

**LAPORAN TUGAS MATA KULIAH PEMROGRAMAN  
BERORIENTASI OBJEK PRAKTIK**

**“Reservasi Klinik Kecantikan”**

**T. A. Semester Genap 2023/2024**



**Nama Anggota :**

- 1. Daffa Nayottama (5230411234)**
- 2. Alfina Dwi Rachmawati (5230411288)**
- 3. Nadia Renitasari Fathur Rohman (5230411319)**
- 4. Daphne Aisya Raflian (5230411325)**

**Kelas : VIII**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS SAINS & TEKNOLOGI  
UNIVERSITAS TEKNOLOGI YOGYAKARTA  
2024**

## **I. Pendahuluan**

Sistem ini merupakan aplikasi web untuk Klinik Aurora yang menyediakan layanan reservasi dan informasi tentang layanan yang ditawarkan. Aplikasi ini dibangun menggunakan PHP untuk backend dan HTML/CSS untuk frontend. Dalam laporan ini, kita akan membahas setiap file yang terlibat dalam sistem ini, menjelaskan fungsinya, dan memberikan gambaran umum tentang bagaimana sistem ini bekerja.

## **II. Latar Belakang**

Klinik estetika semakin populer di kalangan masyarakat yang ingin memperbaiki penampilan dan kesehatan kulit. Dengan meningkatnya permintaan akan layanan estetika, penting bagi klinik untuk memiliki platform yang memudahkan komunikasi dengan pelanggan. Proyek ini dikembangkan untuk memenuhi kebutuhan tersebut, dengan menyediakan fitur-fitur yang memungkinkan pengguna untuk mendapatkan informasi, melakukan reservasi, dan memberikan tanggapan. Dengan menggunakan teknologi web modern, aplikasi ini diharapkan dapat menjangkau lebih banyak pelanggan dan meningkatkan kepuasan mereka.

## **III. Penjelasan Fitur**

Aplikasi Klinik Aurora memiliki beberapa fitur utama yang dirancang untuk memberikan kemudahan bagi pengguna:

- Halaman Utama (Hero Section): Menyediakan informasi singkat tentang klinik dan layanan yang ditawarkan, serta tombol untuk melakukan reservasi.
- Tentang Kami: Menyajikan informasi mengenai klinik, visi, dan misi, serta layanan yang tersedia.
- Layanan Kami: Menampilkan berbagai layanan yang ditawarkan oleh klinik, lengkap dengan deskripsi dan gambar untuk memberikan gambaran yang jelas kepada pengguna.

- Form Reservasi: Memungkinkan pengguna untuk melakukan reservasi dengan mengisi form yang mencakup nama, nomor telepon, email, dan alamat. Setelah pengisian form, pengguna akan menerima notifikasi mengenai status reservasi.
- Tanggapan: Pengguna dapat memberikan feedback atau tanggapan mengenai layanan yang diterima melalui form yang disediakan. Tanggapan ini akan disimpan dalam database untuk analisis lebih lanjut.
- Sistem Login dan Registrasi: Memungkinkan pengguna untuk membuat akun dan login untuk mengakses fitur-fitur tertentu, seperti melihat riwayat reservasi atau mengedit profil.
- Notifikasi: Memberikan umpan balik kepada pengguna setelah melakukan tindakan, seperti mengirimkan tanggapan atau reservasi, dengan menggunakan notifikasi yang muncul di halaman.
- Responsiveness: Desain aplikasi yang responsif memastikan bahwa pengguna dapat mengakses aplikasi dengan baik di berbagai perangkat, baik desktop maupun mobile.

#### **IV. Struktur File**

Struktur File Berikut adalah struktur file yang terlibat dalam sistem ini:

- styles.css: File CSS untuk styling tampilan.
- db.class.php: Kelas untuk mengelola koneksi database.
- index.php: Halaman utama aplikasi.
- login.php: Halaman untuk login pengguna.
- logout.php: Halaman untuk logout pengguna.
- register.php: Halaman untuk pendaftaran pengguna baru.
- reservasi.php: Halaman untuk melakukan reservasi.
- tanggapan.php: Halaman untuk mengirimkan tanggapan dari pengguna.
- database\_klinik.sql: File SQL untuk struktur dan data awal database.
- Gambar: Berbagai gambar yang digunakan dalam aplikasi.

