

# AquaParadise

Autorzy:

Bartosz Dziekan

Paweł Helowicz

# Spis treści:

Wstęp .....	3
Opis ogólny.....	4
Struktura strony internetowej .....	5
Elementy statyczne .....	14
JavaScript.....	14
Elementy dynamiczne (PHP) .....	15
Baza danych .....	16
Bezpieczeństwo .....	18
Możliwości rozbudowy.....	19
Załączniki .....	20

# Wstęp

## 1.1 Cel projektu

Celem projektu jest utworzenie strony internetowej dla kompleksu relaksacyjnego AquaParadise, która daje możliwość poznania informacji na temat ośrodka, wystawić opinie i przede wszystkim zarezerwować pobyt na różne atrakcje.

## 1.2 Zakres projektu

Pracując w dwuosobowej grupie, podzieliliśmy się na konkretne zadania, ewentualnie wprowadzaliśmy poprawki lub zmiany do naszych prac w razie potrzeby i porozumiewaliśmy się w razie potrzeby co potrzeba jeszcze dodać i zrobić.

## 1.3 Użytkownicy systemu

## Opis ogólny

### 2.1 Przeznaczenie witryny

Witryna ma na celu danie użytkownikowi możliwości poznania każdej części kompleksu, dowiedzenia się szczegółowych informacji, oraz zakup/rezerwacje konkretnych usług. Natomiast dla pracowników i administratorów również mamy panel administracyjny do wygodnego zarządzania danymi.

### 2.2 Główne funkcjonalności

Dostęp do 6 podstron przedstawiających strone główną, 4 strefy kompleksu i opinie, możliwość logowania lub rejestracji, dla administracji i pracowników panel administracyjny do zarządzania danymi, możliwość rezerwacji i zakupów na konkretnych stronach.

### 2.3 Wymagania techniczne

Uruchomienie MySQL w programie Xampp oraz importowanie bazy danych znajdującej się w repozytorium. Uruchomienie strony poprzez localhost/repozytorium na przeglądarce internetowej.

### 2.4 Struktura katalogów i plików projektu

W folderze repozytorium znajdują się pliki głównych stron i podstron witryny, oraz plik bazy danych, plik skryptu(java script), plik stylu(css), podzielono również na katalogi admin zawierającego pliki dotyczące panelu admina i obsługi go, foldery z nazwą podstrony i końcówką \_assets zawierające zdjęcia i pliki potrzebne konkretnym podstronom, katalogi config, src, includes w

których znajdują się pliki do których odwołujemy się w głównych plikach aby załączyć strukturę konkretnych elementów itp.

## Struktura strony internetowej

### 3.1 Strona główna (index.php)

```
<?php
// index.php
require_once __DIR__ . '/config/init.php';

$page_title = "AquaParadise - Witaj w Raju!"; // Ustaw tytuł strony

include BASE_PATH . '/includes/header.php'; // Dołącz główny nagłówek
?>

<header class="main-header">
  <div class="header-content">
    <h1>Witaj w AquaParadise!</h1>
    <p class="subtitle">Twoje miejsce ucieczki do świata wodnych przygód, luksusowego wypoczynku i niezapomnianych chwil. Odkryj magię, która czeka na Ciebie.</p>
    <a href="#explore" class="cta-button header-cta">Odkryj Nasze Światy</a>
  </div>
</header>

<?php
// Dołącz dynamiczną nawigację
include BASE_PATH . '/includes/navigation.php';
?>
```

Powtarzające elementy dla wszystkich 6 głównych podstron takie jak header, nawigacja i stopka pobieramy z zewnętrznych plików zawartych w folderach includes, config i src, ewentualnie dodajemy dodatkowe elementy w razie potrzeb tak jak ten header.

```
<div class="page-wrapper" id="explore">
  <section class="section-title-container">
    <h2>Eksploruj Nasze Światy</h2>
  </section>

  <section class="explore-worlds">
    <div class="world-card">
      <div class="world-card-image">
        
      </div>
      <div class="world-card-content">
        <h3><i class="fas fa-water"></i> Niesamowity Aquapark</h3>
        <p>Zanurz się w ekscytującym świecie zjeżdżałni, basenów z falami i stref relaksu dla całej rodziny. Adrenalina i zabawa gwarantowana!</p>
      </div>
    </div>
  </section>
</div>
```

Tutaj przycisk przenoszący do niższej części strony oraz powtarzające się części sekcji z zdjęciami i przenośnikami do konkretnych podstron

```
<section class="cta-section-main" id="contact">
  <h2>Gotowy na Przygodę Życia?</h2>
  <p>Zaplanuj swój pobyt w AquaParadise już dziś i stwórz wspomnienia, które zostaną z Tobą na zawsze. Czekamy na Ciebie!</p>
  <a href="aquapark.php#bilety" class="cta-button">Kup Bilety lub Zobacz Ofertę</a>
</section>
</div>
```

