

LAMPIRAN

1. Kartu Monitoring Proposal



KARTU MONITORING BIMBINGAN
MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

PROPOSAL

MAHASISWA: SAHRUL	Pembimbing I : Marlina,S.Kom.,M.Kom
NIM : 220280190	Pembimbing II : Muh.BASRI,ST., MT
Judul Skripsi : SISTEM INFORMASI PEMESANAN TIKET KERETA API BERBASIS WEB DENGAN METODE WATERWALL	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 <i>Muhammad Syarif</i>		Konsultasi 1 - Perbaiki Format Penulisan - Cari referensi minimal tahun 2018. - Beri penjelasan pd Desain Sistem.	10/1/24 <i>Muhammad Syarif</i>
Konsultasi 2		Konsultasi 2 - Perbaiki Daftar Pustaka	<i>Muhammad Syarif</i>
Konsultasi 3		Konsultasi 3 - Perbaiki Daftar Pustaka	17/1/24. <i>Muhammad Syarif</i>
Konsultasi 4 <i>ACC</i>		Konsultasi 4 <i>ACC</i>	<i>Muhammad Syarif</i>
Konsultasi 5		Konsultasi 5	

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

2. Kartu Monitoring Skripsi



KARTU MONITORING BIMBINGAN
MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

HASIL

MAHASISWA: SAHRUL	Pembimbing I : Marlina,S.Kom.,M.Kom
NIM : 220280190	Pembimbing II : Muh.BASRI,ST., MT
Judul Skripsi : SISTEM INFORMASI PEMESANAN TIKET BUS BERBASIS WEB PO. ANUGRAH TRANS	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 Perbaiki Daftar Pustaka		Konsultasi 1	
Konsultasi 2 Semua bhs Asing dicetak Ming		Konsultasi 2	
Konsultasi 3		Konsultasi 3	
Konsultasi 4 Ace Ujian Hasil.		Konsultasi 4 Ganti Hasil	
Konsultasi 5		Konsultasi 5	

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa setiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

3. Kartu Monitoring Skripsi



KARTU MONITORING BIMBINGAN
MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

SKRIPSI

MAHASISWA: SAHRUL	Pembimbing I : Marlina,S.Kom.,M.Kom
NIM : 220280190	Pembimbing II : Muh.BASRI,ST., MT
Judul Skripsi : SISTEM INFORMASI PEMESANAN TIKET BUS BERBASIS WEB PO. ANUGRAH TRANS	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 - Bab 4, jelaskan cara pembuatan & desain Aplikasi		Konsultasi 1	
Konsultasi 2 - Jelaskan pengelompokan tabel data base & diagram.		Konsultasi 2	
Konsultasi 3 - Relasi tabel di bab IV		Konsultasi 3	
Konsultasi 4 Acc Turun		Konsultasi 4 Acc Ujian Turun	
Konsultasi 5		Konsultasi 5	

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

4. Dokumentasi Penelitian



BUS PARIWISATA		RUTE PINRANG / MALILI	
PINRANG, SIDRAP, PAREPARE		KHUSUS BUS MALILI / BUS PARI	
DAFTAR TARIF		TARIF HARGA	
BULUKUMBA	Rp. 5.000.000	PINRANG KE MALILI	Rp. 150.000
SINJAI	Rp. 4.000.000	PINRANG KE BONEBONE	Rp. 150.000
JENEPONTO	Rp. 3.500.000	PINRANG KE MASAMBA	Rp. 150.000
TORAJA	Rp. 3.500.000	PINRANG KE PALOPO	Rp. 150.000
MALINO	Rp. 4.000.000	PINRANG KE BELOPA	Rp. 150.000
TOPE JAWA	Rp. 3.000.000	PINRANG KE SIWA	Rp. 100.000
GRAND MALL	Rp. 2.500.000	RAPPANG KE MALILI	Rp. 150.000
BANTIMURUNG	Rp. 2.500.000	SIDRAP KE MALILI	Rp. 150.000
BANDARA	Rp. 2.350.000	TAMRU TEDONG KE MALILI	Rp. 150.000
MAKASSAR	Rp. 2.500.000	TURUMPAGE KE MALILI	Rp. 150.000
MAJENE	Rp. 2.500.000		
PUNGAK	Rp. 1.500.000		
BUGIS	Rp. 2.500.000		
DIANA BARRU	Rp. 2.000.000		
POLEWALI MANDAR	Rp. 1.000.000		
PALOPO	Rp. 3.000.000		

5. Index.php (form login)

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="utf-8" />
  <meta name="viewport"
content="width=device-width, initial-
scale=1" />
  <title>Pemesanan Bis</title>
  <link
href="https://cdn.jsdelivrivr.net/npm/bootstrap
@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
QWTKZyjpPEjISv5WaRU9OFeRpok6Yctn
YmDr5pNlyT2bRjXh0JMhY6hW+ALEwI
H" crossorigin="anonymous" />
  <link rel="stylesheet"
href="https://cdn.jsdelivrivr.net/npm/bootstrap
-icons@1.11.3/font/bootstrap-
icons.min.css" />
  <script defer
src="https://cdn.jsdelivrivr.net/npm/alpinejs@
3.14.0/dist/cdn.min.js"></script>
</head>
<style>
* {
  padding: 0;
  margin: 0;
}

body {
  max-height: 100vh;
}

img {
  opacity: 0.5;
  width: 100%;
  height: 100vh;
}

img:hover {
```

```
  opacity: 1;
}
</style>

<body>
  <section id="login">
    <div class="card text-bg-dark g-0 text-
center">
      
      <div class="card-img-overlay"
style="margin-top: 5rem;">
        <div class="row d-flex justify-content-
center">
          <div class="col-md-4 col-md-4 col-
12">
            <h2 class="fw-bold fst-italic mb-
4">ANUGRAH <span class="text-
info">TRANS</span></h2>
            <div class="card-body p-5 border
border-warning text-center">
              <div class="card-title text-center">
                <h3 class="pb-3">
                  LOGIN
                </h3>
                <?php
                if (isset($_GET["msg"])) {
                  if ($_GET["msg"] == 1) {
                    echo '<div class="alert alert-
success" role="alert">Create Account
Success</div>';
                  } else if ($_GET["msg"] == 2) {
                    echo '<div class="alert alert-
danger" role="alert">Password Or Email Is
Wrong</div>';
                  } else if ($_GET["msg"] == 3) {
                    echo '<div class="alert alert-
warning" role="alert">Email Is Already
u</div>';
                  } else {
                    echo '';
                  }
                }
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </section>
</body>
</html>
```



```

integrity="sha384-
YvpcrYf0tY3lHB60NNkmXc5s9fDVZLES
aAA55NDzOxhy9GkcIdslK1eN7N6jleHz"
crossorigin="anonymous"></script>

```

```
</body>
```

```
</html>
```

6. Action.php

```

<?php
include("assets/conn.php");
session_start();
if (isset($_POST["submit"])) {
    $id = $_POST["id"];
    $nama = $_POST["nama"];
    $email = $_POST["email"];
    $no_hp = $_POST["no_hp"];
    $tanggal = $_POST["tanggal"];
    $waktu = $_POST["waktu"];
    $waktu = new DateTime($waktu);
    $waktu = $waktu->format("H:i:s");
    $order = date("Y-m-d H:i:s");
    $Keberangkatan = $tanggal . " " . $waktu;
    if ($data) {
        header("location:users/index.php");
    } else {
        header("location:users/index.php");
    }
}

if (isset($_POST["login"])) {
    $email = $_POST["email"];
    $password = $_POST["password"];
    $data = $conn->query("SELECT * FROM
tbl_users WHERE email='$email' and
password='$password'");
    if ($data->num_rows === 1) {
        $data = $data->fetch_assoc();
        if ($data["kode"] == 2) {
            $_SESSION["admin"] = $data;
            $_SESSION["users"] = "";
            header("location:admin/index.php");
        } else {
            $_SESSION["admin"] = "";
            $_SESSION["users"] = $data;

```

```

        header("location:users/index.php");
    }
} else {
    header("location:index.php?msg=2");
}
}

if (isset($_POST["signup"])) {
    $email = $_POST["email"];
    $nama = $_POST["nama"];
    $password = $_POST["password"];
    $no_hp = $_POST["no_hp"];
    $data = $conn->query("SELECT * FROM
tbl_users WHERE email='$email'");
    if ($data->num_rows === 0) {
        $conn->query("INSERT INTO tbl_users
VALUES ('', '$nama', '$email', '$password'
,1,'$no_hp')");
        header("location:index.php?msg=1");
    } else {
        header("location:index.php?msg=3");
    }
}

if (isset($_POST["logout"])) {
    if ($_SESSION["users"]["id"] ==
$_POST["users"]) {
        $_SESSION["users"] = "";
        session_destroy();
        header("location:index.php");
    } else {
        $_SESSION["admin"] = "";
        session_destroy();
        header("location:index.php");
    }
}
}

```



```

if (isset($_POST["complete"])) {
    $id = $_POST["id"];
    $data = $conn->query("SELECT quantity,
id_schedule, slot FROM tbl_order INNER
JOIN          tbl_schedule          ON
tbl_schedule.id=tbl_order.id_schedule
WHERE tbl_order.id='$id' and status<3");
    $data = $data->fetch_assoc();
    $quantity = $data["quantity"] +
$data["slot"];
    $id_schedule = $data["id_schedule"];
    $conn->query("UPDATE      tbl_schedule
SET      slot      ='$quantity'      WHERE
id='$id_schedule'");
    $conn->query("UPDATE      tbl_order      SET
status=3 WHERE id='$id'");
    header("location:admin/index.php");
}

```

```

if (isset($_POST["deleteOrder"])) {
    $id = $_POST["id"];
    $data = $conn->query("SELECT quantity,
id_schedule, slot FROM tbl_order INNER
JOIN          tbl_schedule          ON
tbl_schedule.id=tbl_order.id_schedule
WHERE tbl_order.id='$id' and status<3");
    $data = $data->fetch_assoc();
    $quantity = $data["quantity"] +
$data["slot"];
    $id_schedule = $data["id_schedule"];
    $conn->query("UPDATE      tbl_schedule
SET      slot      ='$quantity'      WHERE
id='$id_schedule'");
    $conn->query("DELETE FROM tbl_order
WHERE id='$id'");
    header("location:admin/index.php");
}

```

```

if (isset($_POST["cancel"])) {
    $id = $_POST["id"];
    $data = $conn->query("SELECT quantity,
id_schedule, slot FROM tbl_order INNER

```

```

JOIN          tbl_schedule          ON
tbl_schedule.id=tbl_order.id_schedule
WHERE tbl_order.id='$id' and status<3");
    $data = $data->fetch_assoc();
    $quantity = $data["quantity"] +
$data["slot"];
    $id_schedule = $data["id_schedule"];
    $conn->query("UPDATE      tbl_schedule
SET      slot      ='$quantity'      WHERE
id='$id_schedule'");
    $conn->query("UPDATE      tbl_order      SET
status=4 WHERE id='$id'");
    header("location:admin/index.php");
}

```

```

if (isset($_POST["cancelTicket"])) {
    $id = $_POST["idOrder"];
    $data = $conn->query("SELECT quantity,
id_schedule, slot FROM tbl_order INNER
JOIN          tbl_schedule          ON
tbl_schedule.id=tbl_order.id_schedule
WHERE tbl_order.id='$id' and status<3");
    $data = $data->fetch_assoc();
    $quantity = $data["quantity"] +
$data["slot"];
    $id_schedule = $data["id_schedule"];
    $conn->query("UPDATE      tbl_schedule
SET      slot      ='$quantity'      WHERE
id='$id_schedule'");
    $conn->query("UPDATE      tbl_order      SET
status=4 WHERE id='$id'");
    header("location:users/index.php");
}

```