#### **V. Penjelasan Kode**

### 3.1. styles.css

```
/* Import Google Fonts */
@import
url('https://fonts.googleapis.com/css2?family=Roboto:wght@400;700&displ
ay=swap');

html {
    scroll-behavior: smooth;
}

/* Reset CSS */
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

/* Typography */
body {
    font-family: 'Roboto', sans-serif;
    line-height: 1.6;
    color: #333;
    background-color: #FFF0F5; /* Background updated */
}

/* Container */
.container {
    width: 90%;
    max-width: 1200px;
    margin: auto;
    overflow: hidden;
}

/* Header */
header {
    background: #FFB6C1; /* Updated header color */
    color: #fff;
    padding: 20px 0;
    position: sticky;
    top: 0;
    z-index: 1000;
}
```

```
.header-container {
  display: flex;
  align-items: center;
  justify-content: space-between;
}

.logo img {
  height: 50px;
}

nav {
  flex: 1;
  text-align: right;
}

.nav-links {
  list-style: none;
  display: flex;
  justify-content: flex-end;
}

.nav-links li {
  margin-left: 20px;
}

.nav-links li a {
  color: #fff;
  text-decoration: none;
  font-weight: 700;
  transition: color 0.3s ease;
}

.nav-links li a:hover {
  color: white;
}

/* Hamburger Menu (Mobile) */
.hamburger {
  display: none;
  flex-direction: column;
  cursor: pointer;
}
```

```
.hamburger span {
  height: 3px;
  width: 25px;
  background: #fff;
  margin-bottom: 4px;
  border-radius: 5px;
  transition: all 0.3s ease;
}

.toggle span:nth-child(1) {
  transform: rotate(-45deg) translate(-5px, 6px);
}

.toggle span:nth-child(2) {
  opacity: 0;
}

.toggle span:nth-child(3) {
  transform: rotate(45deg) translate(-5px, -6px);
}

/* Hero Section */
.hero {
  background: #f4f4f4;
  padding: 60px 0;
}

.hero-container {
  display: flex;
  align-items: center;
  justify-content: space-between;
  flex-wrap: wrap;
}

.hero-text {
  width: 50%;
}

.hero-text h1 {
  font-size: 40px;
  margin-bottom: 20px;
}
```

```
.hero-text h2 {
  font-size: 24px;
  font-weight: 400;
}

.hero-text p {
  font-size: 20px;
  margin-bottom: 30px;
}

.hero-text .btn {
  background: #FFB6C1; /* Updated button color */
  color: white;
  padding: 15px 30px;
  text-decoration: none;
  border-radius: 50px;
  font-weight: bold;
  transition: background 0.3s ease;
}

.hero-text .btn:hover {
  background: #FFE4E1; /* Pink muda saat hover */
  color: #333;
}

.hero-image {
  width: 45%;
  max-width: 500px;
  text-align: center;
}

.hero-image img {
  width: 100%;
  height: auto;
  border-radius: 10px;
}

/* Sections */
section {
  padding: 60px 0;
}
```

```
.about, .services, .contact {
    background: #f4f4f4;
}

.about h2, .services h2, .contact h2 {
    text-align: center;
    margin-bottom: 40px;
    font-size: 32px;
    position: relative;
}

.about h2::after, .services h2::after, .contact h2::after {
    content: '';
    width: 60px;
    height: 4px;
    background: #FFB6C1; /* Updated color */
    display: block;
    margin: 10px auto 0;
    border-radius: 2px;
}

/* Tentang Kami */
.about-content {
    display: flex;
    align-items: center;
    justify-content: space-between;
    flex-wrap: wrap;
}

.about-image {
    width: 45%;
    border-radius: 10px;
    margin-bottom: 20px;
}

.about-content p {
    width: 50%;
    font-size: 18px;
}

/* Layanan */
.services-grid {
    display: flex;
```



```

        justify-content: space-between;
        flex-wrap: wrap;
    }

    .service-item {
        background: #fff;
        padding: 20px;
        width: 30%;
        margin-bottom: 20px;
        text-align: center;
        border-radius: 10px;
        box-shadow: 0 0 10px rgba(0,0,0,0.1);
        transition: transform 0.3s ease, box-shadow 0.3s ease;
    }

    .service-image {
        width: 100%;
        height: 200px;
        object-fit: cover;
        border-radius: 10px 10px 0 0;
    }

    .service-item h3 {
        margin: 20px 0 10px;
        font-size: 24px;
        color: #FFB6C1; /* Updated color */
    }

    .service-item p {
        font-size: 16px;
        color: #666;
    }

    .service-item:hover {
        transform: translateY(-10px);
        box-shadow: 0 10px 20px rgba(0,0,0,0.2);
    }

    /* Form Kontak / Form Reservasi */
    .contact form {
        max-width: 600px;
        margin: auto;
        background: #fff;
    }

```

```
padding: 30px;
border-radius: 10px;
box-shadow: 0 0 10px rgba(0,0,0,0.1);
}

.form-group {
margin-bottom: 20px;
}

.form-group label {
display: block;
margin-bottom: 10px;
font-weight: bold;
}

.form-group input,
.form-group textarea {
width: 100%;
padding: 15px;
border: 1px solid #ccc;
border-radius: 5px;
font-size: 16px;
}

.contact button {
width: 100%;
padding: 15px;
background: #FFB6C1; /* Updated button color */
color: #fff;
border: none;
border-radius: 5px;
font-size: 18px;
cursor: pointer;
transition: background 0.3s ease;
}

.contact button:hover {
background: #FFE4E1;
color: #333;
}

/* Kotak Notifikasi */
.notification-box {
```

```
position: relative;
margin-bottom: 20px; /* Sesuaikan jika diperlukan */
}

/* Notifikasi */
.notification {
position: absolute;
top: -10px; /* Sesuaikan dengan posisi yang diinginkan */
left: 50%;
transform: translateX(-50%);
padding: 10px 20px;
border-radius: 5px;
font-size: 14px;
box-shadow: 0 2px 6px rgba(0, 0, 0, 0.15);
z-index: 999;
width: 90%; /* Sesuaikan lebar sesuai kebutuhan */
max-width: 400px; /* Batas maksimum lebar notifikasi */
text-align: center; /* Pusatkan teks */
opacity: 0;
transition: opacity 0.5s ease, transform 0.5s ease;
}

/* Notifikasi Sukses */
.notification-success {
background-color: #C8E6C9; /* Hijau muda */
color: #256029;
}

/* Notifikasi Gagal */
.notification-error {
background-color: #F8D7DA; /* Merah muda */
color: #721C24;
}

/* Animasi Masuk */
.notification-show {
opacity: 1;
transform: translateX(-50%) translateY(0);
}

/* Animasi Keluar */
.notification-hide {
opacity: 0;
}
```

```

        transform: translateX(-50%) translateY(-20px);
    }

    /* Footer */
    footer {
        background: #FFB6C1; /* Updated footer color */
        color: #fff;
        text-align: center;
        padding: 20px 0;
    }

    .footer-container p {
        font-size: 16px;
    }

    /* Responsiveness */
    @media (max-width: 992px) {
        .hero-container,
        .about-content,
        .services-grid {
            flex-direction: column;
            align-items: center;
        }

        .hero-text,
        .hero-image,
        .about-content p,
        .service-item {
            width: 80%;
        }
    }

    @media (max-width: 768px) {
        .nav-links {
            position: absolute;
            top: 80px;
            right: 0;
            width: 100%;
            height: calc(100vh - 80px);
            background: #FFB6C1; /* Updated nav background color */
            flex-direction: column;
            align-items: center;
            justify-content: center;
        }
    }