Elegancka reklama i przycisk przenoszący do zakupu biletów

### 3.2 Podstrona hotelu (hotel.php)

```
?>
<header class="page-header" style="background-image: linear-gradient(135deg, rgba(0, 123, 255, 0.7) 0%, rgba(0, 86, 179, 0.85) 100%), url('hotel_header.jpg');">
  <div class="content">
    <h1>Luksusowy Hotel AquaParadise</h1>
    <p>Odkryj komfort i elegancję w sercu wodnego raju. Idealne miejsce na Twój wypoczynek.</p>
  </div>
</header>
<?php
```

#### Nagłówek dla hotelu

```
<div class="page-wrapper">
  <section class="section-title-container">
    <h2>Komfortowe Pokoje i Apartamenty</h2>
  </section>

  <section class="rooms-list">
    <div class="room-card">
      <div class="room-img">
        
      </div>
      <div class="room-content">
        <h3>Pokój Standard</h3>
        <p>Elegancki i funkcjonalny pokój, zapewniający komfortowy wypoczynek po dniu pełnym wrażeń.</p>
        <div class="features-tags">
          <span class="feature-tag">Łóżko Małżeńskie</span>
          <span class="feature-tag">Widok na Ogród</span>
          <span class="feature-tag">Klimatyzacja</span>
          <span class="feature-tag">TV SAT</span>
        </div>
        <p><strong>Od 450 PLN / noc</strong></p>
        <a href="hotel_assets/hotel_b.php" class="cta-button">Zarezerwuj Teraz</a>
      </div>
    </div>
  </section>
</div>
```

Powtarzająca się sekcja z kategoriami pokoi hotelu, odnośnikami do rezerwacji oraz informacjami

```

<section class="section-title-container amenities-title-container">
  <h2>Nasze Udogodnienia</h2>
</section>
<section class="amenities-grid">
  <div class="amenity-card">
    <i class="fas fa-concierge-bell"></i>
    <h3>Obsługa Concierge</h3>
  </div>
  <div class="amenity-card">
    <i class="fas fa-dumbbell"></i>
    <h3>Centrum Fitness</h3>
  </div>
  <div class="amenity-card">
    <i class="fas fa-wifi"></i>
    <h3>Bezpłatne Wi-Fi</h3>
  </div>
</section>
</div>

```

Dodatkowa sekcja z informacjami o udogodnieniach hotelu

### 3.3 Podstrona aquapark

```

<header class="page-header" style="background-image: linear-gradient(135deg, rgba(0, 123, 255, 0.7) 0%, rgba(0, 86, 179, 0.85) 100%); url('=php echo BASE_URL; ?/aquapark_assets/...')">
  <div class="content">
    <h1>Nasz Niezwykły Aquapark</h1>
    <p>Zanurz się w świecie wodnych emocji i niezapomnianej zabawy dla całej rodziny!</p>
  </div>
</header>
<?php

```

Nagłówek strony

```

<div class="page-wrapper">
  <section class="section-title-container">
    <h2>Główne Atrakcje Aquaparku</h2>
  </section>
  <section class="attractions-list horizontal-layout">
    <div class="attraction-card">
      <div class="attraction-img"></div>
      <div class="attraction-content"><h3>Basen z Falą Morską</h3><p>Poczuj prawdziwą morską bryzę i daj się ponieść falom w naszym imponującym basenie.</p>
      <div class="features-tags"><span class="feature-tag">Dla Rodzin</span><span class="feature-tag">Fale co 10 min</span></div>
    </div>
  </section>

```

Sekcja z głównymi atrakcjami i powtarzający się blok zawierający zdjęcia atrakcji i informacje

```

<section class="stats-counter-grid">
  <div class="counter-item"><i class="fas fa-swimming-pool"></i><div class="stat-number">12</div><div class="stat-label">Unikalnych Basenów</div></div>
  <div class="counter-item"><i class="fas fa-tint"></i><div class="stat-number">20</div><div class="stat-label">Zjeżdżalni Wodnych</div></div>
  <div class="counter-item"><i class="fas fa-child"></i><div class="stat-number">3</div><div class="stat-label">Interaktywne Strefy Dziecięce</div></div>
  <div class="counter-item"><i class="fas fa-users"></i><div class="stat-number">1000+</div><div class="stat-label">Zadowolonych Gości Dziennie</div></div>
</section>

```

Sekcja z statystykami aquaparku

```

<section class="section-title-container" id="bilety">
    <h2>Kup Bilety Online</h2>
</section>
<div class="tickets-section">
    <?php
    if(isset($pdo)) {
        try {
            $categoryNameToFilter = 'Aquapark - Bilety';
            $productNameToExclude = 'Rezerwacja Stolika w Restauracji'; // Nazwa produktu do wykluczenia

            $stmtTickets = $pdo->prepare(
                "SELECT p.product_id, p.name, p.description, p.price, p.image_url
                FROM Products p
                JOIN Categories c ON p.category_id = c.category_id
                WHERE c.name = :category_name
                AND p.name != :product_name_to_exclude -- DODANY WARUNEK WYKLUCZAJĄCY
                AND p.is_active = TRUE
                ORDER BY p.price ASC"
            );
            $stmtTickets->bindParam(':category_name', $categoryNameToFilter, PDO::PARAM_STR);
            $stmtTickets->bindParam(':product_name_to_exclude', $productNameToExclude, PDO::PARAM_STR); // Bindowanie nazwy do wykluczenia
            $stmtTickets->execute();
            $tickets = $stmtTickets->fetchAll();

            if ($tickets):
                foreach ($tickets as $ticket):

