VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE I KOMUNIKACIONE TEHNOLOGIJE

INTERNET TEHNOLOGIJE

DOKUMENTACIJA

pozoriste

Predmet: Web programiranje PHP1 Student: Darija Šošdean 18/16

Beograd, jun 2022

Sadržaj

Uvod....5
Korišćeni programski jezici i tehnologije:....5
Opis funkcionalnosti....5
Template....6
Organizacija....6
Organizaciona šema....6
Šema baze....6
Stranice....7
Home....7
Performances....8
Performance....8
Repertoire....9
Kupovina....10
Tickets....11
Kontakt....11
Registracija I logovanje....12
Ativacija....12
Admin Performances....13
Admin New Performance....14
Admin Edit Performance....14
Admin Repertoire....15
Admin Add Repertoire....15
Korisnici....16
Messages....16
Anketa....17
Autor....17
Kodovi....18
Root....18
Index.php....18
Activation....18
Performance.php....19

Performances.php....19
Repertoire.php....19
Tickets.php....20
Seats.php....20
Contact.php....21
Author.php....21
Login.php....21
Activation.php....22
View/fixed....23
Footer.php....23
Head.php....24
Header.php....24
View/pages....25
Index.php....25
Performance.php....27
Performances.php....28
Repertoire.php....28
Tickets.php....29
Seats.php....30
Contact.php....34
Author.php....37
Activation.php....39
Models....40
Activation.php....40
addPerformance.php....41
addReprtoire.php....43
allPerformances.php....45
buyTickets.php....45
cancelTickets.php....47
connection.php....47
deleteMessage.php....48
deletePerformance.php....49

deleteRepertoire.php 49
disablePoll.php 50
editPerfromance.php 51
enablePoll.php 54
getEditPerformance.php 54
getMessagesAdmin.php 55
getNav.php 56
getTickets.php 56
getUsersAdmin.php 57
login.php 58
message.php 60
next.php 61
parameters.php 62
performance.php 62
performances.php 63
poll.php 64
pollRez.php 65
register.php 66
repertoire.php 68
repertoireAdmin.php 69
seats.php 70
vote.php 71
Main.js 72
Style.css 98

Uvod

Korišćeni programski jezici i tehnologije:

HTML, CSS, JavaScript, Jquery,PHP, Bootstrap

Opis funkcionalnosti

1. Dinamičko ispisivanje menija iz baze na svim stranama.
2. Dinamičko ispisivanje footer-a na svim stranama.
3. Ispisivanje sledeće predstave na repertoaru.
4. Pretraga i straničenje svih predstava.
5. Prikaz pojedinačne predstave
6. Prikaz repertoara za tekući mesec.
7. Mogućnost kupovine (otkazivanja) karata za ulogovane korisnike.
8. Prikaz ankete na kontakt strani. Samo ulogovani korisnici mogu glasati (samo jednom). Ukoliko je korisnik već glasao, dugme za glasanje je disable-ovano.
9. Kontakt strana za slanje poruke administratoru. Vrši se klijentska i serverska validacija podataka(ispisivanje poruke o greškama).
10. Logovanje i registracija korisnika(ispisivanje poruke o greškama).Slanje mejla za aktivaciju naloga(ne radi na web000).
11. Admin panel-prikaz svih predstava sa linkovima za brisanje i ažuriranje. Klikom na link za ažuriranje prikazuje se forma sa popunjenim podacima za odabranu predstavu. Mogućnost dodavanja predstave.
12. Admin panel-prikaz repertoara sa linkom za brisanje. Klikom na link za ažuriranje prikazuje se forma sa popunjenim podacima. Mogućnost dodavanja predstave na repertoar.
13. Admin panel-prikaz rezulata ankete (mogućnost disable-ovanja)
14. Admin panel-prikaz svih korisnika.
15. Admin panel-prikaz svih poruka (mogućnost brisanja).

Template

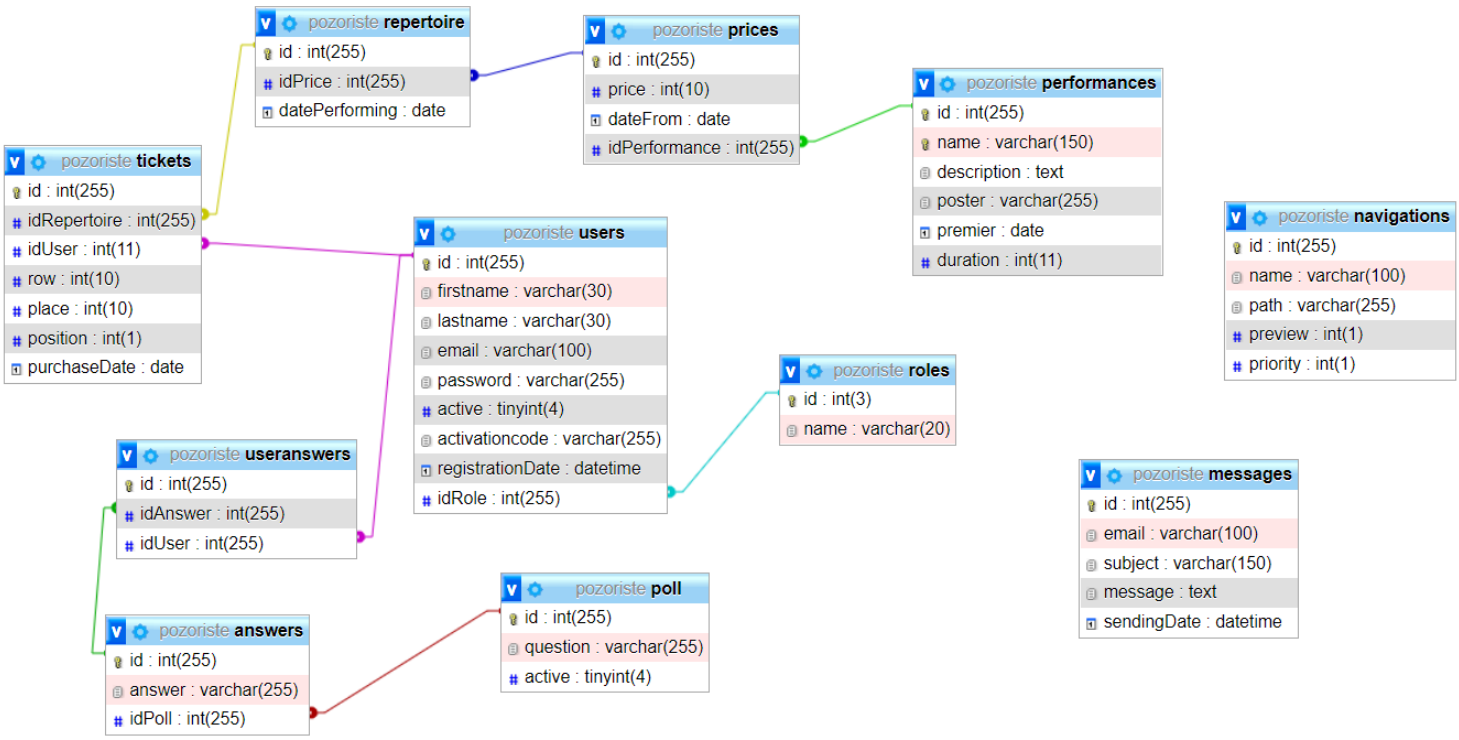
Template je koriscen za admin panel.

Organizacija

Organizaciona šema

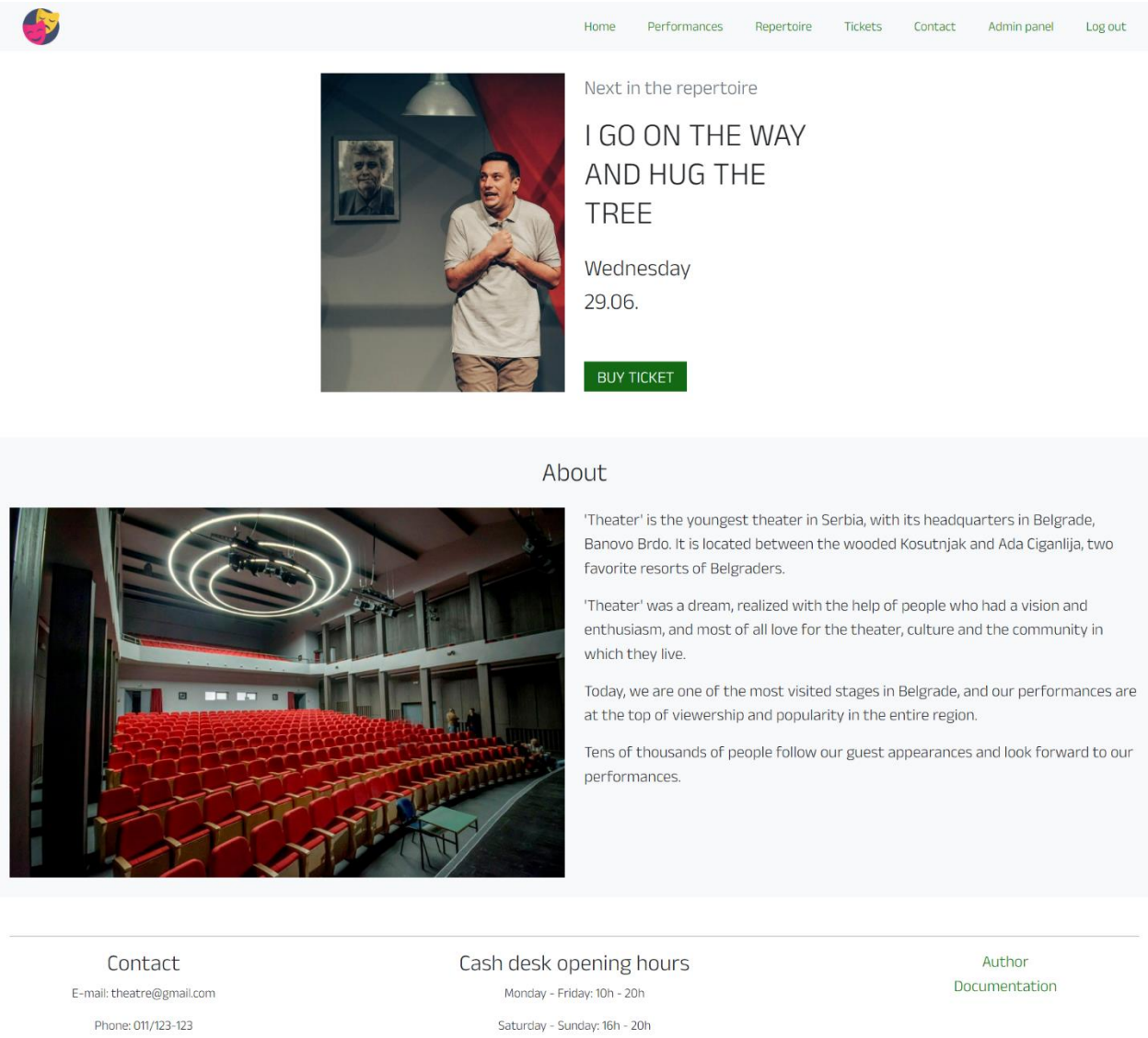
Svakoj stranici može se pristupiti sa svake stranice, pojedinačnoj predstavi je moguće pristupiti sa stranice sa svim predstavama. Pristup stranicama zavisi od uloge korisnika.

Šema baze



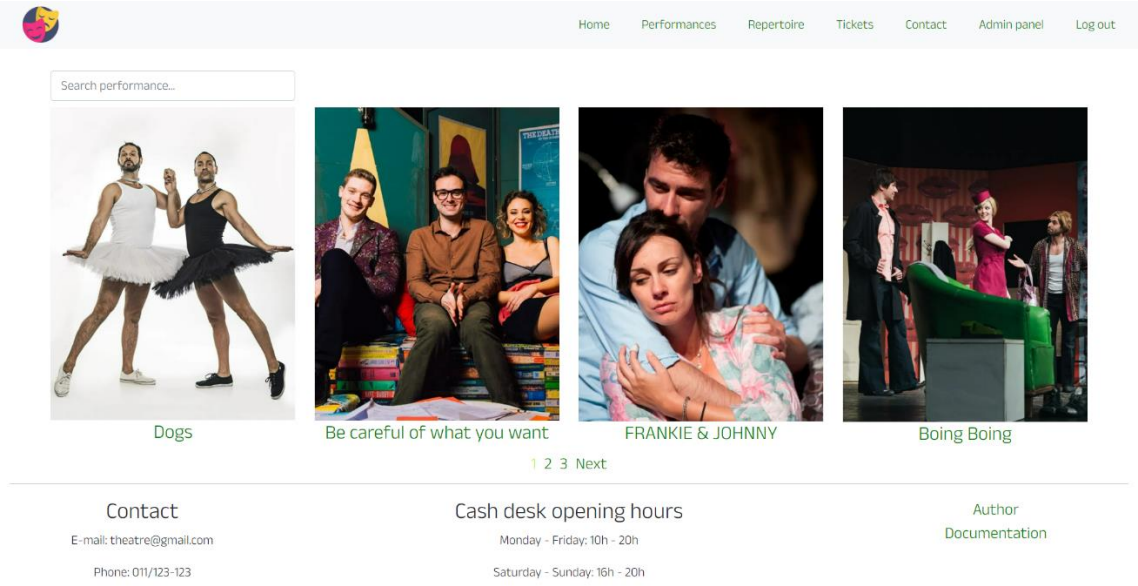
Stranice

Home



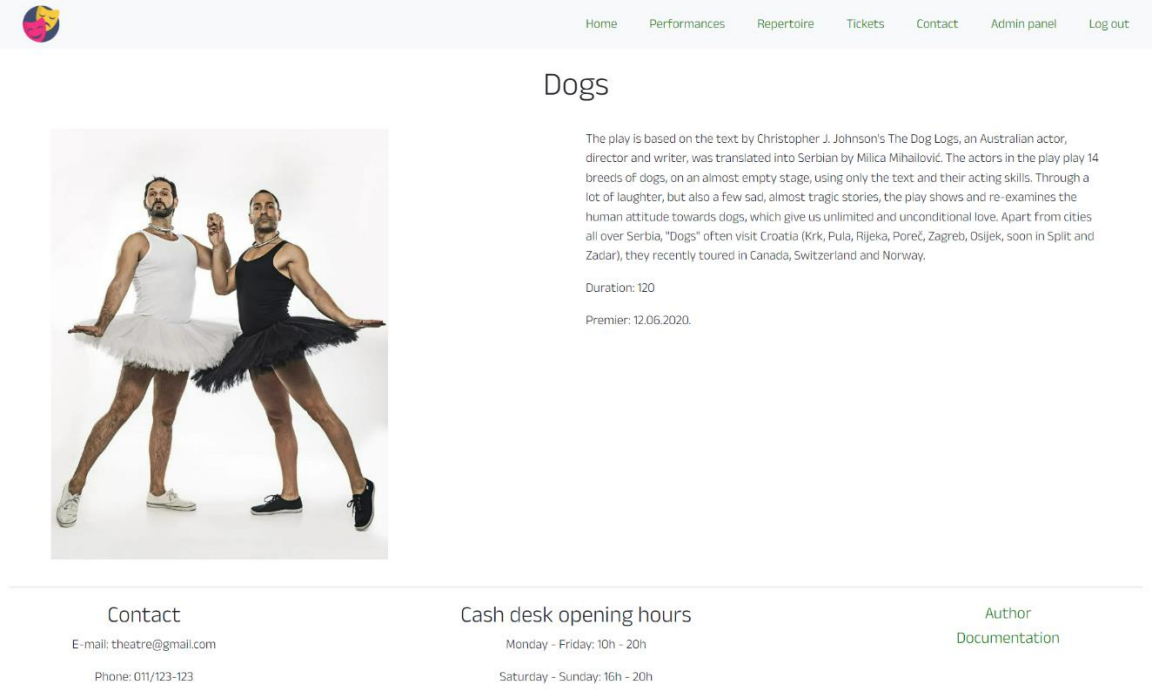
Pocetna strana sa prikazom naredne predstave na repertoaru, dinamički ispis hedera(sa navigacijom) i futera (na svim stranama)