```

```
        transform: translateX(100%);
        transition: transform 0.5s ease-in;
    }

    .nav-links.active {
        transform: translateX(0);
    }

    .nav-links li {
        margin: 20px 0;
    }

    .hamburger {
        display: flex;
    }

    .hero-text h1 {
        font-size: 28px;
    }

    .hero-text p {
        font-size: 16px;
    }

    .about-content p,
    .service-item {
        width: 100%;
    }

    .hero-text,
    .hero-image {
        width: 100%;
    }

    .hero-image img {
        max-width: 100%;
    }

    /* Pastikan notifikasi responsif */
    .notification {
        width: 95%;
        max-width: none;
    }
}
```

```
}
```

- Fungsi: Mengatur tampilan dan layout dari aplikasi.
- Isi:
  - Mengimpor font dari Google Fonts.
  - Mengatur gaya dasar untuk elemen HTML, termasuk reset CSS.
  - Menentukan gaya untuk header, footer, dan berbagai bagian seperti hero section, layanan, dan form.
  - Menyediakan responsivitas untuk tampilan di perangkat mobile.

### 3.2. db.class.php

```
<?php
class Database {
    private $host = 'localhost';
    private $db   = 'klinik_aurora';
    private $user = 'root';
    private $pass = 'Dfaaoam17022005.';
    public $conn;

    public function __construct() {
        try {
            $this->conn = new PDO(
                "mysql:host=$this->host;dbname=$this->db",
                $this->user,
                $this->pass
            );
            $this->conn->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
        } catch (PDOException $e) {
            echo "Koneksi gagal: " . $e->getMessage();
        }
    }

    public function getConnection() {
        return $this->conn;
    }
}
?>
```

- Fungsi: Mengelola koneksi ke database MySQL.
- Isi:
  - Kelas Database yang memiliki properti untuk host, nama database, username, dan password.
  - Metode \_\_construct() untuk membuat koneksi menggunakan PDO dan menangani kesalahan jika koneksi gagal.
  - Metode getConnection() untuk mengembalikan objek koneksi.

### 3.3. index.php

```
<?php
session_start();
if (!isset($_SESSION['logged_in']) || $_SESSION['logged_in'] !== true)
{
    header("Location: login.php");
    exit;
}
?>

<!DOCTYPE html>
<html lang="id">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Klinik Aurora</title>
    <!-- Link ke Google Fonts -->
    <link href=
"https://fonts.googleapis.com/css2?family=Roboto:wght@400;700&display=s
wap" rel="stylesheet">
    <!-- Pastikan file styles.css Anda sudah benar path-nya -->
    <link rel="stylesheet" href="css/styles.css">
</head>

<body>
    <!-- Header -->
    <header>
        <div class="container header-container">
            <div class="logo">
                
            </div>
```

```

        <nav>
            <ul class="nav-links">
                <li><a href="#beranda">Beranda</a></li>
                <li><a href="#tentang">Tentang</a></li>
                <li><a href="#layanan">Layanan</a></li>
                <li><a href="#tanggapan">Tanggapan</a></li>
                <!-- Menu Logout di kanan -->
                <li><a href="logout.php" style="margin-left:
20px;">Logout</a></li>
            </ul>
        </nav>
        <div class="hamburger">
            <span></span>
            <span></span>
            <span></span>
        </div>
    </div>
</header>

<!-- Hero Section -->
<section class="hero" id="beranda">
    <div class="container hero-container">
        <div class="hero-text">
            <h1>Because Your Beauty Deserves to Be Ineffable</h1>
            <h2>One Stop Beauty</h2>
            <p>Instant Glowing Kulit Bebas Jerawat</p>
            <a href="reservasi.php" class="btn">RESERVASI
DISINI!</a>
        </div>
        <div class="hero-image">
            
        </div>
    </div>
</section>

<!-- Tentang Kami -->
<section class="about" id="tentang">
    <div class="container about-container">
        <h2>Tentang Kami</h2>
        <div class="about-content">
            
            <p>

