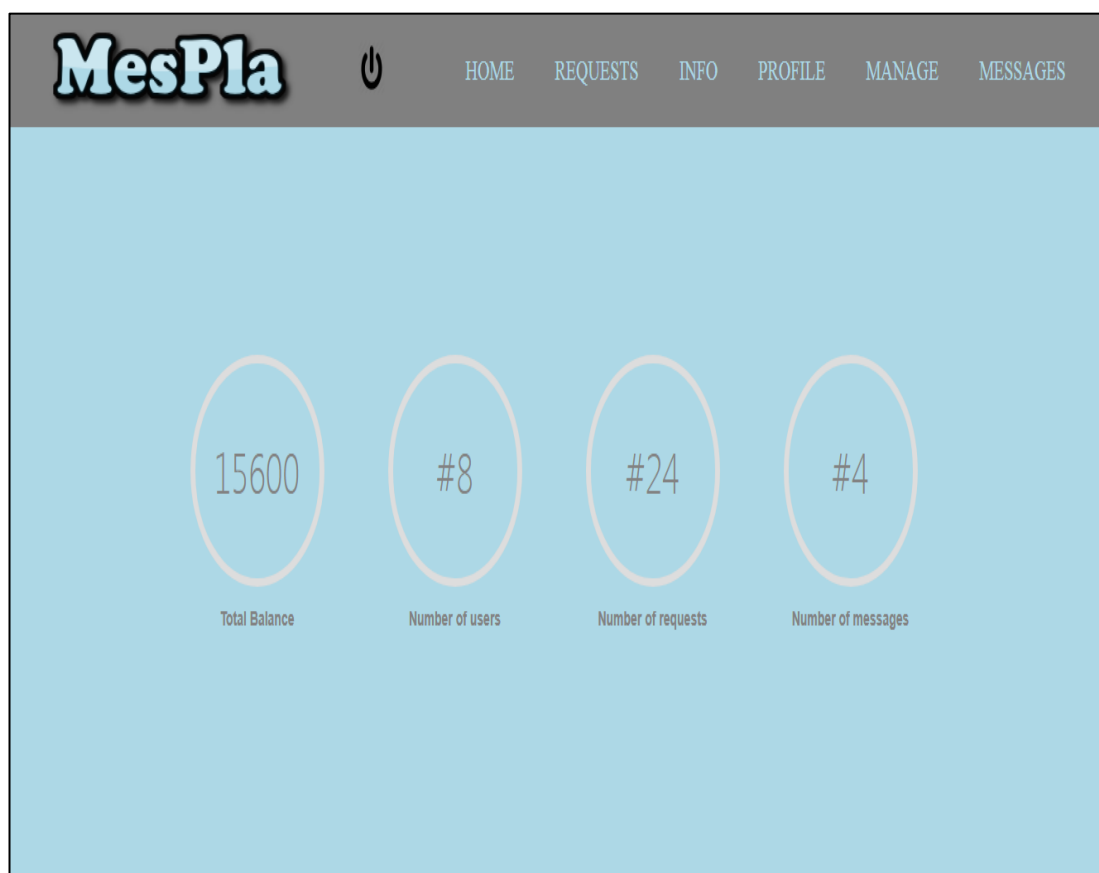


Figura 36: Pagina de informații pentru utilizator

Zona în care este afișată numărul total de cereri oferă și posibilitatea de a deschide într-o nouă pagină, cu ajutorul unei funcții javascript (figura 37), un raport detaliat legat de aceste cereri. Acest raport va conține toate serviciile oferite în cadrul aplicației, respectiv numărul total de solicitări al fiecăruia. În acest mod administratorul poate avea o imagine de ansamblu asupra cărui serviciu ar trebui să se axeze pe viitor, pe baza evoluției numărului total de solicitări primite (figura 38).

```

</script>
function details() {
    window.open("Services.php", "_blank", "toolbar=yes,scrollbars=yes,resizable=yes,top=100,left=400,width=700,height=500");
}
</script>