Performances



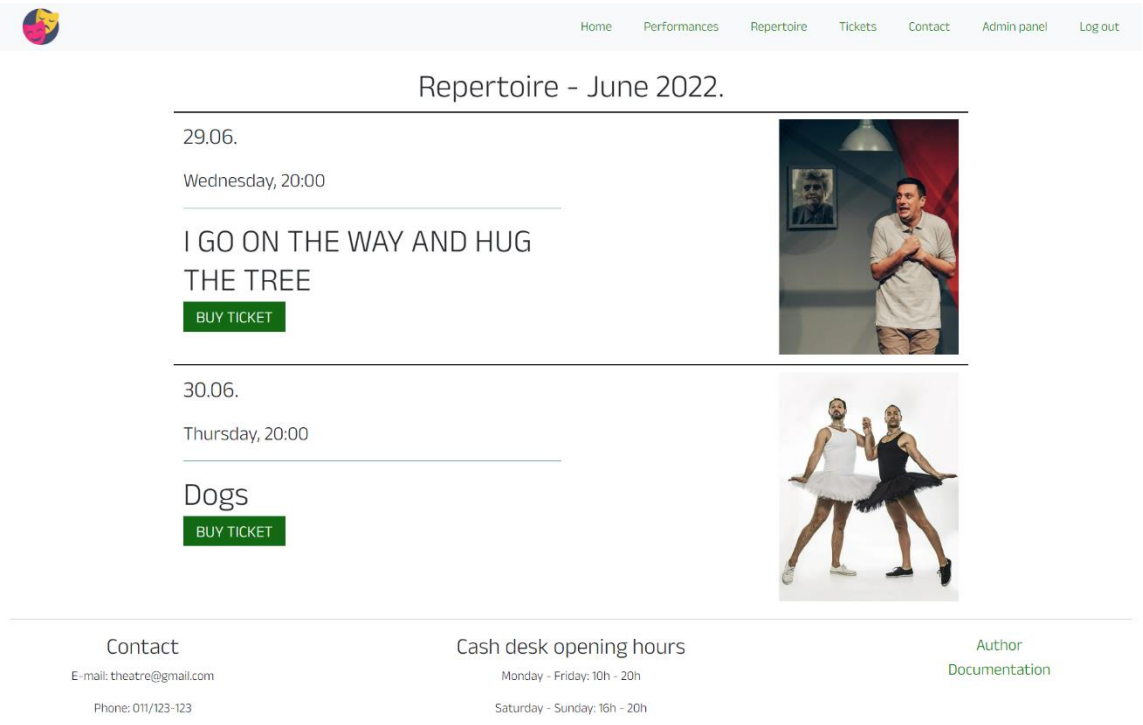
Prikaz svih predstava uz mogućnost straničenja I pretrage

Performance



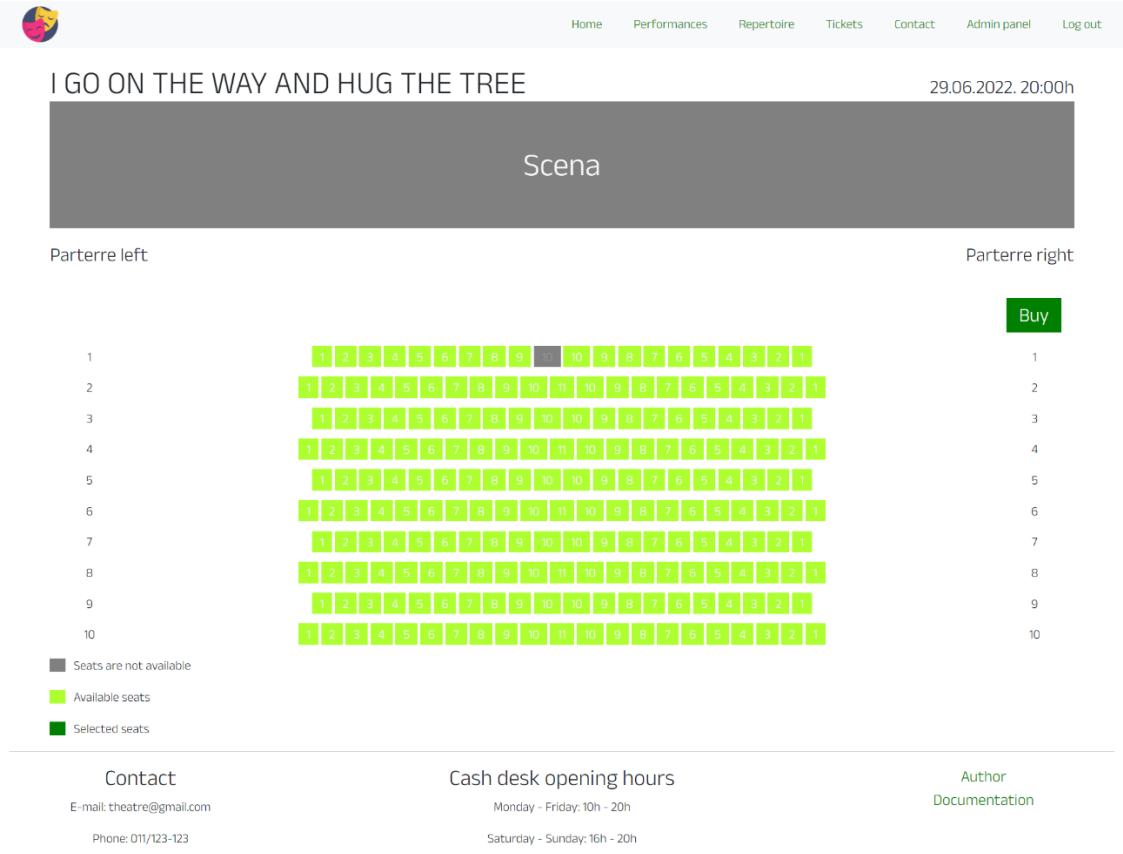
Prikaz jedne predstave sa njenim detaljima (naziv, slika, opis, trajanje I datum premijere)

Repertoire

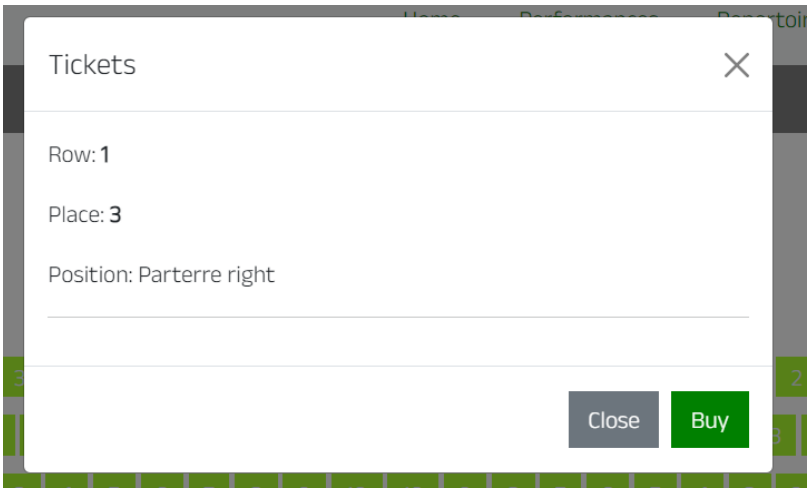


Prikaz repertoara za tekući mesec sa linkom za kupovinu karte

Kupovina

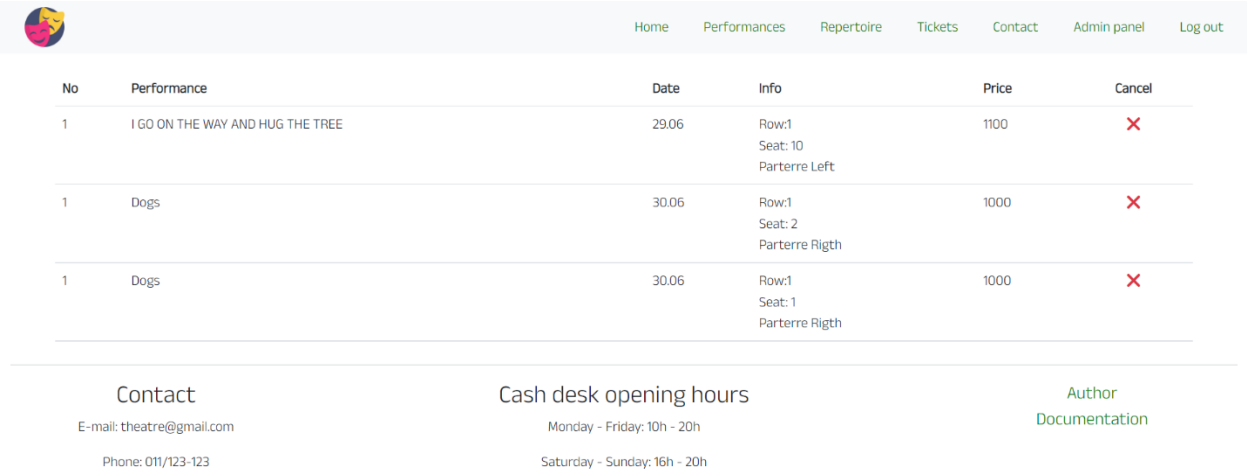


Prikaz ‘forme’ za kupovinu karata. Prikazana su sedista u pozoristu. Prelaskom misa preko sedista, prikazuju se njegovi detalji (red, pozicija I broj sedista). Klikom na dugme ‘Buy’ prikazuje se pop up sa podacima o izabranim mestim I klikom na dugme ‘buy’ kupuju se karte.



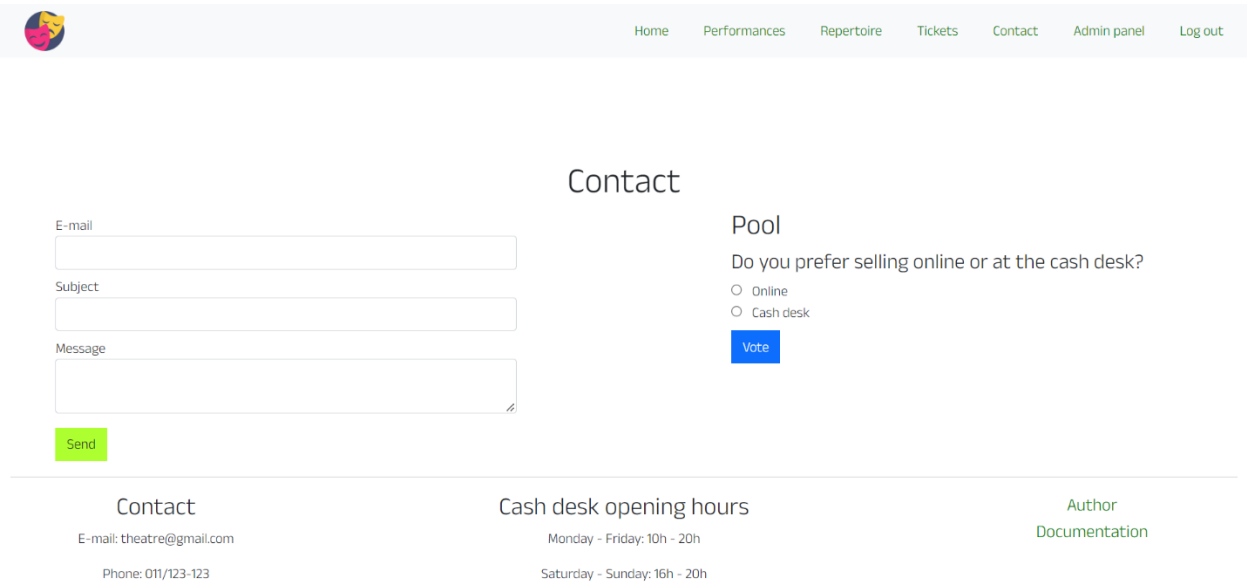
Prikaz pop up-a za kupovinu. Ukoliko korisnik nije izabrao sedista ili nije ulogovan ispisuje se odgovarajuca poruka. Nakon kupovine, ta sedista vise nije moguce kupiti.

Tickets



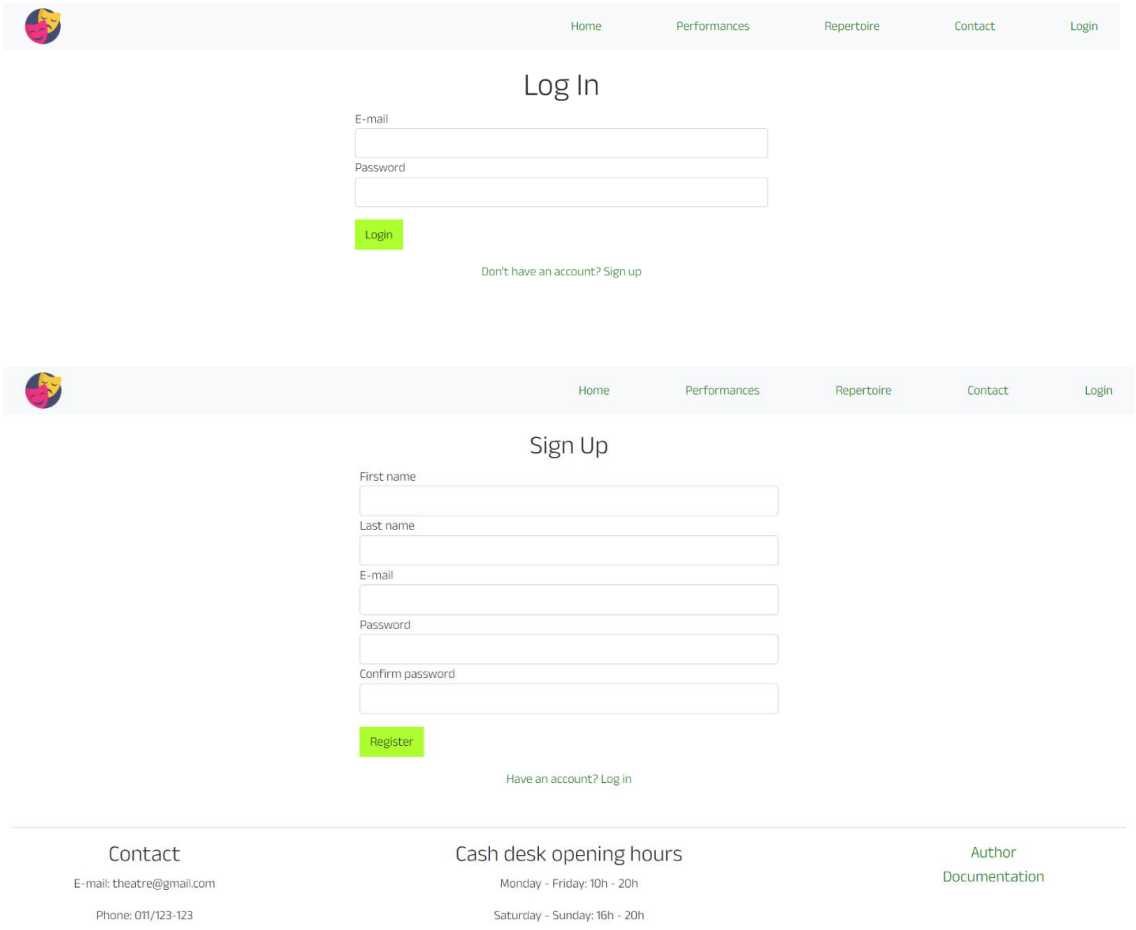
Prikaz kupljenih karata, moguce je otkazivanje kupovine.

Kontakt



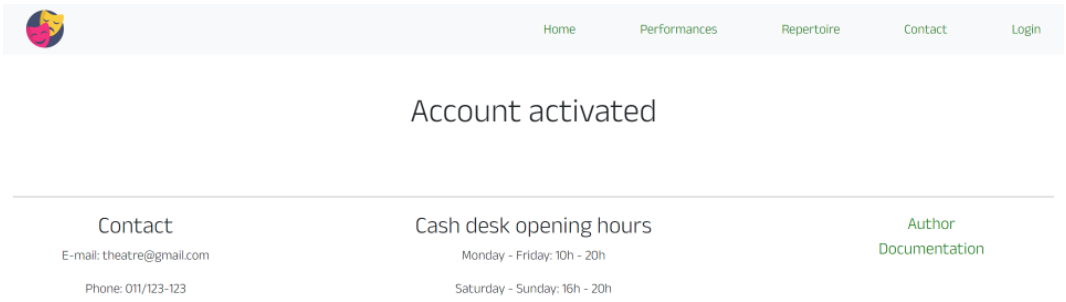
Prikaz kontakt forme za slanje poruke adminu . Popunjavanjem forme ispisuje se odgovarajuća poruka. Vrši se klijentska I serverska validacija podataka. Prikaz ankete. Mogu glasati samo ulogovani korisnici, kada korisnik glasa, ispise mu se poruka I on ne moze ponovo da glasa.

