



UNIVERSITÀ DEGLI STUDI DI MESSINA

Professor Armando Ruggeri

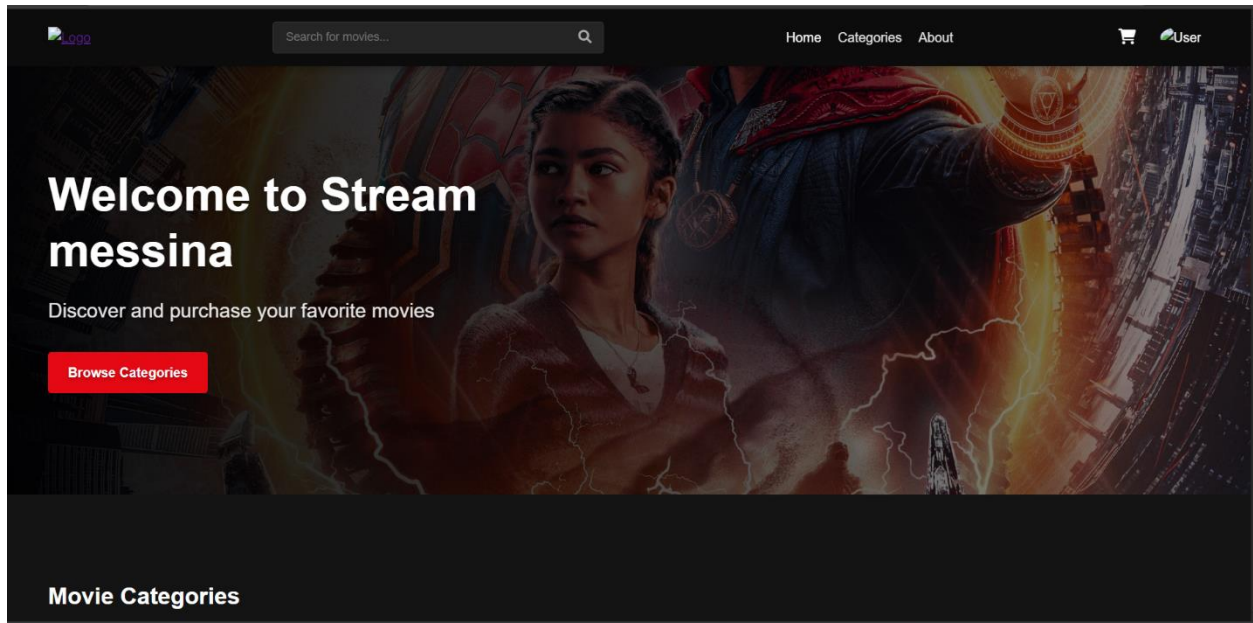
Web programming Project

Sadegh Mohammadi (533033)

Content:

1. Website introduction
2. Way of implementing
3. locally creating database
4. Explanation of the codes

Stream movie website



This website is a stream movie website, costumer can login to website make their credential and choosing from different categories of movies one, and after making the purchasing process by their cart, they can start watching their move.

The website contain different parts for interaction of the user including:

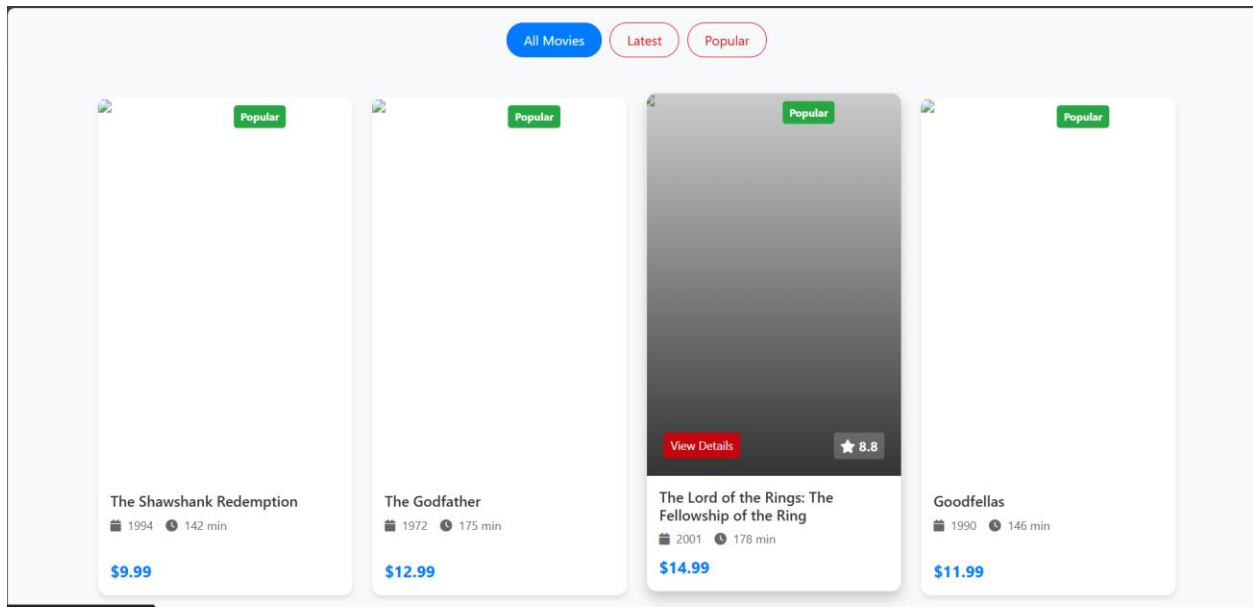
The login-logout-register

The search bar for making the search process for costumer easy.

The profile and setting part so user can modify their credential

The cart, payment, purchase basket and final confirmation for making the payment possible

Here bellow I attach some of them and explain more in detail.