```

Blok dla zakupu biletów oraz kod php służący do sprawdzania i ewentualnie wykluczania biletów które są niedostępne, czyli wyprzedane

```

                if ($tickets):
                    foreach ($tickets as $ticket):
                >>
                <div class="ticket-card">
                    <h4><?php echo e($ticket['name']); ?></h4>
                    <p><?php echo e($ticket['description']); ?></p>
                    <div class="price"><?php echo number_format($ticket['price'], 2, ',', ' '); ?> PLN</div>
                    <form action="<?php echo BASE_URL; ?>cart_actions.php" method="POST" class="add-to-cart-form">
                        <input type="hidden" name="action" value="add_to_cart">
                        <input type="hidden" name="product_id" value="<?php echo $ticket['product_id']; ?>">
                        <label for="quantity"><?php echo $ticket['product_id']; ?>>Ilość:</label>
                        <input type="number" id="quantity_<?php echo $ticket['product_id']; ?>" name="quantity" value="1" min="1" required>
                        <button type="submit" class="cta-button cta-button-small">Dodaj do koszyka</button>
                    </form>
                </div>
                <?php
                endforeach;
            else:
                >>
                <p>Przepraszamy, obecnie brak dostępnych biletów online dla kategorii "<?php echo e($categoryNameToFilter); ?>". Skontaktuj się z nami.</p>
                <?php
                endif;
            } catch (PDOException $e) {
                error_log("Błąd pobierania biletów w aquapark.php: " . $e->getMessage());
                echo "<p>Wystąpił błąd podczas ładowania oferty biletów. Prosimy spróbować później.</p>";
            }
        } else {
            echo "<p>Błąd konfiguracji bazy danych (brak obiektu PDO).</p>";
        }
    }
    >>
</div>

```

Tutaj kod do generowania kart zakupów konkretnych biletów oraz obsługa błędów

### 3.4 Podstrona restauracji

```

<header class="page-header" style="background-image: linear-gradient(135deg, rgba(0, 123, 255, 0.7) 0%, rgba(0, 86, 179, 0.85) 100%); url('restaurant_assets/lounge.jpg');">
    <div class="content">
        <h1>Kulinarne Doznania w AquaParadise</h1>
        <p>Odkryj bogactwo smaków w naszej eleganckiej restauracji i przytulnym barze. Świeże składniki, autorskie dania.</p>
    </div>
</header>

```

Nagłówek



```

<div class="page-wrapper">
  <section class="section-title-container">
    <h2>Karta Dań Restauracji AquaTaste</h2>
  </section>

  <section class="menu-sections">
    <div class="menu-card horizontal-card">
      <div class="menu-img">
        
      </div>
      <div class="menu-content">
        <h3><i class="fas fa-pepper-hot"></i> Wyśmienite Przystawki</h3>
        <p>Idealny początek kulinarnej podróży, lekkie i pełne smaku kompozycje.</p>
        <ul class="menu-item-list">
          <li><i class="fas fa-utensils"></i>Bruschetta z pomidorami concasse i bazylią - 28 PLN</li>
          <li><i class="fas fa-utensils"></i>Chrupiące kalmary z sosem aioli - 38 PLN</li>
          <li><i class="fas fa-utensils"></i>Carpaccio z polędwicy wołowej z parmezanem i rukolą - 45 PLN</li>
        </ul>
      </div>
    </div>
  </section>
</div>

```

Sekcja menu i powtarzające się bloki z kategoriami jedzenia i daniami

```

<section class="cta-section">
  <a href="restaurant_assets/restaurant_b.php" class="cta-button">Zarezerwuj Stolik Online <i class="far fa-calendar-alt"></i></a>
</section>
</div>

```

Przycisk do zamówienia stolika online

### 3.5 Podstrona spa

```

<header class="page-header" style="background-image: linear-gradient(135deg, rgba(0, 123, 255, 0.7) 0%, rgba(0, 86, 179, 0.85) 100%), url('spa_assets/back.jpg');">
  <div class="content">
    <h1>Oaza Relaksu: Spa & Wellness AquaParadise</h1>
    <p>Zanurz się w świecie spokoju, odprężenia i regeneracji. Czekają na Ciebie luksusowe zabiegi i terapie.</p>
  </div>
</header>