```



Aurora Klinik merupakan pionir klinik estetika terbaik di Indonesia yang berfokus pada contouring wajah. Tidak hanya perawatan contouring, Aurora Klinik juga menawarkan perawatan mulai dari rambut, kulit, dan wajah.

```
</p>
</div>
</div>
</section>

<!-- Layanan -->
<section class="services" id="layanan">
  <div class="container services-container">
    <h2>Layanan Kami</h2>
    <div class="services-grid">
      <div class="service-item">
        
        <h3>Aurora Facial</h3>
        <p>
          Aurora Facial adalah suatu metode perawatan
          kulit wajah yang membersihkan pori-pori kulit wajah
          secara menyeluruh. Dengan membersihkan kulit,
          facial dapat membuat kulit wajah bebas dari
          kotoran sehingga dapat mengurangi adanya
          jerawat.
        </p>
      </div>
      <div class="service-item">
        
        <h3>Dermaroller</h3>
        <p>
          Dermaroller adalah suatu alat perawatan yang
          memiliki roller kecil dan ampuh untuk atasi bekas
          jerawat. Roller pada alat tersebut terdapat
          jarum-jarum yang sangat kecil. Penerapan alat ini
          bisa dilakukan pada permukaan wajah yang
          diinginkan agar kembali mulus.
        </p>
      </div>
      <div class="service-item">
```

```

        
        <h3>Face Massage</h3>
        <p>
            Face massage adalah cara untuk mengurangi
garis-garis halus, pori-pori, kerutan & mengencangkan
            kulit wajah. Gerakan pijat wajah merangsang
drainase limfatik, yang meningkatkan sirkulasi darah
            dan mengeluarkan racun, serta mengurangi
pembengkakan.
        </p>
    </div>
</div>
</div>
</section>

<!-- Tanggapan -->
<section class="contact" id="tanggapan">
    <div class="container contact-container">
        <!-- Kotak Kosong untuk Notifikasi -->
        <div class="notification-box">
            <!-- Notifikasi -->
            <?php if (isset($_GET['msg'])): ?>
                <?php if ($_GET['msg'] === 'tanggapan_sukses'): ?>
                    <div class="notification notification-success">
                        Tanggapan berhasil dikirim!
                        <span class="close-btn">&times;</span>
                    </div>
                <?php elseif ($_GET['msg'] === 'tanggapan_gagal'):
?>
                    <div class="notification notification-error">
                        Tanggapan gagal dikirim!
                        <span class="close-btn">&times;</span>
                    </div>
                <?php endif; ?>
            <?php endif; ?>
        </div>

        <form action="tanggapan.php" method="POST">
            <!-- Field Nama -->
            <div class="form-group">
                <label for="nama">Nama:</label>
                <input type="text" id="nama" name="nama" required>

```

```

        </div>
        <!-- Field Email -->
        <div class="form-group">
            <label for="email">Email:</label>
            <input type="email" id="email" name="email"
required>

        </div>
        <!-- Field Pesan -->
        <div class="form-group">
            <label for="pesan">Pesan:</label>
            <textarea id="pesan" name="pesan" rows="5"
required></textarea>
        </div>
        <button type="submit" class="btn">Kirim</button>
    </form>
</div>
</section>

<!-- Footer -->
<footer>
    <div class="container footer-container">
        <p>&copy; 2020 PT. Beauty Gemilang Indonesia</p>
    </div>
</footer>

<!-- JavaScript untuk Menghilangkan Notifikasi secara Otomatis dan
Hapus Parameter msg -->
<script>
    window.addEventListener('load', () => {
        const notifications =
document.querySelectorAll('.notification');
        notifications.forEach(notification => {
            // Tambahkan kelas show untuk animasi masuk
            setTimeout(() => {
                notification.classList.add('notification-show');
            }, 100); // Delay kecil untuk memastikan transition
berjalan

            // Atur waktu sebelum notifikasi mulai menghilang
            setTimeout(() => {
                notification.classList.add('notification-hide');
                // Setelah animasi keluar selesai, sembunyikan
elemen

```

```

        notification.addEventListener('transitionend', ()
=> {
            notification.style.display = 'none';
        });
    }, 5000); // 5 detik

    // Tambahkan event listener untuk tombol tutup
    const closeBtn =
notification.querySelector('.close-btn');
    if (closeBtn) {
        closeBtn.addEventListener('click', () => {
notification.classList.add('notification-hide');
            notification.addEventListener('transitionend',
() => {
                notification.style.display = 'none';
            });
        });
    }
});

    // Hapus parameter msg dari URL
    const url = new URL(window.location.href);
    if (url.searchParams.has('msg')) {
        url.searchParams.delete('msg'); // Hapus parameter msg
        window.history.replaceState(null, '', url.toString());
    // Update URL tanpa refresh
    }
});
</script>
</body>
</html>

```

- Fungsi: Halaman utama yang diakses setelah login.
- Isi:
  - Memulai sesi dan memeriksa apakah pengguna sudah login.
  - Menampilkan header, hero section, tentang kami, layanan, dan form tanggapan.
  - Menyediakan notifikasi untuk tanggapan yang berhasil atau gagal.

### 3.4. login.php

```
<?php
session_start();

// Jika sudah login, redirect ke index
if (isset($_SESSION['logged_in']) && $_SESSION['logged_in'] === true) {
    header("Location: index.php");
    exit;
}

// Import file database
require_once 'db.class.php';

$error = "";

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

    // Koneksi ke DB
    $database = new Database();
    $conn = $database->getConnection();

    try {
        // Ambil data user dari tabel 'users'
        $sql = "SELECT * FROM users WHERE username = :username LIMIT
1";

        $stmt = $conn->prepare($sql);
        $stmt->bindParam(':username', $username);
        $stmt->execute();
        $user = $stmt->fetch(PDO::FETCH_ASSOC);

        if ($user) {
            // Verifikasi password
            // Kita bandingkan input password dengan hash di database
            if (password_verify($password, $user['password'])) {
                // Berhasil login
                $_SESSION['logged_in'] = true;
                $_SESSION['username'] = $user['username'];
                header("Location: index.php");
                exit;
            } else {
```

```

        // Password salah
        $error = "Username atau password salah!";
    }
} else {
    // user tidak ditemukan
    $error = "Username atau password salah!";
}
} catch (PDOException $e) {
    $error = "Error: " . $e->getMessage();
}
}
?>
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Klinik Aurora - Login</title>
    <!-- Link ke CSS yang sama -->
    <link
href="https://fonts.googleapis.com/css2?family=Roboto:wght@400;700&disp
lay=swap" rel="stylesheet">
    <link rel="stylesheet" href="css/styles.css">
    <style>
        body {
            background-color: #f8f9fa;
            font-family: 'Roboto', sans-serif;
            margin: 0;
            padding: 0;
        }
        .login-container {
            max-width: 400px;
            margin: 80px auto;
            background: #fff;
            padding: 30px;
            border-radius: 8px;
            box-shadow: 0 2px 8px rgba(0,0,0,0.1);
        }
        .login-container h2 {
            text-align: center;
            margin-bottom: 20px;
            color: #FFB6C1;