Here we can see the different category of movie in different ganer.

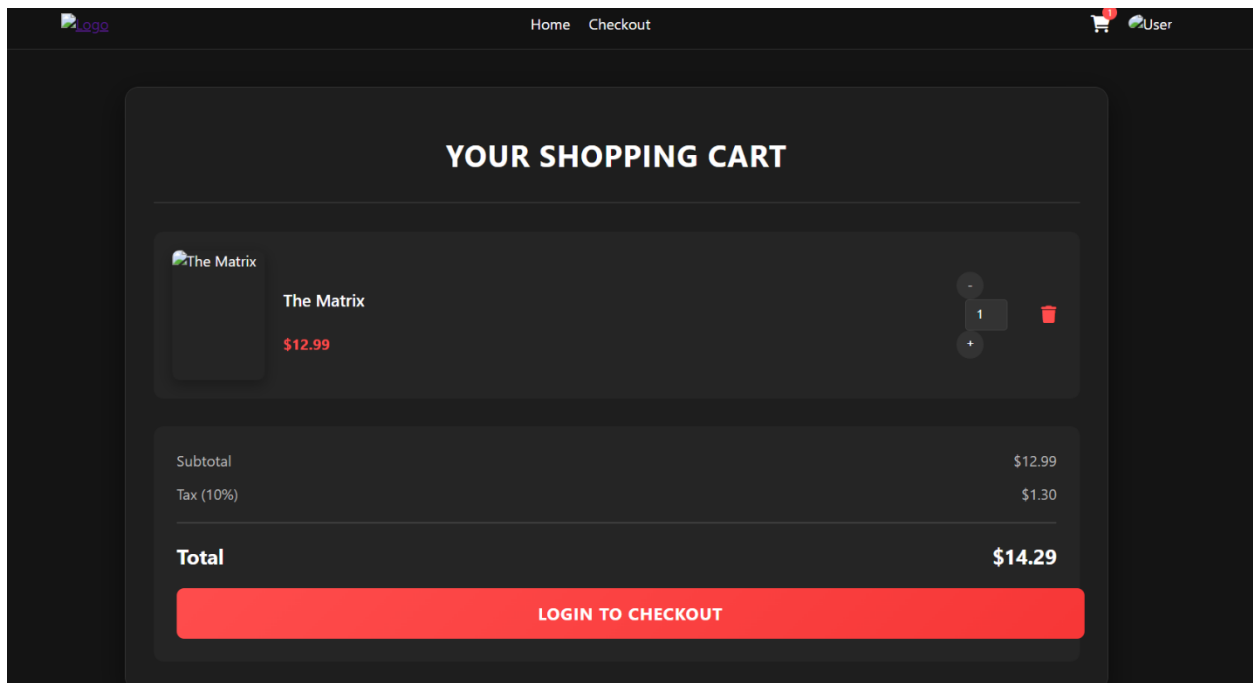
Each movie items associates with 2 bottom

One bottom for play and one for adding it to cart

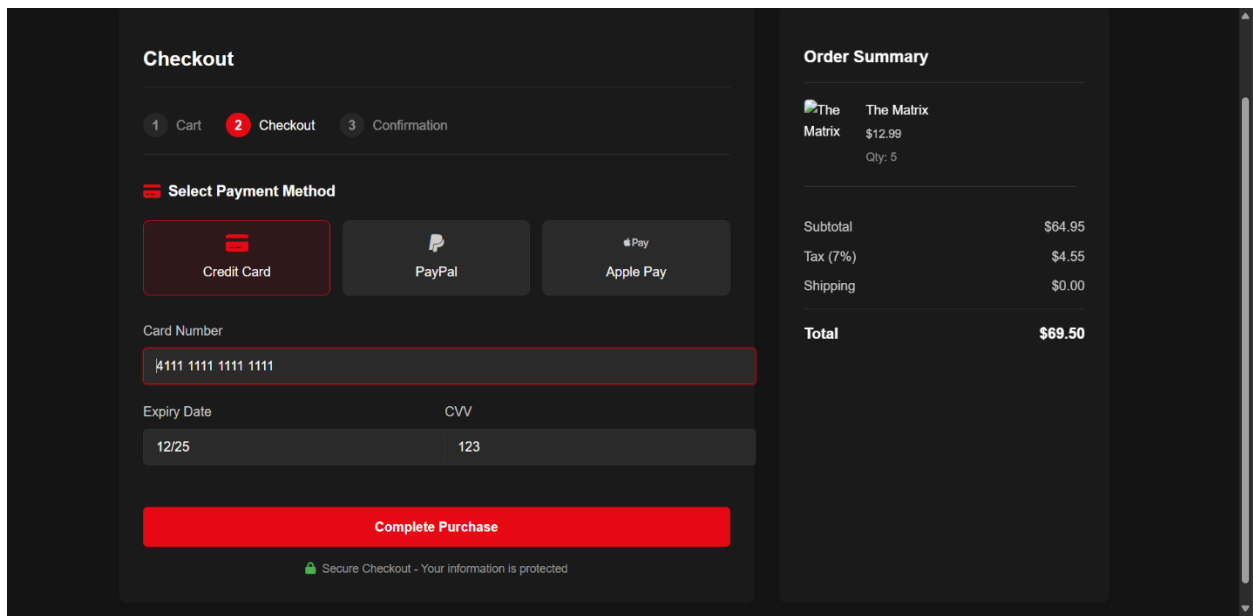
The costumer for the purpose of play the movie first need to make the purchase

After login and completing the purchase process he can start watching it.