```

Header

```

<div class="page-wrapper">
  <section class="section-title-container">
    <h2>Masza Oferta Zabiegów Spa</h2>
  </section>

  <section class="spa-sections cards-grid">
    <div class="spa-service-card card">
      <div class="spa-img card-image">
        
      </div>
      <div class="spa-content card-content">
        <h3><i class="fas fa-hand-holding-heart"></i> Kojące Terapie Masażu</h3>
        <p>Odprężające masaże dostosowane do Twoich potrzeb, od klasycznych po egzotyczne rytuały.</p>
        <ul class="feature-list">
          <li><i class="fas fa-spa"></i> Masaż Relaksacyjny Klasyczny - 250 PLN</li>
          <li><i class="fas fa-spa"></i> Masaż Lomi Lomi Nui - 320 PLN</li>
          <li><i class="fas fa-spa"></i> Masaż Gorącymi Kamieniami Wulkanicznymi - 350 PLN</li>
        </ul>
        <a href="spa_assets/spa_b.php" class="cta-button cta-button-small" style="margin-top: auto; align-self: flex-start;">Szczegóły i Rezerwacja <i class="fas fa-a
      </div>
    </div>
  </section>
</div>

```

Sekcja ofert spa i powtarzające się bloki z ofertami i informacjami

```

<section class="cta-section">
  <a href="spa_assets/spa_b.php" class="cta-button">Zarezerwuj Swoj Zabieg Juz Dziś <i class="far fa-calendar-check"></i></a>
</section>

```

Przycisk przenoszący do rezerwacji zabiegu

### 3.6 Podstrona opinii (opinions\_view.php)

```
<?php
require_once __DIR__ . '/config/init.php'; //

$page_title = "Opinie Klientów - AquaParadise";
define('INITIAL_OPINIONS_COUNT', 5); // Ile opinii załadować na start

$opinions = [];
$has_more_initial_opinions = false;

try {
    // Zapytanie o początkową partię opinii
    $stmt = $pdo->prepare("
        SELECT o.*, u.first_name, u.last_name, p.name as product_name
        FROM opinions o
        LEFT JOIN users u ON o.user_id = u.user_id
        LEFT JOIN products p ON o.product_id = p.product_id
        WHERE o.status = 'Zaakceptowana'
        ORDER BY o.created_at DESC
        LIMIT :limit
    "); //
    $limit = INITIAL_OPINIONS_COUNT;
    $stmt->bindParam(':limit', $limit, PDO::PARAM_INT);
    $stmt->execute();
    $opinions = $stmt->fetchAll(PDO::FETCH_ASSOC); //

    // Sprawdzenie, czy jest więcej opinii niż początkowa partia
    $stmt_count_total = $pdo->prepare("SELECT COUNT(*) FROM opinions WHERE status = 'Zaakceptowana'");
    $stmt_count_total->execute();
    $total_opinions_count = (int)$stmt_count_total->fetchColumn();
    if ($total_opinions_count > INITIAL_OPINIONS_COUNT) {
        $has_more_initial_opinions = true;
    }
} catch (PDOException $e) {
    error_log("Error fetching opinions: " . $e->getMessage());
}

include BASE_PATH . '/includes/header.php';
?>
```

Pobranie opinii z bazy danych oraz obsługa błędów dla systemu paginacji aby ograniczyć ilość opinii wyświetlanych na raz na stronie

```
<header class="page-header" style="background-image: linear-gradient(135deg, rgba(0, 123, 255, 0.7) 0%, rgba(0, 86, 179, 0.85) 100%)">
    <div class="content">
        <h1>Opinie Naszych Gości</h1>
        <p>Zobacz, co mówią o nas nasi klienci i podziel się swoją opinią!</p>
    </div>
</header>
```

Header

```

<div id="opinionsListContainer" class="opinions-list-new">
  <?php if (empty($opinions)): ?>
    <p class="no-opinions-new">Nie ma jeszcze żadnych opinii. Bądź pierwszy!</p>
  <?php else: ?>
    <?php foreach ($opinions as $opinion): ?>
      <div class="opinion-card-new">
        <div class="opinion-header">
          <strong class="opinion-author">
            <i class="fas fa-user"></i>
            <?php
              if (!empty($opinion['user_id']) && !empty($opinion['first_name'])) { //
                echo htmlspecialchars($opinion['first_name']) . (isset($opinion['last_name']) ? ' ' . $opinion['last_name'][0] . ' ' : ''); //
              } elseif (!empty($opinion['guest_name'])) { //
                echo htmlspecialchars($opinion['guest_name']); //
              } else {
                echo 'Anonimowy Gość'; //
              }
            <?php
          </strong>
          <span class="opinion-date"><?php echo date("d.m.Y", strtotime($opinion['created_at'])); ?></span>
        </div>
        <?php if ($opinion['rating']): ?>
          <div class="opinion-rating-display">
            <?php for ($i = 1; $i <= 5; $i++): ?>
              <i class="fas fa-star"><?php echo ($i <= $opinion['rating']) ? 'filled' : ''; ?></i>
            <?php endfor; ?>
          </div>
        <?php endif; ?>
        <?php if ($opinion['product_name']): ?>
          <p class="opinion-product">Dotyczy: <strong><?php echo htmlspecialchars($opinion['product_name']); ?></strong></p>
        <?php endif; ?>
        <p class="opinion-comment-text"><?php echo nl2br(htmlspecialchars($opinion['comment'])); ?></p>
        <?php if (!empty($opinion['admin_comment'])): ?>
          <div class="admin-reply">
            <p><strong><i class="fas fa-headset"></i> Odpowiedź AquaParadise:</strong></p>
            <p><?php echo nl2br(htmlspecialchars($opinion['admin_comment'])); ?></p>
          </div>
        <?php endif; ?>
      </div>
    <?php endforeach; ?>
  <?php endif; ?>
</div>