```

if (isset($_POST["ubahSchedule"])) {
    $id = $_POST["id"];
    $arah = $_POST["arah"];
    $price = $_POST["price"];
    $jadwal = strtoupper($_POST["jadwal"]);
    $tipe = $_POST["tipe"];
    $slot = $_POST["slot"];
    if ($arah) {
        $set[] = "arah='$arah'";

```

```

    }
    if ($price) {
        $set[] = "price='$price'";
    }
    if ($jadwal) {
        $set[] = "jadwal='$jadwal'";
    }
    if ($tipe) {
        $set[] = "tipe='$tipe'";
    }
    if ($slot) {

```

7. Midtrans.php

```

<?php
require      dirname(__FILE__)
'/assets/Midtrans-php-master/Midtrans.php';
include("assets/conn.php");
\Midtrans\Config::$serverKey = "SB-Mid-
server-FJARwt_FNRmtUinwinGt2rDf";
\Midtrans\Config::$isProduction = false;
\Midtrans\Config::$isSanitized = true;
\Midtrans\Config::$is3ds = true;

if ($_POST["tmd"] != null) {
    $argv = $_POST["tmd"];
    $no_kursi = implode(",", $argv);
    $quantity = sizeof($_POST["tmd"]);
} else {
    $quantity = $_POST["quantity"];
    $no_kursi = null;
}
$total = ($quantity * $_POST["price"]);
$nama = $_POST["nama"];
$email = $_POST["email"];
$no_hp = $_POST["no_hp"];
$id = $_POST["id"];
$waktu = $_POST["waktu"];
$tanggal = $_POST["tanggal"];
$order_at = date("Y-m-d H:i:s");
$id_order = rand();

```

```

    $set[] = "slot='$slot'";
}
$conn->query("UPDATE      tbl_schedule
SET " . implode(",", $set) . "WHERE
id='$id'");
if ($tipe == 2 && $slot > 0) {
    $conn->query("UPDATE      tbl_schedule
SET slot='$slot' WHERE tipe=2");
}
header("location:admin/index.php");
}

```

```

$tanggal = $tanggal . " " . $waktu;
$conn->query("INSERT INTO      tbl_order
VALUES ('$id_order', '$nama', '$email',
'$no_hp','$id','$quantity',
'$order_at','$tanggal','$no_kursi', 1)");

```

```

$data = $conn->query("SELECT * FROM
tbl_schedule WHERE id='$id'");
if ($data->num_rows) {
    $ds = $data->fetch_assoc();
    $stok = $ds["slot"] - $quantity;
    if ($ds["tipe"] == 1 && $ds["slot"] != 0) {
        $conn->query("UPDATE      tbl_schedule
SET slot='$stok' WHERE id='$id'");
    } else {
        $conn->query("UPDATE      tbl_schedule
SET slot='$stok' WHERE tipe=2");
    }
    $params = array(
        'transaction_details' => array(
            'order_id' => $id_order,
            'gross_amount' => $total,
        ),
        'customer_details' => array(
            'first_name' => $_POST['nama'],
            'email' => $_POST['email'],
            'phone' => $_POST['no_hp'],
        ),
    );
}

```

```

);
$snapToken =
\Midtrans\Snap::getSnapToken($params);
echo $snapToken;
8. Payment.php
<?php
require dirname(__FILE__) .
'/assets/Midtrans-php-master/Midtrans.php';
include("assets/conn.php");
\Midtrans\Config::$serverKey = "SB-Mid-
server-FJARwt_FNRmtUinwinGt2rDf";
\Midtrans\Config::$isProduction = false;
\Midtrans\Config::$isSanitized = true;
\Midtrans\Config::$is3ds = true;
$id = $_POST["id"];
$data = $conn->query("SELECT *,
tbl_order.id AS IDS FROM tbl_order
INNER JOIN tbl_schedule ON
tbl_order.id_schedule=tbl_schedule.id
WHERE tbl_order.id='$id'");
if ($data->num_rows === 1) {
    $data = $data->fetch_assoc();
    $params = array(
9. CompletePayment.php
<?php
include("../assets/conn.php");
if (isset($_POST["completePayment"])) {
    $id = $_POST["id"];
    $data = $conn->query("SELECT *,
tbl_order.id AS IDS FROM tbl_order INNER
JOIN tbl_schedule ON
tbl_order.id_schedule=tbl_schedule.id
WHERE tbl_order.id='$id'");
    $data = $data->fetch_assoc();
    $jadwal = explode(" ",
$data["keberangkatan"]);
    $time = $jadwal[1];
    $tanggal = $jadwal[0];
} ?>
<!DOCTYPE html>
<html lang="en">

} else {
    header("locatino:index.php");
}

'transaction_details' => array(
    'order_id' => $id . "-" . rand(100, 999),
    'gross_amount' => $data["quantity"] *
$data["price"],
    ),
    'customer_details' => array(
        'first_name' => $data['nama'],
        'email' => $data['email'],
        'phone' => $data['no_hp'],
    ),
);
$snapToken =
\Midtrans\Snap::getSnapToken($params);
echo $snapToken;
} else {
    header("locatino:index.php");
}

<head>
<meta charset="utf-8" />
<meta name="viewport"
content="width=device-width, initial-
scale=1" />
<title>Pemesanan Bis</title>
<link
href="https://cdn.jsdelivrivr.net/npm/bootstrap
@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
QWTKZyjpPEjISv5WaRU9OFeRpok6Yctn
YmDr5pNlyT2bRjXh0JMHjY6hW+ALEwI
H" crossorigin="anonymous" />
<link rel="stylesheet"
href="https://cdn.jsdelivrivr.net/npm/bootstrap
-icons@1.11.3/font/bootstrap-
icons.min.css" />

```

```

<script defer
src="https://cdn.jsdelivr.net/npm/alpinejs@
3.14.0/dist/cdn.min.js"></script>
<script type="text/javascript"
src="https://app.sandbox.Midtrans.com/sna
p/snap.js" data-client-key="SB-Mid-server-
FJARwt_FNRmtUinwinGt2rDf"></script>
</head>

<body>
<div class="card ">
<div class="container d-flex justify-
content-center align-items-center"
style="min-height:100vh;">
<div class="col-md-6">
<div class="card-title text-center">
<h2>Detail Pembayaran</h2>
</div>
<div class="card-body">
<div class="card shadow p-3 mb-2 bg-
body-tertiary rounded">
<div class="card-body">
<div id="detail">
<form id="payment" class="text-
center" method="post">
<label for="quantity">Data
Diri</label>
<input type="hidden" name="id"
value="<?=$data["IDS"]; ?>">
<input required type="text"
name="nama" id="nama" class="form-
control my-2" placeholder="Masukkan
Nama Anda" value="<?=$data["nama"];
?>">
<input required type="text"
name="email" id="email" class="form-

```

```

control my-2" placeholder="Masukkan
Email Anda" value="<?=$data["email"];
?>">
<input required type="text"
name="no_hp" id="no_hp" class="form-
control my-2" placeholder="Masukkan
Nomor Handphone" value="<?=$
data["no_hp"]; ?>">
<input type="time"
name="waktu" class="form-control my-2"
value="<?(isset($time) ? $time : ""); ?>">
<input required type="text"
name="price" placeholder="Harga"
class="form-control my-2" value="<?=$
data["price"]; ?>">
<input type="date"
name="tanggal" class="form-control my-2"
required value="<?=$tanggal; ?>">
<button type="submit"
class="btn btn-success">Done</button>
<a href="index.php" class="btn
btn-danger">Cancel</a>
</form>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<script src="../assets/js/test.js?v=<%= new
Date() %>"></script>
</body>

</html>

```

10. Index.php (User)

```

<?php include("../assets/conn.php");
session_start();

```

```

if (empty($_SESSION['users']) ||
$_SESSION["users"] == "") {
header("location:../index.php");
}

```

```

if (isset($_GET["status"])) {
    $status = $_GET["status"];
    $id = $_GET["id"];
    if (explode("-", $id)) {
        $id = explode("-", $id);
        $id = $id[0];
    }
    if ($status == "settlement") $status = 1;
    else if ($status == "pending") $status = 0;
    else $status = 2;
    $conn->query("UPDATE tbl_order SET
status='$status' WHERE id='$id'");
}

```

```

$db = $conn->query("SELECT * FROM
tbl_schedule");
$totalTicket = 0;
while ($rows = $db->fetch_assoc()) {
    $id = $rows["id"];
    $dv = $conn->query("SELECT
quantity,slot FROM tbl_order INNER JOIN
tbl_schedule WHERE status=3 AND
tbl_schedule.id='$id'");
    while ($row = $dv->fetch_assoc()) {
        if ($rows["tipe"] == 1) {
            if ($row["slot"] == 0) {
                $totalTicket += $row["quantity"];
                if ($totalTicket == 35) {
                    $conn->query("UPDATE
tbl_schedule SET slot=35 WHERE
id='$id'");
                }
            }
        } else {
            if ($row["slot"] == 0) {
                $totalTicket += $row["quantity"];
                if ($totalTicket == 5) {
                    $conn->query("UPDATE
tbl_schedule SET slot=5 WHERE tipe=2");
                }
            }
        }
    }
}

```

```

    }
}
}

?>

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

<head>
    <meta charset="utf-8" />
    <meta name="viewport"
content="width=device-width, initial-
scale=1" />
    <title>Pemesanan Bis</title>
    <link
href="https://cdn.jsdelivrivr.net/npm/bootstrap
@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
QWTKZyjpPEjISv5WaRU9OFeRpok6Yctn
YmDr5pNlyT2bRjXh0JMhY6hW+ALEwI
H" crossorigin="anonymous" />
    <link rel="stylesheet"
href="https://cdn.jsdelivrivr.net/npm/bootstrap
-icons@1.11.3/font/bootstrap-
icons.min.css" />
    <script defer
src="https://cdn.jsdelivrivr.net/npm/alpinejs@
3.14.0/dist/cdn.min.js"></script>
    <link rel="stylesheet"
href="https://cdn.jsdelivrivr.net/npm/bootstrap
-icons@1.11.3/font/bootstrap-
icons.min.css">
</head>
<style>
    * {
        padding: 0;
        margin: 0;
    }
</style>

<body>

```

```

<nav class="navbar navbar-expand-lg
shadow p-3 mb-2 bg-body-tertiary rounded
fixed-top opacity-75" data-bs-
theme="dark">
  <div class="container d-flex align-items-
center">
    <a class="navbar-brand fst-italic fw-
bold" href="#">ANUGRAH <span
class="text-info">TRANS</span></a>
    <button class="navbar-toggler"
type="button" data-bs-toggle="collapse"
data-bs-target="#navbarNavAltMarkup"
aria-controls="navbarNavAltMarkup" aria-
expanded="false" aria-label="Toggle
navigation">
      <span class="navbar-toggler-
icon"></span>
    </button>
    <div class="collapse navbar-collapse"
id="navbarNavAltMarkup">
      <div class="navbar-nav ms-auto">
        <li class="nav-item">
          <a class="nav-link"
href="#hero">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link"
href="#schedule">Jadwal</a>
        </li>
        <li class="nav-item">
          <a class="nav-link"
href="#contact">Kontak Kami</a>
        </li>
        <li class="nav-item dropdown">
          <form action="../action.php"
method="post">
            <a class="nav-link dropdown-
toggle" role="button" data-bs-
toggle="dropdown" aria-
expanded="false">Akun </a>

```

```

      <ul class="dropdown-menu">
        <input type="hidden"
name="users" value="<?=$_SESSION["users"]["id"]; ?>">
        <li><input type="submit"
name="logout" value="Log out"
class="dropdown-item"></li>
      </ul>
    </form>
  </li>
</div>
</div>
</div>
</nav>

<main>
  <section id="hero" class="shadow mb-3
bg-body-tertiary rounded">
    <div class="card text-bg-dark g-0 text-
center">
      