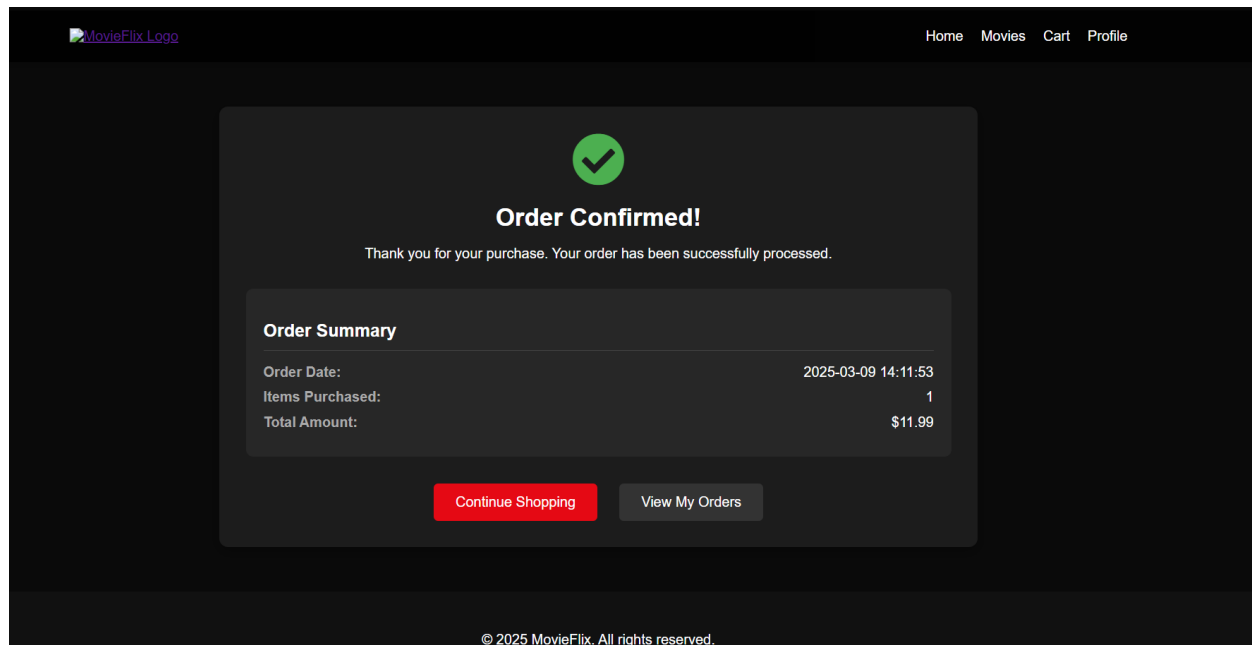
Other bottom as we can see help the customer to add it directly to cart payment part



Here we have a cart part which show the use the info on purchase like the number and the cost which he or she has to pay.



Here we are at checkout page. Which make the different option for payment for costumer . User can choice the method of payment which is easier for him. Also the right side of the page we see the order summary, I and the total to pay.



Here we see the final step, which confirm the payment of movie for user.

It shows some more info like the data of pay and the amount of pay.

	id	user_id	total_amount	status	payment_method	payment_status	created_at	updated_at
<input type="checkbox"/>	13	4	12.99	completed	NULL	pending	2025-04-13 16:23:09	2025-04-13 16:23:09
<input type="checkbox"/>	14	4	12.99	completed	NULL	pending	2025-04-13 16:31:04	2025-04-13 16:31:04
<input type="checkbox"/>	15	4	8.99	completed	NULL	pending	2025-04-13 16:31:12	2025-04-13 16:31:12
<input type="checkbox"/>	16	4	9.99	completed	NULL	pending	2025-04-13 16:33:57	2025-04-13 16:33:57
<input type="checkbox"/>	17	4	14.99	completed	NULL	pending	2025-04-13 17:43:14	2025-04-13 17:43:14
<input type="checkbox"/>	20	4	12.99	completed	NULL	pending	2025-04-24 08:49:28	2025-04-24 08:49:28

All the order sent to the DB in phpMyAdmin. And reserved there by name and date.

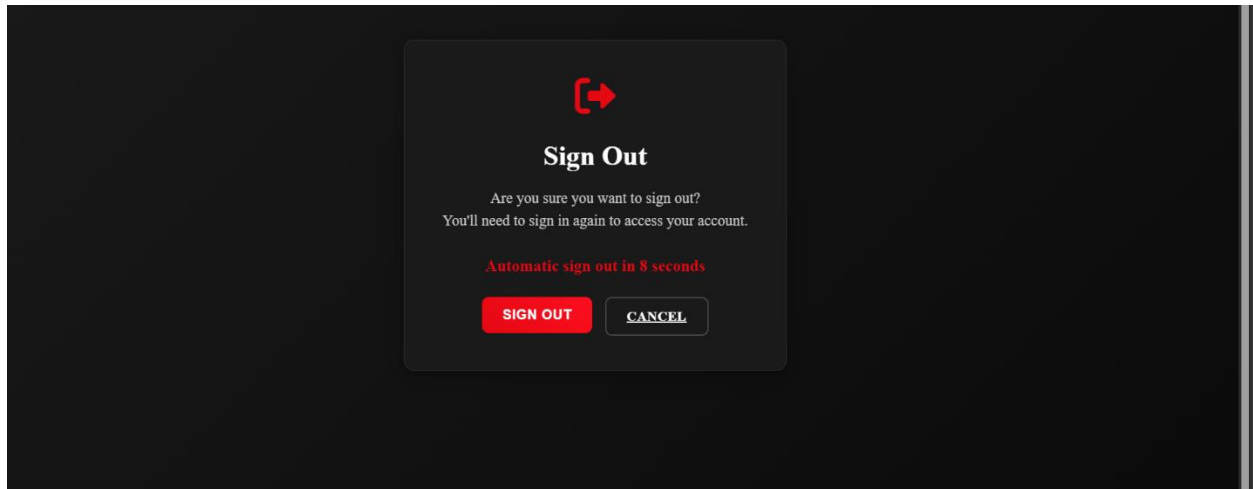
The screenshot shows the 'Create Account' page of the StreamFlix application. At the top, there is a navigation bar with the 'StreamFlix Logo' on the left, and 'Home', 'Movies', and 'About' links in the center. On the right side of the navigation bar are 'Sign In' and 'Help' buttons. The main content area features a dark gray card with the title 'Create Account' in white. Below the title are four input fields: 'Username', 'Email', 'Password', and 'Confirm Password'. A prominent red 'Register' button is positioned below these fields. A link that says 'Already have an account? [Login here](#)' is located at the bottom of the card. In the bottom right corner of the page, a small status bar indicates 'Battery status: 31% remaining' and '38min'.