```

Ta część kodu służy do wyświetlania pobranych z bazy opinii

### 3.7 Plik load\_more\_opinions.php

Obsługuje przycisk na podstronie opinii rozwijająca dodatkowy panel z formularzem do wprowadzania opinii

### 3.8 Plik submit\_opinion.php

Waliduje i pobiera wprowadzone dane z podstrony load\_more\_opinions.php do bazy danych

### 3.9 Podstrona cart\_view.php

Ogólnie to cały kod php na tej stronie jest od sprawdzania jakie rzeczy aktualnie znajdują się w koszyku dla aktualnie zalogowanego użytkownika, oraz pobierają te dane aby móc je wyświetlić

Później w części z html są wyświetlane te wszystkie elementy

### 3.10 Plik cart\_actions.php

Ten plik obsługuje akcje wykonywane w koszyku takie jak usuwanie produktu itp, tak żeby odpowiednio dane były zmieniane w bazie danych

### 3.11 Podstrona login.php

Część php zawiera obsługę sesji i przydzielania jej odpowiedniej wartości w zależności od typu użytkownika który się zaloguje, oraz druga część php zawiera pobieranie danych z koszyka logującego się użytkownika

```
<?php echo e($page_title); ?></title>
<div class="auth-page-container">
  <div class="return-button-container">
    <a href="index.php" class="return-button"><i class="fas fa-arrow-left"></i> Powrót do strony głównej</a>
  </div>

  <div class="form-container-wrapper">
    <div class="form-container auth-form">
      <h2>Logowanie</h2>

      <?php if ($error_message): ?>
        <div class="flash-message error" style="padding: 10px; margin-bottom: 15px; border: 1px solid var(--border-color); border-left-width: 5px; border-left-color: red;
      <?php endif; ?>
      <?php if ($success_message): ?>
        <div class="flash-message success" style="padding: 10px; margin-bottom: 15px; border: 1px solid var(--border-color); border-left-width: 5px; border-left-color: gre
      <?php endif; ?>

      <form action="<?php echo e($form_action_url); ?>" method="POST" id="loginForm">
        <div class="form-group">
          <label for="login-email"><i class="fas fa-envelope"></i> Adres Email</label>
          <input type="text" id="login-email" name="email_login" value="<?php echo e($_POST['email_login'] ?? ''); ?>" required>
        </div>
        <div class="form-group">
          <label for="login-password"><i class="fas fa-lock"></i> Hasło</label>
          <input type="password" id="login-password" name="password" required>
        </div>
        <div class="form-options">
          <span></span>
          <a href="forgot_password.php" class="form-link">Nie pamiętam hasła</a>
        </div>
        <button type="submit" class="cta-button form-submit-button">Zaloguj się</button>
      </form>
      <p class="form-switch">Nie masz jeszcze konta? <a href="signup.php"><?php echo (empty($form_action_params) ? "?" : http_build_query($form_action_params)); ?>" cla
    </div>
  </div>
</div>

<?php include BASE_PATH . '/includes/footer.php'; ?>
```