```

Figura 37: Funcția pentru a deschide numărul serviciilor într-o fereastră nouă

SERVICES REQUESTS	
Services	Requested
Personal measeurement	10
House measeurement	4
Land inside	5
Terrain outside the city	2
Marking private property limit	1
House measeurement (planning construction)	2
Road measeurement (small) (planning construction)	0

Figura 38: Numărul de servicii cerute

#### 2.2.4 Pagina de profil

La fel ca pagina de informații, pagina de profil pentru administrator este relativ asemănătoare cu cea de profil pentru client. Această pagină se divide în două părți:

- Partea superioară, în care este afișat username-ul, acesta nefiind editabil

- Partea mediană, în care se află informații despre utilizator și de asemenea oferă posibilitatea de a depune o sumă de bani ce poate fi utilizată în cadrul aplicației

Câmpurile editabile sunt distribuite astfel:

- Prenume, este deja populat cu prenumele curent
- Nume, este deja populat cu numele curent
- Telefon, este deja populat cu numărul de telefon curent
- Cont, inițial este format din 16 cifre – 0000000000000000
- Parolă, câmp ce nu este populat dar pe care există o validare referitoare la lungimea parolei ce trebuie setată – lungimea trebuie să aibă minim cinci caractere
- Imagine de profil, inițial va fi afișată o imagine default, aceasta putând fi oricând schimbată

Orice modificare a informațiilor personale ale administratorului va declanșa afișarea unui mesaj personalizat ce informează că datele au fost salvate cu succes. Acest lucru se poate vedea cu ușurință în figura 39.



Figura 39: Mesaj disponibil în momentul în care datele au fost actualizate

### 2.2.5 Pagina de management

Fiind administrator, acesta beneficiază în cadrul acestei pagini de o suită de funcționalități ce țin de managementul aplicației. În primul rând se pot adăuga noi resurse pentru o anumită dată, dacă se găsesc și cererea este mare. De asemenea orice utilizator obișnuit poate fi exclus de pe site, iar într-un final, administratorul poate retrage o anumită sumă de bani disponibilă pe site figura (40).



Figura 40: Pagina de management

#### **Adăugarea de noi resurse:**

Să presupunem că putem întâlni următorul scenariu. Pentru data de 30-06-2017 există predefinit un număr de 8 resurse (cele disponibile în cadrul firmei). Pentru această dată există un număr total de 10 cereri ce cumulează nu mai puțin de 12 resurse. Este destul de clar că firma nu poate onora toate aceste cereri însă, nici nu dorește să refuze unii clienți. Tocmai de aceea, administratorul poate contacta în particular, până la luarea unei decizii, câteva firme, ingineri, etc. care pot fi disponibili la acea dată și cu care să se poată împărți câștigul prin oferirea serviciilor lor.



Caseta *Add* oferă o soluție în cadrul aplicației tocmai la această problemă. Trecerea cursorului deasupra ei va declanșa afișarea unui mesaj sugestiv ce va ajuta administratorul oferindu-i informații suplimentare legate de funcția care este oferită.

O nouă fereastră va fi deschisă și va conține următoarele elemente:

- ✓ Un câmp în care se poate număra de resurse dorit
- ✓ Data pentru care s-au obținut resursele
- ✓ Butonul ce preia datele din formular și le trimite mai departe

Odată apăsat butonul *Add resource* numărul de resurse inserat va fi adăugat la numărul de resurse existente în momentul trimerii datelor. Dacă informațiile au fost actualizate cu succes un mesaj de notificare va apărea în partea centrală a paginii (figura 41).

```
<?php
if (isset($increased) && $increased == true) { include 'ResourceUser.php'; ?>
<script>
    $( function() {
        $( "#dialog-message-resourceAdd").dialog({
            modal: true,
            width: 300,
            dialogClass: "no-close",
            buttons: {
                OK: function() {
                    $( this ).dialog( "close" );
                }
            }
        });
    } );
</script>
<?php } ?>
```

Figura 41: Mesaj personalizat pentru adăugarea de resurse

### **Excluderea unor utilizatori:**

Fiind un instrument online, este clar că interacțiunea dintre clienți și administrator se realizează tot în mediul online (cel puțin prima). Este posibil ca anumiți utilizatori să nu folosească tocmai un limbaj adecvat vis-à-vis de conducerea firmei sau pot pur și simplu dăuna imaginii firmei din diverse motive. Tocmai de aceea, cea de-a doua funcționalitate pusă la dispoziție de aplicația MesPla, în cadrul paginii de management este aceea de a exclude anumiți utilizatori de pe site.