```

```
}  
.form-group {  
  margin-bottom: 15px;  
}  
label {  
  display: block;  
  margin-bottom: 5px;  
  font-weight: bold;  
}  
input[type="text"],  
input[type="password"] {  
  width: 100%;  
  padding: 10px;  
  border: 1px solid #ccc;  
  border-radius: 5px;  
}  
.btn-login {  
  width: 100%;  
  background-color: #FFB6C1;  
  color: #fff;  
  border: none;  
  padding: 12px;  
  border-radius: 5px;  
  cursor: pointer;  
  font-weight: bold;  
}  
.btn-login:hover {  
  background-color: #FFE4E1;  
  color: #333;  
}  
.error {  
  color: red;  
  margin-bottom: 15px;  
  text-align: center;  
}  
p.register-link {  
  margin-top: 10px;  
  text-align: center;  
}  
</style>  
</head>  
<body>
```

```

<div class="login-container">
  <h2>Login Klinik Aurora</h2>
  <?php if(!empty($error)): ?>
    <div class="error"><?= $error ?></div>
  <?php endif; ?>
  <form method="POST" action="">
    <div class="form-group">
      <label for="username">Username:</label>
      <input
        type="text"
        name="username"
        id="username"
        placeholder="Masukkan username"
        required
      >
    </div>
    <div class="form-group">
      <label for="password">Password:</label>
      <input
        type="password"
        name="password"
        id="password"
        placeholder="Masukkan password"
        required
      >
    </div>
    <button type="submit" class="btn-login">Login</button>
  </form>
  <p class="register-link">
    Belum punya akun? <a href="register.php">Daftar di sini</a>.
  </p>
</div>

</body>
</html>

```

- Fungsi: Halaman untuk login pengguna.
- Isi:
  - Memulai sesi dan memeriksa apakah pengguna sudah login.
  - Mengambil input username dan password, memvalidasi, dan memeriksa ke database.



- Jika login berhasil, pengguna diarahkan ke index.php, jika gagal, pesan kesalahan ditampilkan.

### 3.5. logout.php

```
<?php
session_start();
session_destroy();
header("Location: login.php");
exit;
```

- Fungsi: Menghapus sesi pengguna dan mengarahkan kembali ke halaman login.
- Isi:
  - Memulai sesi, menghancurkan sesi, dan mengarahkan pengguna ke login.php.

### 3.6. register.php

```
<?php
session_start();

// Jika user sudah login, redirect ke index
if (isset($_SESSION['logged_in']) && $_SESSION['logged_in'] === true) {
    header("Location: index.php");
    exit;
}

// Import file class database
require_once 'db.class.php';

// Inisialisasi pesan error/sukses
$error = "";
$success = "";

// Cek jika form disubmit
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Ambil data form
    $username = trim($_POST['username']);
    $password = trim($_POST['password']);
    $confirm = trim($_POST['confirm']);
```

```

// Validasi sederhana
if ($username === "" || $password === "" || $confirm === "") {
    $error = "Semua field wajib diisi.";
} elseif ($password !== $confirm) {
    $error = "Password dan konfirmasi password tidak sama!";
} else {
    // Koneksi ke DB
    $database = new Database();
    $conn = $database->getConnection();

    // 1. Cek apakah username sudah ada
    try {
        $sql_check = "SELECT COUNT(*) FROM users WHERE username = :username";
        $stmt_check = $conn->prepare($sql_check);
        $stmt_check->bindParam(':username', $username);
        $stmt_check->execute();
        $count = $stmt_check->fetchColumn();

        if ($count > 0) {
            // Username sudah dipakai
            $error = "Username sudah terdaftar, silakan gunakan username lain.";
        } else {
            // 2. Insert user baru dengan password hash
            $hashed_password = password_hash($password, PASSWORD_DEFAULT);

            $sql_insert = "INSERT INTO users (username, password) VALUES (:username, :password)";
            $stmt_insert = $conn->prepare($sql_insert);
            $stmt_insert->bindParam(':username', $username);
            $stmt_insert->bindParam(':password', $hashed_password);

            if ($stmt_insert->execute()) {
                $success = "Pendaftaran berhasil! Silakan <a href='login.php'>login di sini</a>.";
            } else {
                $error = "Terjadi kesalahan saat menyimpan data user.";
            }
        }
    } catch (PDOException $e) {

```

```

        $error = "Error: " . $_e->getMessage();
    }
}
?>
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Klinik Aurora - Register</title>
    <!-- Link ke CSS yang sama agar nuansa selaras -->
    <link
href="https://fonts.googleapis.com/css2?family=Roboto:wght@400;700&disp
lay=swap" rel="stylesheet">
    <link rel="stylesheet" href="css/styles.css">

    <style>
        body {
            background-color: #f8f9fa;
            font-family: 'Roboto', sans-serif;
            margin: 0;
            padding: 0;
        }
        .register-container {
            max-width: 400px;
            margin: 60px auto;
            background: #fff;
            padding: 30px;
            border-radius: 8px;
            box-shadow: 0 2px 8px rgba(0,0,0,0.1);
        }
        .register-container h2 {
            text-align: center;
            margin-bottom: 20px;
            color: #FFB6C1;
        }
        .form-group {
            margin-bottom: 15px;
        }
        label {
            display: block;