Tutaj mamy formularz do logowania się użytkownika

### 3.12 Podstrona signup.php

Ta strona robi prawie to samo co login.php tylko służy do insertowania nowych wartości do bazy i tworzenia nowych użytkowników, oczywiście jest tu walidacja czy wszystkie dane są wypełnione oraz czy wypełnione są poprawnie

```

<form action="php echo e($form_action_url_signup); ?" method="POST" id="signupForm">
  <div class="form-group">
    <label for="register-firstname"><i class="fas fa-user"></i> Imię</label>
    <input type="text" id="register-firstname" name="firstname" value="php echo e($POST['firstname'] ?? ''); ?&gt;" required&gt;
  &lt;/div&gt;
  &lt;div class="form-group"&gt;
    &lt;label for="register-lastname"&gt;&lt;i class="fas fa-user-tag"&gt;&lt;/i&gt; Nazwisko&lt;/label&gt;
    &lt;input type="text" id="register-lastname" name="lastname" value="<?php echo e($POST['lastname'] ?? ''); ?&gt;" required&gt;
  &lt;/div&gt;
  &lt;div class="form-group"&gt;
    &lt;label for="register-email"&gt;&lt;i class="fas fa-envelope"&gt;&lt;/i&gt; Adres Email&lt;/label&gt;
    &lt;input type="email" id="register-email" name="email" value="<?php echo e($POST['email'] ?? ''); ?&gt;" required&gt;
  &lt;/div&gt;
  &lt;div class="form-group"&gt;
    &lt;label for="register-password"&gt;&lt;i class="fas fa-lock"&gt;&lt;/i&gt; Hasło&lt;/label&gt;
    &lt;input type="password" id="register-password" name="password" required&gt;
  &lt;/div&gt;
  &lt;div class="form-group"&gt;
    &lt;label for="register-confirm-password"&gt;&lt;i class="fas fa-redo-alt"&gt;&lt;/i&gt; Potwierdź Hasło&lt;/label&gt;
    &lt;input type="password" id="register-confirm-password" name="confirm_password" required&gt;
  &lt;/div&gt;
  &lt;div class="form-group"&gt;
    &lt;label for="register-phone"&gt;&lt;i class="fas fa-phone"&gt;&lt;/i&gt; Numer Telefonu (opcjonalnie)&lt;/label&gt;
    &lt;input type="tel" id="register-phone" name="phone" value="<?php echo e($POST['phone'] ?? ''); ?&gt;"&gt;
  &lt;/div&gt;
  &lt;div class="form-group checkbox-group"&gt;
    &lt;label class="checkbox-label"&gt;
      &lt;input type="checkbox" name="terms_agree" required &lt;?php echo (isset($POST['terms_agree']) ? 'checked' : ''); ?&gt;
      Akceptuję &lt;a href="regulamin.php" target="_blank" class="form-link"&gt;regulamin serwisu&lt;/a&gt;*
    &lt;/label&gt;
  &lt;/div&gt;
  &lt;div class="form-group checkbox-group"&gt;
    &lt;label class="checkbox-label"&gt;
      &lt;input type="checkbox" name="newsletter_agree" &lt;?php echo (isset($POST['newsletter_agree']) ? 'checked' : ''); ?&gt;
      Chcę otrzymywać informacje o promocjach i nowościach (newsletter)
    &lt;/label&gt;
  &lt;/div&gt;
  &lt;button type="submit" class="cta-button form-submit-button"&gt;Zarejestruj się&lt;/button&gt;
&lt;/form&gt;
&lt;p class="form-switch"&gt;Masz już konto? &lt;a href="login.php"&gt;&lt;?php echo (empty($login_link_params) ? "?" . $login_link_params : ""); ?&gt;" class="form-link"&gt;Zaloguj się!&lt;/p&gt;
&lt;/div&gt;
</pre

```

I oczywiście formularz

### 3.13 Podstrona spa\_b.php

Ta podstrona jest od rezerwacji na sekcji spa. Zawiera ona po prostu formularz i obsługę wprowadzania danych do bazy w ph.

### 3.14 Podstrona restaurant\_b.php

Tak samo jak dla spa zawiera formularz i obsługę wprowadzania danych do rezerwacji stolika w restauracji

### 3.15 Podstrona checkout.php

Jest to strona do której przenosimy się z koszyka gdy chcemy dokonać płatności

Ogólnie kod php polega na tym żeby pobrać wszystkie dane z koszyka, jest też formularz wyświetlający dane zamówienia i ewentualnie możliwość dodania uwagi do zamówienia i wyboru kodów rabatowych i metod płatności. Podczas złożenia zamówienia dane wysyłane są do bazy jako zamówienie a koszyk jest zerowany.

### 3.16 Podstrona hotel\_b.php

To samo co poprzednie strony do rezerwacji

Podstrona index.php w folderze admin

Zawiera index dla panelu admina, pobiera dane i sprawdza które powinny być wyświetlane dla typu użytkownika pobranego z sesji, oraz dodatkowe funkcje.

## Elementy statyczne

### 4.1 struktura dokumentów

Dla każdej z głównych 6 podstron znaleźliśmy powtarzające się elementy takie jak nagłówek, nawigacja i stopka dla których utworzyliśmy oddzielne pliki zawarte w folderze includes do których się odwołujemy na każdej stronie.

### 4.2 style (css)

Mamy tylko jeden plik do stylów podłączany do każdej podstrony, ewentualne konkretne style dla elementów są dorzucane na stronie

## JavaScript

Poza jakimiś pojedynczymi obsługami żeby się strony nie odswieżały przy kliknięciu przycisków albo przenośniki

### 5.1 Płynne przewijanie

```
document.querySelectorAll('a[href^="#"]').forEach(anchor => {
  anchor.addEventListener('click', function (e) {
    const hrefAttribute = this.getAttribute('href');
    if (hrefAttribute.length > 1 && hrefAttribute.startsWith('#')) {
      try {
        const targetElement = document.querySelector(hrefAttribute);
        if (targetElement) {
          e.preventDefault();
          targetElement.scrollIntoView({ behavior: 'smooth' });
        }
      } catch (error) {
        console.error("Błąd w selektorze dla płynnego przewijania:", error);
      }
    }
  });
});
```

## 5.2 Walidacja

Jakoś od 81 linijki aż do samego końca czyli około 714 linijki mamy walidacje dla stron rejestracji, płatności, rezerwacji itp do formularzy aby nie dało się wprowadzić pustych wartości lub złych i żeby wszystko było wypełnione.