      <div class="card-img-overlay position-
absolute top-50 start-50 translate-middle">
        <h1 class="card-body mt-5">
          ANUGRAH <span class="text-
info">TRANS</span>
        </h1>
        <p>Kami Siap Mengantar Anda
Kemanapun</p>
        <a href="#schedule" class="btn btn-
outline-info px-3 py-2">Order Ticket
Now</a>
      </div>
    </div>
  </section>

  <section id="schedule">
    <?php
      $nama = $_SESSION["users"]["nama"];

```



```

        <div class="modal-
body">
            <input
type="hidden" name="id" value="<?=$row["id"]; ?>">
            <input required
type="text" placeholder="Nama"
name="nama" class="form-control my-2"
value="<?=$ _SESSION["users"]["nama"];
?>">
            <input required
type="text" placeholder="Email"
name="email" class="form-control my-2"
value="<?=$ _SESSION["users"]["email"];
?>">
            <input required
type="text" placeholder="Nomor
Handphone" name="no_hp" class="form-
control my-2" value="<?=$
_SESSION["users"]["no_hp"]; ?>">
            <textarea
name="msg" class="form-control"
placeholder="Optional"
rows="2"></textarea>
            <?php if
($row["tipe"] == 1) { ?>
                <h4 class="my-
2">Jadwal Keberangkatan</h4>
                <div class="form-
check form-check-inline">
                    <input
class="form-check-input" type="radio"
name="jadwal" id="inlineRadio1"
value="10:00" checked>
                    <label
class="form-check-label"
for="inlineRadio1">10:00</label>
                </div>
                <div class="form-
check form-check-inline">
                    <input
class="form-check-input" type="radio"

```

```

name="jadwal" id="inlineRadio2"
value="20:00">
                    <label
class="form-check-label"
for="inlineRadio2">20:00</label>
                </div>
                <?php } ?>
            </div>
            <div class="modal-
footer d-flex justify-content-center">
                <button
type="button" class="btn btn-secondary"
data-bs-dismiss="modal">Close</button>
                <button
type="submit" name="submit" class="btn
btn-primary">Done</button>
            </div>
        </div>
    </div>
    </div>
    <?php } else { ?>
        <td>
            <a href="#" class="btn
btn-danger"><i class="bi bi-x "></i></a>
        </td>
    </tr>
    </form>
    <?php } ?>
</tbody>
</table>
</div>
</div>
</div>
<?php } else { ?>
    <div class="card mb-2">
        <div class="container" style="margin-
top: 4.5rem;">
            <div class="card-title text-center">
                <h2>List Ticket</h2>
            </div>

```

```

<div class="card-body">
  <div class="table-responsive">
    <table class="table">
      <thead>
        <th>No</th>
        <th>Nama</th>
        <th>Tanggal</th>
        <th>Whatsapp</th>
        <th>Total Ticket</th>
        <th>Expired Tiket</th>
        <th>Tujuan</th>
        <th>Status</th>
        <th>Cek Ticket</th>
      </thead>
      <tbody>
        <?php
          $email =
$_SESSION["users"]["email"];
          $data = $conn-
>query("SELECT *, tbl_order.id as IDS
FROM tbl_order INNER JOIN tbl_schedule
ON tbl_order.id_schedule=tbl_schedule.id
WHERE email='$email' and (status=1 OR
status=0) ORDER BY order_at");
          $i = 1;
          while ($row = $data-
>fetch_assoc()) {
            if ($row["status"] == 0) $status
= '<div class="badge alert alert-success"
role="alert">'. "Pending" . '</div>';
            if ($row["status"] == 1) $status
= '<div class="badge alert alert-success"
role="alert">'. "Paid" . '</div>';
            ?>
            <tr class="align-middle">
              <th scope="row"><?=$i++>
?></th>
              <td><?=$
strtoupper($row["nama"]); ?></td>
              <td><?=$row["order_at"];
?></td>
              <td><?=$row["no_hp"];
?></td>

```

```

<?php
if ($row["status"] == 0) {
  $expiration_date = new
DateTime($row['order_at']);
  $expiration_date-
>modify('+3 hours');
}
if ($row["status"] == 1) {
  $expiration_date = new
DateTime($row['keberangkatan']);
  $expiration_date-
>modify('+1 days');
}
$new = $expiration_date-
>format('Y-m-d H:i:s'); ?>
<td><?=$
rupiah($row["price"]); ?></td>
<td><?=$new; ?></td>
<td><?=$row["arah"];
?></td>
<td><?=$status; ?></td>
<?php if ($row["status"] ==
1) { ?>
  <form
action="../action.php" method="post">
    <td>
      <button type="button"
class="btn btn-outline-success" data-bs-
toggle="modal" data-bs-target="#detail<?=$
row["id"] ?>">
        Cek Tiket
      </button>
    </td>
    <div class="modal modal-
lg fade" id="detail<?=$row["id"] ?>"
tabindex="-1" aria-
labelledby="exampleModalLabel" aria-
hidden="true">
      <div class="modal-dialog
modal-dialog-centered">
        <div class="modal-
content">

```

```

<div class="modal-
header">
    <h1 class="modal-
title fs-5" id="exampleModalLabel">Tiket
<?= strtoupper($row["arah"]); ?></h1>
    <button
type="button" class="btn-close" data-bs-
dismiss="modal" aria-
label="Close"></button>
</div>
<div class="modal-
body text-center">
    <div class="card text-
bg-dark g-0 text-center" style="background-
image: url(../assets/img/hero.png);
background-repeat: no-repeat; background-
position: center;">
        <div class="card-
body h4">
            <div class="row">
                <div class="col-
5">Nama</div>
                <div class="col-
7"><?= strtoupper($row["nama"]); ?></div>
            </div>
            <div class="row">
                <div class="col-
5">Tujuan</div>
                <div class="col-
7"><?= $row["arah"]; ?></div>
            </div>
            <div class="row">
                <div class="col-
5">Harga</div>
                <div class="col-
7"><?= rupiah($row["price"] *
$row["quantity"]); ?></div>
            </div>
            <div class="row">
                <div class="col-
5">Lokasi Tempat Duduk</div>
                <div class="col-
7"><?= $row["no_kursi"]; ?></div>

```

```

</div>
</div>
</div>
</div>
<div class="modal-
footer d-flex justify-content-center">
    <button
type="button" class="btn btn-secondary"
data-bs-dismiss="modal">Close</button>
    <button
type="button" class="btn btn-danger" data-
bs-toggle="modal" data-bs-
target="#verifyModal<?= $row["IDS"] ?>">
        Cancel Tiket
    </button>
</div>
</div>
</div>
</div>
<div class="modal fade"
id="verifyModal<?= $row["IDS"] ?>"
tabindex="-1" aria-
labelledby="exampleModalLabel" aria-
hidden="true">
    <div class="modal-dialog
modal-dialog-centered">
        <div class="modal-
content">
            <div class="modal-
header">
                <h1 class="modal-
title fs-5"
id="exampleModalLabel">Validasi
Pembatalan Tiket</h1>
                <button
type="button" class="btn-close" data-bs-
dismiss="modal" aria-
label="Close"></button>
            </div>
            <div class="modal-
body text-center">

```

```

                <h5>Apakah Anda
Setuju Untuk Membatalkan Tiket
Keberangkatan Anda?</h5>
                <h5>Jika Anda Setuju
Maka Pengembalian Dana Anda Akan Kami
Potong Sebesar 50%</h5>
                <input type="hidden"
value="<?=$row["IDS"]; ?>"
name="idOrder">
            </div>
            <div class="modal-
footer d-flex justify-content-center">
                <button
type="button" class="btn btn-secondary"
data-bs-dismiss="modal">Close</button>
                <button
type="submit" name="cancelTicket"
class="btn btn-danger">Cancel</button>
            </div>
        </div>
    </div>
</form>
<?php }
if ($row["status"] == 0) { ?>
    <form
action="completePayment.php"
method="post">
        <td>
            <button type="button"
class="btn btn-primary" data-bs-
toggle="modal" data-bs-
target="#completePayment<?=$row["status"] ?>">
                Bayar Tiket
            </button>
        </td>
        <div class="modal fade"
id="completePayment<?=$row["status"]
?>" tabindex="-1" aria-
labelledby="exampleModalLabel" aria-
hidden="true">

```

```

                <div class="modal-dialog
modal-dialog-centered">
                    <div class="modal-
content">
                        <div class="modal-
header">
                            <h1 class="modal-
title fs-5"
id="exampleModalLabel">Selesaikan
Pembayaran</h1>
                            <button
type="button" class="btn-close" data-bs-
dismiss="modal" aria-
label="Close"></button>
                        </div>
                        <div class="modal-
body">
                            <input type="hidden"
name="id" class="form-control"
value="<?=$row["IDS"]; ?>">
                            Apakah Anda Ingin
Melanjutkan Pembayaran
                        </div>
                        <div class="modal-
footer">
                            <button
type="button" class="btn btn-secondary"
data-bs-dismiss="modal">Close</button>
                            <button
type="submit" name="completePayment"
class="btn btn-success">Bayar</button>
                        </div>
                    </div>
                </div>
            </div>
        </form>
        <?php } ?>
    </tr>
    <?php } ?>
</tbody>
</table>
</div>
</div>

```

```

    </div>
  </div>
  <?php } ?>
</section>
<section id="contact" class="shadow p-3
mb-5 bg-body-tertiary rounded">
  <div class="card mb-2">
    <div class="container" style="margin-
top: 4.5rem;">
      <div class="card-title text-center">
        <h2>Kontak Kami</h2>
      </div>
      <div class="card-body">
        <div class="row text-center g-0
justify-content-center align-items-center">
          <div class="col-md-8">
            <iframe
src="https://www.Google.com/maps/embed
?pb=!1m18!1m12!1m3!1d3519.657955437
6713!2d119.66274197438153!3d-
3.810617343594667!2m3!1f0!2f0!3f0!3m2!
1i1024!2i768!4f13.1!3m3!1m2!1s0x2d944c
c469019a23%3A0xcd10ed125cae7691!2sP
o%20Anugrah!5e1!3m2!1sid!2sid!4v17168
97237699!5m2!1sid!2sid" width="100%"
height="450" style="border:0;"
allowfullscreen="" loading="lazy"
referrerpolicy="no-referrer-when-
downgrade"></iframe>
          </div>
          <div class="col-md-4">
            <h5>PO ANUGRAH
TRANS</h5>
            <p>Jl. Serigala, Macorawalie,
Kec. Paleteang, Kabupaten Pinrang,
Sulawesi Selatan 91215</p>

```

11. Payment.php (user)

```

<?php
include("../assets/conn.php");
if (isset($_POST["submit"])) {
  $id = $_POST["id"];
  $nama = $_POST["nama"];

```

```

  <a class="btn btn-success"
href="http://wa.me/+6282242700793"
target="_blank">
    <i class="bi bi-whatsapp"></i>
    +62 822-4270-0793
  </a>
</div>
</div>
</div>
</div>
</div>
</div>
</section>
</main>
<footer class="bg-dark text-light py-4">
  <div class="container">
    <div class="row mt-3">
      <div class="col text-center">
        <p class="mb-0">&copy; 2024
Company Name. All rights reserved.</p>
      </div>
    </div>
  </div>
</footer>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap
@5.3.3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-
YvpcrYf0tY3lHB60NNkmXc5s9fDVZLES
aAA55NDzOxhy9GkcIdslK1eN7N6jIeHz"
crossorigin="anonymous"></script>
</body>
</html>

```

```

$email = $_POST["email"];
$no_hp = $_POST["no_hp"];
(isset($_POST["jadwal"])) ? $jadwal =
$_POST["jadwal"] : "";

```