Registracija I logovanje



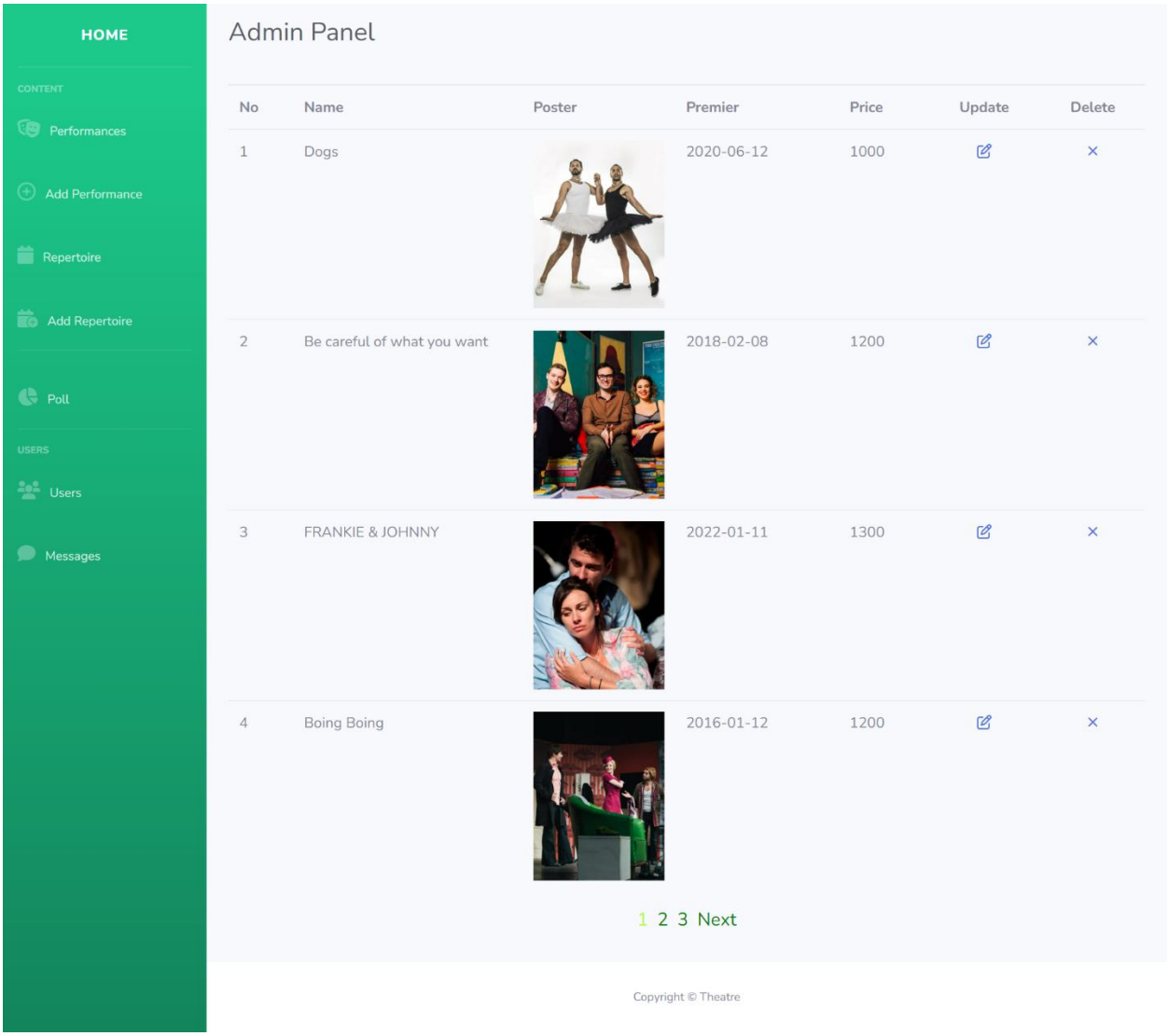
Prikaz formi za logovanje I registraciju. Vrsi se klijenska I serverska validacija I ispisuju odgovarajuce poruka. Slanje mejla ne radi na web000 pa je nalog odmah aktivan.

Ativacija



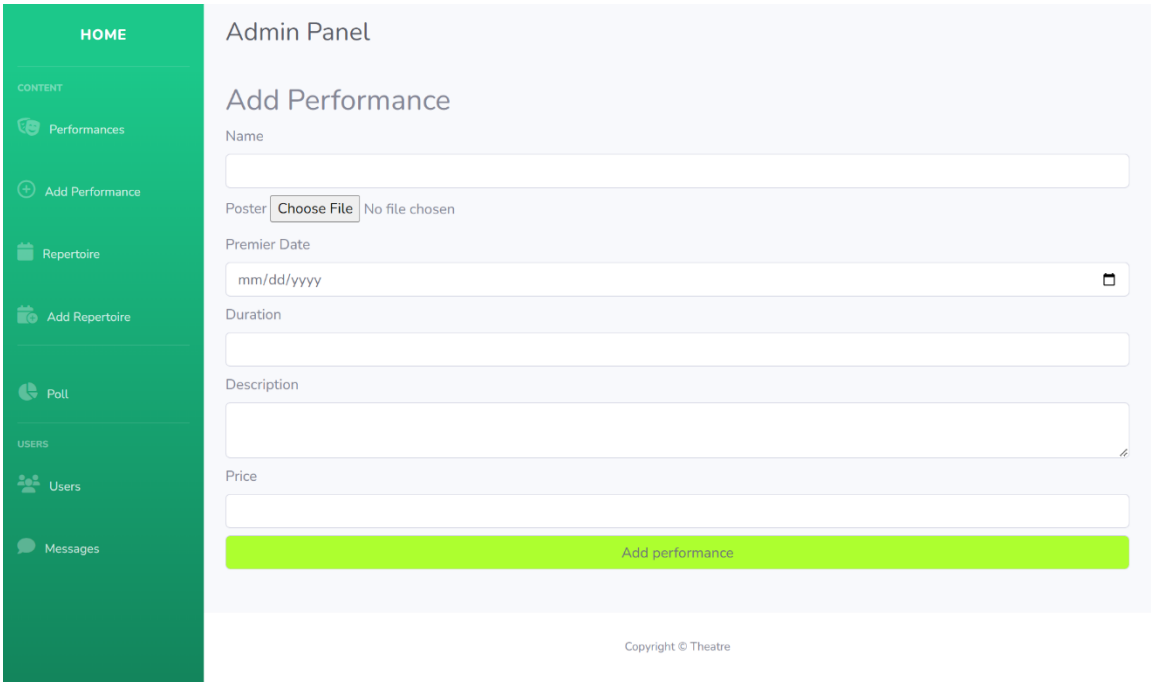
Prikaz stranice za aktivaciju naloga

Admin Performances



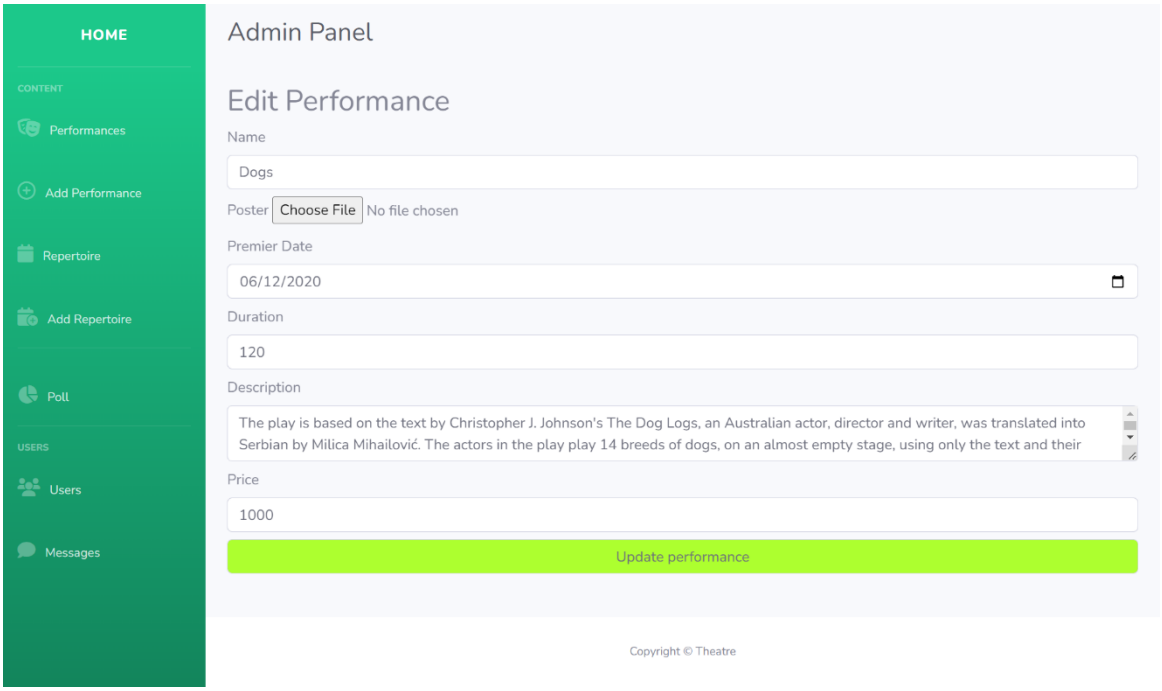
Prikaz svih predstava. Predstava se moze azurirati (klikom na link za azuriranje ispisuje se forma za odabranu predstavu) i obrisati.

Admin New Performance



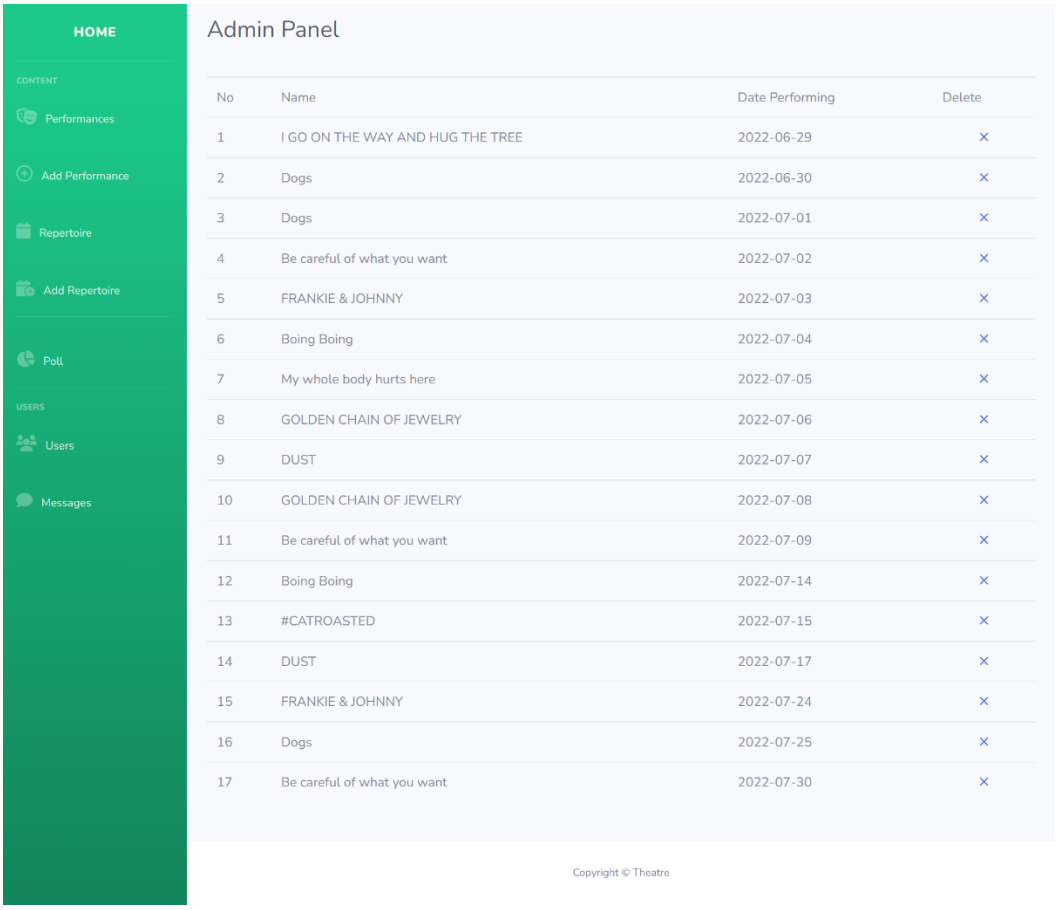
Prikaz forme za dodavanje predstave. Sva polja su obavezna

Admin Edit Performance



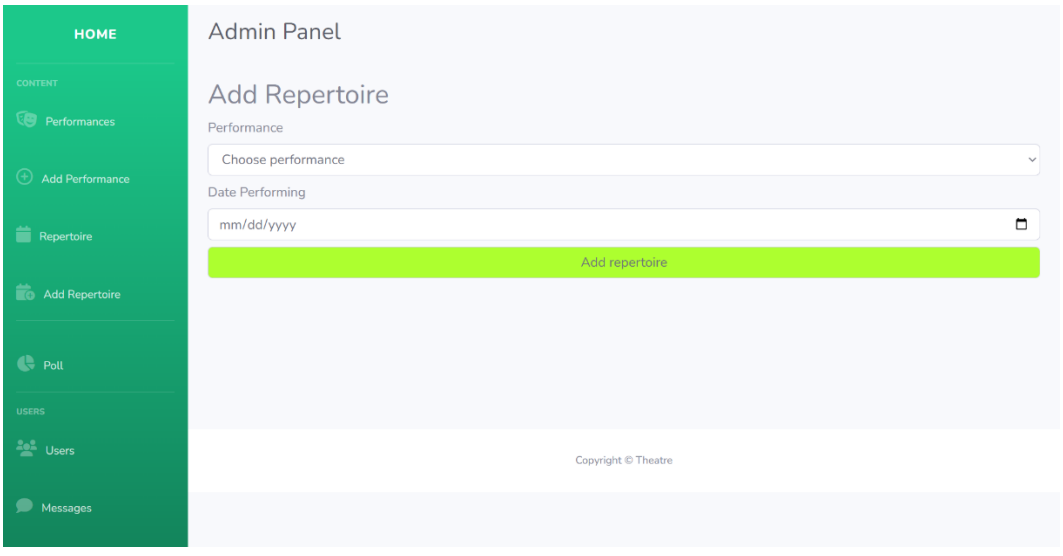
Prikaz forme za editovanje predstave. Sva polja su obavezna osim slike

Admin Repertoire



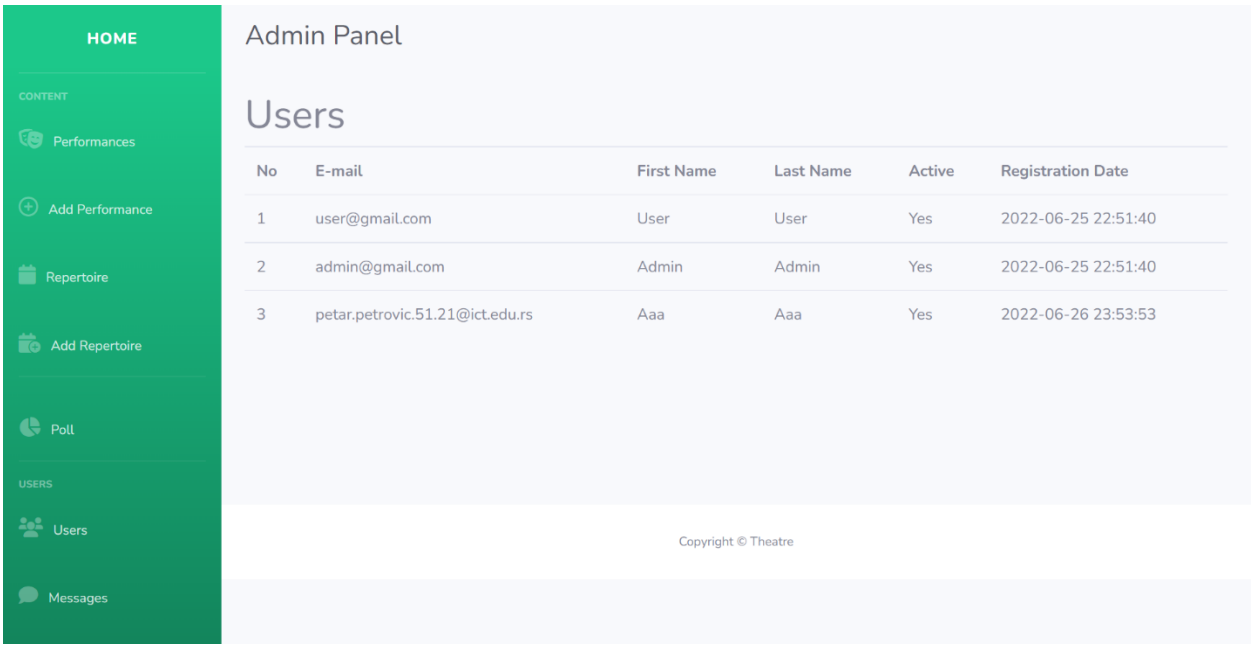
Prikaz repertoara za predstave koje sledece. Moguce je skloniti predstavu sa repertoara

Admin Add Repertoire



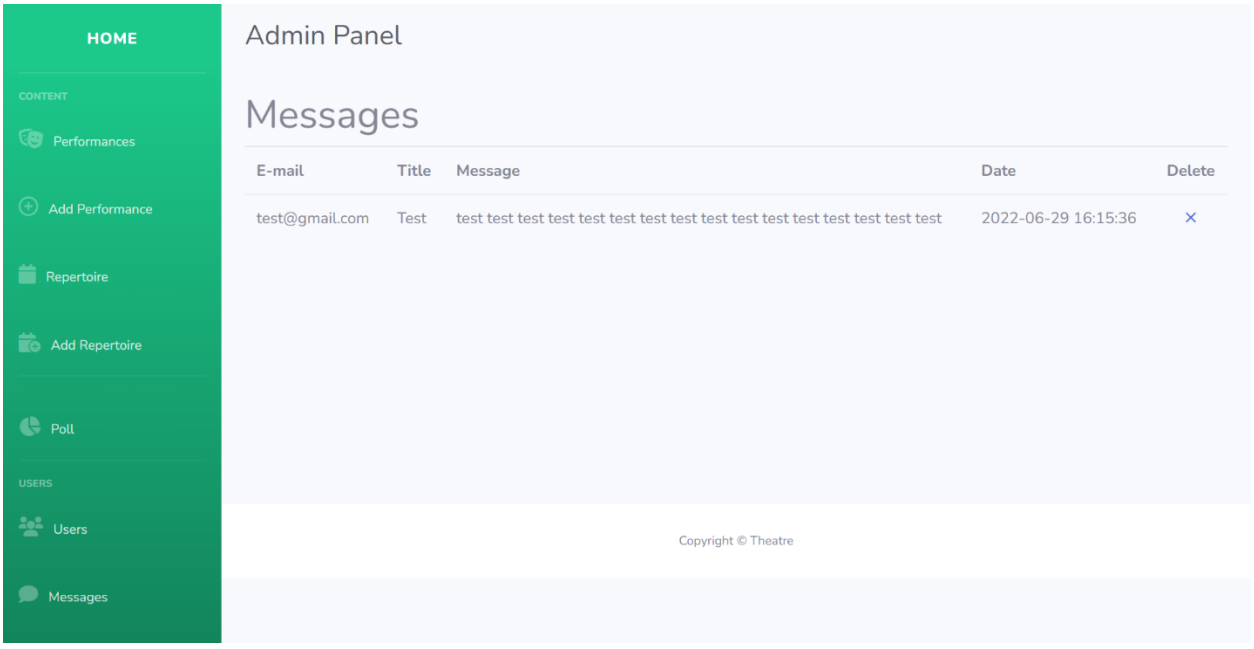
Forma za dodavanje predstave na repertoar. Vrsi se provera da li se tog dana vec ne igra neka predstava