```

```

        margin-bottom: 5px;
        font-weight: bold;
    }
    input[type="text"],
    input[type="password"] {
        width: 100%;
        padding: 10px;
        border: 1px solid #ccc;
        border-radius: 5px;
    }
    .btn-register {
        width: 100%;
        background-color: #FFB6C1;
        color: #fff;
        border: none;
        padding: 12px;
        border-radius: 5px;
        cursor: pointer;
        font-weight: bold;
    }
    .btn-register:hover {
        background-color: #FFE4E1;
        color: #333;
    }
    .error {
        color: red;
        margin-bottom: 15px;
        text-align: center;
    }
    .success {
        color: green;
        margin-bottom: 15px;
        text-align: center;
    }
}
</style>
</head>
<body>
<div class="register-container">
    <h2>Register Klinik Aurora</h2>
    <?php if(!empty($error)): ?>
        <div class="error"><?= $error ?></div>
    <?php endif; ?>
    <?php if(!empty($success)): ?>

```

```
<div class="success"><?= $success ?></div>
<?php endif; ?>

<form method="POST" action="">
  <div class="form-group">
    <label for="username">Username:</label>
    <input
      type="text"
      name="username"
      id="username"
      placeholder="Masukkan username"
      required
    >
  </div>
  <div class="form-group">
    <label for="password">Password:</label>
    <input
      type="password"
      name="password"
      id="password"
      placeholder="Masukkan password"
      required
    >
  </div>
  <div class="form-group">
    <label for="confirm">Konfirmasi Password:</label>
    <input
      type="password"
      name="confirm"
      id="confirm"
      placeholder="Ulangi password"
      required
    >
  </div>
  <button type="submit" class="btn-register">Daftar</button>
</form>

<p style="margin-top: 10px; text-align: center;">
  Sudah punya akun? <a href="login.php">Login di sini</a>.
</p>
</div>
</body>
</html>
```

- Fungsi: Halaman untuk pendaftaran pengguna baru.
- Isi:
  - Memeriksa apakah pengguna sudah login.
  - Mengambil input username dan password, memvalidasi, dan menyimpan ke database jika valid.
  - Menampilkan pesan sukses atau kesalahan berdasarkan hasil pendaftaran.

### 3.7. reservasi.php

```
<?php
// reservasi.php

// Mulai sesi jika diperlukan (untuk notifikasi atau fitur lainnya)
session_start();

// Sertakan kelas Database
require_once 'db.class.php';

// Inisialisasi objek Database
$db = new Database();
$conn = $db->getConnection();

// Inisialisasi variabel pesan
$msg = '';

// Cek apakah form telah disubmit
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Ambil dan sanitasi input
    $nama      = trim($_POST['nama']);
    $telepon   = trim($_POST['telepon']);
    $email     = trim($_POST['email']);
    $alamat    = trim($_POST['alamat']);

    // Validasi sederhana (pastikan semua field terisi)
    if (empty($nama) || empty($telepon) || empty($email) ||
empty($alamat)) {
        // Redirect dengan pesan gagal
        header("Location: reservasi.php?msg=reservasi_gagal");
        exit();
    }
}
```

```

try {
    // Siapkan statement SQL dengan prepared statements
    $stmt = $conn->prepare("INSERT INTO reservasi (nama,
no_telepon, email, alamat) VALUES (:nama, :telepon, :email, :alamat)");

    // Bind parameter dan eksekusi
    $stmt->execute([
        ':nama'      => $nama,
        ':telepon'   => $telepon,
        ':email'     => $email,
        ':alamat'    => $alamat,
    ]);

    // Redirect dengan pesan sukses
    header("Location: reservasi.php?msg=reservasi_sukses");
    exit();
} catch (PDOException $e) {
    // Log error jika diperlukan
    error_log($e->getMessage());

    // Redirect dengan pesan gagal
    header("Location: reservasi.php?msg=reservasi_gagal");
    exit();
}
?>

<!DOCTYPE html>
<html lang="id">

<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1"
/>
    <title>Form Reservasi - Klinik Aurora</title>
    <!-- Bootstrap CSS -->
    <link
href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.m
in.css" rel="stylesheet">
    <!-- Link ke CSS yang sama agar konsisten tampilan -->
    <link rel="stylesheet" href="../../css/styles.css">
    <style>
        /* Agar tinggi halaman 100% untuk flexbox */