Here we see the registr part which user need to register in order to get access for login and make the purchasing process.

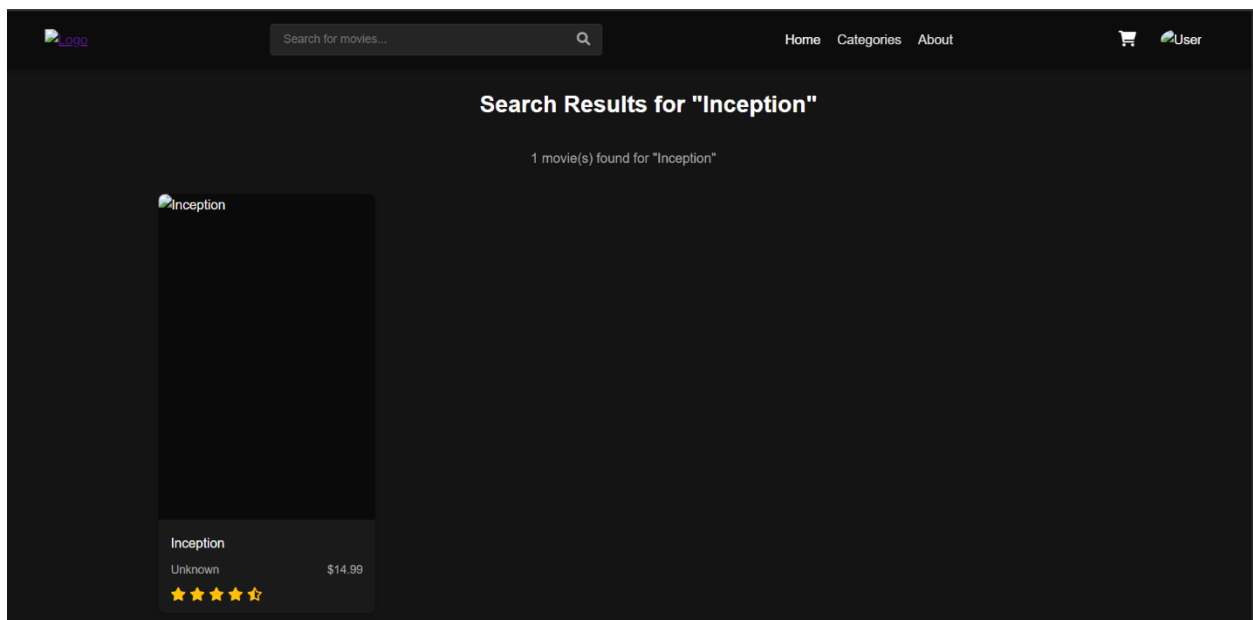
The screenshot displays the 'Welcome Back' login page of the StreamFlix application. The page has a dark background with a central dark gray card. The card is titled 'Welcome Back' in white. It contains two input fields: 'Email Address' and 'Password'. The 'Email Address' field has an envelope icon on the right, and the 'Password' field has an eye icon. Below the password field is a link that says 'Forgot your password?'. A large red 'SIGN IN' button is centered below the input fields. Underneath the button, the text 'or continue with' is followed by two buttons: 'Google' (with the Google 'G' logo) and 'Facebook' (with the Facebook 'f' logo). At the bottom of the card, there is a link that says 'New to StreamFlix? [Create an account](#)'. The overall design is clean and modern with a dark theme.

The login part after registration done.

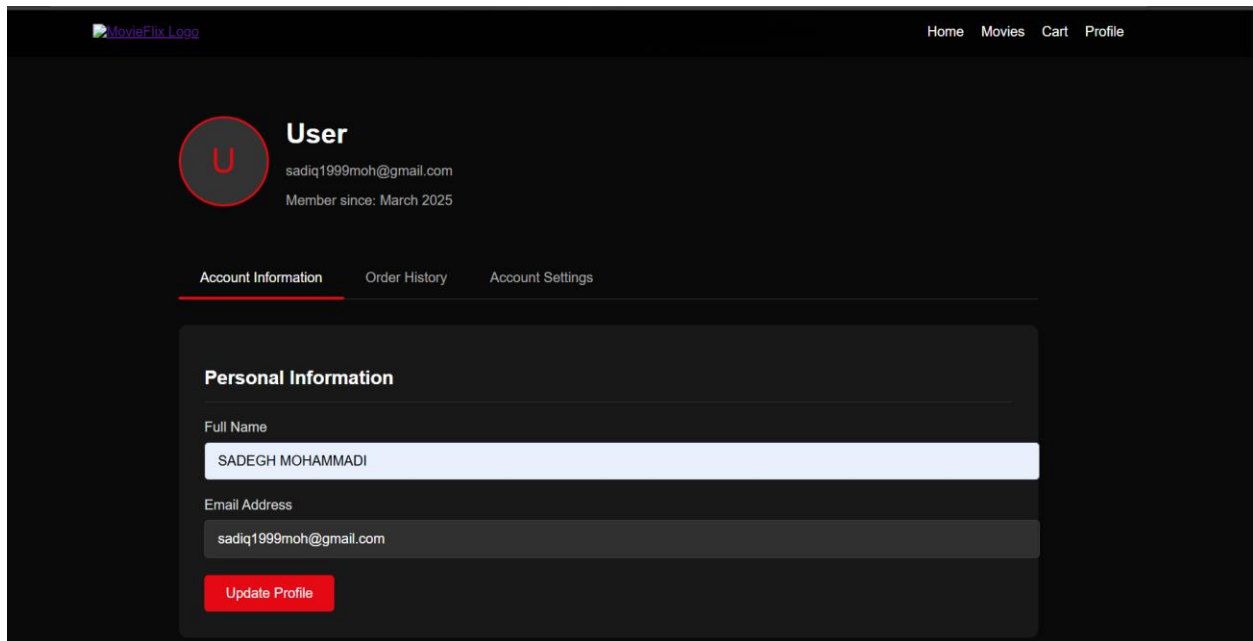
User use his credential to enter.