Korisnici



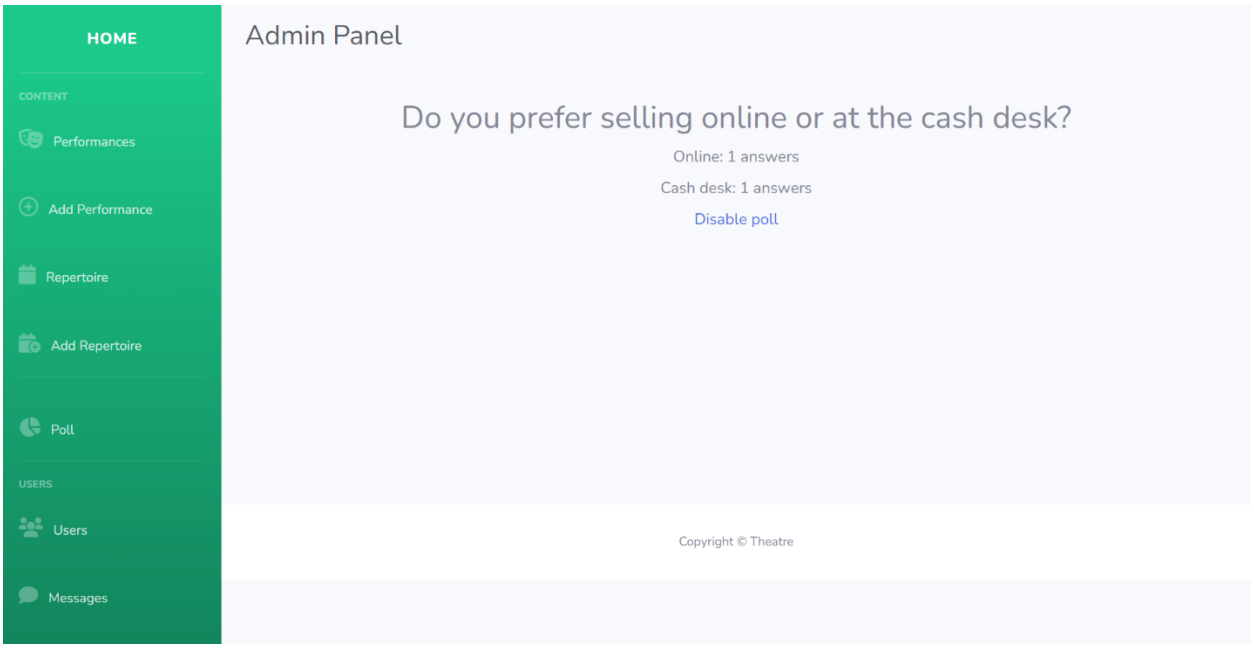
Prikaz svih korisnika

Messages



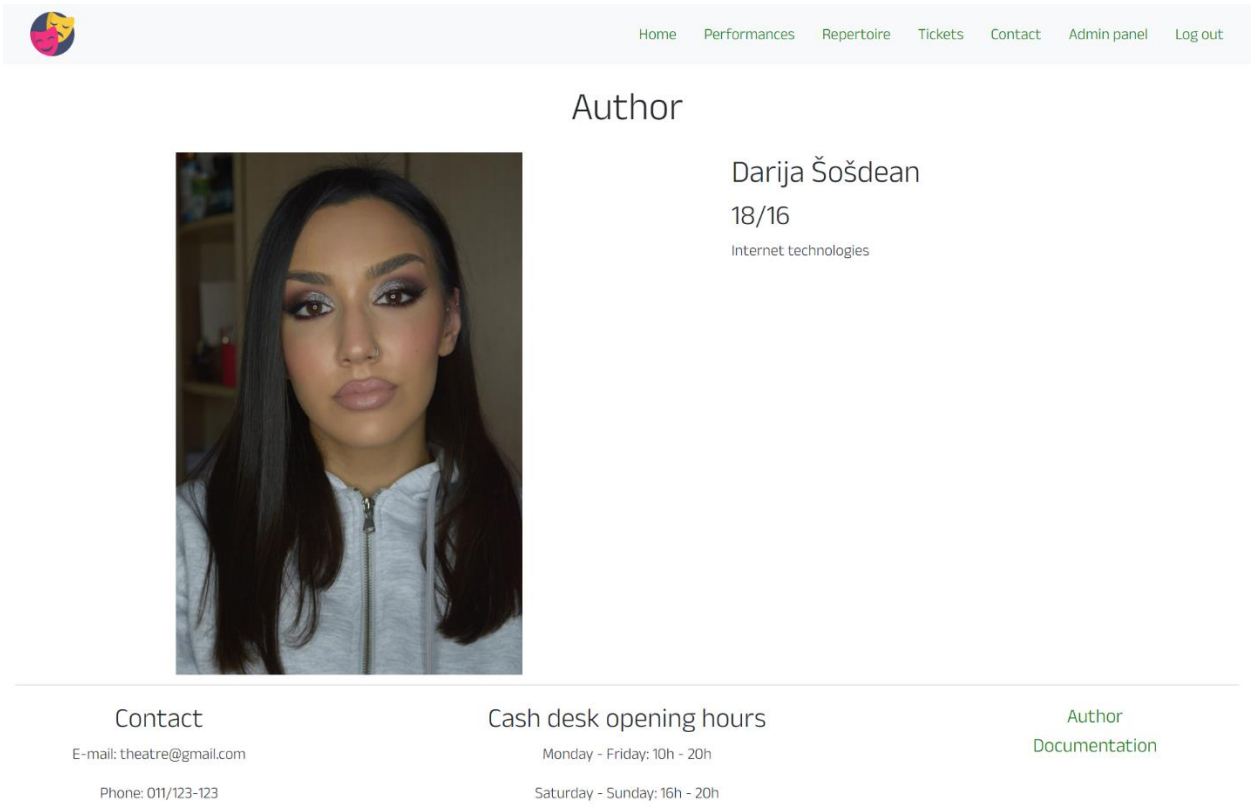
Prikaz poruka poslatih adminu preko kontakt forme sa linkom za brisanje

Anketa



Prikaz rezultata ankete. Moze se disejblovati i enejblovati

Autor



Kodovi

Root

Index.php

```php
<?php

    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    require_once "view/pages/index.php";

    require_once "view/fixed/footer.php";

?>
```

Activation.php

```php
<?php


    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    if(!isset($_SESSION['user'])){
        require_once "view/pages/activation.php";
    }
    else{
        echo "<main><h1 class='text-center my-5'>No Access</h1></main>";
    }


    require_once "view/fixed/footer.php";

?>
```

Performance.php

```php
<?php

    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    require_once "view/pages/performance.php";

    require_once "view/fixed/footer.php";

?>
```

Performances.php

```php
<?php


    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    require_once "view/pages/performances.php";

    require_once "view/fixed/footer.php";

?>
```

Repertoire.php

```php
<?php


    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    require_once "view/pages/repertoire.php";

    require_once "view/fixed/footer.php";

?>
```

Tickets.php

```php
<?php

    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    if(isset($_SESSION['user'])){
        require_once "view/pages/tickets.php";
    }
    else{
        echo "<main><div class='text-center'><h1>No access</h1></div></main>";
    }

    require_once "view/fixed/footer.php";

?>
```

Seats.php

```php
<?php


    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    require_once "view/pages/seats.php";

    require_once "view/fixed/footer.php";

?>
```

Contact.php

```php
<?php

    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    require_once "view/pages/contact.php";

    require_once "view/fixed/footer.php";

?>
```

Author.php

```php
<?php

    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    require_once "view/pages/author.php";

    require_once "view/fixed/footer.php";

?>
```

Login.php

```php
<?php


    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    if(isset($_SESSION['user'])){
        echo "<h1 class='text-center my-3'>No access</h1>";
```

```
    }
    else{
        require_once "view/pages/login.php";
    }

    require_once "view/fixed/footer.php";

?>
```

Activation.php

```
<?php


    require_once "view/fixed/head.php";

    require_once "view/fixed/header.php";

    if(!isset($_SESSION['user'])){
        require_once "view/pages/activation.php";
    }
    else{
        echo "<main><h1 class='text-center my-5'>No Access</h1></main>";
    }


    require_once "view/fixed/footer.php";

?>
```

View/fixed

Footer.php

```
<?php?>
<footer>

    <div class="container-fluid">
        <div class="row justify-content-between text-center">
            <div class="col-12">
                <hr class="m-0">
            </div>
            <div class="col-3 py-3">
                <h3>Contact</h3>
                <p>E-mail: theatre@gmail.com</p>
                <p>Phone: 011/123-123</p>
            </div>
            <div class="col-3 py-3">
                <h3>Cash desk opening hours</h3>
                <p>Monday - Friday: 10h - 20h</p>
                <p>Saturday - Sunday: 16h - 20h</p>
            </div>
            <div class="col-3 py-3">
                <a href="author.php" class="fs-5">Author</a>
                <br>
                <a href="doc.pdf" class="fs-5">Documentation</a>
            </div>
        </div>
    </div>

</footer>


    <script src="https://code.jquery.com/jquery-3.5.1.min.js" integrity="sha256-
9/aliU8dGd2tb6OSsuzixeV4y/faTqgFtohetphbbj0=" crossorigin="anonymous"></script>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js
" integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
crossorigin="anonymous"></script>
    <script src="assets/js/main.js" type="text/javascript"></script>
</body>
</html>
```

Head.php

```
<?php
    session_start();
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
    <link
href="https://fonts.googleapis.com/css2?family=Anek+Latin:wght@300&display=swap"
rel="stylesheet">
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
crossorigin="anonymous">
    <link rel="stylesheet" href="https://site-
assets.fontawesome.com/releases/v6.1.1/css/all.css">
    <link rel="stylesheet" href="assets/css/style.css" type="text/css">
    <meta name="keywords" content="Theatre, Performances, Repertoire, Tickets,
Belgrade"/>
    <meta name="description" content="'Theater' is the youngest theater in
Serbia, with its headquarters in Belgrade, Banovo Brdo."/>
    <title>Theatre</title>
    <link rel="shortcut icon" href="assets/images/logo.png">
</head>
<body>
```

Header.php

```
<?php
    require_once "models/getNav.php";
?>
<div class="container-fluid bg-light fixed-top">
    <div class="row py-2 px-3 justify-content-between align-items-center">
        <div class="col-5" id="logo">
            <a href="index.php" class="text-decoration-none text-dark">
                <img class="img-fluid" src="assets/images/logo.png" alt="Logo
image"/>
            </a>
        </div>
        <div class="col-6">
```

```
            <ul class="d-flex justify-content-between align-items-center m-0
nav">
                <?php
                    foreach($nav as $n):
                ?>
                    <li class="nav-item">
                        <a href="<?=$n['path']?>" class="text-decoration-
none"><?=$n['name']?></a>
                    </li>
                <?php
                    endforeach;
                ?>
            </ul>
        </div>
    </div>
</div>
```

View/pages

Index.php

```
<?php
    require_once "models/next.php";
?>

<main>
    <div class="container">
        <div class="row justify-content-center p-2">
            <?php
                if($next):
            ?>
                <div class="col-3">
                    <img src="assets/images/<?=$next['poster']?>" class="img-
fluid" alt="<?=$next['name']?>"/>
                </div>
                <div class="col-3 next">
                    <div>
                        <p class="text-secondary fs-4">Next in the repertoire</p>
                        <h1 class="mt-2 mb-4"><?=$next['name']?></h1>
                        <?php
                            $date=new DateTime($next['datePerforming']);
                        ?>
                        <p class="m-0 fs-3"><?=$date->format('l')?></p>
```

```
                        <p class="fs-3"><?=$date->format('d.m.')?></p>
                    </div>
                    <div class="my-1">
                    <a class="fs-5 text-light py-2 px-3 buy text-uppercase"
href="seats.php?id=<?=$next['id']?>">Buy ticket</a>
                    </div>
                </div>
            <?php
                else:
            ?>
                <p><?=$data?></p>
            <?php
                endif;
            ?>
        </div>
    </div>

    <div class="container-fluid bg-light py-4 my-5">
        <div class="row justify-content-between">
            <div class="col-12 text-center mb-3">
                <h2>About</h2>
            </div>
            <div class="col-6">
                <img src="assets/images/teatar.jpg" alt="Theatre"/>
            </div>
            <div class="col-6">
                <p class="fs-5">'Theater' is the youngest theater in Serbia, with
its headquarters in Belgrade, Banovo Brdo. It is located between the wooded
Kosutnjak and Ada Ciganlija, two favorite resorts of Belgraders.</p>
                <p class="fs-5">'Theater' was a dream, realized with the help of
people who had a vision and enthusiasm, and most of all love for the theater,
culture and the community in which they live.</p>
                <p class="fs-5">Today, we are one of the most visited stages in
Belgrade, and our performances are at the top of viewership and popularity in the
entire region.</p>
                <p class="fs-5">Tens of thousands of people follow our guest
appearances and look forward to our performances.</p>
            </div>
        </div>
    </div>
</main>
```

Performance.php

```php
<?php
    require_once "models/performance.php";
?>

<main>
    <div class="container">
        <div class="row justify-content-between">
                <?php
                    if($code!=200):
                ?>
                    <div class="col-12 text-center">
                        <h1><?=$data?></h1>
                    </div>
                <?php
                    else:
                ?>
                    <div class="col-12 text-center">
                        <h1>
                            <?=$data['name']?>
                        </h1>
                    </div>

                    <div class="col-4 my-4">
                        <img class="img-fluid"
src="assets/images/<?=$data['poster']?>" alt="<?=$data['name']?>"/>
                    </div>

                    <div class="col-6 my-4">
                        <p><?=$data['description']?></p>
                        <p>Duration: <?=$data['duration']?></p>
                            <?php
                                $date=new DateTime($data['premier']);
                            ?>
                        <p>Premier: <?=$date->format('d.m.Y.')?></p>
                    </div>

                <?php
                    endif;
                ?>
        </div>
    </div>
</main>
```

Performances.php

```
<?php
?>

<main>
    <div class="container">
        <div class="row">
            <div class="col-3 my-2">
                <form action="#">
                    <input id="search" class="form-control" type="search"
placeholder="Search performance..."/>
                </form>
            </div>
        </div>
        <div class="row" id="performances">
        </div>
        <div class="row">
            <div class="col-10 text-center mx-auto" id="pagination">
            </div>
        </div>
    </div>
</main>
```

Repertoire.php

```
<?php
    require_once "models/repertoire.php";
?>

<main>
    <div class="container">
        <div class="row">
            <?php
                if($code==200):
                    $getMont = new DateTime();
            ?>
            <div class="col-12 text-center">
                <h1>Repertoire - <?=$getMont->format('F Y.')?></h1>
            </div>
            <?php
                foreach($repertoire as $r):
            ?>
                <div class="col-9 mx-auto repertoire mb-1 py-2">
                    <div class="row justify-content-between">
                        <div class="col-6">
```

```
                            <?php
                                $date=new DateTime($r['datePerforming']);
                            ?>
                            <p class="fs-3"><?=$date->format('d.m.')?></p>
                            <p class="fs-4"><?=$date->format('l')?>, 20:00</p>
                            <hr class="text-success border-3 opacity-75"/>
                            <h2 class="fs-1"><?=$r['name']?></h2>
                            <a class="fs-5 text-light py-2 px-3 buy text-
uppercase" href="seats.php?id=<?=$r['id']?>">Buy ticket</a>
                        </div>
                        <div class="col-3">
                            <img class="img-fluid"
src="assets/images/<?=$r['poster']?>" alt="<?=$p['name']?>"/>
                        </div>
                    </div>
                </div>
            <?php
                endforeach;
            ?>
            <?php
                else:
            ?>
                <p><?=$message?></p>
            <?php
                endif;
            ?>
        </div>
    </div>
</main>
```

Tickets.php

```
<?php

?>


<main>
    <div class="container">
        <div class="row">
            <div class="col-12" id="table">

            </div>
        </div>
    </div>
</main>
```