```

    $data = $conn->query("SELECT * FROM
tbl_schedule WHERE id='$id'");
    $data = $data->fetch_assoc();
}
function numberToAlphabet($number)
{
    $alphabet = range('A', 'Z');
    $result = "";

    while ($number > 0) {
        $index = ($number - 1) % 26;
        $result = $alphabet[$index] . $result;
        $number = (int)((($number - 1) / 26);
    }

    return $result;
}
?>
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8" />
    <meta
        name="viewport"
        content="width=device-width, initial-
scale=1" />
    <title>Pemesanan Bis</title>
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap
@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
QWTKZyjpPEjISv5WaRU9OFeRpok6Yctn
YmDr5pNlyT2bRjXh0JMhY6hW+ALEwI
H" crossorigin="anonymous" />
    <link
rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap
-icons@1.11.3/font/bootstrap-
icons.min.css" />
    <script
defer
src="https://cdn.jsdelivr.net/npm/alpinejs@
3.14.0/dist/cdn.min.js"></script>
    <script
type="text/javascript"
src="https://app.sandbox.Midtrans.com/sna

```

```

p/snap.js" data-client-key="SB-Mid-server-
FJARwt_FNRmtUinwinGt2rDf"></script>
</head>

<body>
    <div class="card">
        <div class="container d-flex justify-
content-center align-items-center"
style="min-height:100vh;">
            <div class="col-md-6">
                <div class="card-title text-center">
                    <h2>Detail Pembayaran</h2>
                </div>
                <?php if ($data["tipe"] == 1) { ?>
                    <div class="card-body">
                        <div class="card shadow p-3 mb-2
bg-body-tertiary rounded">
                            <div class="card-body">
                                <div id="detail">
                                    <form id="complete-payment"
class="text-center">
                                        <label
for="quantity">Data
Diri</label>
                                            <input
type="hidden"
name="id" value="<?= $data["id"]; ?>">
                                                <input required type="text"
name="nama" id="nama" class="form-
control my-2" placeholder="Masukkan
Nama Anda" value="<?= $nama; ?>">
                                                    <input required type="text"
name="email" id="email" class="form-
control my-2" placeholder="Masukkan
Email Anda" value="<?= $email; ?>">
                                                        <input required type="text"
name="no_hp" id="no_hp" class="form-
control my-2" placeholder="Masukkan
Nomor Handphone" value="<?= $no_hp;
?>">
                                                            <input
type="time"
name="waktu" class="form-control my-2"
value="<?= (isset($jadwal) ? $jadwal : "");
?>">

```

```

        <input required type="text"
name="price" placeholder="Harga"
class="form-control my-2" value="<?=$data["price"]; ?>">
        <input type="date"
name="tanggal" class="form-control my-2"
required>
        <?php if ($data["tipe"] == 1) {
?>
        <label for="quantity">Detail
Tempat Duduk</label>
        <div class="container my-2">
        <div class="row">
        <?php
        $res =
numberToAlphabet($data["id"]);
        $x = null;
        $slot = 0;
        $kursi = array();
        $no_kursi = $conn-
>query("SELECT no_kursi FROM tbl_order
WHERE status < 2");
        if ($no_kursi->num_rows >
0) {
            while ($row = $no_kursi-
>fetch_assoc()) {
                $arr = explode(",",
$row["no_kursi"]);
                for ($i = 0; $i <
sizeof($arr); $i++) {
                    array_push($kursi,
$arr[$i]);
                }
            }
            $total_kursi =
sizeof($kursi) + $data["slot"];
            for ($i = 0; $i <
$total_kursi; $i++) {
                $rows = $res . ($i + 1);
                for ($j = 0; $j <
sizeof($kursi); $j++) {
                    if ($kursi[$j] == $rows)
{

```

```

        $x = $rows ?>
        <div class="col-md-
3">
            <input
type="checkbox" class="btn-check"
id="btn-check-<?=$i; ?>" value="<?=$rows; ?>" autocomplete="off">
            <label class="btn btn-
success my-2" for="btn-check-<?=$i;
?>">Kursi <?=$rows; ?></label>
        </div>
        <?php }
        }
        if ($x != $rows) {
        ?>
        <div class="col-md-3">
            <input
type="checkbox" name="tmd[]" class="btn-
check" id="btn-check-<?=$i; ?>"
value="<?=$rows; ?>"
autocomplete="off">
            <label class="btn btn-
outline-primary my-2" for="btn-check-<?=$i; ?>">Kursi <?=$rows; ?></label>
        </div>
        <?php
        }
        } else {
            for ($i = 0; $i <
$data["slot"]; $i++) {
                ?>
                <div class="col-md-3">
                    <input type="checkbox"
name="tmd[]" class="btn-check" id="btn-
check-<?=$i; ?>" value="<?=$res . ($i + 1);
?>" autocomplete="off">
                    <label class="btn btn-
outline-primary my-2" for="btn-check-<?=$i; ?>">Kursi <?=$res . ($i + 1); ?></label>
                </div>
                <?php
                }
            }
        }
    }
}

```

```

    }
    ?>
  </div>
</div>
<?php } ?>
<button type="submit"
class="btn btn-success">Done</button>
  <a href="index.php" class="btn
btn-danger">Cancel</a>
</form>
</div>
</div>
</div>
</div>
<?php } else { ?>
  <div class="card-body">
    <div class="card shadow p-3 mb-2
bg-body-tertiary rounded">
      <div class="card-body">
        <div id="detail">
          <form id="complete-payment"
class="text-center">
            <label for="quantity">Data
Diri</label>
            <input type="hidden"
name="tmd" value="<?= null; ?>">
            <input type="hidden"
name="id" value="<?= $data["id"]; ?>">
            <input required type="text"
name="nama" id="nama" class="form-
control my-2" placeholder="Masukkan
Nama Anda" value="<?= $nama; ?>">
            <input required type="text"
name="email" id="email" class="form-
control my-2" placeholder="Masukkan
Email Anda" value="<?= $email; ?>">
            <input required type="text"
name="no_hp" id="no_hp" class="form-

```

```

control my-2" placeholder="Masukkan
Nomor Handphone" value="<?= $no_hp;
?>">

```

```

    <label for="quantity">Detail
Per Bus</label>

```

```

    <input required type="number"
name="quantity" id="quantity" class="form-
control my-2" value="1" min="1"
max="<?= (isset($data["slot"])) ?
$data["slot"] : 5); ?>">

```

```

    <input required type="text"
name="price" placeholder="Harga"
class="form-control my-2" value="<?=
$data["price"]; ?>">

```

```

    <input type="time"
name="waktu" class="form-control my-2">

```

```

    <input type="date"
name="tanggal" class="form-control my-
2">

```

```

    <button type="submit"
class="btn btn-success">Done</button>

```

```

    <a href="index.php" class="btn
btn-danger">Cancel</a>

```

```

  </form>

```

```

</div>

```

```

</div>

```

```

</div>

```

```

</div>

```

```

<?php } ?>

```

```

</div>

```

```

</div>

```

```

</div>

```

```

<script src=" ../assets/js/app.js?v=<%= new
Date() %>"></script>

```

```

</body>

```

```

</html>

```

12. DaftarOrder.php (admin)

```

<div class="container-fluid">

```

```

  <div class="card shadow px-3 mb-3
rounded">

```



```

        <div class="modal-
header">
            <h1 class="modal-title
fs-5" id="exampleModalLabel">Complete
The Order</h1>
            <button type="button"
class="btn-close" data-bs-dismiss="modal"
aria-label="Close"></button>
        </div>
        <div class="modal-
body">
            Are You Sure The
            Complete The Order ?
            <input type="hidden"
name="id" value="<?= $row["ids"]; ?>">
        </div>
        <div class="modal-
footer">
            <button type="submit"
class="btn btn-secondary"
name="cancel">Cancel</button>
            <button type="submit"
name="complete" class="btn btn-
primary">Update</button>
        </div>
    </div>
</div>
</form>
<button type="button"
class="btn btn-danger mt-2" data-bs-
toggle="modal" data-bs-
target="#dOrder<?= $row["ids"]; ?>">
    Delete Order
</button>
<form action=" ../action.php"
method="post">
    <div class="modal fade"
id="dOrder<?= $row["ids"]; ?>"
tabindex="-1" aria-
labelledby="exampleModalLabel" aria-
hidden="true">

```

```

        <div class="modal-dialog
modal-dialog-centered">
            <div class="modal-
content">
                <div class="modal-
header">
                    <h1 class="modal-title
fs-5" id="exampleModalLabel">Hapus
Order</h1>
                    <button type="button"
class="btn-close" data-bs-dismiss="modal"
aria-label="Close"></button>
                </div>
                <div class="modal-
body">
                    Apakah Anda Ingin
                    Menghapus Orderan Ini ?
                    <input type="hidden"
name="id" value="<?= $row["ids"]; ?>">
                </div>
                <div class="modal-footer
d-flex justify-content-center">
                    <button type="submit"
class="btn btn-secondary"
name="cancel">Cancel</button>
                    <button type="submit"
name="deleteOrder" class="btn btn-
danger">Delete</button>
                </div>
            </div>
        </div>
    </div>
</form>
</td>
</tr>
<?php } ?>
</tbody>
</table>
</div>
</div>
</div>
</div>

```

</div>

13. DaftarSchedule.php (admin)

```
<section id="schedule">
  <div class="card g-0 shadow bg-body-
    tertiary rounded">
    <div class="container" style="margin-
      top: 4.5rem;">
      <div class="card-title text-center mt-3
        mb-2">
        <h2>Waktu Keberangkatan</h2>
      </div>
      <div class="card-body">
        <div class="card p-3">
          <div class="card-title text-center">
            <h3>BUS MALAM / PAGI</h3>
          <div class="table-responsive">
            <table class="table text-center
              datatable">
              <thead class="table-info">
                <tr>
                  <th scope="col">Jadwal</th>
                  <th scope="col">Arah</th>
                  <th scope="col">Harga</th>
                  <th scope="col">Stok
                    Ticket</th>
                  <th scope="col">Tipe</th>
                  <th scope="col">Action</th>
                </tr>
              </thead>
              <tbody class="align-middle">
                <?php $data = $conn-
                  >query("SELECT * FROM tbl_schedule
                    ORDER BY tipe ");
                  while ($row = $data-
                    >fetch_assoc()) {
                    if ($row["tipe"] == 1) $tipe =
                      "BUS MALAM / PAGI";
                    else if ($row["tipe"] == 2) $tipe
                      = "BUS PARIWISATA";
                    ?>
```

```
                  <form action="../action.php"
                    method="post">
                    <tr>
                      <td><?= $row["jadwal"];
                        ?></td>
                      <td><?= $row["arah"];
                        ?></td>
                      <td><?=
                        rupiah($row["price"]); ?></td>
                      <td><?= $row["slot"];
                        ?></td>
                      <td><?= $tipe; ?></td>
                      <td>
                        <button type="button"
                          class="btn btn-warning" data-bs-
                            toggle="modal" data-bs-
                              target="#updateSchedule<?= $row["id"];
                                ?>">
                          <i class="bi bi-gear-
                            fill"></i>
                        </button>
                        <div class="modal fade"
                          id="updateSchedule<?= $row["id"]; ?>"
                          tabindex="-1" aria-
                            labelledby="exampleModalLabel" aria-
                              hidden="true">
                          <div class="modal-dialog
                            modal-dialog-centered">
                            <div class="modal-
                              content">
                              <div class="modal-
                                header">
                                  <h1 class="modal-title
                                    fs-5" id="exampleModalLabel">Update
                                      Jadwal</h1>
                                  <button type="button"
                                    class="btn-close" data-bs-dismiss="modal"
                                      aria-label="Close"></button>
                                </div>
                              </div>
```