Here is the sign out part as well.

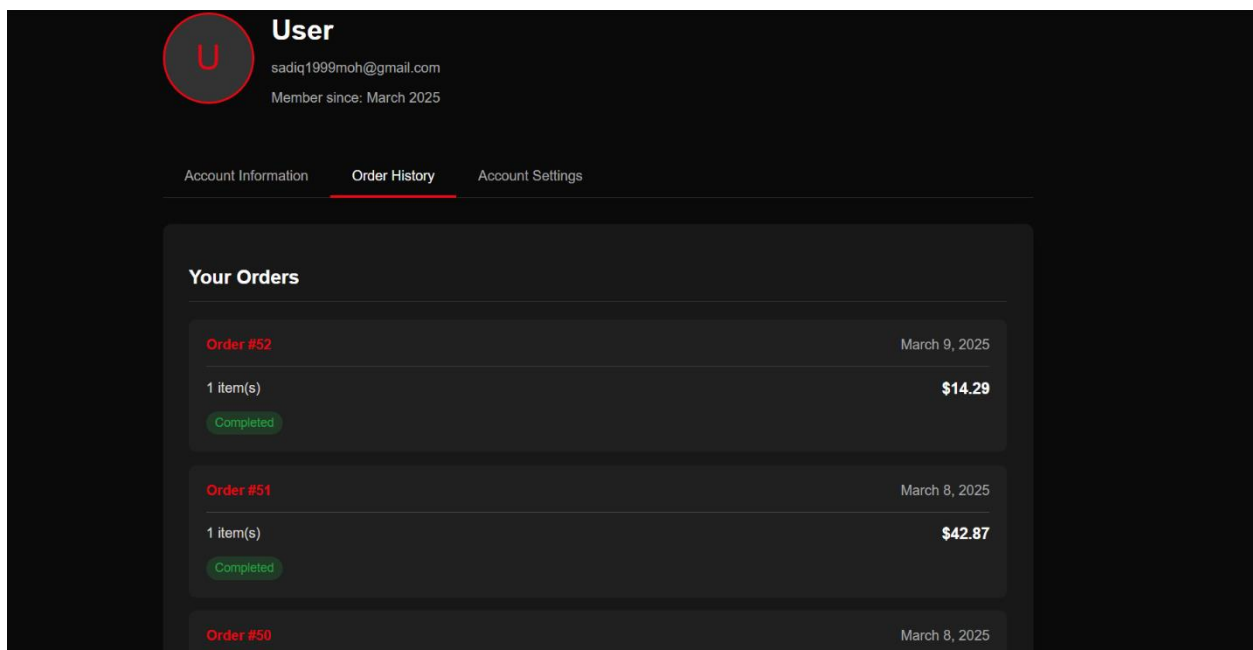


Here I add a search bar by JavaScript, user can enter the name of the movie they looking for and immediately find it within different category.



Here we have profile part of the user.

User can see, change and update his information here.



At here we see the purchase history of the customer.