Seats.php

```
<?php
    require_once "models/seats.php";

?>
<main>
    <div class="container">
        <div class="row">
            <?php
                if($code!=200):
            ?>
                <h1 class="text-center"><?=$data?></h1>
            <?php
                else:
            ?>
            <div class="col-12">
                <div class="row justify-content-between align-items-end">
                    <div class="col-6">
                        <h1 class="m-0"><?=$performance['name']?></h1>
                    </div>
                    <div class="col-6 text-end">
                        <?php
                            $date=new DateTime($performance['datePerforming']);
                        ?>
                        <p class="m-0 fs-4"><?=$date->format('d.m.Y.')?>
20:00h</p>
                    </div>
                </div>
            </div>
            <div class="col-12">
                <div class="scena">
                    <p class="text-light fs-1 m-0">Scena</p>
                </div>
            </div>
            <div class="col-12 parter mx-auto my-3">
                <div>
                    <p class="fs-4">Parterre left</p>
                </div>
                <div>
                    <p class="fs-4">Parterre right</p>
                </div>
            </div>
            <div class="col-12 mx-auto text-end">
                    <form action="#">
                        <div class="px-3 py-1">
```

```
                            <input type="button" id="checkSeats" class="fs-4
text-bold px-3 py-1 kupi text-light border-0" value="Buy" data-bs-toggle="modal"
data-bs-target="#modalTickets">
                            <input type="hidden" id="repertoire"
value="<?=$_GET['id']?>"/>
                        </div>
                    </form>
            </div>
            <div class="col-12 mx-auto">
                <?php
                    for($i=1;$i<=10;$i++):
                        $jMax=10;
                    ?>

                    <div class="redDrzac">
                        <div class="red">
                            <span><?=$i?></span>
                        </div>

                        <div id="places">
                    <?php
                        for($j=1;$j<=$jMax;$j++):

                ?>
                <!-- POSITION!!!!! -->
                <!-- 1 levo -->
                <!-- 2 desno -->
                <!-- 3 sredina -->


                            <a href="#" class="places
                                <?php
                                foreach($data as $d):
                                    if($d['position']==1 && $d['row']==$i &&
$d['place']==$j):

                                ?>
                                    purchased
                                <?php
                                    endif;
                                endforeach;
                                ?>
                            " data-position="1" data-row="<?=$i?>" data-
place="<?=$j?>"><?=$j?><span class="popup">
                                Row: <span class="font-weight-
bold"><?=$i?></span><br>
```

```
                                Seat: <span class="font-weight-
bold"><?=$j?></span><br>
                                Parterre left <br>
                                Price : <?=$performance['price']?>
                            </span></a>


                <?php
                        endfor;
                ?>
                <?php
                        if($i%2==0):
                ?>
                        <a href="#" class="places
                                <?php
                                foreach($data as $d):
                                    if($d['position']==3 && $d['row']==$i &&
$d['place']==$j):
                                ?>
                                    purchased
                                <?php
                                    endif;
                                endforeach;
                                ?>
                            " data-position="3" data-row="<?=$i?>" data-
place="11">11<span class="popup">
                                Row: <span class="font-weight-
bold"><?=$i?></span><br>
                                Seat: <span class="font-weight-
bold">11</span><br>
                                Middle <br>
                                Price : <?=$performance['price']?>
                            </span></a>
                <?php
                        endif;
                ?>
                <?php
                        for($j=$jMax;$j>=1;$j--):
                ?>

                            <a href="#" class="places
                                <?php
                                foreach($data as $d):
                                    if($d['position']==2 && $d['row']==$i &&
$d['place']==$j):
```

```
                                ?>
                                    purchased
                                <?php
                                    endif;
                                endforeach;
                                ?>" data-position="2" data-row="<?=$i?>" data-
place="<?=$j?>"><?=$j?><span class="popup">
                                Row: <span class="font-weight-
bold"><?=$i?></span><br>
                                Seat: <span class="font-weight-
bold"><?=$j?></span><br>
                                Parterre right <br>
                                Price : <?=$performance['price']?>
                            </span></a>




                <?php

                        endfor;

                ?>
                        </div>

                        <div class="red desno">
                            <span><?=$i?></span>
                        </div>
                    </div>
                <?php
                    endfor;
                ?>
            </div>


            <!-- Modal -->
            <div class="modal fade" id="modalTickets" tabindex="-1" aria-
labelledby="exampleModalLabel" aria-hidden="true">
            <div class="modal-dialog">
                <div class="modal-content">
                <div class="modal-header">
                    <h5 class="modal-title" id="exampleModalLabel">Tickets</h5>
                    <button type="button" class="btn-close" data-bs-
dismiss="modal" aria-label="Close"></button>
                </div>
```

```
                <div class="modal-body">
                    <div id="seatsMessage">

                    </div>
                    <div id="buyMessage"></div>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-bs-
dismiss="modal">Close</button>
                    <button type="button" id="buy" class="btn
btnBuy">Buy</button>
                </div>
                </div>
            </div>
            </div>

            <?php
                endif;
            ?>
        </div>
        <?php
            if($code==200):
        ?>
        <div class="col-12">
            <p><span class="boja nedostupna"></span> Seats are not available</p>
            <p><span class="boja dostupna"></span> Available seats</p>
            <p><span class="boja izabrana"></span> Selected seats</p>
        </div>

        <?php
            endif;
        ?>
    </div>
</main>
```

Contact.php

```
<?php
    require_once "models/poll.php";
?>
<main>
    <div class="container pt-5">
        <div class="row pt-5 text-center">
            <h1>Contact</h1>
        </div>
```

```
        <div class="row justify-content-between">
            <div class="col-5">
                <form action="#" onSubmit="return sendMessage()">
                    <div class="row messageDiv">
                        <div class="col-12 mt-2">
                            <label for="email">E-mail</label>
                            <input type="text" id="email" class="form-control">
                            <p>E-mail formats: petar.petrovic.123.21@ict.edu.rs
or petar@gmail.com</p>
                        </div>
                        <div class="col-12 mt-2">
                            <label for="title">Subject</label>
                            <input type="text" id="subject" class="form-control">
                            <p>You have to write subject</p>
                        </div>
                        <div class="col-12 mt-2">
                            <label for="message">Message</label>
                            <textarea class="col-12 form-control"
id="message"></textarea>
                            <p>Message has 10 words minimum</p>
                        </div>
                        <div class="col-12 mt-2">
                        </div>
                        <div class="col-12 mt-2">
                            <input type="submit" id="btnMessage" class="btn"
value="Send">
                            <div class="my-2">
                                <p id="alertMessage"></p>
                            </div>
                        </div>
                    </div>
                </form>
            </div>
            <div class="col-5">
                <h2>Pool</h2>
                <?php
                    if(isset($_SESSION['user'])):
                ?>
                <form action="#">

                    <?php
                        if(isset($message)):
                    ?>
                            <p><?=$message?></p>
                    <?php
```

```
                        else:
                    ?>
                    <h4><?=$aktivna['question']?></h4>
                    <?php
                        foreach($odgovori as $o):
                    ?>
                        <input type="radio" value="<?=$o['id']?>" name="odg"
class="odg"/>
                        <span class="mx-2"><?=$o['answer']?></span>
                        <br/>
                    <?php
                        endforeach;
                    ?>
                        <input type="button" value="Vote" id="vote"
                    <?php
                        if(isset($obavestenje)):
                        ?>
                            disabled="disabled"
                        <?php
                        endif;
                        ?>
                            class="btn btn-primary mt-2">
                        <?php
                            if(isset($obavestenje)):
                            ?>
                                <p><?=$obavestenje?></p>
                            <?php
                            endif;
                        ?>
                            <p id="obavestenje"></p>
                    <?php
                        endif;
                    ?>
                </form>
                <?php
                    else:
                ?>
                    <h3>Log in to see our poll!</h3>
                <?php
                    endif;
                ?>
            </div>
        </div>
    </div>
</main>
```

Author.php

```
<?php?>

<main>
    <div class="container">
        <div class="row my-3">
            <div class="col-12 text-center">
                <h1>Author</h1>
            </div>
        </div>
        <div class="row justify-content-around">
            <div class="col-4">
                <img class="img-fluid" src="assets/images/author.jfif"
alt="Author">
            </div>
            <div class="col-4">
                <h2>Darija Šošdean</h2>
                <h3>18/16</h3>
                <p>Internet technologies</p>
            </div>
        </div>
    </div>
</main>
```

Login.php

```
<?php


?>

<main>
    <div class="container my-5 ">
        <div class="row justify-content-center divLogReg">
            <div class="col-5 mx-auto" id="logDiv">
                <h1 class="text-center">Log In</h1>
                <form action="#" onSubmit="return login()">
                    <div>
                        <label for="email">E-mail</label>
                        <input type="text" id="email" class="form-control">
                        <p>E-mail formats: petar.petrovic.123.21@ict.edu.rs or
petar@gmail.com.</p>
                    </div>
                    <div>
                        <label for="pass">Password</label>
```

```
                        <input type="password" id="password" class="form-
control">
                        <p>Password must be at least 8 characters long.</p>
                    </div>
                    <div id="messageLogin" class="my-2"></div>
                    <div class="my-3">
                        <input type="submit" id="btnLogin" value="Login"
class="btn">
                    </div>
                    <div class="text-center">
                        <a href="#" id="regBtn" class="green">Don't have an
account? Sign up</a>
                    </div>
                </form>
            </div>

            <div class="col-5 mx-auto d-none" id="regDiv">
                <h2 class="text-center">Sign Up</h2>
                <form action="#" onSubmit="return register()">
                    <div>
                        <label for="fname">First name</label>
                        <input type="text" id="firstName" class="form-control">
                        <p>First name has capital letter,3 letter minimum,
30maximum</p>
                    </div>
                    <div>
                        <label for="lName">Last name</label>
                        <input type="text" id="lastName" class="form-control">
                        <p>Last name has capital letter,3 letter minimum,
30maximum</p>
                    </div>
                    <div>
                        <label for="email">E-mail</label>
                        <input type="text" id="emailReg" class="form-control">
                        <p>E-mail formats: petar.petrovic.123.21@ict.edu.rs or
petar@gmail.com</p>
                    </div>
                    <div>
                        <label for="pass">Password</label>
                        <input type="password" id="pass" class="form-control">
                        <p>Password must be at least 8 characters long.</p>
                    </div>
                    <div>
                        <label for="passConfirm">Confirm password</label>
```

```
                        <input type="password" id="passConfirm" class="form-
control">
                        <p>Passwords do not match</p>
                    </div>
                    <div id="messageRegister" class="my-2"></div>
                    <div class="my-3">
                        <input type="submit" id="btnRegister" value="Register"
class="btn">
                    </div>
                    <div class="text-center">
                        <a href="#" id="logBtn" class="green">Have an account?
Log in</a>
                    </div>
                </form>
            </div>
        </div>
    </div>
</main>
```

Activation.php

```
<?php

    require_once "models/activation.php";
?>

<main>
    <div class="container my-5">
        <div class="row my-4 text-center">
            <div class="col-12 my-4">
                <h1><?=$message?></h1>
            </div>
        </div>
    </div>
</main>
```

Models

Activation.php

```php
<?php
    require_once "connection.php";

    if(isset($_GET['code'])){
        $code=$_GET['code'];

        $query="SELECT * FROM users WHERE activationcode=:code";

        $prepare=$connection->prepare($query);
        $prepare->bindParam(":code",$code);
        try{
            $prepare->execute();
            if($prepare->rowCount()==1){
                $korisnik=$prepare->fetch();
                $aktivan=1;
                $query2="UPDATE users SET active=:active WHERE
activationcode=:code";
                $prepare2=$connection->prepare($query2);
                $prepare2->bindParam(":active",$aktivan);
                $prepare2->bindParam(":code",$code);
                try{
                    $prepare2->execute();
                    $message="Account activated";
                }
                catch(PDOException $e){
                    $message="Server error";
                }
            }
            else{
                $message="Code Not Found";
            }
        }
        catch(PDOException $e){
            // $message="Server error";
            $message=$e->getMessage();
        }
    }
    else{
        $message="No Access";
    }

?>
```

addPerformance.php

```php
<?php
    require_once "connection.php";

    $code=200;
    $data="Inserted";


    $name=$_POST['name'];
    $duration=$_POST['duration'];
    $premier=$_POST['premier'];
    $price=$_POST['price'];
    $description=$_POST['description'];

    $err=0;
    if(strlen($name)<1){
        $data="Name has 1 letter minimum";
        $err++;
    }
    if(!isset($_FILES['file'])){
        $data="You have to choose image";
        $err++;
    }
    if(strlen($description)<15){
        $data="Description has 15 letter minimum";
        $err++;
    }
    if($duration>180 || $duration==""){
        $data="Duration is number (minutes), values 180 maximum";
        $err++;
    }
    if($premier==""){
        $data="You have to choose premier date";
        $err++;
    }
    if($price==""){
        $data="You have to choose price";
        $err++;
    }

    if($err==0){
        $code=201;

        $ifNameExists="SELECT * FROM performances WHERE name like :name";
```

```
        $prepareTest=$connection->prepare($ifNameExists);

        $namePrepare="%".$name."%";
        $prepareTest->bindParam(":name",$namePrepare);


        try{
            $prepareTest->execute();

            if($prepareTest->rowCount()==1){
                $code=200;
                $data="Name already exists";
            }
            else{
                $slika=$_FILES['file'];

                $tmpName=$slika['tmp_name'];
                $size=$slika['size'];
                $tip=$slika['type'];
                $nameImage=$slika['name'];
                $naziv=time().$nameImage;
                $putanja="../assets/images/$naziv";
                move_uploaded_file($tmpName,$putanja);

                $query="INSERT INTO performances VALUES
(null,:name,:description,:poster,:premier,:duration) ";

                $prepare=$connection->prepare($query);

                $prepare->bindParam(":name",$name);
                $prepare->bindParam(":description",$description);
                $prepare->bindParam(":poster",$naziv);
                $prepare->bindParam(":premier",$premier);
                $prepare->bindParam(":duration",$duration);

                $prepare->execute();

                $id=$connection->lastInsertId();


                $inserQuery="INSERT INTO prices VALUES
(null,:price,:dateFrom,:idPerformance)";

                $prepare2=$connection->prepare($inserQuery);
```

```
                $date=date("Y-m-d");

                $prepare2->bindParam(":price",$price);
                $prepare2->bindParam(":dateFrom",$date);
                $prepare2->bindParam(":idPerformance",$id);
                $prepare2->execute();
                $data="Inserted";
            }

        }
        catch(PDOException $e){
            $data=$e->getMessage();
            $code=500;
        }
    }


    echo json_encode($data);
    http_response_code($code);
?>
```

addReprtoire.php

```
<?php
    require_once "connection.php";


    if(isset($_POST['btn'])){
        $code=200;



        $idPerformance=$_POST['idPerformance'];

        $err=0;
        if($idPerformance==0){
            $data="You have to choose performance";
            $err++;
        }
        if(!isset($_POST['date']) || strtotime($_POST['date'])<time()){
            $data="You have to choose date in future";
            $err++;
        }

        if($err==0){
            $date=$_POST['date'];
```

```
            $query="SELECT * FROM repertoire WHERE
datePerforming=:datePerforming";

            $prepare=$connection->prepare($query);

            $prepare->bindParam(":datePerforming",$date);

            try{
                $prepare->execute();
                if($prepare->rowCount()==1){
                    $data="Date already reserved!";
                }
                else{
                    $query2="INSERT INTO repertoire  VALUES (NULL, :id,
:datePerforming)";

                    $prepare2=$connection->prepare($query2);

                    $prepare2->bindParam(":id",$idPerformance);
                    $prepare2->bindParam(":datePerforming",$date);

                    $prepare2->execute();
                    $code=201;
                    $data="Added to repertoire";
                }

            }
            catch(Exception $e){
                $code=500;
                $data="Server error";
            }
        }

    }
    else{
        $code=404;
        $data="Not found";
    }


echo json_encode($data);
http_response_code($code);
?>
```

allPerformances.php

```php
<?php
    require_once "connection.php";


    if(isset($_GET['btn'])){
        $code=200;




        $query="SELECT *,pr.id FROM performances p INNER JOIN prices pr
                ON p.id=pr.idPerformance";


        try{
            $performances=$connection->query($query);
            $data=$connection->query($query)->fetchAll();

        }
        catch(Exception $e){
            $code=500;
            $data="Server error";
        }
    }
    else{
        $code=404;
        $data="Not found";
    }


echo json_encode($data);
http_response_code($code);
?>
```

buyTickets.php

```php
<?php
session_start();
    require_once "connection.php";

    $data="";
    $code=200;

    if(isset($_POST['btnBuy'])){
```

```
        $tickets=$_POST['tickets'];
        if(isset($_SESSION['user'])){
            $idUser=$_SESSION['user']['id'];
            $query="INSERT INTO tickets VALUES (NULL, :idRepertoire, :idUser,
:placeRow, :place, :position, :purchaseDate)";

            foreach($tickets as $t){

                try{
                    $prepare=$connection->prepare($query);
                    $date=date("Y-m-d H:i:s");

                    $prepare->bindParam(":idRepertoire",$t['idRepertoire']);
                    $prepare->bindParam(":idUser",$idUser);
                    $prepare->bindParam(":placeRow",$t['row']);
                    $prepare->bindParam(":place",$t['place']);
                    $prepare->bindParam(":position",$t['position']);
                    $prepare->bindParam(":purchaseDate",$date);

                    $prepare->execute();
                    $code=201;
                    $data="Tickets were successfully purchased";
                }
                catch(PDOException $e){
                    $data="Server error";
                    $code=500;
                }
            }
        }
        else{
           $data="Login to buy";
           $code=401;
        }


    }
    else{
        $data= "NOT FOUND";
        $code=404;
    }


echo json_encode($data);
http_response_code($code);
```

```
?>
```