```

<div class="card-body">
  <div class="row">
    <div class="col-md-4 mt-2">
      <div class="card shadow p-3 mb-2
rounded text-bg-success">
        <div class="card-header">
          <h3>
            <i class="bi bi-arrow-left-right
text-bg-light me-3 rounded-circle p-1 px-
2"></i>
              Order Selesai
            </h3>
          </div>
          <div class="card-body text-center">
            <blockquote class="blockquote
mb-0">
              <?php $data = $conn-
>query("SELECT * FROM tbl_order
INNER JOIN tbl_schedule ON
tbl_order.id_schedule=tbl_schedule.id
WHERE status=3");
              echo $data->num_rows;
              ?>
            </blockquote>
          </div>
        </div>
      </div>
    <div class="col-md-4 mt-2">
      <div class="card shadow p-3 mb-2
rounded text-bg-info">
        <div class="card-header">
          <h3>
            <i class="bi bi-coin text-bg-light
me-3 rounded-circle p-1 px-2"></i>
              Penghasilan
            </h3>
          </div>
          <div class="card-body text-center">
            <blockquote class="blockquote
mb-0">

```

```

        <?php $data = $conn-
>query("SELECT sum(quantity*price) as
val FROM tbl_order INNER JOIN
tbl_schedule ON
tbl_order.id_schedule=tbl_schedule.id
WHERE status=3");
        echo rupiah($data-
>fetch_assoc()["val"]);
        ?>
      </blockquote>
    </div>
  </div>
</div>
<div class="col-md-4 mt-2">
  <div class="card shadow p-3 mb-2
rounded text-bg-warning">
    <div class="card-header">
      <h3>
        <i class="bi bi-box-arrow-in-
down text-bg-light me-3 rounded-circle p-1
px-2"></i>
          Total Pesanan
        </h3>
      </div>
      <div class="card-body text-center">
        <blockquote class="blockquote
mb-0">
          <?php $data = $conn-
>query("SELECT * FROM tbl_order
INNER JOIN tbl_schedule ON
tbl_order.id_schedule=tbl_schedule.id");
          echo $data->num_rows;
          ?>
        </blockquote>
      </div>
    </div>
  </div>
</div>
</div>
</div>

```

```

<?php include("../assets/conn.php");
session_start();
if ($_SESSION["admin"] == "") {
    header("location:../index.php");
} else {
?>

<!doctype html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta name="viewport"
content="width=device-width, initial-
scale=1">
    <title>Anugrah Trans</title>
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap
@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
QWTKZyjpPEjISv5WaRU9OFeRpok6Yctn
YmDr5pNlyT2bRjXh0JMhY6hW+ALEwI
H" crossorigin="anonymous">
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap
-icons@1.11.3/font/bootstrap-
icons.min.css">
</head>

<body>
    <nav class="navbar navbar-expand-lg
shadow p-3 mb-2 bg-body-tertiary rounded
fixed-top opacity-75" data-bs-
theme="dark">
        <div class="container">
            <a class="navbar-brand fst-italic fw-
bold" href="#">ANUGRAH <span
class="text-info">TRANS</span></a>
            <button class="navbar-toggler"
type="button" data-bs-toggle="collapse"
data-bs-target="#navbarNavAltMarkup"
aria-controls="navbarNavAltMarkup" aria-

```

```

expanded="false" aria-label="Toggle
navigation">
            <span class="navbar-toggler-
icon"></span>
            </button>
            <div class="collapse navbar-collapse"
id="navbarNavAltMarkup">
                <div class="navbar-nav ms-auto">
                    <li class="nav-item">
                        <a class="nav-link active" aria-
current="page"
href="#transaksi">Home</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link"
href="#order">Pesanan</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link"
href="#schedule">Jadwal</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link"
href="../index.php">Logout</a>
                    </li>
                </div>
            </div>
        </nav>
        <main>
            <section id="hero" class="shadow mb-3
bg-body-tertiary rounded">
                <div class="card text-bg-dark g-0 text-
center">
                    
                    <div class="card-img-overlay
position-absolute top-50 start-50 translate-
middle">
                        <h1 class="card-body mt-5">
                            ANUGRAH <span class="text-
info">TRANS</span>

```



```

        console.log(err.message);
    }
});

```

17. Conn.php (asset)

```

<?php
$conn = mysqli_connect("localhost", "root",
"", "db_transport");
setExpiredTicket();
setExpiredOrderTicket();
function rupiah($amount)
{
    $formatter = new
    NumberFormatter('id_ID',
    NumberFormatter::CURRENCY);
    return $formatter-
    >formatCurrency($amount, 'IDR');
}

```

```

function setExpiredTicket()
{
    global $conn;
    $data = $conn->query("SELECT * FROM
tbl_order");
    $today = date('Y-m-d');
    if ($data->num_rows > 0) {
        while ($row = $data->fetch_assoc()) {
            $expiration_date = new
            DateTime($row['keberangkatan']);
            $expiration_date->modify('+1 days');
            $new = $expiration_date->format('Y-m-
d H:i:s');
            if ($today >= $new && ($row["status"]
== 1)) {
                $id = $row["id"];
                $data = $conn->query("SELECT
quantity, id_schedule, slot FROM tbl_order
INNER JOIN tbl_schedule ON
tbl_schedule.id=tbl_order.id_schedule
WHERE tbl_order.id='$id' and status<3");
                $data = $data->fetch_assoc();
                $quantity = $data["quantity"] +
                $data["slot"];

```

```

            });
            $id_schedule = $data["id_schedule"];
            $conn->query("UPDATE tbl_schedule
SET slot = '$quantity' WHERE
id='$id_schedule'");
            $conn->query("UPDATE tbl_order
SET status=4 WHERE id='$id'");
        }
    }
}

```

```

function setExpiredOrderTicket()
{
    global $conn;
    $data = $conn->query("SELECT * FROM
tbl_order WHERE status=0");
    $now = date("Y-m-d H:i:s");
    if ($data->num_rows > 0) {
        while ($row = $data->fetch_assoc()) {
            $expired = new
            DateTime($row["order_at"]);
            $expired->modify("+50 seconds");
            $new = $expired->format("Y-m-d
H:i:s");
            if ($now >= $new) {
                $id = $row["id"];
                $orderData = $conn->query("SELECT
quantity, id_schedule, slot FROM tbl_order
INNER JOIN tbl_schedule ON
tbl_schedule.id=tbl_order.id_schedule
WHERE tbl_order.id='$id' and status<3");
                $orderData = $orderData-
                >fetch_assoc();
                $quantity = $orderData["quantity"] +
                $orderData["slot"];
                $id_schedule =
                $orderData["id_schedule"];

```



```

$conn->query("UPDATE tbl_schedule
SET slot = '$quantity' WHERE
id='$id_schedule'");
$conn->query("UPDATE tbl_order
SET status=4 WHERE id='$id'");

```

18. *Midtrans.php* (asset)

```

<?php
/**
 * Check PHP version.
 */
if (version_compare(PHP_VERSION, '5.4',
'<')) {
    throw new Exception('PHP version >= 5.4
required');
}

// Check PHP Curl & json decode
capabilities.
if (!function_exists('curl_init') ||
!function_exists('curl_exec')) {
    throw new Exception('Midtrans needs the
CURL PHP extension. ');
}
if (!function_exists('json_decode')) {

```

19. *Test.php*

```

<?php
require dirname(__FILE__) .
'./assets/Midtrans-php-
master/Midtrans.php';
include("assets/conn.php");
\Midtrans\Config::$serverKey = "SB-Mid-
server-FJARwt_FNRmtUinwinGt2rDf";
\Midtrans\Config::$isProduction = false;
\Midtrans\Config::$isSanitized = true;
\Midtrans\Config::$is3ds = true;
function getTransactionStatus($orderId)
{
    try {
        $status =
\Midtrans\Transaction::status($orderId);

```

```

    }
    }
    }

    throw new Exception('Midtrans needs the
JSON PHP extension. ');
}

// Configurations
require_once 'Midtrans/Config.php';

// Midtrans API Resources
require_once 'Midtrans/Transaction.php';

// Plumbing
require_once 'Midtrans/ApiRequestor.php';
require_once 'Midtrans/Notification.php';
require_once 'Midtrans/CoreApi.php';
require_once 'Midtrans/Snap.php';

// Sanitization
require_once 'Midtrans/Sanitizer.php';

```

```

        return $status;
    } catch (Exception $e) {
        return null;
    }
}

function showPendingTransactions()
{
    $orderId = [
        '1897972141',
    ];

    foreach ($orderId as $orderId) {
        $status = getTransactionStatus($orderId);

```

```

        $status =
    json_decode(json_encode($status), true);
    if ($status &&
    $status["transaction_status"] == 'pending') {
        echo json_encode($status);
    }
}
}

```

```

showPendingTransactions();

```

20. Db_transport.sql (database)

```

-- phpMyAdmin SQL Dump
-- version 5.1.1
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: Jul 30, 2024 at 01:49
PM
-- Server version: 10.4.22-MariaDB
-- PHP Version: 7.4.26

```

```

SET SQL_MODE =
"NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

```

```

/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@
@CHARACTER_SET_CLIENT */;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=
@@CHARACTER_SET_RESULTS */;
/*!40101 SET
@OLD_COLLATION_CONNECTION=@
@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

```

```

--
-- Database: `db_transport`
--

```

```

-----
-----
--

```

```

-- Table structure for table `tbl_order`
--

```

```

CREATE TABLE `tbl_order` (
  `id` int(11) NOT NULL,
  `nama` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
  `no_hp` varchar(100) NOT NULL,
  `id_schedule` int(11) NOT NULL,
  `quantity` int(11) NOT NULL,
  `order_at` datetime DEFAULT NULL,
  `keberangkatan` datetime DEFAULT
NULL,
  `no_kursi` text NOT NULL,
  `status` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT
CHARSET=utf8mb4;

```

```

-----
-----

```

```

--
-- Table structure for table `tbl_schedule`
--

```

```

CREATE TABLE `tbl_schedule` (
  `id` int(11) NOT NULL,
  `arah` varchar(100) NOT NULL,
  `jadwal` varchar(100) NOT NULL,
  `price` varchar(100) NOT NULL,
  `tipe` varchar(100) NOT NULL,
  `slot` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT
CHARSET=utf8mb4;

```

```
--
-- Dumping data for table `tbl_schedule`
--
```

```
INSERT INTO `tbl_schedule` (`id`, `arah`,
`jadwal`, `price`, `tipe`, `slot`) VALUES
(1, 'PINRANG KE MALILI', 'PAGI /
MALAM', '170000', '1', 35),
(2, 'PINRANG KE MASAMBA', 'PAGI /
MALAM', '150000', '1', 35),
(3, 'PINRANG KE PALOPO', 'PAGI /
MALAM', '130000', '1', 35),
(4, 'PINRANG KE BELOPA', 'PAGI /
MALAM', '130000', '1', 35),
(5, 'PINRANG KE SIWA', 'PAGI /
MALAM', '100000', '1', 35),
(6, 'RAPPANG KE MALILI', 'PAGI /
MALAM', '170000', '1', 35),
(7, 'SIDRAP KE MALILI', 'PAGI /
MALAM', '150000', '1', 35),
(8, 'TANRU TEDONG KE MALILI', 'PAGI
/ MALAM', '150000', '1', 35),
(9, 'TURUMPAKE KE MALILI', 'PAGI /
MALAM', '150000', '1', 35),
(10, 'PINRANG KE BONE-BONE', 'PAGI /
MALAM', '150000', '1', 35),
(11, 'BULUKUMBA', 'OPTIONAL',
'5000000', '2', 5),
(12, 'SINJAI', 'OPTIONAL', '4000000', '2',
5),
(13, 'JENEPONTO', 'OPTIONAL',
'3500000', '2', 5),
(14, 'TORAJA', 'OPTIONAL', '3500000', '2',
5),
(15, 'MALINO', 'OPTIONAL', '4000000', '2',
5),
(16, 'TOPE JAWA', 'OPTIONAL',
'3000000', '2', 5),
(17, 'GRAND MALL', 'OPTIONAL',
'2500000', '2', 5),
(18, 'BANTIMURUNG', 'OPTIONAL',
'2500000', '2', 5),
```