U

User

sadiq1999moh@gmail.com

Member since: March 2025

Account Information

Order History

Account Settings

Account Settings

Change Password

New password

Confirm Password

Confirm new password

Update Password

Danger Zone

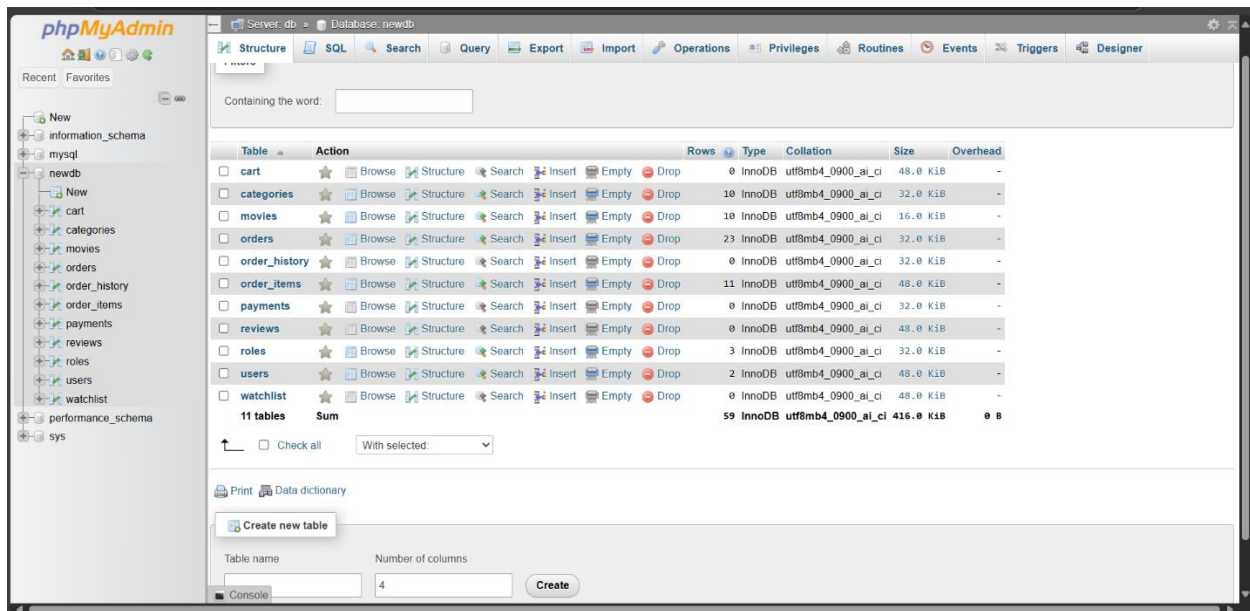
Delete Account

And finally the account setting, which user here has the right to delete his account or modify the password.

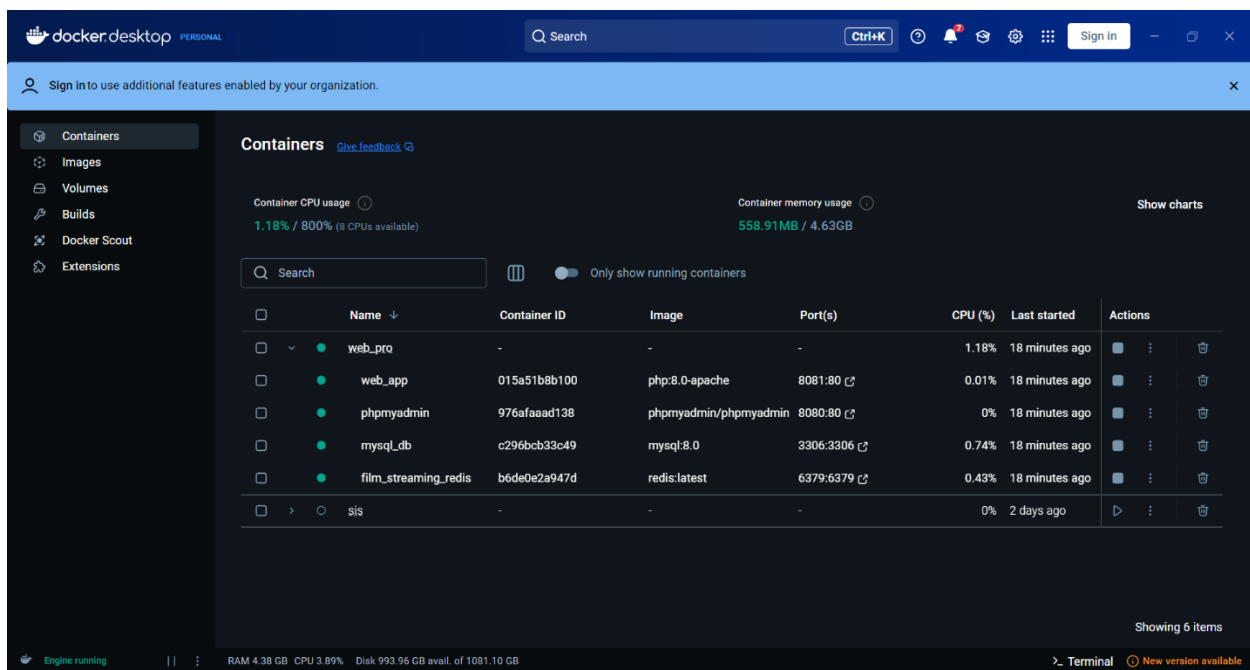
The screenshot shows the phpMyAdmin interface with the 'users' table selected. The table structure and data are as follows:

	id	username	email	password	role_id	avatar	created_at	updated
<input type="checkbox"/>	3	admin	admin@example.com	\$2y\$10\$xtz18jl74lWSSkdw3vklm.JgUfGGJbyk0iHgGhBN3E...	1	assets/images/default-avatar.png	2025-04-13 13:39:58	2025-04-
<input type="checkbox"/>	4	sadeh	sadiq1999moh@gmail.com	\$2y\$10\$H7FM7GfyS9owKMGuTBwu7pqARViyAm8lSPZcq6/aD...	1	assets/images/default-avatar.png	2025-04-13 13:40:07	2025-04-

User information and his register details all saved here at db.