cancelTickets.php

```php
<?php
    require_once "connection.php";

    $code=200;

    if(isset($_POST['cancelBtn'])){
        $query="DELETE FROM tickets WHERE id=:id";

        $priprema=$connection->prepare($query);

        $priprema->bindParam(":id",$_POST['id']);

        try{
            $priprema->execute();
            $data="Deleted";
        }
        catch(PDOException $e){
            $data="Server error";
            $code=500;
            // $data=$e->getMessage();
        }
    }
    else{
        $code=404;
        $data="Not found";
    }


echo json_encode($data);
http_response_code($code);
?>
```

connection.php

```php
<?php
    require_once "parameters.php";

    try{
```

```
        $connection=new
PDO("mysql:host=$host;dbname=$baza",$user,$pass,array(PDO::MYSQL_ATTR_INIT_COMMAN
D=>"SET NAMES 'utf8' COLLATE 'utf8_unicode_ci'"));
        $connection->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    }
    catch (PDOException $e)
    {
        echo $e->getMessage();
    }
?>
```

deleteMessage.php

```
<?php
    require_once "connection.php";

    $code=200;

    if(isset($_POST['id'])){
        $query="DELETE FROM messages WHERE id=:id";

        $priprema=$connection->prepare($query);

        $priprema->bindParam(":id",$_POST['id']);

        try{
            $priprema->execute();
            $data="Deleted";
        }
        catch(PDOException $e){
            $data="Server error";
            $code=500;
            // $data=$e->getMessage();
        }
    }
    else{
        $code=404;
        $data="Not found";
    }


echo json_encode($data);
http_response_code($code);
?>
```

deletePerformance.php

```php
<?php
    require_once "connection.php";

    $code=200;

    if(isset($_POST['id'])){
        $query="DELETE FROM performances WHERE id=:id";

        $priprema=$connection->prepare($query);

        $priprema->bindParam(":id",$_POST['id']);

        try{
            $priprema->execute();
            $data="Deleted";
        }
        catch(PDOException $e){
            $data="Server error";
            $code=500;
            // $data=$e->getMessage();
        }
    }
    else{
        $code=404;
        $data="Not found";
    }


echo json_encode($data);
http_response_code($code);
?>
```

deleteRepertoire.php

```php
<?php
    require_once "connection.php";

    $code=200;

    if(isset($_POST['id'])){
        $query="DELETE FROM repertoire WHERE id=:id";

        $priprema=$connection->prepare($query);

        $priprema->bindParam(":id",$_POST['id']);

        try{
            $priprema->execute();
            $data="Deleted";
        }
        catch(PDOException $e){
            $data="Server error";
            $code=500;
            // $data=$e->getMessage();
        }
    }
    else{
        $code=404;
        $data="Not found";
    }


echo json_encode($data);
http_response_code($code);
?>
```

disablePoll.php

```php
<?php

require_once "connection.php";

    $code=200;
    $data="";

    $upit="UPDATE poll set active=0";

    try{
```

```
        $connection->query($upit);
        $code=204;
    }
    catch(PDOException $e){
        $code=500;
        $data="Server error";
    }


echo json_encode($data);
http_response_code($code);
?>
```

editPerfromance.php

```
<?php
    require_once "connection.php";

    $code=200;
    $data="Inserted";


    $name=$_POST['name'];
    $duration=$_POST['duration'];
    $premier=$_POST['premier'];
    $price=$_POST['price'];
    $description=$_POST['description'];
    $id=$_POST['id'];

    $err=0;
    if(strlen($name)<1){
        $data="Name has 1 letter minimum";
        $err++;
    }
    if(strlen($description)<15){
        $data="Description has 15 letter minimum";
        $err++;
    }
    if($duration>180 || $duration==""){
        $data="Duration is number (minutes), values 180 maximum";
        $err++;
    }
    if($premier==""){
        $data="You have to choose premier date";
        $err++;
```

```
    }
    if($price==""){
        $data="You have to choose price";
        $err++;
    }

    if($err==0){
        $code=201;

        $ifNameExists="SELECT * FROM performances WHERE name like :name AND
id!=:id";

        $prepareTest=$connection->prepare($ifNameExists);

        $namePrepare="%".$name."%";
        $prepareTest->bindParam(":name",$namePrepare);
        $prepareTest->bindParam(":id",$id);


        try{
            $prepareTest->execute();

            if($prepareTest->rowCount()==1){
                $code=200;
                $data="Name already exists";
            }
            else{

                if(isset($_FILES['file'])){
                    $slika=$_FILES['file'];

                    $tmpName=$slika['tmp_name'];
                    $size=$slika['size'];
                    $tip=$slika['type'];
                    $nameImage=$slika['name'];
                    $naziv=time().$nameImage;
                    $putanja="../assets/images/$naziv";
                    move_uploaded_file($tmpName,$putanja);

                    ///upis sa slikom
                    $query="UPDATE performances set
name=:name,description=:description,poster=:poster,premier=:premier,duration=:dur
ation where id=:id";
                    $prepare=$connection->prepare($query);
                    $prepare->bindParam(":poster",$naziv);
```

```
                }
                else{
                    ///upit bez slike
                    $query="UPDATE performances set
name=:name,description=:description,premier=:premier,duration=:duration where
id=:id";
                    $prepare=$connection->prepare($query);
                }



                $prepare->bindParam(":name",$name);
                $prepare->bindParam(":description",$description);
                $prepare->bindParam(":premier",$premier);
                $prepare->bindParam(":duration",$duration);
                $prepare->bindParam(":id",$id);

                $prepare->execute();



                $priceQuery="UPDATE prices SET price=:price,dateFrom=:dateFrom
WHERE idPerformance=:id";

                $prepare2=$connection->prepare($priceQuery);

                $date=date("Y-m-d");

                $prepare2->bindParam(":price",$price);
                $prepare2->bindParam(":dateFrom",$date);
                $prepare2->bindParam(":id",$id);
                $prepare2->execute();
                $data="Updated";
            }

        }
        catch(PDOException $e){
            $data=$e->getMessage();
            $code=500;
        }
    }
    echo json_encode($data);
    http_response_code($code);
?>
```

enablePoll.php

```php
<?php

require_once "connection.php";

    $code=200;
    $data="";

    $upit="UPDATE poll set active=1";

    try{
        $connection->query($upit);
        $code=204;
    }
    catch(PDOException $e){
        $code=500;
        $data="Server error";
    }


echo json_encode($data);
http_response_code($code);
?>
```

getEditPerformance.php

```php
<?php
    require_once "connection.php";


    if(isset($_POST['id'])){
        $id=$_POST['id'];

        $query="SELECT *,p.id FROM performances p INNER JOIN prices pr
                ON p.id=pr.idPerformance
               WHERE p.id=:id";

        $prepare=$connection->prepare($query);
        $prepare->bindParam(":id",$id);
        try{
            $prepare->execute();
            if($prepare->rowCount()==1){
                $data=$prepare->fetch();
                $code=200;
            }
```

```
            else{
                $data="No access";
                $code=404;
            }
        }
        catch(PDOException $e){
            $data="Server error";
            // $data=$e->getMessage();
            $code=500;
        }




    }
    else{
        $code=404;
        $data="No access";
    }
    echo json_encode($data);
    http_response_code($code);
?>
```

getMessagesAdmin.php

```
<?php
    require_once "connection.php";

    $code=200;

    $query="select * from messages";

    try{
        $data=$connection->query($query)->fetchAll();



    }
    catch(PDOException $e){
        $data="Server error";
        $code=500;
        // $data=$e->getMessage();
    }

echo json_encode($data);
http_response_code($code);
```

```
?>
```

getNav.php

```
<?php
    require_once "connection.php";

    $query="SELECT * FROM navigations WHERE preview=1";

    if(isset($_SESSION['user'])){
        $query.=" OR preview=3";

        if($_SESSION['user']['idRole']==1){
        $query.=" OR preview=4";
        }
    }
    else{
        $query.=" OR preview=2";
    }

    $query.=" ORDER BY priority";

    try{
        $nav=$connection->query($query)->fetchAll();
    }
    catch(PDOException $e){
        $nav=$e->getMessage();
    }
?>
```

getTickets.php

```
<?php
session_start();
    require_once "connection.php";

    if(isset($_POST['ticketsBtn'])){

        $query="SELECT *,t.id FROM tickets t INNER JOIN repertoire r
                ON r.id=t.idRepertoire INNER JOIN prices p
                ON r.idPrice=p.id INNER JOIN performances pe
                ON pe.id=p.idPerformance
                WHERE t.idUser=:idUser
                AND r.datePerforming>=CURRENT_DATE
```

```
                ORDER BY datePerforming";

        $idUser=$_SESSION['user']['id'];


        $prepare=$connection->prepare($query);
        $prepare->bindParam(":idUser",$idUser);


        try{
            $prepare->execute();

            $data=$prepare->fetchAll();
            $code=200;


        }
        catch(Exception $e){
            $code=500;
            // $data="Server error";
            $data=$e->getMessage();
        }
    }
    else{
        $data="No access";
        $code=404;
    }



echo json_encode($data);
http_response_code($code);
?>
```

getUsersAdmin.php

```
<?php
    require_once "connection.php";

    $code=200;

    $query="select * from users";

    try{
        $data=$connection->query($query)->fetchAll();
```

```
    }
    catch(PDOException $e){
        $data="Server error";
        $code=500;
        // $data=$e->getMessage();
    }

echo json_encode($data);
http_response_code($code);

?>
```

login.php

```
<?php
    session_start();

    require_once "connection.php";

    $data="";
    $code=200;
    if(isset($_POST['btnLogin'])){

        $mail=$_POST['email'];
        $pass=$_POST['password'];

        $regEmail="/^\w[.\d\w]*\@[a-z]{2,10}(\.[a-z]{2,3})+$/";
        $regPass="/^.{8,50}$/";

        $greske=[];

        if(!preg_match($regEmail,$mail)){
            array_push($greske,"E-mail formats: petar.petrovic.123.21@ict.edu.rs
or petar@gmail.com. <br>");
        }
        if(!preg_match($regPass,$pass) && strlen($pass)<8){
            array_push($greske,"Password must be at least 8 characters long.
<br>");
        }

        if(count($greske)==0){
            $aktivan=1;
```

```
            $daLiPostojiKorisnik="SELECT * FROM users WHERE email=:email AND
active=:aktivan";
            $priprema=$connection->prepare($daLiPostojiKorisnik);
            $priprema->bindParam(":email",$mail);
            $priprema->bindParam(":aktivan",$aktivan);
            try{
                $priprema->execute();
                if($priprema->rowCount()==1){
                    $upit="SELECT *,u.id FROM users u INNER JOIN roles r ON
u.idrole=r.id WHERE email=:email AND password=:pass";
                    $pass=md5($pass);
                    $priprema2=$connection->prepare($upit);
                    $priprema2->bindParam(":email",$mail);
                    $priprema2->bindParam(":pass",$pass);
                    try{
                        $priprema2->execute();
                        $code=200;
                        if($priprema2->rowCount()==1){
                            $korisnik=$priprema2->fetch();
                            $_SESSION['user']=$korisnik;
                            $code=200;
                            $data="ok";
                        }
                        else{
                            $data="The password is incorrect";
                        }
                    }
                    catch(PDOException $e){
                        $code=500;
                        $data=$e->getMessage();
                    }
                }
                else{
                    $code=200;
                    $data="No user with that email address was found or the
account was not activated";
                }
            }
            catch(PDOException $e){
                $code=500;
                $data="Server error";
            }
        }
        else{
            $data=$greske;
```

```
            $code=422;
        }

    }
    else{
        $data="Sorry bit the page you are looking for does not exist, have been
removed or you have to log in.";
        $code=404;
    }



echo json_encode($data);
http_response_code($code);

?>
```

message.php

```
<?php

    require_once "connection.php";

if(isset($_POST["btnMessage"])){
    $code=200;
    $email=$_POST["email"];
    $subject=$_POST["subject"];
    $message=$_POST["message"];
    $regEmail="/^\w[.\d\w]*\@[a-z]{2,10}(\.[a-z]{2,3})+$/";
    $regSubject="/^[\w\s]+$/";
    $error=0;
    if(!preg_match($regEmail,$email)){
        $error++;
        $data="E-mail format: pera.peric.55.19@ict.edu.rs or pera@gmail.com";
    }
    if(!preg_match($regSubject,$subject)){
        $error++;
        $data="You have to write title";
    }
    if(count($message)<10){
        $error++;
        $data="Message have 10 words minimum";
    }
    if($error==0){
        $message=implode($message," ");
```

```
        $datum=date("Y-m-d H:i:s");
        $insert="INSERT INTO messages
VALUES(null,:email,:subject,:message,:datum)";
        $prepare=$connection->prepare($insert);
        $prepare->bindParam(":email",$email);
        $prepare->bindParam(":subject",$subject);
        $prepare->bindParam(":message",$message);
        $prepare->bindParam(":datum",$datum);
        try{
            $prepare->execute();
            $date="Message successfuly sent";
            $code=201;
        }
        catch(PDOException $e){
            $date="Server error";
            $code=500;
        }
    }
    else{
        $code=422;
        $date=$data;
    }
}
else{
    $code=404;
    $date="Error";
}

echo json_encode($date);
http_response_code($code);
?>
```

next.php

```
<?php
    require_once "connection.php";
    $next=null;

    $query="SELECT *,re.id FROM performances p INNER JOIN prices pr
            ON p.id=pr.idPerformance INNER JOIN repertoire re
            ON re.idPrice=pr.id
            WHERE datePerforming>=CURRENT_DATE ORDER BY datePerforming LIMIT 1";


    try{
```

```
        $data=$connection->query($query);
        if($data->rowCount()==0){
            $data="There is nothing in the repertoire";
        }
        else{
            $next=$data->fetch();
        }
        $code=200;
    }
    catch(PDOException $e){
        $data="Server error";
        $code=500;
    }
?>
```

parameters.php

```
<?php

    $baza="pozoriste";
    $user="root";
    $pass="";
    $host="localhost";

?>
```

performance.php

```
<?php
    require_once "connection.php";


    if(isset($_GET['id'])){
        $id=$_GET['id'];

        $query="SELECT * FROM performances WHERE id=:id";

        $prepare=$connection->prepare($query);
        $prepare->bindParam(":id",$id);
        try{
            $prepare->execute();
            if($prepare->rowCount()==1){
                $data=$prepare->fetch();
                $code=200;
            }
```

```
            else{
                $data="No access";
                $code=404;
            }
        }
        catch(PDOException $e){
            $date="Server error";
            $code=500;
        }




    }
    else{
        $code=404;
        $data="No access";
    }
?>
```

performances.php

```
<?php
    require_once "connection.php";


    if(isset($_GET['btn'])){
        $code=200;

        $limit=4;
        $offset=0;
        $page=0;
        $search="";

        if(isset($_GET['page'])){
            $page=$_GET['page']-1;
        }
        if(isset($_GET['search'])){
            $search=$_GET['search'];
        }

        $sum=0;
        $offset=$page*$limit;
```

```
        $query="SELECT *,p.id FROM performances p INNER JOIN prices pr
                ON p.id=pr.idPerformance  WHERE name like '%$search%'";


        try{
            $performances=$connection->query($query);
            if($performances->rowCount()!=0){
                $sum=$performances->rowCount();
                $data['sumPages']=ceil($sum/$limit);

                $query.=" LIMIT $limit OFFSET $offset";
                $data['performances']=$connection->query($query)->fetchAll();

                $data['currentPage']=$page+1;
            }
            else{
                $data['performances']=[];
            }
        }
        catch(Exception $e){
            $code=500;
            $data="Server error";
        }
    }
    else{
        $code=404;
        $data="Not found";
    }


echo json_encode($data);
http_response_code($code);
?>
```

poll.php

```
<?php
require_once "connection.php";
$upit = "SELECT * FROM poll WHERE active=1";
try
{
    $anketa = $connection->query($upit);
    if ($anketa->rowCount() == 1)
    {
```

```
        $aktivna = $anketa->fetch();
        $idAnkete = $aktivna['id'];
        $upit2 = "SELECT * FROM answers WHERE idPoll=$idAnkete";
        $odg = $connection->query($upit2);
        if ($odg->rowCount() != 0)
        {
            $odgovori = $odg->fetchAll();
            if (isset($_SESSION['user']))
            {
                $user = $_SESSION['user'];
                $id = $user['id'];
                $upit3 = "SELECT * FROM useranswers ua INNER JOIN answers a on
ua.idAnswer=a.id WHERE idPoll=$idAnkete AND idUser=$id";
                $odgovorKorisnika = $connection->query($upit3);
                if ($odgovorKorisnika->rowCount() == 1)
                {
                    $test = $odgovorKorisnika->fetch();
                    $obavestenje = "You have already voted!";
                }
            }
            else
            {
                $obavestenje = "Login to vote!";
            }
        }
    }
    else
    {
        $message = "There is no active poll currently";
    }
}
catch(PDOException $e)
{
    $message = "Server error";
}
?>
```

pollRez.php

```
<?php
    require_once "connection.php";

    $upit="SELECT * FROM poll";
```

```
    $upit2="SELECT a.answer,COUNT(ua.idAnswer) AS answers FROM answers a LEFT
JOIN  useranswers ua ON a.id=ua.idAnswer GROUP BY a.id";
    $code=200;

    try{
        $question=$connection->query($upit)->fetch();
        $data['question']=$question['question'];
        $data['active']=$question['active'];

        $data['voting']=$connection->query($upit2)->fetchAll();
    }
    catch(PDOException $e){
        $code=500;
        $data=$e->getMessage();
    }




echo json_encode($data);
http_response_code($code);
?>
```