```
(19, 'BANDARA', 'OPTIONAL', '2350000',
'2', 5),
(20, 'MAKASSAR', 'OPTIONAL',
'2700000', '2', 5),
(21, 'MAJENE', 'OPTIONAL', '2500000', '2',
5),
(22, 'PUNCAK', 'OPTIONAL', '1800000',
'2', 5),
(23, 'BUGIS', 'OPTIONAL', '2700000', '2',
5),
(24, 'DIANA BARRU', 'OPTIONAL',
'2000000', '2', 5),
(25, 'POLEWALI MANDAR',
'OPTIONAL', '1800000', '2', 5),
(26, 'PALOPO', 'OPTIONAL', '3000000', '2',
5);
```

```
-----
-----
```

```
--
-- Table structure for table `tbl_users`
--
```

```
CREATE TABLE `tbl_users` (
  `id` int(11) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `password` varchar(50) NOT NULL,
  `kode` int(11) NOT NULL,
  `no_hp` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT
CHARSET=utf8mb4;
```

```
--
-- Dumping data for table `tbl_users`
--
```

```
INSERT INTO `tbl_users` (`id`, `nama`,
`email`, `password`, `kode`, `no_hp`)
VALUES
(1, 'admin', 'admin@admin.com', 'admin', 2,
"),
```

```

(2, 'user', 'user@user.com', 'user', 1, ''),
--
-- Indexes for dumped tables
--
--
-- Indexes for table `tbl_order`
--
ALTER TABLE `tbl_order`
  ADD PRIMARY KEY (`id`);
--
-- Indexes for table `tbl_schedule`
--
ALTER TABLE `tbl_schedule`
  ADD PRIMARY KEY (`id`);
--
-- Indexes for table `tbl_users`
--
ALTER TABLE `tbl_users`
  ADD PRIMARY KEY (`id`);
--
--
-- AUTO_INCREMENT for dumped tables
--
--
-- AUTO_INCREMENT for table
`tbl_order`
--
ALTER TABLE `tbl_order`

```

```

  MODIFY `id` int(11) NOT NULL
  AUTO_INCREMENT,
  AUTO_INCREMENT=2128979525;
--
-- AUTO_INCREMENT for table
`tbl_schedule`
--
ALTER TABLE `tbl_schedule`
  MODIFY `id` int(11) NOT NULL
  AUTO_INCREMENT,
  AUTO_INCREMENT=27;
--
-- AUTO_INCREMENT for table
`tbl_users`
--
ALTER TABLE `tbl_users`
  MODIFY `id` int(11) NOT NULL
  AUTO_INCREMENT,
  AUTO_INCREMENT=9;
COMMIT;

/*!40101 SET
CHARACTER_SET_CLIENT=@OLD_CH
ARACTER_SET_CLIENT */;
/*!40101 SET
CHARACTER_SET_RESULTS=@OLD_
CHARACTER_SET_RESULTS */;
/*!40101 SET
COLLATION_CONNECTION=@OLD_C
OLLATION_CONNECTION */;

```

21. *Midtrans.php* (asset)

```

<?php
/**
 * Check PHP version.
 */
if (version_compare(PHP_VERSION, '5.4',
'<')) {
    throw new Exception('PHP version >= 5.4
required');
}

```

```

}

// Check PHP Curl & json decode
capabilities.
if (!function_exists('curl_init')) ||
!function_exists('curl_exec')) {
    throw new Exception('Midtrans needs the
CURL PHP extension.');
```

```

    }
    if (!function_exists('json_decode')) {
        throw new Exception('Midtrans needs the
JSON PHP extension.');
```

```

    }

    // Configurations
    require_once 'Midtrans/Config.php';
```

```

    // Midtrans API Resources
```

22. ApiRequestor.php (Asset)

```

<?php
```

```

namespace Midtrans;
```

```

use Exception;
```

```
/**
```

```

 * Send request to Midtrans API
 * Better don't use this class directly, please
use CoreApi, Snap, and Transaction instead
 */
```

```

class ApiRequestor
```

```
{
```

```
    /**
```

```
    * Send GET request
```

```
    *
```

```
    * @param string $url
```

```
    * @param string $server_key
```

```
    * @param mixed[] $data_hash
```

```
    * @return mixed
```

```
    * @throws Exception
```

```
    */
```

```

    public static function get($url,
$server_key, $data_hash)
```

```
    {
```

```

        require_once 'Midtrans/Transaction.php';
```

```

        // Plumbing
```

```

        require_once 'Midtrans/ApiRequestor.php';
```

```

        require_once 'Midtrans/Notification.php';
```

```

        require_once 'Midtrans/CoreApi.php';
```

```

        require_once 'Midtrans/Snap.php';
```

```

        // Sanitization
```

```

        require_once 'Midtrans/Sanitizer.php';
```

```

        return self::remoteCall($url,
$server_key, $data_hash, 'GET');
    }
```

```
    /**
```

```
    * Send POST request
```

```
    *
```

```
    * @param string $url
```

```
    * @param string $server_key
```

```
    * @param mixed[] $data_hash
```

```
    * @return mixed
```

```
    * @throws Exception
```

```
    */
```

```

    public static function post($url,
$server_key, $data_hash)
```

```
    {
```

```

        return self::remoteCall($url,
$server_key, $data_hash, 'POST');
```

```
    }
```

```
    /**
```

```
    * Send PATCH request
```

```
    *
```

```
    * @param string $url
```

```
    * @param string $server_key
```

```
    * @param mixed[] $data_hash
```

```
    * @return mixed
```

```

* @throws Exception
*/
public static function patch($url,
$server_key, $data_hash)
{
    return self::remoteCall($url,
$server_key, $data_hash, 'PATCH');
}

/**
 * Actually send request to API server
 *
 * @param string $url
 * @param string $server_key
 * @param mixed[] $data_hash
 * @param bool $post
 * @return mixed
 * @throws Exception
 */
public static function remoteCall($url,
$server_key, $data_hash, $method)
{
    $ch = curl_init();

    if (!$server_key) {
        throw new Exception(
            'The ServerKey/ClientKey is null,
            You need to set the server-key from Config.
            Please double-check Config and ServerKey
            key. ' .
            'You can check from the Midtrans
            Dashboard. ' .
            'See
            https://docs.Midtrans.com/en/Midtrans-
            account/overview?id=retrieving-api-access-
            keys ' .
            'for the details or contact support at
            support@Midtrans.com if you have any
            questions.'
        );
    }

```

```

);
} else {
    if ($server_key == '') {
        throw new Exception(
            'The ServerKey/ClientKey is
            invalid, as it is an empty string. Please
            double-check your ServerKey key. ' .
            'You can check from the
            Midtrans Dashboard. ' .
            'See
            https://docs.Midtrans.com/en/Midtrans-
            account/overview?id=retrieving-api-access-
            keys ' .
            'for the details or contact support
            at support@Midtrans.com if you have any
            questions.'
        );
    } elseif
(preg_match('/s/', $server_key)) {
        throw new Exception(
            'The ServerKey/ClientKey is
            contains white-space. Please double-check
            your API key. Please double-check your
            ServerKey key. ' .
            'You can check from the
            Midtrans Dashboard. ' .
            'See
            https://docs.Midtrans.com/en/Midtrans-
            account/overview?id=retrieving-api-access-
            keys ' .
            'for the details or contact support
            at support@Midtrans.com if you have any
            questions.'
        );
    }
}

$curl_options = array(

```

```

CURLOPT_URL => $url,
CURLOPT_HTTPHEADER =>
array(
    'Content-Type: application/json',
    'Accept: application/json',
    'User-Agent:      Midtrans-php-
v2.5.2',
    'Authorization:  Basic ' .
base64_encode($server_key . ':')
),
CURLOPT_RETURNTRANSFER
=> 1
);

```

```

// Set append notification to header
if (Config::$appendNotifUrl)
Config::$curlOptions[CURLLOPT_HTTPHE
ADER][] = 'X-Append-Notification: ' .
Config::$appendNotifUrl;
// Set override notification to header
if (Config::$overrideNotifUrl)
Config::$curlOptions[CURLLOPT_HTTPHE
ADER][] = 'X-Override-Notification: ' .
Config::$overrideNotifUrl;
// Set payment idempotency-key to
header
if (Config::$paymentIdempotencyKey)
Config::$curlOptions[CURLLOPT_HTTPHE
ADER][] = 'Idempotency-Key: ' .
Config::$paymentIdempotencyKey;

```

```

// merging with Config::$curlOptions
if (count(Config::$curlOptions)) {
    // We need to combine headers
manually, because it's array and it will no be
merged
    if
(Config::$curlOptions[CURLLOPT_HTTPH
EADER]) {

```

```

        $mergedHeaders
        =
array_merge($curl_options[CURLLOPT_HT
TPHEADER],
Config::$curlOptions[CURLLOPT_HTTPHE
ADER]);
        $headerOptions
        =
array(CURLLOPT_HTTPHEADER =>
$mergedHeaders);
        } else {
            $mergedHeaders = array();
            $headerOptions
            =
array(CURLLOPT_HTTPHEADER =>
$mergedHeaders);
        }

```

```

        $curl_options
        =
array_replace_recursive($curl_options,
Config::$curlOptions, $headerOptions);
    }

```

```

    if ($method != 'GET') {

        if ($data_hash) {
            $body
            =
json_encode($data_hash);

```

```

        $curl_options[CURLLOPT_POSTFIELDS] =
$body;
        } else {

```

```

        $curl_options[CURLLOPT_POSTFIELDS] =
"";
    }

```

```

    if ($method == 'POST') {
        $curl_options[CURLLOPT_POST]
= 1;
    } elseif ($method == 'PATCH') {

```

```

        curl_setopt($ch,
CURLOPT_CUSTOMREQUEST,
'PATCH');
    }
}

curl_setopt_array($ch, $curl_options);

// For testing purpose
if (class_exists("\Midtrans\MT_Tests")
&& MT_Tests::$stubHttp) {
    $result =
self::processStubed($curl_options, $url,
$server_key, $data_hash, $method);
} else {
    $result = curl_exec($ch);
    // curl_close($ch);
}

if ($result === false) {
    throw new Exception('CURL Error: '
. curl_error($ch), curl_errno($ch));
} else {
    try {
        $result_array =
json_decode($result);
    } catch (Exception $e) {
        throw new Exception("API
Request Error unable to json_decode API
response: ".$result . ' | Request url: '.$url);
    }
    $statusCode = curl_getinfo($ch,
CURLINFO_HTTP_CODE);

```

```

        if (isset($result_array->status_code)
&& $result_array->status_code >= 401 &&
$result_array->status_code != 407) {
            throw new Exception('Midtrans
API is returning API error. HTTP status
code: ' . $result_array->status_code . ' API
response: ' . $result, $result_array-
>status_code);
        } elseif ($statusCode >= 400) {
            throw new Exception('Midtrans
API is returning API error. HTTP status
code: ' . $statusCode . ' API response: ' .
$result, $statusCode);
        } else {
            return $result_array;
        }
    }
}

```

```

private static function
processStubed($curl, $url, $server_key,
$data_hash, $method)
{
    MT_Tests::$lastHttpRequest = array(
        "url" => $url,
        "server_key" => $server_key,
        "data_hash" => $data_hash,
        $method => $method,
        "curl" => $curl
    );

    return MT_Tests::$stubHttpResponse;
}
}

```

23. Config.php (Asset)

```
<?php
```

```
namespace Midtrans;
```

```
/**
```

```
* Midtrans Configuration
```



```

*/
class Config
{

    /**
     * Your merchant's server key
     *
     * @static
     */
    public static $serverKey;
    /**
     * Your merchant's client key
     *
     * @static
     */
    public static $clientKey;
    /**
     * True for production
     * false for sandbox mode
     *
     * @static
     */
    public static $isProduction = false;
    /**
     * Set it true to enable 3D Secure by
    default
     *
     * @static
     */
    public static $is3ds = false;
    /**
     * Set Append URL notification
     *
     * @static
     */
    public static $appendNotifUrl;
    /**
     * Set Override URL notification
     *

```