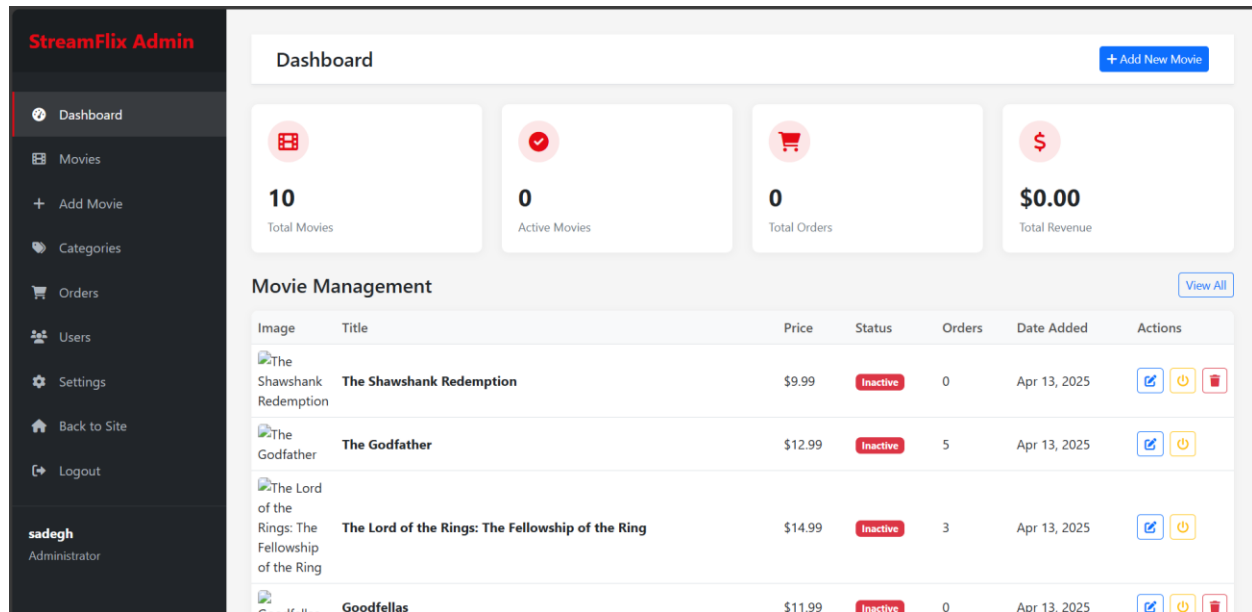


For this project I used the db. on MySQL server. Here we can see the general structure of db with its associated tables and values.



All db. connections and webpage and SQL code running on docker desktop application, I made different container for different part to handle it better.

Here below we have a admin panel which give the access to the platform admin to add new product or remove some product from site or change the information related to the product.



Code explanation

Here below is the related to the main page or index page, including the starting php code followed by the header and main section and footer part of the page, also the css embedded within.

```

</style>
</head>
<body>
  <!-- Navigation Bar -->
  <nav class="navbar navbar-expand-lg navbar-dark">
    <div class="container">
      <a class="navbar-brand" href="index.php">Movie Messina</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarNav">
        <ul class="navbar-nav me-auto">
          <li class="nav-item">
            <a class="nav-link" href="index.php">Home</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="movies.php">Movies</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="categories.php">Categories</a>
          </li>
        </ul>
        <form class="d-flex me-3" action="search.php" method="GET">
          <input class="form-control me-2" type="search" name="q" placeholder="Search movies..." aria-label="Search">
          <button class="btn btn-outline-light" type="submit"><i class="fas fa-search"></i></button>
        </form>
        <ul class="navbar-nav">
          <?php if (isLoggedIn()): ?>
            <li class="nav-item dropdown">
              <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-bs-toggle="dropdown" aria-expanded="false">
                <?php if (empty($user['profile_image'])): ?>
                  
                <?php else: ?>
                  <i class="fas fa-user-circle"></i>
                <?php endif; ?>
              <?php echo htmlspecialchars($user['username']); ?>
            </li>
          </?php>
        </ul>
      </div>
    </div>
  </nav>

```

```

.nav-link {
  color: var(--text-color) !important;
  margin: 0 0.5rem;
  transition: color 0.3s ease;
}

.nav-link:hover {
  color: var(--primary-color) !important;
}

.btn-primary {
  background-color: var(--primary-color);
  border-color: var(--primary-color);
}

.btn-primary:hover {
  background-color: #c4070f;
  border-color: #c4070f;
}

.btn-outline-primary {
  color: var(--primary-color);
  border-color: var(--primary-color);
}

.btn-outline-primary:hover {
  background-color: var(--primary-color);
  border-color: var(--primary-color);
}

.user-avatar {
  width: 32px;
  height: 32px;
  border-radius: 50%;
}

```

```

    </div>
</section>
<?php else: ?>

<!-- Movies Available Section -->
<section class="movies-available py-5">
    <div class="container">
        <div class="section-header text-center mb-5">
            <h2 class="section-title display-4 fw-bold">Movies Available</h2>
            <p class="lead text-muted">Discover our latest and most popular movies</p>
            <div class="filter-buttons mt-4">
                <button class="btn btn-outline-primary active" data-filter="all">All Movies</button>
                <button class="btn btn-outline-primary" data-filter="latest">Latest</button>
                <button class="btn btn-outline-primary" data-filter="popular">Popular</button>
            </div>
        </div>
        <div class="row g-4">
            <?php
            try {
                // Get both latest and popular movies
                $latestMovies = $db->query("SELECT * FROM movies ORDER BY created_at DESC LIMIT 8");
                $popularMovies = $db->query("SELECT * FROM movies ORDER BY rating DESC LIMIT 8");

                $allMovies = [];

                // Combine and deduplicate movies
                while ($movie = $latestMovies->fetch(PDO::FETCH_ASSOC)) {
                    $movie['type'] = 'latest';
                    $allMovies[$movie['id']] = $movie;
                }

                while ($movie = $popularMovies->fetch(PDO::FETCH_ASSOC)) {
                    if (!isset($allMovies[$movie['id']])) {
                        $movie['type'] = 'popular';
                        $allMovies[$movie['id']] = $movie;
                    }
                }
            } catch (Exception $e) {
                // Handle database error
            }
            </?php>
        </div>
    </div>
</section>