Modul în care această funcționalitate operează este similar cu cel al adăugării de noi resurse, până la un punct. Trecerea cursorului peste casetă va genera de asemenea un mesaj auxiliar pentru a ajuta administratorul să înțeleagă ce se întâmplă prin apăsarea ei. O nouă pagină se va deschide conținând un tabel ce va cuprinde:

- Id-ul utilizatorului
- Username-ul
- Numele clientului
- Prenumele clientului
- Emailul clientului
- Imaginea de profil a clientului

Vor fi listați atât utilizatorii obișnuiți cât și administratorul, însă doar rândul utilizatorului va conține un checkbox ce poate fi selectat (Figura 42).

DELETE FROM SYSTEM

	ID	Username	Surname	Firstname	Email	Picture
<input type="checkbox"/>	32	aldo	Grigurescu	Adrian	sdfsdf@asdka.com	
<input type="checkbox"/>	26	new000	Moromete 2	Ilie	cosminaloredana20@gmail.com	
<input type="checkbox"/>	34	lucian	Popescu	Adelin	adelin@yahoo.com	
	9	ionut	Preda	Ionut	preda.ionut.iulian@gmail.com	
<input checked="" type="checkbox"/>	25	new123	Ranete	Cosmina	cosminaloredana20@gmail.com	

Remove

Figura 42: Pagina de eliminare a utilizatorilor

În background, afișarea tabelului cu utilizatori, ce pot fi excluși se formează cum este exemplificat în figura numărul 43:

```

<form action="" method="post">
<!-- TABLE -->
<table class="table table-action">
  <thead>
    <tr>
      <th class="t-small"></th>
      <th class="t-small">ID</th>
      <th class="t-medium">Username</th>
      <th class="t-medium">Surname</th>
      <th class="t-medium">Firstname</th>
      <th class="t-medium">Email</th>
      <th>Picture</th>
    </tr>
  </thead>
  <tbody>
    <?php if ($result && $result->num_rows > 0) {
      while($row = $result->fetch_assoc()) { <?>
        <tr>
          <td><label> <?php if ($_SESSION["id"] != $row["Id"]) { <?>
            <input name="num[]" type="checkbox" value="<?php echo $row['Id'];?>"><?php } ?>
          </label></td>
          <td><?php echo $row["Id"] ?></td>
          <td><?php echo $row["Username"] ?></td>
          <td><?php echo $row["Surname"] ?></td>
          <td><?php echo $row["Firstname"] ?></td>
          <td><?php echo $row["Email"] ?></td>
          <td><?php if (is_null($row["Image"]) || $row["Image"] == "") {
            echo "<img id= 'profilePicture' src='profile/default.png' alt='No picture'>";
          } else { echo "<img id= 'profilePicture' src='profile/' . $row["Image"] . ". " alt='No picture'>"; }?></td>
        </tr>
      <?php
    }
  }
  <?>
</tbody>
</table>

```

Figura 43: Secvență de cod pentru afișarea tabelului cu utilizatori

### **Transferul unei sume de bani:**

Ultima funcție ce o poate folosi administratorul, în cadrul paginii de management este cea prin care se realizează transferul unei sume de bani de pe site într-un cont specificat de acesta. Fiind specificat și anterior că prezenta aplicație doar simulează procesarea plăților pagina de transfer va conține următoarele trei elemente:

- ✓ Suma de ban ce se dorește a fi transferată
- ✓ Contul în care sa fie tranferată
- ✓ Butonul de transfer

Validarea în cadrul acestui formular se va face atât la nivelul sumei de bani introduse cât și la nivelul contului specificat. Suma de bani nu trebuie să fie mai mare decât balanța actuală

a administratorului iar contul trebuie sa fie atât diferit de cel predefinit (000000000000000000) cât și să aibă o lungime de exact 16 cifre (figura 44).

```
<?php
session_start();
include 'OwnFunctions.php';
if ($_SESSION["isLogged"] == true) {
if(isset($_POST["subBtn"])) {
    include 'Connection.php';
    $mon = $_POST["money"];
    $acc = $_POST["account"];
    $sqlTotalMoney = "Select * from accounts where Username_id =' " . $_SESSION["id"] . "'";
    $resultTotalMoney= $conn->query($sqlTotalMoney);
    $rowT = $resultTotalMoney->fetch_assoc();
    if ($acc != $_defaultAccount && strlen($acc) == 16 && $mon <= $rowT["Money"]) {
        $change = true;
    } else {
        $change = false;
    }
    if ($change == true ) {
        $newBalance = $rowT["Money"] - $mon;
        $stmt = $conn->prepare("UPDATE accounts SET Money = ? WHERE Username_id = ?");
        $stmt->bind_param("ss", $newMoney, $token);
        $newMoney = $newBalance;
        $token = $_SESSION["id"];
        if ($stmt->execute() == true) {
            $updated = true;
        } else {
            $updated = false;
        }
    } else {
        $updated = false;
    }
}
} else {
    $newURL = "http://localhost/deton/login.php";
    header('Location: '.$newURL);
}
```

Figura 44: Validări ale celor două câmpuri din pagina de transfer a banilor

Schema de utilizare a celor trei funcționalități descrise mai sus poate fi consultată analizând figura 45:

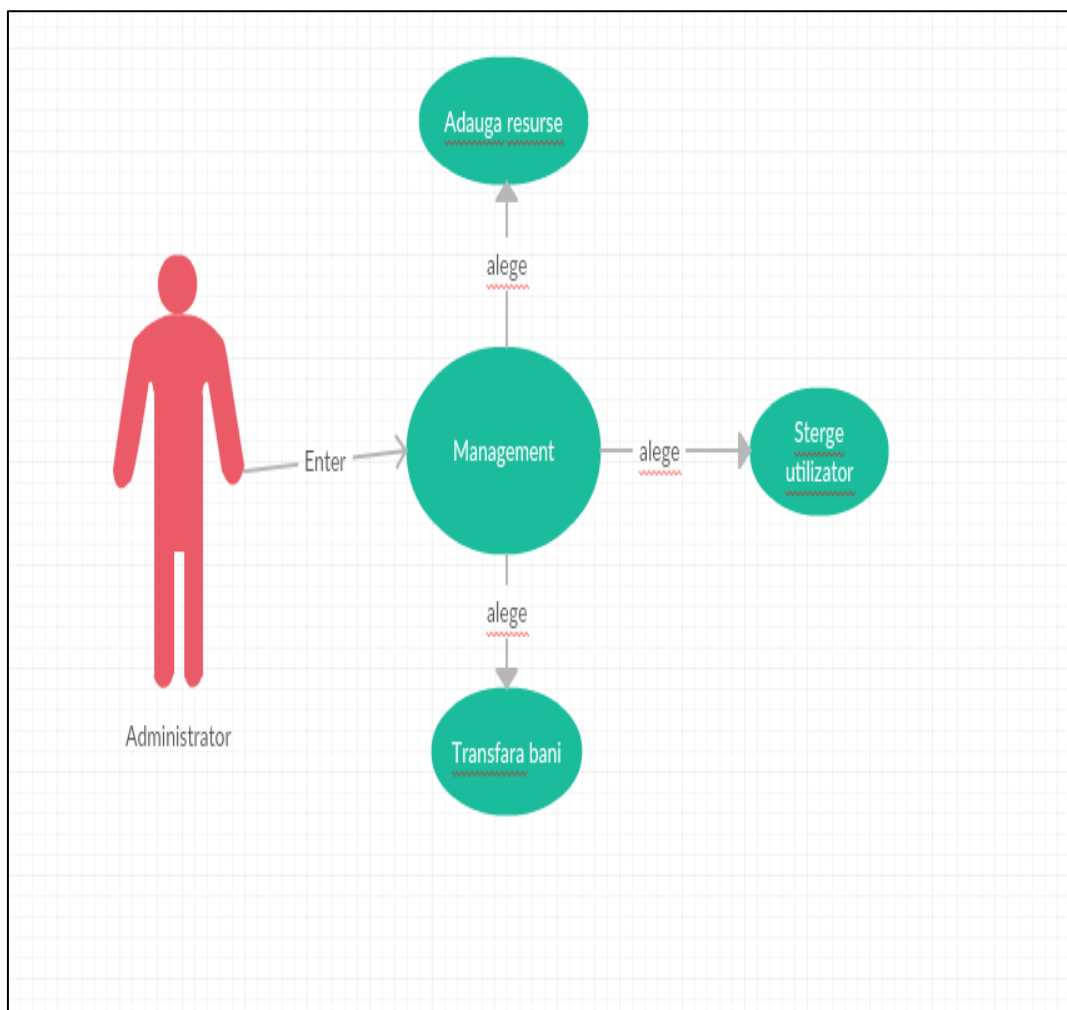


Figura 45: Schema funcționalităților de pe pagina de management

### 2.2.6 Pagina de mesaje

Ultima funcționalitate pusă la dispoziția administratorului este cea de vizualizarea a tuturor mesajelor trimise de către utilizatorii aplicației. Pagina este reprezentată printr-un tabel în care se găsesc exact elementele prezentate în cadrul paginii de contact, disponibilă utilizatorului obișnuit:

- Id-ul mesajului trimis
- Utilizatorul care l-a trimis
- Subiectul mesajului
- Textul propriu zis
- Data la care a fost trimis

Odată citite mesajele pot fi șterse, un istoric nefiind ținut (figura 46).

```

<input type="submit" value="Delete Messages" name="del" id="delBtn" />
</form>
<?php
    if(isset($_POST["del"])) {
        $box=$_POST["num"];
        while(list($key,$val) = @each ($box)) {
            $sql="DELETE FROM messages WHERE Message_id = '" . $val . "'";
            if ($conn->query($sql) === TRUE) {
                $ok=true;
            } else {
                echo "Error deleting record: " . $conn->error;
            }
        }
    }

?>
<script type="text/javascript">
window.location.href=window.location.href;
</script>
<?php
}

?>

```

Figura 46: Stergerea mesajelor din lista administratorului

### 3. Dezvoltări ale aplicației

După cum se poate observa, scopul acestei lucrări este acela de a pune bazele unui proiect mult mai amănunțit, proiect ce trebuie să definească și o paradigmă de securitate ce trebuie urmată, având în vedere că există destule date sensibile ce interacționează.

Cu siguranță prezenta aplicație poate fi îmbunătățită printr-o serie de noi integrări ale unor funcționalități existente:

- Pe baza sistemului care în prezent doar simulează procesarea plăților se poate integra un sistem real ce comunică cu alte aplicații (validare conturi, verificare identitate, etc.)
- Se poate integra un chat-box care să permită transmiterea informațiilor cu și mai multă rapiditate decât cea din prezent. Momentan este implementat sistemul Contact – Messages
- Se poate implementa un sistem de rating pus la dispoziția utilizatorilor pentru a nota profesionalismul lucrărilor

De asemenea, nu doar că se pot dezvolta funcționalități deja existente, ci se pot dezvolta și unele noi, cum ar fi:

- Corelarea aplicației cu un API Google Maps, care să permită pe baza reperelor geografice (latitudine și longitudine) determinarea și validarea unei adrese reale pentru care se efectuează o cerere
- Dezvoltarea unui sistem complex de notificări automate, ce urmărește modificarea informațiilor în timp real – orice cerere înaintată să fie prezentă atât într-un *dashboard* al administratorului cât și al utilizatorului; la fel pentru orice aprobare sau refuz al unei cereri

## Aplicații similare

Una din aplicațiile similare de pe piață este aplicația *firmecadastru* (<https://www.firmecadastru.ro/>). Aceasta este o aplicație WEB în care sunt grupate mai multe persoane ce oferă servicii cadastrale sau chiar firme ce prestează astfel de servicii (figura numărul 47).



Figura 47 – Aplicație similară

Principale avantaje ale aplicației MesPla față de ceea ce există pe piață, în aceeași industrie, sunt următoarele:

- ✓ Niciuna nu oferă posibilitatea înregistrării în aplicație
- ✓ Majoritatea nu dispun de efectuarea unei cereri "la distanță"
- ✓ Nu dispun de un sistem de plată online



## Concluzii

În cadrul procesului de concepere a aplicației *MesPla – Measurements and Planning*, *Aplicație WEB pentru servicii cadastrale*, am întâmpinat o serie de provocări cum ar fi următoarele:

- Odată hotărât în a concepe această aplicație (întrucât a trebuit să apelez la o serie de servicii cadastrale iar timpul de așteptare a fost mult prea mare) a trebuit să mă consult cu un specialist pentru a înțelege și a avea o imagine de ansamblu a acestui domeniu
- Pentru a crește productivitatea aplicației am fost nevoit să învăț cum se utilizează biblioteca JQuery pentru a realiza call-uri AJAX (Asynchronous JavaScript And XML) – astfel pagina WEB nu trebuie reîncărcată în întregime de fiecare dată când este nevoie

Având în vedere, toate cele prezentate mai sus, consider că aplicația *MesPla – Measurements and Planning* este capabilă să deschidă noi orizonturi în această industrie, care încă o dată, cel puțin vorbind de domeniul serviciilor cadastrale, nu este foarte puternic digitalizată. Așadar *MesPla*:

- Prin abordarea ei poate reduce extraordinar de mult timpul de așteptare pentru a solicita un serviciu cadastral
- Propune o soluție de plată online, integrată în aplicație fiind o noutate în cadrul aplicațiilor de acest tip
- Serviciile cadastrale pot fi solicitate și din alt județ, nu doar Iași (Neamț, Botoșani, Vaslui, Bacău, Suceava, Harghita, Covasna, Vrancea, Galați)
- Firma își poate crește numărul colaboratorilor, pentru anumite date, astfel putând să onoreze mai multe cereri

## Bibliografie

- [1] - Rasmus Lerdorf, Kevin Tatroe (2013). **Programming PHP, 3rd Edition. Creating Dynamic Web Pages**, Editura O Reilly Media, California
- [2] James Wallace (2015). **Coding HTML and CSS**, Editura Flame Tree Publishing Co Ltd, London
- [3] Robin Nixon (2015). **JavaScript**, Editura Mcgraw- Hill Professional, California
- [4] Paul DuBois (2010). **MySQL**, Editura Addison-Wesley, Boston
- [5] Chris Shiflett, *SQL Injection*, Publicat în PHP Architect 2004
- [6] Bara de navigare - <https://codepen.io/>
- [7] Crearea unui dialog modal - <https://jqueryui.com/>
- [8] Calcularea unor valori în timp real - <https://jquery.com/>
- [9] Trimiterea automată de mailuri - <https://github.com/PHPMailer/PHPMailer>
- [10] Generarea statisticilor - <https://codepen.io/BoiseDigital/pen/Cbxst>
- [11] Crearea unui tabel - <https://codepen.io/>
- [12] Cursuri Tehnolгии WEB - <https://profs.info.uaic.ro/~busaco/>
- [13] Tutorial javascript - <https://www.tutorialspoint.com/>
- [14] Proiect tehnologii web - Deton