register.php

```
<?php

    require_once "connection.php";

    $data="";

    if(isset($_POST['btnRegister'])){


        $firstName=$_POST['firstName'];
        $lastName=$_POST['lastName'];
        $email=$_POST['email'];
        $password=$_POST['password'];
        $passwordConf=$_POST['passConfirm'];

        $regName="/^[A-ZZŠĐŽČĆ][a-zzšđžčć]{2,29}$/";
        $regEmail="/^\w[.\d\w]*\@[a-z]{2,10}(\.[a-z]{2,3})+$/";
        $regPass="/^.{8,50}$/";
        $greske=[];
        if(!preg_match($regName,$firstName)){
```

```
            array_push($greske,"First name has capital letter,3 letter minimum,
30maximum<br>");
        }
        if(!preg_match($regName,$lastName)){
            array_push($greske,"Last name has capital letter,3 letter minimum,
30maximum<br>");
        }
        if(!preg_match($regEmail,$email)){
            array_push($greske,"E-mail formats: petar.petrovic.123.21@ict.edu.rs
or petar@gmail.com<br>");
        }
        if(!preg_match($regPass,$password)){
            array_push($greske,"Password must be at least 8 characters
long<br>");
        }
        if($passwordConf!=$password){
           array_push($greske,"Passwords do not match<br>");
        }

        $mailProvera="SELECT email FROM users WHERE email=:mail";
        $priprema=$connection->prepare($mailProvera);
        $priprema->bindParam(":mail",$email);
        try{
            $priprema->execute();
            $rez=$priprema->fetch();
            if($priprema->rowCount()==1){
                $data="There is already a user with that email address";
                $code=200;
            }
            else{
                if(count($greske)==0){

                    $insert="INSERT INTO users
VALUES(NULL,:ime,:prezime,:mail,:pass,:aktivan,:aktivacionikod,:regDate,:iduloge)
";
                    $password=md5($password);
                    $uloga=2;
                    $aktivan=1;
                    $kod=md5(time().md5($email));

                    //mail($mail,"Account
activation","http://127.0.0.1/pozoriste/activation.php?code=".$kod);

                    $date=date("Y-m-d H:i:s");
```

```
                    $priprema2=$connection->prepare($insert);
                    $priprema2->bindParam(":ime",$firstName);
                    $priprema2->bindParam(":prezime",$lastName);
                    $priprema2->bindParam(":mail",$email);
                    $priprema2->bindParam(":pass",$password);
                    $priprema2->bindParam(":aktivan",$aktivan);
                    $priprema2->bindParam(":aktivacionikod",$kod);
                    $priprema2->bindParam(":regDate",$date);
                    $priprema2->bindParam(":iduloge",$uloga);
                    try{
                        $uspesno=$priprema2->execute();
                        $code=201;
                        $data="You have successfully registered, check your email
to activate your account";
                    }
                    catch(PDOException $e){
                        $code=500;
                        $data="Server error";
                    }
                }
                else{
                    $data=$greske;
                    $code=422;
                }
            }
        }
        catch(PDOException $e){
            $data= "Server error";
            $code=500;
        }

    }
    else{
        $code=404;
        $data="Error";
    }

echo json_encode($data);
http_response_code($code);
?>
```

repertoire.php

```
<?php
    require_once "connection.php";
```

```
    $query="SELECT *,r.id FROM repertoire r INNER JOIN prices p
    ON r.idPrice=p.id INNER JOIN performances pe
    ON p.idPerformance=pe.id
    WHERE MONTH(datePerforming)=MONTH(CURRENT_DATE) AND
datePerforming>=CURRENT_DATE
    ORDER BY datePerforming";


    $code=200;
    try{
        $repertoire=$connection->query($query)->fetchAll();
    }
    catch(Exception $e){
        $code=500;
        // $message="Server error";
        $message=$e->getMessage();
    }
?>
```

repertoireAdmin.php

```
<?php
    require_once "connection.php";

    if(isset($_GET['btn'])){
        $query="SELECT *,r.id FROM repertoire r INNER JOIN prices p
        ON r.idPrice=p.id INNER JOIN performances pe
        ON p.idPerformance=pe.id
        WHERE datePerforming>=CURRENT_DATE
        ORDER BY datePerforming";


        $code=200;
        try{
            $data=$connection->query($query)->fetchAll();
        }
        catch(Exception $e){
            $code=500;
            $data="Server error";
            // $data=$e->getMessage();
        }
    }
    else{
        $code=404;
```

```
        $data="Not found";
    }


echo json_encode($data);
http_response_code($code);

?>
```

seats.php

```
<?php
    require_once "connection.php";

    if(isset($_GET['id'])){
        $id=$_GET['id'];
        $query="SELECT * FROM tickets WHERE idRepertoire=:id";
        $query2="SELECT * FROM performances p INNER JOIN prices pr
                ON p.id=pr.idPerformance INNER JOIN repertoire re
                ON re.idPrice=pr.id WHERE re.id=:id
                AND datePerforming>=CURRENT_DATE";

        $prepare=$connection->prepare($query);
        $prepare2=$connection->prepare($query2);

        $prepare->bindParam(":id",$id);
        $prepare2->bindParam(":id",$id);


        $code=200;

        try{
            // $performances=$connection->query($query)->fetchAll();
            $prepare->execute();
            $prepare2->execute();
            if($prepare2->rowCount()==0){
                $data="No access";
                $code=404;
            }
            else{
                $code=200;

                $data=$prepare->fetchAll();
                $performance=$prepare2->fetch();
            }
```

```
        }
        catch(Exception $e){
            $code=500;
            $message="Server error";
        }
    }
    else{
        $code=404;
        $message="Page not found";
    }
?>
```

vote.php

```
<?php
session_start();
require_once "connection.php";
    $code = 200;

    $id = $_POST['id'];
    $user = $_SESSION['user'];
    $idUser = $user['id'];

    $upit = "INSERT INTO useranswers VALUES (null,:id,:idUser)";
    $priprema = $connection->prepare($upit);
    $priprema->bindParam(":id", $id);
    $priprema->bindParam(":idUser", $idUser);
    try
    {
        $priprema->execute();
        $code = 201;
        $data = "Successfully voted!";
    }
    catch(PDOException $e)
    {
        $code = 500;
        $data = $e->getMessage();
    }
http_response_code($code);
echo json_encode($data);
?>
```