```

     * @static
     */
    public static $overrideNotifUrl;
    /**
     * Set Payment IdempotencyKey
     *      for details (http://api-
    docs.Midtrans.com/#idempotent-requests)
     *
     * @static
     */
    public static $paymentIdempotencyKey;
    /**
     * Enable request params sanitizer
    (validate and modify charge request params).
     * See Midtrans_Sanitizer for more details
     *
     * @static
     */
    public static $isSanitized = false;
    /**
     * Default options for every request
     *
     * @static
     */
    public static $curlOptions = array();

    const SANDBOX_BASE_URL =
    'https://api.sandbox.Midtrans.com';
    const PRODUCTION_BASE_URL =
    'https://api.Midtrans.com';
    const SNAP_SANDBOX_BASE_URL =
    'https://app.sandbox.Midtrans.com/snap/v1';
    const
    SNAP_PRODUCTION_BASE_URL =
    'https://app.Midtrans.com/snap/v1';

    /**
     * Get baseUrl
     *

```

```

        * @return string Midtrans API URL,
        depends on $isProduction
        */
        public static function getBaseUrl()
        {
            return Config::$isProduction ?
                Config::PRODUCTION_BASE_URL :
                Config::SANDBOX_BASE_URL;
        }

        /**
         * Get snapBaseUrl
         */

```

24. CoreApi.php (Asset)

```

<?php

namespace Midtrans;

use Exception;

/**
 * Provide charge and capture functions for
 * Core API
 */
class CoreApi
{
    /**
     * Create transaction.
     *
     * @param mixed[] $params Transaction
     * options
     * @return mixed
     * @throws Exception
     */
    public static function charge($params)
    {
        $payloads = array(
            'payment_type' => 'credit_card'
        );

```

```

        * @return string Snap API URL, depends
        on $isProduction
        */
        public static function getSnapBaseUrl()
        {
            return Config::$isProduction ?

            Config::SNAP_PRODUCTION_BASE_URL
            :
            Config::SNAP_SANDBOX_BASE_URL;
        }
    }

```

```

        if (isset($params['item_details'])) {
            $gross_amount = 0;
            foreach ($params['item_details'] as
                $item) {
                $gross_amount +=
                    $item['quantity'] * $item['price'];
            }

            $payloads['transaction_details']['gross_amo
            unt'] = $gross_amount;
        }

        $payloads =
            array_replace_recursive($payloads,
                $params);

        if (Config::$isSanitized) {
            Sanitizer::jsonRequest($payloads);
        }

        return ApiRequestor::post(
            Config::getBaseUrl() . '/v2/charge',
            Config::$serverKey,
            $payloads

```

```

    );
}

/**
 * Capture pre-authorized transaction
 *
 * @param string $param Order ID or
transaction ID, that you want to capture
 * @return mixed
 * @throws Exception
 */
public static function capture($param)
{
    $payloads = array(
        'transaction_id' => $param,
    );

    return ApiRequestor::post(
        Config::getBaseUrl() . '/v2/capture',
        Config::$serverKey,
        $payloads
    );
}

/**
 * Do `/v2/card/register` API request to
Core API
 *
 * @param $cardNumber
 * @param $expMoth
 * @param $expYear
 * @return mixed
 * @throws Exception
 */
public static function
cardRegister($cardNumber, $expMoth,
$expYear)
{

```

```

    $path = "/card/register?card_number="
. $cardNumber
    . "&card_exp_month=" . $expMoth
    . "&card_exp_year=" . $expYear
    . "&client_key="
    . Config::$clientKey;

    return ApiRequestor::get(
        Config::getBaseUrl() . "/v2" . $path,
        Config::$clientKey,
        false
    );
}

/**
 * Do `/v2/token` API request to Core API
 *
 * @param $cardNumber
 * @param $expMoth
 * @param $expYear
 * @param $cvv
 * @return mixed
 * @throws Exception
 */
public static function
cardToken($cardNumber, $expMoth,
$expYear, $cvv)
{
    $path = "/token?card_number="
. $cardNumber
    . "&card_exp_month=" . $expMoth
    . "&card_exp_year=" . $expYear
    . "&card_cvv=" . $cvv
    . "&client_key="
    . Config::$clientKey;

    return ApiRequestor::get(
        Config::getBaseUrl() . "/v2" . $path,
        Config::$clientKey,

```

```

        false
    );
}

/**
 * Do `/v2/point_inquiry/<tokenId>` API
request to Core API
 *
 * @param string tokenId - tokenId of
credit card (more params detail refer to:
https://api-docs.Midtrans.com)
 * @return mixed
 * @throws Exception
 */
public static function
cardPointInquiry($tokenId)
{
    return ApiRequestor::get(
        Config::getBaseUrl()
        .
        '/v2/point_inquiry/' . $tokenId,
        Config::$serverKey,
        false
    );
}

/**
 * Create `/v2/pay/account` API request to
Core API
 *
 * @param string create pay account
request (more params detail refer to:
https://api-docs.Midtrans.com/#create-pay-account)
 * @return mixed
 * @throws Exception
 */
public static function
linkPaymentAccount($param)
{

```

```

        return ApiRequestor::post(
            Config::getBaseUrl()
            .
            '/v2/pay/account',
            Config::$serverKey,
            $param
        );
    }

    /**
     * Do `/v2/pay/account/<accountId>` API
request to Core API
     *
     * @param string accountId (more params
detail refer to: https://api-docs.Midtrans.com/#get-pay-account)
     * @return mixed
     * @throws Exception
     */
    public static function
    getPaymentAccount($accountId)
    {
        return ApiRequestor::get(
            Config::getBaseUrl()
            .
            '/v2/pay/account/' . $accountId,
            Config::$serverKey,
            false
        );
    }

    /**
     *
     * @param string Unbind
`/v2/pay/account/<accountId>/unbind` API
request to Core API
     *
     * @param string accountId (more params
detail refer to: https://api-docs.Midtrans.com/#unbind-pay-account)
     * @return mixed
     * @throws Exception
    }

```

```

    */
    public          static          function
    unlinkPaymentAccount($accountId)
    {
        return ApiRequestor::post(
            Config::getBaseUrl()
            .
            '/v2/pay/account/' . $accountId . '/unbind',
            Config::$serverKey,
            false
        );
    }

    /**
     * Create `/v1/subscription` API request to
     Core API
     *
     * @param string create subscription
     request (more params detail refer to:
     https://api-docs.Midtrans.com/#create-
     subscription)
     * @return mixed
     * @throws Exception
     */
    public          static          function
    createSubscription($param)
    {
        return ApiRequestor::post(
            Config::getBaseUrl()
            .
            '/v1/subscriptions',
            Config::$serverKey,
            $param
        );
    }

    /**
     *
     Do
     `/v1/subscription/<subscription_id>` API
     request to Core API
     *

```

```

    * @param string get subscription request
    (more params detail refer to: https://api-
    docs.Midtrans.com/#get-subscription)
    * @return mixed
    * @throws Exception
    */
    public          static          function
    getSubscription($SubscriptionId)
    {
        return ApiRequestor::get(
            Config::getBaseUrl()
            .
            '/v1/subscriptions/' . $SubscriptionId,
            Config::$serverKey,
            false
        );
    }

    /**
     *
     Do
     `/v1/subscription/<subscription_id>/disable`
     API request to Core API
     *
     * @param string disable subscription
     request (more params detail refer to:
     https://api-docs.Midtrans.com/#disable-
     subscription)
     * @return mixed
     * @throws Exception
     */
    public          static          function
    disableSubscription($SubscriptionId)
    {
        return ApiRequestor::post(
            Config::getBaseUrl()
            .
            '/v1/subscriptions/' . $SubscriptionId .
            '/disable',
            Config::$serverKey,
            false
        );
    }

```

```

    }

    /**
     *          Do          enable
     *`/v1/subscription/<subscription_id>/enable`
     *API request to Core API
     *
     * @param string enable subscription
     * request (more params detail refer to:
     * https://api-docs.Midtrans.com/#enable-subscription)
     * @return mixed
     * @throws Exception
     */
    public          static          function
    enableSubscription($SubscriptionId)
    {
        return ApiRequestor::post(
            Config::getBaseUrl() .
            '/v1/subscriptions/' . $SubscriptionId .
            '/enable',
            Config::$serverKey,
            false
        );
    }
}

```

25. Notification.php (Asset)

```

<?php

namespace Midtrans;

/**
 * Read raw post input and parse as JSON.
 * Provide getters for fields in notification
 * object
 *
 * Example:
 *
 * ```php

```

```

    /**
     *          Do          update          subscription
     *`/v1/subscription/<subscription_id>` API
     *request to Core API
     *
     * @param string update subscription
     * request (more params detail refer to:
     * https://api-docs.Midtrans.com/#update-subscription)
     * @return mixed
     * @throws Exception
     */
    public          static          function
    updateSubscription($SubscriptionId,
    $param)
    {
        return ApiRequestor::patch(
            Config::getBaseUrl() .
            '/v1/subscriptions/' . $SubscriptionId,
            Config::$serverKey,
            $param
        );
    }
}

```

```

*
* namespace Midtrans;
*
* $notif = new Notification();
* echo $notif->order_id;
* echo $notif->transaction_status;
* ```
*/
class Notification
{
    private $response;

```

```

        public function
        __construct($input_source = "php://input")
        {
            $raw_notification =
            json_decode(file_get_contents($input_source), true);
            $status_response =
            Transaction::status($raw_notification['transaction_id']);
            $this->response = $status_response;
        }

```

26. Zanitizer.php (Asset)

```

<?php

namespace Midtrans;

/**
 * Request params filters.
 *
 * It truncate fields that have length limit,
 * remove not allowed characters from other
 * fields
 *
 * This feature is optional, you can control it
 * with Config::$isSanitized (default: false)
 */
class Sanitizer
{
    private $filters;

    public function __construct()
    {
        $this->filters = array();
    }

    /**
     * Validates and modify data
     *

```

```

        public function __get($name)
        {
            if (isset($this->response->$name)) {
                return $this->response->$name;
            }
        }

        public function getResponse()
        {
            return $this->response;
        }
    }

    * @param mixed[] $json
    */
    public static function
    jsonRequest(&$json)
    {
        $keys = array('item_details',
        'customer_details');
        foreach ($keys as $key) {
            if (!isset($json[$key])) continue;
            $camel =
            static::upperCamelize($key);
            $function = "field$camel";
            static::$$function($json[$key]);
        }
    }

    private static function
    fieldItemDetails(&$items)
    {
        foreach ($items as &$item) {
            $id = new self;
            $item['id'] = $id
            ->maxLength(50)
            ->apply($item['id']);
            $name = new self;
            $item['name'] = $name

```

```

        ->maxLength(50)
        ->apply($item['name']);
    }
}

private static function
fieldCustomerDetails(&$field)
{
    if (isset($field['first_name'])) {
        $first_name = new self;
        $field['first_name'] = $first_name-
>maxLength(255)-
>apply($field['first_name']);
    }

    if (isset($field['last_name'])) {
        $last_name = new self;
        $field['last_name'] = $last_name-
>maxLength(255)-
>apply($field['last_name']);
    }

    if (isset($field['email'])) {
        $email = new self;
        $field['email'] = $email-
>maxLength(255)->apply($field['email']);
    }

    if (isset($field['phone'])) {
        $phone = new self;
        $field['phone'] = $phone-
>maxLength(255)->apply($field['phone']);
    }

    if (!empty($field['billing_address']) ||
!empty($field['shipping_address'])) {
        $keys = array('billing_address',
'shipping_address');
        foreach ($keys as $key) {

```

```

        if (!isset($field[$key])) continue;