```

```
html,
body {
    height: 100%;
    margin: 0;
    padding: 0;
}

/* Jadikan body sebagai flex container */
body {
    display: flex;
    flex-direction: column;
    background-color: #f8f9fa;
    /* warna latar */
    font-family: 'Roboto', sans-serif;
}

/* Wrapper untuk konten utama (agar footer tetap di bawah) */
main {
    flex: 1;
}

/* Header styling */
header {
    background-color: #FFB6C1;
    /* Pink */
    padding: 10px 0;
}

.header-container {
    display: flex;
    align-items: center;
    /* Ubah "space-between" menjadi "center" */
    justify-content: center;
    max-width: 1200px;
    margin: 0 auto;
    padding: 0 15px;
}

.header-container .logo img {
    height: 50px;
    width: auto;
}
```



```
/* Form Container */
.form-container {
    max-width: 600px;
    margin: 40px auto;
    background-color: #ffffff;
    border-radius: 8px;
    box-shadow: 0 2px 8px rgba(0, 0, 0, 0.1);
    padding: 30px;
    position: relative; /* Untuk notifikasi absolute
positioning */
}

.title {
    text-align: center;
    margin-bottom: 1.5rem;
    color: #FFB6C1;
    font-weight: bold;
}

.btn-pink {
    background-color: #FFB6C1;
    border-color: #FFB6C1;
}

.btn-pink:hover {
    background-color: #FFE4E1;
    border-color: #FFE4E1;
}

/* Footer selalu di bawah */
footer {
    background-color: #FFB6C1;
    text-align: center;
    color: #fff;
    padding: 15px 0;
}

.footer-container {
    max-width: 1200px;
    margin: 0 auto;
    padding: 0 15px;
}
```

```
/* Notifikasi */
.notification {
    position: absolute;
    top: 600px;
    right: 150px;
    padding: 10px 20px;
    border-radius: 5px;
    font-size: 14px;
    box-shadow: 0 2px 6px rgba(0, 0, 0, 0.15);
    z-index: 999;
    width: 300px;
    max-width: 400px;
    text-align: center;
    opacity: 0;
    transform: translateY(-20px);
    transition: opacity 0.5s ease, transform 0.5s ease;
}

/* Notifikasi Sukses */
.notification-success {
    background-color: #C8E6C9;
    /* Hijau muda */
    color: #256029;
}

/* Notifikasi Gagal */
.notification-error {
    background-color: #F8D7DA;
    /* Merah muda */
    color: #721C24;
}

/* Animasi Masuk */
.notification-show {
    opacity: 1;
    transform: translateY(0);
}

/* Animasi Keluar */
.notification-hide {
    opacity: 0;
    transform: translateY(-20px);
}
```

```

    /* Responsiveness */
    @media (max-width: 768px) {
        .notification {
            width: 90%;
            left: 50%;
            right: auto;
            transform: translateX(-50%);
        }
    }

    /* Styling Form */
    .form-group {
        margin-bottom: 1rem;
    }

    .form-group label {
        display: block;
        margin-bottom: 0.5rem;
        font-weight: 600;
    }

    .form-group input,
    .form-group textarea {
        width: 100%;
        padding: 0.5rem;
        border-radius: 4px;
        border: 1px solid #ced4da;
    }

    .btn {
        padding: 0.75rem;
        font-size: 1rem;
        cursor: pointer;
    }
</style>
</head>

<body>
    <!-- HEADER -->
    <header>
        <div class="header-container">
            <div class="logo">

```

```

        
    </div>
</div>
</header>

<!-- WRAPPER KONTEN UTAMA -->
<main>
    <div class="form-container">
        <h2 class="title">Form Reservasi</h2>

        <!-- Notifikasi -->
        <div class="notification-box">
            <?php
                if (isset($_GET['msg'])) {
                    switch ($_GET['msg']) {
                        case 'reservasi_sukses':
                            echo '<div class="notification
notification-success">
                                Reservasi berhasil dikirim!
                                <span class="close-btn"
style="cursor:pointer; float:right;">&times;</span>
                                </div>';
                            break;
                        case 'reservasi_gagal':
                            echo '<div class="notification
notification-error">
                                Reservasi gagal dikirim! Silakan coba
lagi.
                                <span class="close-btn"
style="cursor:pointer; float:right;">&times;</span>
                                </div>';
                            break;
                        default:
                            // Tidak melakukan apa-apa
                            break;
                    }
                }
            ?>
        </div>

        <form action="reservasi.php" method="POST">
            <div class="form-group">
                <label for="nama">Nama</label>

```

```

        <input type="text" id="nama" name="nama"
placeholder="Masukkan Nama Anda" required />
    </div>
    <div class="form-group">
        <label for="telepon">No Telepon</label>
        <input type="tel" id="telepon" name="telepon"
placeholder="Masukkan No Telepon Anda" required />
    </div>
    <div class="form-group">
        <label for="email">Alamat Email</label>
        <input type="email" id="email" name="email"
placeholder="Masukkan Email Anda" required />
    </div>
    <div class="form-group">
        <label for="alamat">Alamat</label>
        <textarea id="alamat" name="alamat" rows="3"
placeholder="Masukkan Alamat Anda" required></textarea>
    </div>
    <button type="submit" class="btn btn-pink w-100">
        Kirim
    </button>
</form>
</div>
</main>

<!-- FOOTER -->
<footer>
    <div class="footer-container">
        <p>&copy; <?php echo date("Y"); ?> PT. Beauty Gemilang
Indonesia</p>
    </div>
</footer>

<!-- JavaScript untuk Menghilangkan Notifikasi secara Otomatis
dengan Animasi dan Tombol Tutup -->
<script>
    window.addEventListener('load', () => {
        const notifications =
document.querySelectorAll('.notification');
        notifications.forEach(notification => {
            // Tambahkan kelas show untuk animasi masuk
            setTimeout(() => {
                notification.classList.add('notification-show');

```

```

        }, 100); // Delay kecil untuk memastikan transition
berjalan

        // Atur waktu sebelum notifikasi mulai menghilang
        setTimeout(() => {
            notification.classList.add('notification-hide');
            // Setelah animasi keluar selesai, sembunyikan
elemen
            notification.addEventListener('transitionend', ()
=> {
                notification.style.display = 'none';
            });
        }, 5000); // 5 detik

        // Tambahkan event listener untuk tombol tutup
        const closeBtn =
notification.querySelector('.close-btn');
        if (closeBtn) {
            closeBtn.addEventListener('click', () => {
notification.classList.add('notification-hide');
                notification.addEventListener('transitionend',
() => {
                    notification.style.display = 'none';
                });
            });
        }
    });
});
</script>

</body>

</html>

```

- Fungsi: Halaman untuk melakukan reservasi.
- Isi:
  - Memulai sesi dan mengelola input dari form reservasi.
  - Memvalidasi input dan menyimpan data reservasi ke database.
  - Menampilkan notifikasi sukses atau gagal berdasarkan hasil reservasi.