## 5.3 Wylogowanie

Od początku do 43 linijki mamy obsługę wiadomości wyświetlanej podczas wylogowania się

# Elementy dynamiczne (PHP)

## 6.1 Obsługa logowania i rejestracji użytkownika

Tu mamy obsługę logowania i rozpoczynania sesji która pobiera między innymi id roli użytkownika dzięki czemu rozróżniamy czy jest on pracownikiem czy nie i jego stanowisko itp.

```
$error_message = '';
$success_message = '';

if (isset($_SESSION['success_message'])) {
    $success_message = $_SESSION['success_message'];
    unset($_SESSION['success_message']);
}

$redirect_url = $_GET['redirect'] ?? 'index.php';
$action_after_login = $_GET['action'] ?? null;
$product_id_after_login = isset($_GET['product_id']) ? (int)$_GET['product_id'] : null;
$details_after_login_json = $_GET['details'] ?? null;

if (isset($_SESSION['user_id'])) {
    header("Location: " . $redirect_url);
    exit;
}

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $email_login = trim($_POST['email_login'] ?? '');
    $password_input = $_POST['password'] ?? '';

    if (empty($email_login) || empty($password_input)) {
        $error_message = "Proszę wypełnić wszystkie pola.";
    } else {
        try {
            $stmt = $pdo->prepare("SELECT u.user_id, u.first_name, u.last_name, u.email, u.password_hash, r.role_name, r.role_id
            FROM Users u
            JOIN Roles r ON u.role_id = r.role_id
            WHERE u.email = :email_login");
            $stmt->bindParam(':email_login', $email_login, PDO::PARAM_STR);
            $stmt->execute();
            $user = $stmt->fetch();

            if ($user && $password_input === $user['password_hash']) { // Porównanie hasła w czystym tekście
                session_regenerate_id(true);

                $_SESSION['user_id'] = $user['user_id'];
                $_SESSION['user_first_name'] = $user['first_name'];
                $_SESSION['user_last_name'] = $user['last_name'];
                $_SESSION['user_email'] = $user['email'];
                $_SESSION['user_role_id'] = (int)$user['role_id'];
                $_SESSION['user_role_name'] = $user['role_name'];

                $_SESSION['Flash_message'] = ['type' => 'success', 'text' => 'Pomyślnie zalogowano. Witaj, ' . htmlspecialchars($user['first_name']) . '!'];
            }
        } catch (PDOException $e) {
            $error_message = "Błąd bazy danych: " . $e->getMessage();
        }
    }
}
```

I tutaj obsługa błędów przy rejestracji oraz insert danych do bazy jako nowy użytkownik

```
if (!$default_role_id) {
    error_log("Krytyczny błąd: Domyślna rola 'Klient' nie znaleziona w bazie danych.");
    $error_message = "Wystąpił błąd systemu podczas rejestracji. Prosimy spróbować później.";
} else {
    $stmt_insert = $pdo->prepare("INSERT INTO Users (first_name, last_name, email, password_hash, phone, role_id)
    VALUES (:firstname, :lastname, :email, :password_to_store, :phone, :role_id)");
    $stmt_insert->bindParam(':firstname', $firstname, PDO::PARAM_STR);
    $stmt_insert->bindParam(':lastname', $lastname, PDO::PARAM_STR);
    $stmt_insert->bindParam(':email', $email, PDO::PARAM_STR);
    $stmt_insert->bindParam(':password_to_store', $password_to_store, PDO::PARAM_STR); // Zapisujemy $password_to_store
    $phone_to_insert = empty($phone) ? $phone : null;
    $stmt_insert->bindParam(':phone', $phone_to_insert, PDO::PARAM_STR);
    $stmt_insert->bindParam(':role_id', $default_role_id, PDO::PARAM_INT);

    if ($stmt_insert->execute()) {
        $new_user_id = $pdo->lastInsertId();
        $_SESSION['success_message'] = "Rejestracja zakończona sukcesem! Możesz się teraz zalogować.";

        $_SESSION['user_id'] = $new_user_id;
        $_SESSION['user_first_name'] = $firstname;
        $_SESSION['user_role_name'] = 'Klient';
    }
}
```