        $camel =
static::upperCamelize($key);
        $function = "field$camel";
        static::$function($field[$key]);
    }
}

private static function
fieldBillingAddress(&$field)
{
    $fields = array(
        'first_name' => 255,
        'last_name' => 255,
        'address' => 255,
        'city' => 255,
        'country_code' => 3
    );

    foreach ($fields as $key => $value) {
        if (isset($field[$key])) {
            $self = new self;
            $field[$key] = $self
                ->maxLength($value)
                ->apply($field[$key]);
        }
    }

    if (isset($field['postal_code'])) {
        $postal_code = new self;
        $field['postal_code'] = $postal_code
            ->whitelist('A-Za-z0-9\\- ')
            ->maxLength(10)
            ->apply($field['postal_code']);
    }

    if (isset($field['phone'])) {

```



```

        static::fieldPhone($field['phone']);
    }
}

private static function
fieldShippingAddress(&$field)
{
    static::fieldBillingAddress($field);
}

private static function
fieldPhone(&$field)
{
    $plus = substr($field, 0, 1) === '+';
    $self = new self;
    $field = $self
        ->whitelist("\d\\-\\(\\) ")
        ->maxLength(19)
        ->apply($field);

    if ($plus) $field = '+' . $field;
    $self = new self;
    $field = $self
        ->maxLength(19)
        ->apply($field);
}

private function maxLength($length)
{
    $this->filters[] = function ($input) use
($length) {

```

```

        return substr($input, 0, $length);
    };
    return $this;
}

private function whitelist($regex)
{
    $this->filters[] = function ($input) use
($regex) {
        return preg_replace("/[^\$regex]/", "",
$input);
    };
    return $this;
}

private function apply($input)
{
    foreach ($this->filters as $filter) {
        $input = call_user_func($filter,
$input);
    }
    return $input;
}

private static function
upperCamelize($string)
{
    return str_replace(' ', ' ',
ucwords(str_replace('_', ' ', $string)));
}
}

```

27. Snap.php (Asset)

```

<?php

namespace Midtrans;

use Exception;

```

```

/**
 * Create Snap payment page and return snap
token
 */
class Snap
{

```

```

/**
 * Create Snap payment page
 *
 * Example:
 *
 * ```php
 *
 * namespace Midtrans;
 *
 * $params = array(
 *     'transaction_details' => array(
 *         'order_id' => rand(),
 *         'gross_amount' => 10000,
 *     )
 * );
 *
 * $paymentUrl =
Snap::getSnapToken($params);
 * ```
 *
 * @param array $params Payment
options
 * @return string Snap token.
 * @throws Exception curl error or
Midtrans error
 */
public static function
getSnapToken($params)
{
    return
(Snap::createTransaction($params)->token);
}

/**
 * Create Snap URL payment
 *
 * Example:
 *
 * ```php
 *

```

```

 * namespace Midtrans;
 *
 * $params = array(
 *     'transaction_details' => array(
 *         'order_id' => rand(),
 *         'gross_amount' => 10000,
 *     )
 * );
 *
 * $paymentUrl =
Snap::getSnapUrl($params);
 * ```
 *
 * @param array $params Payment
options
 * @return string Snap redirect url.
 * @throws Exception curl error or
Midtrans error
 */
public static function
getSnapUrl($params)
{
    return
(Snap::createTransaction($params)-
>redirect_url);
}

/**
 * Create Snap payment page, with this
version returning full API response
 *
 * Example:
 *
 * ```php
 *
 * $params = array(
 *     'transaction_details' => array(
 *         'order_id' => rand(),
 *         'gross_amount' => 10000,
 *     )
 * );
 *

```

```

*           $paymentUrl      =
Snap::getSnapToken($params);
* ''
*
* @param array $params Payment
options
* @return object Snap response (token
and redirect_url).
* @throws Exception curl error or
Midtrans error
*/
public static function
createTransaction($params)
{
    $payloads = array(
        'credit_card' => array(
            // 'enabled_payments' =>
array('credit_card'),
            'secure' => Config::$is3ds
        )
    );

    if (isset($params['item_details'])) {
        $gross_amount = 0;
        foreach ($params['item_details'] as
$item) {

```

```

        $gross_amount      +=
$item['quantity'] * $item['price'];
    }

    $params['transaction_details']['gross_ammoun
t'] = $gross_amount;
    }

    if (Config::$isSanitized) {
        Sanitizer::jsonRequest($params);
    }

    $params =
array_replace_recursive($payloads,
$params);

    return ApiRequestor::post(
        Config::getSnapBaseUrl()
        .
'/transactions',
        Config::$serverKey,
        $params
    );
}
}

```

28. SnapApiRequestor.php (Asset)

```
<?php
```

```
namespace Midtrans;
```

```

/**
 * Send request to Snap API
 * Better don't use this class directly, use
Snap
 * @deprecated this class already
deprecated. We will deleted on the next
major release. We have been centralized the

```

```

* requestor via ApiRequestor.php
*/

```

```

class SnapApiRequestor
{
    /**
     * Send GET request
     *
     * @param string $url
     * @param string $server_key
     * @param mixed[] $data_hash

```

```

        */
        public static function get($url,
        $server_key, $data_hash)
        {
            return self::remoteCall($url,
        $server_key, $data_hash, false);
        }

        /**
        * Send POST request
        *
        * @param string $url
        * @param string $server_key
        * @param mixed[] $data_hash
        */
        public static function post($url,
        $server_key, $data_hash)
        {
            return self::remoteCall($url,
        $server_key, $data_hash, true);
        }

        /**
        * Actually send request to API server
        *
        * @param string $url
        * @param string $server_key
        * @param mixed[] $data_hash
        * @param bool $post
        */
        public static function remoteCall($url,
        $server_key, $data_hash, $post = true)
        {
            $ch = curl_init();

            $curl_options = array(
                CURLOPT_URL => $url,
                CURLOPT_HTTPHEADER => array(
                    'Content-Type: application/json',

```

```

                    'Accept: application/json',
                    'Authorization: Basic ' .
                        base64_encode($server_key . ':')
                ),
                CURLOPT_RETURNTRANSFER =>
                    1,
                // CURLOPT_CAINFO =>
                    dirname(__FILE__) . '/../data/cacert.pem'
            );

            // merging with Config::$curlOptions
            if (count(Config::$curlOptions)) {
                // We need to combine headers
                manually, because it's array and it will no be
                merged
                if
                (Config::$curlOptions[CURLOPT_HTTPH
                    EADER]) {
                    $mergedHeders =
                        array_merge($curl_options[CURLOPT_HT
                            TPHEADER],
                            Config::$curlOptions[CURLOPT_HTTPPHE
                                ADER]);
                    $headerOptions = array(
                        CURLOPT_HTTPHEADER =>
                            $mergedHeders );
                } else {
                    $mergedHeders = array();
                }

                $curl_options =
                    array_replace_recursive($curl_options,
                        Config::$curlOptions, $headerOptions);
            }

            if ($post) {
                $curl_options[CURLOPT_POST] =
                    1;

```

```

        if ($data_hash) {
            $body =
            json_encode($data_hash);

            $curl_options[CURLOPT_POSTFIELDS] =
            $body;
        } else {

            $curl_options[CURLOPT_POSTFIELDS] =
            "";
        }

        curl_setopt_array($ch, $curl_options);

        // For testing purpose
        if (class_exists('\Midtrans\VT_Tests')
        && VT_Tests::$stubHttp) {
            $result =
            self::processStubed($curl_options, $url,
            $server_key, $data_hash, $post);
            $info = VT_Tests::$stubHttpStatus;
        } else {
            $result = curl_exec($ch);
            $info = curl_getinfo($ch);
            // curl_close($ch);
        }

        if ($result === false) {
            throw new \Exception('CURL Error:
            ' . curl_error($ch), curl_errno($ch));
        } else {
            try {
                $result_array =
                json_decode($result);

```

```

            } catch (\Exception $e) {
                $message = "API Request Error
                unable to json_decode API response:
                ".$result . ' | Request url: '.$url;
                throw new \Exception($message);
            }
            if ($info['http_code'] != 201) {
                $message = 'Midtrans Error (' .
                $info['http_code'] . '): '
                . $result . ' | Request url: '.$url;
                throw new \Exception($message,
                $info['http_code']);
            } else {
                return $result_array;
            }
        }
    }

    private static function
    processStubed($curl, $url, $server_key,
    $data_hash, $post)
    {
        VT_Tests::$lastHttpRequest = array(
            "url" => $url,
            "server_key" => $server_key,
            "data_hash" => $data_hash,
            "post" => $post,
            "curl" => $curl
        );

        return VT_Tests::$stubHttpResponse;
    }
}

```

29. TransAction.php (Asset)
 <?php

namespace *Midtrans*;

```

use Exception;
/**
 * API methods to get transaction status,
 * approve and cancel transactions
 */
class Transaction
{
    /**
     * Retrieve transaction status
     *
     * @param string $id Order ID or
     * transaction ID
     *
     * @return mixed[]
     * @throws Exception
     */
    public static function status($id)
    {
        return ApiRequestor::get(
            Config::getBaseUrl() . '/v2/' . $id .
            '/status',
            Config::$serverKey,
            false
        );
    }

    /**
     * Retrieve B2B transaction status
     *
     * @param string $id Order ID or
     * transaction ID
     *
     * @return mixed[]
     * @throws Exception
     */
    public static function statusB2b($id)
    {
        return ApiRequestor::get(

```

```

            Config::getBaseUrl() . '/v2/' . $id .
            '/status/b2b',
            Config::$serverKey,
            false
        );
    }

    /**
     * Approve challenge transaction
     *
     * @param string $id Order ID or
     * transaction ID
     *
     * @return string
     * @throws Exception
     */
    public static function approve($id)
    {
        return ApiRequestor::post(
            Config::getBaseUrl() . '/v2/' . $id .
            '/approve',
            Config::$serverKey,
            false
        )->status_code;
    }

    /**
     * Cancel transaction before it's settled
     *
     * @param string $id Order ID or
     * transaction ID
     *
     * @return string
     * @throws Exception
     */
    public static function cancel($id)
    {
        return ApiRequestor::post(

```

```

        Config::getBaseUrl() . '/v2/' . $id .
'/cancel',
        Config::$serverKey,
        false
    )->status_code;
}

/**
 * Expire transaction before it's settled
 *
 * @param string $id Order ID or
transaction ID
 *
 * @return mixed[]
 * @throws Exception
 */
public static function expire($id)
{
    return ApiRequestor::post(
        Config::getBaseUrl() . '/v2/' . $id .
'/expire',
        Config::$serverKey,
        false
    );
}

/**
 * Transaction status can be updated into
refund
 * if the customer decides to cancel
completed/settlement payment.
 * The same refund id cannot be reused
again.
 *
 * @param string $id Order ID or
transaction ID
 *
 * @param $params
 * @return mixed[]

```

```

 * @throws Exception
 */
public static function refund($id,
$params)
{
    return ApiRequestor::post(
        Config::getBaseUrl() . '/v2/' . $id .
'/refund',
        Config::$serverKey,
        $params
    );
}

/**
 * Transaction status can be updated into
refund
 * if the customer decides to cancel
completed/settlement payment.
 * The same refund id cannot be reused
again.
 *
 * @param string $id Order ID or
transaction ID
 *
 * @return mixed[]
 * @throws Exception
 */
public static function refundDirect($id,
$params)
{
    return ApiRequestor::post(
        Config::getBaseUrl() . '/v2/' . $id .
'/refund/online/direct',
        Config::$serverKey,
        $params
    );
}

/**

```

```
* Deny method can be triggered to
immediately deny card payment transaction
* in which fraud_status is challenge.
*
* @param string $id Order ID or
transaction ID
*
* @return mixed[]
* @throws Exception
*/
```

```
public static function deny($id)
{
    return ApiRequestor::post(
        Config::getBaseUrl() . '/v2/' . $id .
        '/deny',
        Config::$serverKey,
        false
    );
}
```