```

```

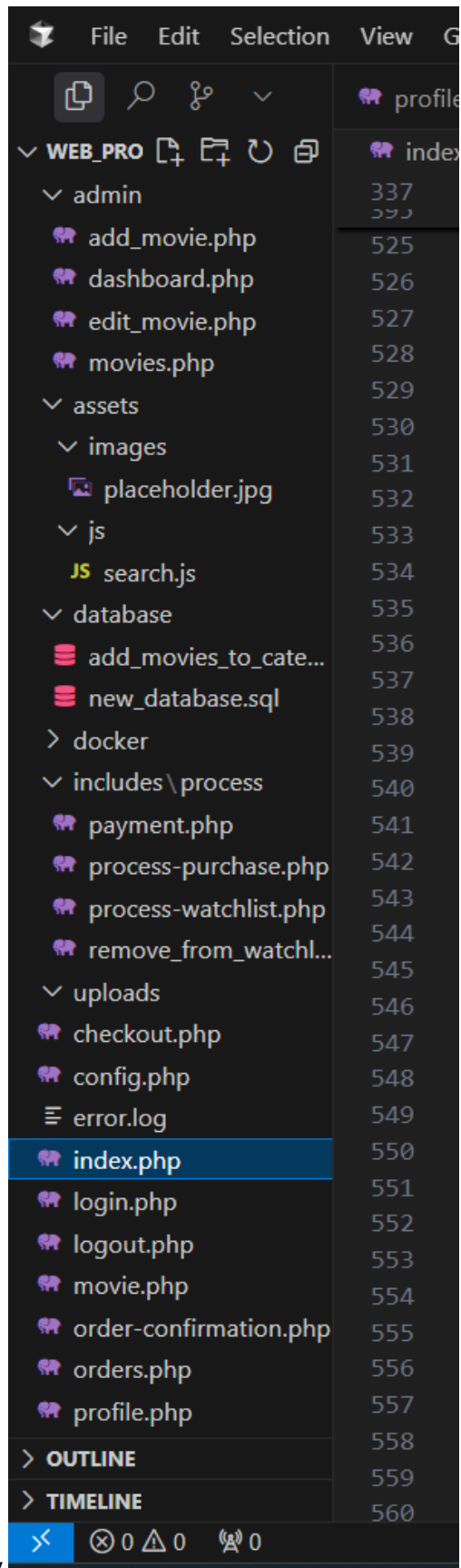
</main>

<!-- Footer -->
<footer class="bg-dark text-white py-5 mt-5">
    <div class="container">
        <div class="row">
            <div class="col-md-4 mb-4 mb-md-0">
                <h5 class="text-primary mb-3"><?php echo SITE_NAME; ?></h5>
                <p class="text-muted">Your ultimate destination for streaming the latest and greatest movies. Watch anytime, anywhere.</p>
                <div class="social-icons mt-3">
                    <a href="#" class="text-white me-3"><i class="fab fa-facebook-f"></i></a>
                    <a href="#" class="text-white me-3"><i class="fab fa-twitter"></i></a>
                    <a href="#" class="text-white me-3"><i class="fab fa-instagram"></i></a>
                    <a href="#" class="text-white"><i class="fab fa-youtube"></i></a>
                </div>
            </div>
            <div class="col-md-4 mb-4 mb-md-0">
                <h5 class="text-white mb-3">Quick Links</h5>
                <ul class="list-unstyled">
                    <li class="mb-2"><a href="index.php" class="text-muted text-decoration-none">Home</a></li>
                    <li class="mb-2"><a href="movies.php" class="text-muted text-decoration-none">Movies</a></li>
                    <li class="mb-2"><a href="categories.php" class="text-muted text-decoration-none">Categories</a></li>
                    <li class="mb-2"><a href="about.php" class="text-muted text-decoration-none">About Us</a></li>
                    <li class="mb-2"><a href="contact.php" class="text-muted text-decoration-none">Contact</a></li>
                </ul>
            </div>
            <div class="col-md-4">
                <h5 class="text-white mb-3">Categories</h5>
                <ul class="list-unstyled">
                    <li class="mb-2"><a href="category.php?slug=action" class="text-muted text-decoration-none">Action</a></li>
                    <li class="mb-2"><a href="category.php?slug=comedy" class="text-muted text-decoration-none">Comedy</a></li>
                    <li class="mb-2"><a href="category.php?slug=drama" class="text-muted text-decoration-none">Drama</a></li>
                    <li class="mb-2"><a href="category.php?slug=horror" class="text-muted text-decoration-none">Horror</a></li>
                    <li class="mb-2"><a href="category.php?slug=sci-fi" class="text-muted text-decoration-none">Sci-Fi</a></li>
                </ul>
            </div>
        </div>
    </div>
</footer>

```

Below is the list of all files including, php, js, sql and docker files. All in their correspondent, all css, html codes are php embedded. Except for JavaScript which has its own separate files.

And at the end the javascript file which I used to make a search bar for items. So user can easily search through the db related table to find his interested item and retrieved in front end part.



directory.

```
document.addEventListener('DOMContentLoaded', function() {

    const searchForm = document.querySelector('form[action="search.php"]');
    const searchInput = searchForm.querySelector('input[name="q"]');
    const resultsContainer = document.createElement('div');

    resultsContainer.className = 'search-results-container';
    resultsContainer.style.display = 'none';
    searchForm.appendChild(resultsContainer);

    resultsContainer.style.position = 'absolute';
    resultsContainer.style.top = '100%';
    resultsContainer.style.left = '0';
    resultsContainer.style.right = '0';
    resultsContainer.style.backgroundColor = '#222';
    resultsContainer.style.borderRadius = '0 0 8px 8px';
    resultsContainer.style.boxShadow = '0 4px 15px rgba(0,0,0,0.3)';
    resultsContainer.style.maxHeight = '350px';
    resultsContainer.style.overflowY = 'auto';
    resultsContainer.style.zIndex = '1000';

    searchInput.addEventListener('focus', handleSearchFocus);
    searchInput.addEventListener('input', debounce(handleSearchInput, 300));
    document.addEventListener('click', function(e) {
        if (!searchForm.contains(e.target)) {
            resultsContainer.style.display = 'none';
        }
    });

    searchForm.addEventListener('submit', function(e) {
        const searchTerm = searchInput.value.trim();
```