## 6.2 Panel administracyjny

```
$tables_to_display = [];
$allowedTablesForAdminRole6 = ['roles', 'products', 'opinions', 'carts', 'users', 'categories', 'orders', 'orderitems', 'cartitems', 'permissions', 'rolepermissions', 'salaries'];

if ($roleId == 7) $tables_to_display = $all_tables_from_db;
elseif ($roleId == 6) $tables_to_display = array_intersect($all_tables_from_db, $allowedTablesForAdminRole6);
else {
    if (user_has_permission($pdo, $userId, 'manage_products_services')) $tables_to_display = array_merge($tables_to_display, array_intersect($all_tables_from_db, ['products', 'carts', 'cartitems']));
    if (user_has_permission($pdo, $userId, 'manage_opinions')) $tables_to_display = array_merge($tables_to_display, array_intersect($all_tables_from_db, ['opinions']));
    if (user_has_permission($pdo, $userId, 'manage_salaries')) $tables_to_display = array_merge($tables_to_display, array_intersect($all_tables_from_db, ['salaries']));
}
$tables_to_display = array_unique($tables_to_display); $tables_to_display = array_values($tables_to_display);
```

Tu sprawdzamy jakie tabele powinny się wyświetlić w zależności od id roli zalogowanego użytkownika

# Baza danych

## 7.1 Projekt struktury bazy danych

Baza ma za zadanie głównie przechowywać informacje o użytkownikach i ich rolach dla obsługi sesji i panelu admina, informacje o produktach, przechowuje wprowadzane opinie, oraz dane dla koszyka i zamówień

## 7.2 Opis tabel i pól

### Tabela users

Zawiera: imię, nazwisko, email, hasło, numer telefonu, id roli, czy jest użytkownikiem vip, data utworzenia, data ostatniego logowania



Tabela roles

Zawiera: nazwe roli, opis

Tabela rolepermissions

Jest to tabela łącząca tabele role i permission

Tabela permissions

Zawiera: nazwe permisji, opis

Tabela orders

Zawiera: id uzytkownika, email, imie, calkowita ilosc zamowienia, status zamowienia, metoda platnosci, status platnosci, transakcje, czy uzyto kod promocyjny, notka, data utworzenia, data modyfikacji

Tabela opinions

Zawiera: user\_id, product id, ocene, komentarz, status, date utworzenia, komentarz admina

Tabela categories

Zawiera: imie kategorii, opis

Tabela carts

Zawiera: user\_id, session\_id, date utworzenia, date modyfikacji

Tabela cartitems

Zawiera: cart\_id, product\_id, ilosc, cena, informacje, date dodania

## 7.3 Relacje między tabelami

Relacja między users a roles poprzez role\_id w tabeli users

Relacja między roles i permissions poprzez tabelę łączącą  
rolepermissions

Relacja między products i categories poprzez category\_id w tabeli  
products

Relacja między opinions i users poprzez user\_id w tabeli opinions

Relacja między opinions i products poprzez product\_id w tabeli  
opinions

Relacja między carts i users poprzez user\_id w tabeli carts

Relacja między cartitems i cart poprzez cart\_id w cartitems

Relacja między cartitems i products poprzez product\_id w cartitems

## 7.4 Przykładowe zapytania SQL

```
SELECT u.user_id, u.first_name, u.last_name, u.email, u.password_hash,  
r.role_name, r.role_id  
FROM Users u  
JOIN Roles r ON u.role_id = r.role_id  
WHERE u.email = :email_login
```

Pobiera informacje potrzebne do logowania oraz role aby  
identyfikować od razu uprawnienia użytkownika

```
INSERT INTO Users (first_name, last_name, email, password_hash, phone, role_id)  
VALUES  
(:firstname, :lastname, :email, :password_to_store, :phone, :role_id)
```

Podczas rejestracji wprowadza dane do bazy danych użytkowników

```
SELECT COUNT(*) FROM opinions WHERE status = 'Zaakceptowana'
```

Wyświetla dane opinii

## Bezpieczeństwo

### 8.1 Walidacja danych wejściowych

Walidacja odbywa się w pliku script.js dzięki bardzo zaawansowanemu skryptowi walidacji złożonemu z ponad 600 linii kodu

## 8.2 Obsługa sesji i uwierzytelnianie

Uwierzytelnianie użytkownika odbywa się w pliku init.php w folderze config a obsługa sesji jest na każdej stronie wykorzystującej sesje

```
<?php
// login.php
require_once __DIR__ . '/config/init.php';

$page_title = "Logowanie - AquaParadise";

$error_message = '';
$success_message = '';

if (isset($_SESSION['success_message'])) {
    $success_message = $_SESSION['success_message'];
    unset($_SESSION['success_message']);
}

$redirect_url = $_GET['redirect'] ?? 'index.php';
$action_after_login = $_GET['action'] ?? null;
$product_id_after_login = isset($_GET['product_id']) ? (int)$_GET['product_id'] : null;
$details_after_login_json = $_GET['details'] ?? null;

if (isset($_SESSION['user_id'])) {
    header("Location: " . $redirect_url);
    exit;
}
```

## Możliwości rozbudowy

### 9.1 Utworzenie aplikacji mobilnej

### 9.2 System wglądu na kamery basenowe na żywo

### 9.3 Możliwość umówienia na nauki pływania

# Załączniki

## 10.1 Diagram bazy danych