### 3.8. Tanggapan.php

```
<?php
require_once 'db.class.php';

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $nama = $_POST['nama'];
    $email = $_POST['email'];
    $pesan = $_POST['pesan'];

    // Inisialisasi Database
    $database = new Database();
    $conn = $database->getConnection();

    $sql = "INSERT INTO tanggapan (nama, email, pesan) VALUES (:nama,
:email, :pesan)";
    $stmt = $conn->prepare($sql);

    $stmt->bindParam(':nama', $nama);
    $stmt->bindParam(':email', $email);
    $stmt->bindParam(':pesan', $pesan);

    if ($stmt->execute()) {
        // Jika sukses, redirect ke index dengan pesan dan fragment
        identifier
        header("Location: index.php?msg=tanggapan_sukses#tanggapan");
        exit;
    } else {
        // Jika gagal, redirect dengan pesan error dan fragment
        identifier
        header("Location: index.php?msg=tanggapan_gagal#tanggapan");
        exit;
    }
} else {
    echo "Metode tidak diperbolehkan!";
}
?>
```

- Fungsi: Mengelola pengiriman tanggapan dari pengguna.
- Isi:
  - Mengambil input dari form tanggapan dan menyimpannya ke database.
  - Mengarahkan kembali ke index.php dengan pesan sukses atau gagal.

### 3.9. database\_klinik.sql

- Fungsi: Menyediakan struktur dan data awal untuk database.
- Isi:
  - Berisi perintah SQL untuk membuat tabel-tabel yang diperlukan, seperti:
    - users: Menyimpan informasi pengguna (username dan password).
    - reservasi: Menyimpan data reservasi (nama, no telepon, email, alamat).
    - tanggapan: Menyimpan tanggapan dari pengguna (nama, email, pesan).

```
-- 1. Buat database 'klinik_aurora'

CREATE DATABASE IF NOT EXISTS klinik_aurora;

USE klinik_aurora;

-- 2. Buat tabel 'users' untuk menyimpan data user yang register

CREATE TABLE IF NOT EXISTS users (

    id INT AUTO_INCREMENT PRIMARY KEY,

    username VARCHAR(50) NOT NULL UNIQUE,

    password VARCHAR(255) NOT NULL,

    created_at DATETIME DEFAULT CURRENT_TIMESTAMP

);
```



*-- 3. Buat tabel 'reservasi' untuk menyimpan data form reservasi*

```
CREATE TABLE IF NOT EXISTS reservasi (  
  
    id INT AUTO_INCREMENT PRIMARY KEY,  
  
    nama VARCHAR(100) NOT NULL,  
  
    no_telepon VARCHAR(20) NOT NULL,  
  
    email VARCHAR(100) NOT NULL,  
  
    alamat TEXT NOT NULL,  
  
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP  
);
```

*-- 4. Buat tabel 'tanggapan' untuk menyimpan data feedback / komentar*

```
CREATE TABLE IF NOT EXISTS tanggapan (  
  
    id INT AUTO_INCREMENT PRIMARY KEY,
```

```
nama VARCHAR(100) NOT NULL,

email VARCHAR(100) NOT NULL,

pesan TEXT NOT NULL,

created_at DATETIME DEFAULT CURRENT_TIMESTAMP

);

-- (Optional) Lihat semua tabel

cSHOW TABLES;

-- (Optional) Cek struktur masing-masing tabel

DESC users;

SELECT * FROM users;

DESC reservasi;

SELECT * FROM reservasi;

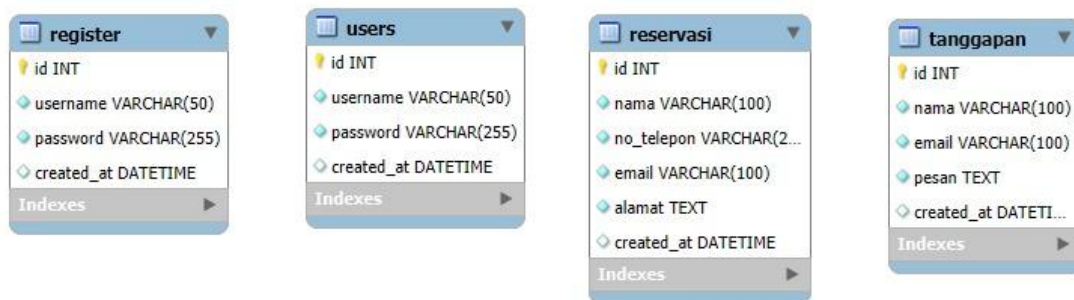
DESC tanggapan;

DELETE FROM tanggapan;
```

```
SELECT * FROM tanggapan;
```

```
SET SQL_SAFE_UPDATES = 0;
```

```
SET SQL_SAFE_UPDATES = 1;
```



- Tabel users menyimpan informasi pengguna yang terdaftar, termasuk username dan password yang di-hash.
- Tabel reservasi menyimpan data reservasi yang dilakukan oleh pengguna, termasuk nama, nomor telepon, email, dan alamat.
- Tabel tanggapan menyimpan tanggapan yang diberikan oleh pengguna, termasuk nama, email, dan pesan.

## VI. Kesimpulan

Sistem reservasi Klinik Aurora adalah aplikasi web yang terintegrasi dengan baik, memungkinkan pengguna untuk melakukan reservasi dan memberikan tanggapan. Dengan menggunakan PHP untuk backend dan HTML/CSS untuk frontend, sistem ini memberikan

pengalaman pengguna yang baik. Setiap file memiliki fungsinya masing-masing, dan bersama-sama membentuk aplikasi yang fungsional dan responsif.

## **VII. Pembagian Tugas Kelompok**

1. Mendesain web dan membuat web : Daffa Nayottama & Daphne Aisya Raflian
2. Membuat Database & UML : Alfina Dwi Rachmawati & Nadia Renitasari Fathur Rohman
3. Menyusun laporan dan PPT : Daffa Nayottama, Alfina Dwi Rachmawati, Nadia Renitasari Fathur Rohman, Daphne Aisya Raflian