Main.js

```
var url=location.href;


if(url.indexOf("seats.php")!=-1){
    $("#buy").click(buyTickets)
    $("#checkSeats").click(checkSeats)
    $(".places").click(markPlace)
}

if(url.indexOf("performances.php")!=-1){
    getPerformances()
    $("#search").keyup(getPerformances)
}
if(url.indexOf("login.php")!=-1){
    $("#logBtn").click(showRegDiv)
    $("#regBtn").click(showLogDiv)
    $(".divLogReg p").hide()
}

if(url.indexOf("tickets.php")!=-1){
    getTickets()
}
if(url.indexOf("contact.php")!=-1){
    $(".messageDiv p").hide()
    $("#vote").click(vote)
}

if(url.indexOf("admin/index.php")!=-1){
    $("#performances").click(adminPerformances)
    $("#addPerformance").click(addPerformance)
    $("#repertoire").click(adminRepertoire)
    $("#addRepertoire").click(addRepertoire)
    $("#users").click(getUsers)
    $("#messages").click(getMessages)
    $("#poll").click(getPoll)
}

function getPoll(e){
    e.preventDefault()
    getPollRes()
}

function adminPerformances(e){
    e.preventDefault()
```

```
    getPerformancesAdmin()
}

function adminRepertoire(e){
    e.preventDefault()
    getRepertoire()
}


function showRegDiv(e){
    e.preventDefault()
    $("#logDiv").removeClass("d-none")
    $("#regDiv").addClass("d-none")
}

function showLogDiv(e){
    e.preventDefault()
    $("#logDiv").addClass("d-none")
    $("#regDiv").removeClass("d-none")
}
let errors=[]
function validate(input,regex){
    let br=0
    if(!regex.test(input.val())){
        input.addClass("border-danger")
        input.removeClass("border-success")
        input.parent().find("p").show()
        br++
    }
    else{
        input.removeClass("border-danger")
        input.addClass("border-success")
        input.parent().find("p").hide()
    }
    return br
}
function login(){
    var email=$("#email")
    var password=$("#password")

    var regEmail=/^\w[.\d\w]*\@[a-z]{2,10}(\.[a-z]{2,3})+$/;
    var regPass=/^.{8,50}$/;

    var error=0
```

```
    error=validate(email,regEmail)
    error=validate(password,regPass)




    if(error==0){
        $.ajax({
            url:'models/login.php',
            method:"POST",
            dataType:"json",
            data:{
                "email":email.val(),
                "password":password.val(),
                "btnLogin":true
            },
            success:function(data){
                if(data=="ok"){
                    window.location.href="index.php"
                }
                else{
                    $("#messageLogin").html(data)
                }
            },
            error:function(xhr){
                var error=JSON.parse(xhr.responseText)
                let print=''
                for(let e of error){
                    print+=`${e}`
                }
                $("#messageLogin").html(print)
            }
        })
    }

    return false;
}


function getPerformances(){
    var search=$("#search").val()
    $.ajax({
        url:'models/performances.php',
        method:"get",
        dataType:"json",
```

```
        data:{
            "btn":true,
            "search":search
        },
        success:function(data){
            if(data['performances'].length!=0){
                printPerformances(data['performances'])
                printPagination(data)
            }
            else{
                $("#performances").html("<h2 class='text-primary'>There is no
matching data</h2>")
            }

        },
        error:function(xhr){

        }
    })
}

function printPerformances(data){
    let print=''
    for(let d of data){
        print+=`<div class="col-3 performance text-center">
        <a href="performance.php?id=${d.id}">
            <img class="img-fluid" src="assets/images/${d.poster}"
alt="${d.name}"/>
            <h2 class="green fs-4">${d.name}</h2>
        </a>
    </div>`
    }

    $("#performances").html(print)
}

function printPagination(data){
    let print='';
    if(data.currentPage!=1){
        print+=`<a href="#" class="page" data-id="${data.currentPage-
1}">Previous</a>`
    }
    if(data.sumPages!=1){
        for(let i=1;i<=data.sumPages;i++){
            if(i<=data.sumPages){
```

```
                print+=`<a href="#" class="page links`
                if(i==data.currentPage){
                    print+=` active`
                }
                print+=`" data-id="${i}">${i}</a>`
            }
        }
    }

    if(data.currentPage!=data.sumPages){
        print+=`<a href="#" class="page" data-
id="${data.currentPage+1}">Next</a>`
    }

    $("#pagination").html(print)
    $(".page").click(changePage)
}

function changePage(e){
    e.preventDefault()
    var page=this.dataset.id
    var search=$("#search").val()
    $.ajax({
        url:"models/performances.php",
        method:"get",
        dataType:"json",
        data:{
            "page": page,
            "search": search,
            "btn":true
        },
        success:function(data){
            if(data.performances.length!=0){
                printPerformances(data.performances)
                printPagination(data)
            }
            else{
                $("#performances").html("<h2 class='text-primary'>There is no
matching data</h2>")
            }
        },
        error:function(xhr){
            //err
        }
    })
```

```
}

function markPlace(e){
    e.preventDefault()
    if($(this).hasClass("addToCart")){
        $(this).removeClass("addToCart")
    }
    else{
        $(this).addClass("addToCart")
    }
}
function checkSeats(){
    var seats=$(".addToCart")
    $("#buyMessage").html("")
    if(seats.length!=0){
        $("#buy").removeAttr("disabled")
        let print=''
        for(let s of seats){
            if(s.dataset.position==1){
                var position="Parterre left"
            }
            else if(s.dataset.position==2){
                var position="Parterre right"
            }
            else{
                var position="Middle"
            }
            print+=`<p>Row: <b>${s.dataset.row}</b></p>
                <p>Place: <b>${s.dataset.place}</b></p>
                <p>Position: ${position}</p>
                <hr>`
        }
        $("#seatsMessage").html(print)
    }
    else{
        $("#buy").attr("disabled","disabled")
        $("#seatsMessage").html("<h3>You have to choose seats</h3>")
    }
}
function buyTickets(){
    var seats=$(".addToCart")
    if(seats.length!=0){
        var repertoire=$("#repertoire").val()
        var seatsForSend=[]
        for(let s of seats){
```

```
            var obj={
                "idRepertoire":repertoire,
                "row":s.dataset.row,
                "place":s.dataset.place,
                "position":s.dataset.position
            }
            seatsForSend.push(obj)
        }


        $.ajax({
            url:'models/buyTickets.php',
            method:"POST",
            dataType:"json",
            data:{
                "tickets":seatsForSend,
                "btnBuy":true
            },
            success:function(data){
                console.log(seats)
                $("#buyMessage").html(data)
                $("#buy").attr("disabled","disabled")
                for(let s of seats){
                    s.classList.add("purchased")
                    s.classList.remove("addToCart")
                }

            },
            error:function(xhr){
                alert(JSON.parse(xhr.responseText))
            }
        })
    }
    else{
        alert("Choose seats")
    }
}


function register(){
    var email=$("#emailReg")
    var password=$("#password")
    var firstName=$("#firstName")
    var lastName=$("#lastName")
    var password=$("#pass")
```

```
    var passConf=$("#passConfirm")

    var regEmail=/^\w[.\d\w]*\@[a-z]{2,10}(\.[a-z]{2,3})+$/;
    var regPass=/^.{8,50}$/;
    var regName=/^[A-ZZŠĐŽČĆ][a-zzšđžčć]{2,29}$/;
    var regPass=/^.{8,50}$/;
    var regPass=/^.{8,50}$/;

    var error=0

    error=validate(email,regEmail)
    error=validate(firstName,regName)
    error=validate(lastName,regName)
    error=validate(password,regPass)

    if(password.val()!=passConf.val() || password.val()==""){
        passConf.addClass("border-danger")
        passConf.removeClass("border-success")
        passConf.parent().find("p").show()
        error++
    }
    else{
        passConf.removeClass("border-danger")
        passConf.addClass("border-success")
        passConf.parent().find("p").hide()
    }

    if(error==0){
        $.ajax({
            url:'models/register.php',
            method:"POST",
            dataType:"json",
            data:{
                "firstName":firstName.val(),
                "lastName":lastName.val(),
                "email":email.val(),
                "password":password.val(),
                "passConfirm":passConf.val(),
                "btnRegister":true
            },
            success:function(data){
                    $("#messageRegister").html(data)
            },
            error:function(xhr){
                var error=JSON.parse(xhr.responseText)
```

```
                let print=''
                  for(let e of error){
                      print+=`${e}`
                  }
                $("#messageRegister").html(print)
            }
        })
    }



    return false;
}

function sendMessage(){

    var email=$("#email")
    var subject=$("#subject")
    var message=$("#message")


    var regEmail=/^\w[.\d\w]*\@[a-z]{2,10}(\.[a-z]{2,3})+$/;
    var regSubject=/^[\w\s]+$/;
    var error=0;
    message=message.val().split(" ")

    error+=validate(email,regEmail)
    error+=validate(subject,regSubject)

    if(message.length<10){
        $("#message").addClass("border-danger")
        $("#message").removeClass("border-success")
        $("#message").parent().find("p").show()
        error++;
    }
    else{
        $("#message").removeClass("border-danger")
        $("#message").addClass("border-success")
        $("#message").parent().find("p").hide()
    }
    if(error==0){
        $.ajax({
            url:'models/message.php',
            method:"POST",
            dataType:"json",
```

```
            data:{
                "email":email.val(),
                "message":message,
                "subject":subject.val(),
                "btnMessage":true
            },
            success:function(data){
                $("#alertMessage").html(data)
                $("#alertMessage").show()

            },
            error:function(xhr){
                $("#alertMessage").html(JSON.parse(xhr.responseText))
                $("#alertMessage").show()
            }
        })
    }

    return false;
}
function getTickets(){
    $.ajax({
        url:'models/getTickets.php',
        method:"POST",
        dataType:"json",
        data:{
            "ticketsBtn":true
        },
        success:function(data){
            printTickets(data)

        },
        error:function(xhr){
            $("#table").html(JSON.parse(xhr.responseText))
        }
    })
}

function printTickets(data){
    let print=''
    let br=1
    if(data.length!=0){
        print+=`
        <table class="table">
            <tr>
```

```
                <th>No</th>
                <th>Performance</th>
                <th>Date</th>
                <th>Info</th>
                <th>Price</th>
                <th class="text-center">Cancel</th>
            </tr>`

        for(let d of data){
            if(d.position==1){
                d.position="Parterre Left"
            }
            else if(d.position==2){
                d.position="Parterre Rigth"
            }
            else{
                d.position="Middle"
            }
            print+=`
                <tr>
                    <td>${br}</td>
                    <td>${d.name}</td>
                    <td>${d.datePerforming.split("-
")[2]}.${d.datePerforming.split("-")[1]}</td>
                    <td>Row:${d.row} <br> Seat: ${d.place} <br>
${d.position}</td>
                    <td>${d.price}</td>
                    <td class="text-center"><a class="cancel" href="#" data-
id="${d.id}"><i class="fa-solid fa-xmark text-danger fs-4"></i></a></td>
                </tr>
            `
        }

        print+=`</table>`
    }
    else{
        print=`<h2 class="text-center">There are no tickets yet!</h2>`
        $("#table").html(print)
    }


    $("#table").html(print)
    $(".cancel").click(cancelTicket)
}
```

```
function cancelTicket(e){
    e.preventDefault()
    var id=this.dataset.id
    var row=$(this).parent().parent()
    $.ajax({
        url:'models/cancelTicket.php',
        method:"POST",
        dataType:"json",
        data:{
            "id":id,
            "cancelBtn":true
        },
        success:function(data){
            row.remove()
        },
        error:function(xhr){
            $("#table").html(JSON.parse(xhr.responseText))
        }
    })
}

function vote(){
    var answer=$(".odg:checked").val()
    if(answer){
        $.ajax({
            url:"models/vote.php",
            method:"post",
            dataType:"json",
            data:{
                "id":answer
            },
            success:function(data){
                $("#obavestenje").html(data)
                $("#vote").attr("disabled","disabled")
            },
            error:function(xhr){
                $("#obavestenje").html(JSON.parse(xhr.responseText))
            }
        })
    }
    else{
        $("#obavestenje").html("You have to choose answer!")
    }

}
```

```
// ADMIN


function getUsers(e){
    e.preventDefault()
    $.ajax({
        url:'../models/getUsersAdmin.php',
        method:"GET",
        dataType:"json",
        success:function(data){
            printUsersAdmin(data)
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })
}

function printUsersAdmin(data){
    var print=''
    print+=`<h1>Users</h1>`
    print+=`<table class="table">
            <tr>
                <th>No</th>
                <th>E-mail</th>
                <th>First Name</th>
                <th>Last Name</th>
                <th>Active</th>
                <th>Registration Date</th>
            </tr>`
    let br=1;
    for(let d of data){
        print+=`<tr>
                    <td>${br++}</td>
                    <td>${d.email}</td>
                    <td>${d.firstname}</td>
                    <td>${d.lastname}</td>
                    <td>`
                    if(d.active==1){
                        print+=`Yes`
                    }
                    else{
                        print+=`No`
```

```
                    }
                print+=`</td>
                <td>${d.registrationDate}</td>
                </tr>`
    }
        print+=`</table>`

    $("#content").html(print)

}

function getMessages(e){
    e.preventDefault()
    $.ajax({
        url:'../models/getMessagesAdmin.php',
        method:"GET",
        dataType:"json",
        success:function(data){
            printMessagesAdmin(data)
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })
}

function printMessagesAdmin(data){
    var print=''
    print+=`<h1>Messages</h1>`
    print+=`<table class="table">
            <tr>
                <th>E-mail</th>
                <th>Title</th>
                <th>Message</th>
                <th>Date</th>
                <th class="text-center">Delete</th>
            </tr>`
    for(let d of data){
        print+=`<tr>
                    <td>${d.email}</td>
                    <td>${d.subject}</td>
                    <td>${d.message}</td>
                    <td>${d.sendingDate}</td>
                    <td class="text-center"><a href="#" class="delete" data-
id="${d.id}"><i class="fa-regular fa-xmark"></i></a></td>
```

```
                </tr>`
    }
        print+=`</table>`

    $("#content").html(print)

    $(".delete").click(deleteMessage)
}
function deleteMessage(e){
    e.preventDefault()
    var id=this.dataset.id
    var row=$(this).parent().parent()
    $.ajax({
        url:'../models/deleteMessage.php',
        method:"POST",
        dataType:"json",
        data:{
            "id":id
        },
        success:function(data){
            row.remove()
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })
}

function getPerformancesAdmin(){
    $.ajax({
        url:'../models/performances.php',
        method:"GET",
        dataType:"json",
        data:{
            "btn":true
        },
        success:function(data){
            printPerformancesAdmin(data)
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })
}
```

```
function printPerformancesAdmin(data){
    let print=`<table class="table">`

    print+=`<tr>
                <th>No</th>
                <th>Name</th>
                <th>Poster</th>
                <th>Premier</th>
                <th>Price</th>
                <th class="text-center">Update</th>
                <th class="text-center">Delete</th>
            </tr>`
    let br=(data.currentPage-1)*4+1
    for(let d of data.performances){
        print+=`<tr>
                    <td>${br++}</td>
                    <td class="col-3">${d.name}</td>
                    <td class="col-2">
                        <img class="img-fluid" src="../assets/images/${d.poster}"
alt="${d.name}"/>
                    </td>
                    <td>${d.premier}</td>
                    <td>${d.price}</td>
                    <td class="text-center"><a href="#" class="update" data-
id="${d.id}"><i class="fa-regular fa-pen-to-square"></i></a></td>
                    <td class="text-center"><a href="#" class="delete" data-
id="${d.id}"><i class="fa-regular fa-xmark"></i></a></td>
                </tr>`
    }

    print+='</table>'

    print+=`<div id="pagination" class="text-center">`

    if(data.currentPage!=1){
        print+=`<a href="#" class="page" data-id="${data.currentPage-
1}">Previous</a>`
    }
    if(data.sumPages!=1){
        for(let i=1;i<=data.sumPages;i++){
            if(i<=data.sumPages){
                print+=`<a href="#" class="page links`
                if(i==data.currentPage){
                    print+=` active`
                }
```

```
                print+=`" data-id="${i}">${i}</a>`
            }
        }
    }

    if(data.currentPage!=data.sumPages){
        print+=`<a href="#" class="page" data-
id="${data.currentPage+1}">Next</a>`
    }

    print+=`</div>`



    $("#content").html(print)

    $(".delete").click(deletePerformance)
    $(".update").click(updatePerformance)

    $(".page").click(changePageAdmin)
}
function updatePerformance(e){
    e.preventDefault()
    var id=this.dataset.id
    $.ajax({
        url:'../models/getEditPerformance.php',
        method:"POST",
        dataType:"json",
        data:{
            "id":id
        },
        success:function(data){
            printFormEditPerformance(data)
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })
}
function printFormEditPerformance(data){
    let print=`<h2>Edit Performance</h2>
                <form action="#" enctype="multipart/form-data">
                    <div>
                        <label for="name">Name</label>
```

```
                        <input type="text" id="name" value="${data['name']}"
class="form-control">
                    </div>
                    <div class="mt-2">
                        <label for="poster">Poster</label>
                        <input type="file" id="file">
                    </div>
                    <div class="mt-2">
                        <label for="premier">Premier Date</label>
                        <input type="date" id="premier"
value="${data['premier']}" class="form-control">
                    </div>
                    <div class="mt-2">
                        <label for="duration">Duration</label>
                        <input type="number" id="duration"
value="${data['duration']}" class="form-control">
                    </div>
                    <div class="mt-2">
                        <label for="description">Description</label>
                        <textarea class="form-control"
id="description">${data['description']}</textarea>
                    </div>
                    <div class="mt-2">
                        <label for="price">Price</label>
                        <input type="number" class="form-control" id="price"
value="${data['price']}">
                    </div>
                    <div class="mt-2">
                        <input type="submit" id="update" value="Update
performance" class="form-control">
                    </div>
                    <div class="mt-2">
                        <p id="messageInsert"></p>
                    </div>
                    <input type="hidden" value="${data['id']}" id="id">
                </form>`


    $("#content").html(print)


    $("#update").click(doUpdatePerformance)

}
```

```
function doUpdatePerformance(){
    var name=$("#name").val()
    var file=$("#file")[0].files[0]
    var premier=$("#premier").val()
    var duration=$("#duration").val()
    var description=$("#description").val()
    var price=$("#price").val()
    var id=$("#id").val()


    var podaciZaSlanje = new FormData()

    podaciZaSlanje.append("name",name)
    podaciZaSlanje.append("file",file)
    podaciZaSlanje.append("premier",premier)
    podaciZaSlanje.append("duration",duration)
    podaciZaSlanje.append("description",description)
    podaciZaSlanje.append("price",price)
    podaciZaSlanje.append("id",id)

    $.ajax({
        url:"../models/editPerformance.php",
        method:"post",
        dataType:"json",
        processData:false,
        contentType:false,
        data:podaciZaSlanje,
        success:function(data){
            $("#messageInsert").html(data)
        },
        error:function(xhr){
            $("#messageInsert").html(JSON.parse(xhr.responseText))
        }
    })

    return false

}
function deletePerformance(e){
    e.preventDefault()
    var id=this.dataset.id
    $.ajax({
        url:'../models/deletePerformance.php',
        method:"POST",
        dataType:"json",
```

```
        data:{
            "id":id
        },
        success:function(data){
            getPerformancesAdmin()
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })
}
function changePageAdmin(e){
    e.preventDefault()
    var page=this.dataset.id
    $.ajax({
        url:"../models/performances.php",
        method:"get",
        dataType:"json",
        data:{
            "page": page,
            "btn":true
        },
        success:function(data){
            if(data.performances.length!=0){
                printPerformancesAdmin(data)
            }
            else{
                $("#content").html("<h2 class='text-primary'>There is no matching
data</h2>")
            }
        },
        error:function(xhr){
            //err
        }
    })
}

function addRepertoire(e){
    e.preventDefault()
    $.ajax({
        url:'../models/allPerformances.php',
        method:"GET",
        dataType:"json",
        data:{
            "btn":true
```

```
        },
        success:function(data){
            printFormRepertoire(data)
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })
}

function printFormRepertoire(data){
    let print=`<h2>Add Repertoire</h2>
    <form action="#">
        <div>
            <label for="performance">Performance</label>
            <select id="idPerformance" class="form-control">
                <option value="0">Choose performance</option>`
            for(let d of data){
                print+=`<option value="${d.id}">${d.name}</option>`
            }
        print+=`</select>
        </div>
        <div class="mt-2">
            <label for="date">Date Performing</label>
            <input type="date" id="date" class="form-control">
        </div>
        <div class="mt-2">
            <input type="submit" id="add" value="Add repertoire" class="form-
control">
        </div>
        <div class="mt-2">
            <p id="messageInsert"></p>
        </div>
    </form>`


    $("#content").html(print)

    $("#add").click(addOnRepertoire)
}

function addOnRepertoire(){
    var idPerformance=$("#idPerformance").val()
    var date=$("#date").val()
    $.ajax({
```

```
        url:'../models/addRepertoire.php',
        method:"POST",
        dataType:"json",
        data:{
            "btn":true,
            "idPerformance":idPerformance,
            "date":date
        },
        success:function(data){
            $("#messageInsert").html(data)
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })

    return false
}

function getRepertoire(){
    $.ajax({
        url:'../models/repertoireAdmin.php',
        method:"GET",
        dataType:"json",
        data:{
            "btn":true
        },
        success:function(data){
            printRepertoire(data)
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })
}

function printRepertoire(data){
    let br=1
    let print=''
    if(data.length==0){
        print="<h2>There's nothing on repertoire</h2>"
    }
    else{
        print=`<table class="table">
            <tr>
```

```
                <td>No</td>
                <td>Name</td>
                <td>Date Performing</td>
                <td>Delete</td>
            </tr>`
        for(let d of data){
            print+=`<tr>
                        <td>${br++}</td>
                        <td>${d.name}</td>
                        <td>${d.datePerforming}</td>
                        <td class="text-center"><a href="#" class="delete" data-
id="${d.id}"><i class="fa-regular fa-xmark"></i></a></td>
                    </tr>`
        }
    }


    $("#content").html(print)

    $(".delete").click(deleteRepertoire)

}
function deleteRepertoire(e){
    e.preventDefault()
    var id=this.dataset.id
    $.ajax({
        url:'../models/deleteRepertoire.php',
        method:"POST",
        dataType:"json",
        data:{
            "id":id
        },
        success:function(data){
            getRepertoire()
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })
}
function getPollRes(){
    $.ajax({
        url:'../models/pollRez.php',
        method:"POST",
        dataType:"json",
```

```
        data:{
            "btn":true
        },
        success:function(data){
            printPoll(data)
        },
        error:function(xhr){
            $("#content").html(JSON.parse(xhr.responseText))
        }
    })
}
function printPoll(data){
    let print="<div class='text-center'>"
    print+=`<h2 class="my-2">${data['question']}</h2>`
    for(let d of data['voting']){
        print+=`<p class="mb-2">${d.answer}: ${d.answers} answers</p>`
    }
    if(data['active']==1){
        print+=`<a href="#" id="disable" class="text-center">Disable poll</a>`
    }
    else{
        print+=`<a href="#" id="enable" class="text-center">Enable poll</a>`
    }
    print+="</div>"
    $("#content").html(print)

    $("#disable").click(disablePoll)
    $("#enable").click(enablePoll)
}
function disablePoll(e){
    e.preventDefault()
    $.ajax({
        url:"../models/disablePoll.php",
        method:"post",
        dataType:"json",
        success:function(data){
            getPollRes()
        },
        error:function(xhr){
            console.log(xhr)
        }
    })
}
function enablePoll(e){
    e.preventDefault()
```

```
    $.ajax({
        url:"../models/enablePoll.php",
        method:"post",
        dataType:"json",
        success:function(data){
            getPollRes()
        },
        error:function(xhr){
            console.log(xhr)
        }
    })
}
function addPerformance(e){
    e.preventDefault()
    let print=`<h2>Add Performance</h2>
                <form action="#" enctype="multipart/form-data">
                    <div>
                        <label for="name">Name</label>
                        <input type="text" id="name" class="form-control">
                    </div>
                    <div class="mt-2">
                        <label for="poster">Poster</label>
                        <input type="file" id="file">
                    </div>
                    <div class="mt-2">
                        <label for="premier">Premier Date</label>
                        <input type="date" id="premier" class="form-control">
                    </div>
                    <div class="mt-2">
                        <label for="duration">Duration</label>
                        <input type="number" id="duration" class="form-control">
                    </div>
                    <div class="mt-2">
                        <label for="description">Description</label>
                        <textarea class="form-control"
id="description"></textarea>
                    </div>
                    <div class="mt-2">
                        <label for="price">Price</label>
                        <input type="number" class="form-control" id="price">
                    </div>
                    <div class="mt-2">
                        <input type="submit" id="add" value="Add performance"
class="form-control">
                    </div>
```

```
                    <div class="mt-2">
                        <p id="messageInsert"></p>
                    </div>
                </form>`


    $("#content").html(print)

    $("#add").click(addNewPerformance)
}

function addNewPerformance(){
    var name=$("#name").val()
    var file=$("#file")[0].files[0]
    var premier=$("#premier").val()
    var duration=$("#duration").val()
    var description=$("#description").val()
    var price=$("#price").val()


    var podaciZaSlanje = new FormData()

    podaciZaSlanje.append("name",name)
    podaciZaSlanje.append("file",file)
    podaciZaSlanje.append("premier",premier)
    podaciZaSlanje.append("duration",duration)
    podaciZaSlanje.append("description",description)
    podaciZaSlanje.append("price",price)

    $.ajax({
        url:"../models/addPerformance.php",
        method:"post",
        dataType:"json",
        processData:false,
        contentType:false,
        data:podaciZaSlanje,
        success:function(data){
            $("#messageInsert").html(data)
        },
        error:function(xhr){
            $("#messageInsert").html(JSON.parse(xhr.responseText))
        }
    })
```

```
    return false
}
```

Style.css

```
*{
    margin: 0px;
    padding: 0px;
    box-sizing: border-box;
}
body{
    font-family: 'Anek Latin', sans-serif;
}
a{
    text-decoration: none;
}
a:hover{
    color: unset;
}
.nav-item a{
    color: green;
}
img{
    width: 100%;
}
#pagination ul li{
    list-style-type: none;
    margin: 0px 5px;
}
#pagination ul li a{
    text-decoration: none;
}
.places{
    padding: 5px 10px;
    background-color: greenyellow;
    position: relative;
}
.places{
    color: aliceblue;
}
.purchased{
    color: rgb(156, 156, 156);
    pointer-events: none;
}
.purchased{
```

```
    background-color: gray;
}
.popup{
    visibility: hidden;
    position: absolute;
    z-index: 99;
    width: 130px;
    background-color: rgb(255, 237, 103);
    padding: 5px;
    margin-left: 5px;
}
.places:hover .popup{
    visibility: visible;
}
.red{
    width: 100px;
    text-align: center;
}
.redDrzac{
    width: 100%;
    display: flex;
    justify-content: space-between;
    margin: 15px 0px;
}
.addToCart{
    background-color: green;
}
.parter{
    display: flex;
    justify-content: space-between;
}
.scena{
    padding: 50px;
    text-align: center;
    background-color: grey;
}
.kupi{
    background-color: green;
}
.font-weight-bold{
    font-weight: bold;
}
.boja{
    padding: 0px 10px;
    margin-right: 10px;
```

```
}
.dostupna{
    background-color: greenyellow;
}
.izabrana{
    background-color: green;
}
.nedostupna{
    background-color: gray;
}
main{
    margin-top: 80px;
    margin-bottom: 10px;
}
.repertoire{
    border-top: 2px solid black;
}
.buy{
    background-color: rgb(20, 106, 20);
}
.next{
    display: flex;
    flex-direction: column;
    justify-content: space-between;
}
#pagination a{
    font-size: 20px;
    margin: 0px 5px;
    color: green;
}
#pagination .active{
    color: greenyellow;
}
input[type=submit]{
    background-color: greenyellow;
}
.green,footer a{
    color: green;
}
.divLogReg{
    min-height: 450px;
}
.sidebar .nav-item .nav-link i{
    font-size: 20px!important;
}
```

```
.sidebar .nav-item .nav-link{
    font-size: 25px;
}
#logo img{
    width: 45px;
}
.mainAdmin{
    min-height: 480px;
}
#pagination a{
    text-decoration: none;
}
.btn{
    border-radius: 0px;
}
.btnBuy{
    background-color: green;
    color: white;
}
.modal .btn:hover{
    color: white;
}
```