

BAB IV

PEMBANGUNAN DAN PENGUJIAN SISTEM

4.1 PENGENALAN

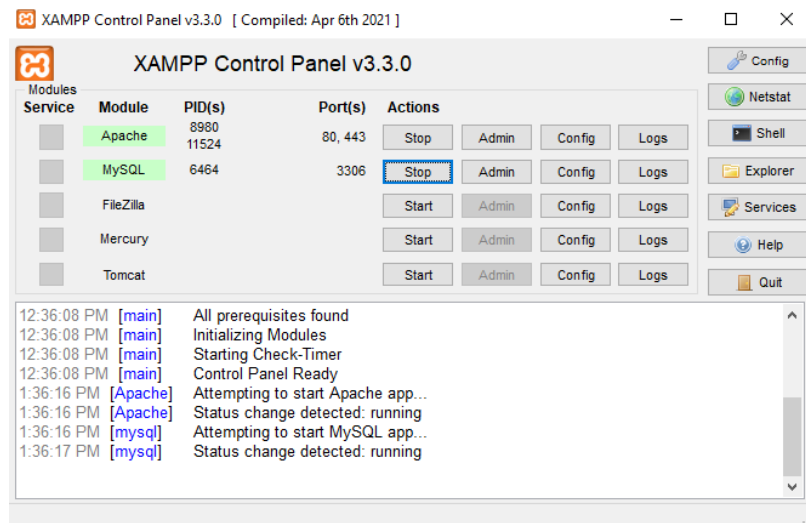
Bab ini membincangkan tentang fasa pembangunan dan pengujian bagi First Step Siswa: Sistem Web Bantuan Awal Siswa. Sistem ini mempermudah pelajar memohon bantuan asas secara talian, memudahkan pihak pentadbiran memantau dan meluluskan permohonan serta menyediakan platform kepada penderma untuk menyumbang keperluan yang diperlukan. Tujuan utama bab ini adalah untuk memberikan gambaran menyeluruh mengenai proses pembangunan dan pengujian sistem dengan memberi fokus kepada modul-modul utama dalam sistem seperti modul permohonan bantuan, modul penajaan penderma, modul pengurusan permohonan oleh pentadbir dan modul pengurusan inventori bantuan. Setiap modul dibangunkan mengikut keperluan pengguna yang telah dikenalpasti dalam fasa awal kajian di bab terdahulu bagi memastikan sistem ini dapat memenuhi keperluan sebenar pengguna.

Bab ini juga menjelaskan setiap komponen telah direka mengikut spesifikasi keperluan fungsian yang telah ditetapkan di bab 3. Reka bentuk antaramuka pengguna juga disesuaikan agar sistem mudah digunakan oleh pelajar, penderma dan pentadbir. Sistem ini dibangunkan menggunakan PHP, HTML, CSS, JavaScript, dan MySQL. Selain itu, bab ini turut memperincikan proses pengujian sistem yang dilaksanakan supaya sistem berfungsi dengan betul dan memenuhi keperluan pengguna. Pengujian ini dilaksanakan secara berperingkat yang merangkumi pengujian fungsi utama, pengujian antara modul dan pengujian keseluruhan sistem. Pelan pengujian yang diterangkan dalam bab ini memberi tumpuan kepada objektif pengujian, kaedah pengujian serta kriteria pengujian yang telah ditetapkan. Proses ini adalah penting bagi memastikan First Step Siswa berfungsi dengan lancar dan efisien.

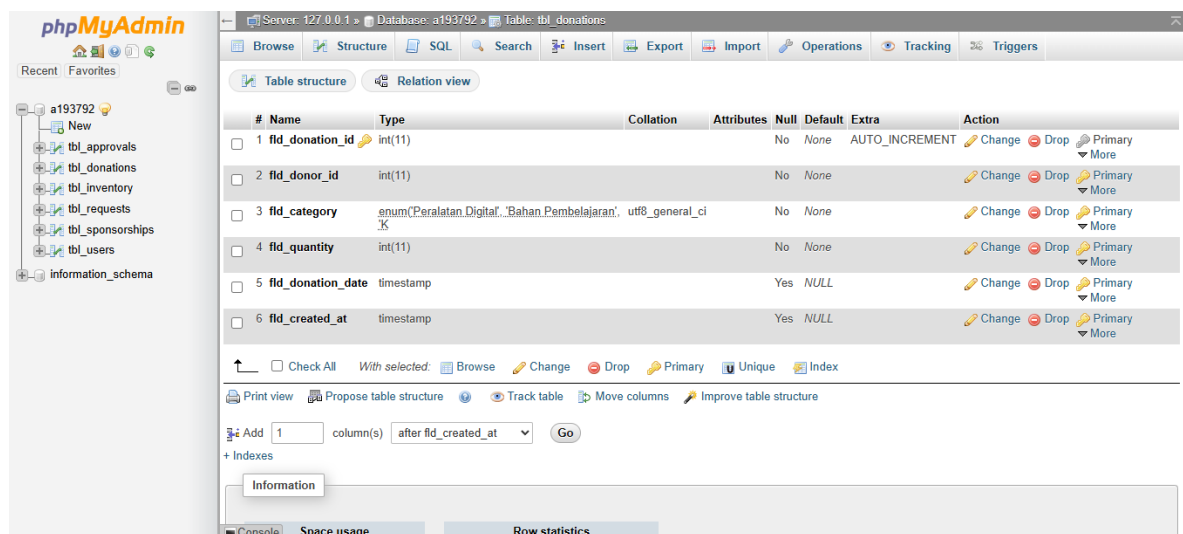
4.2 PROSES PEMBANGUNAN

Sistem First Step Siswa: Sistem Web Bantuan Awal Siswa dibangunkan secara berperingkat mengikut keperluan pengguna. Proses pembangunan dimulakan dengan reka bentuk antara muka pengguna yang mudah digunakan oleh pelajar, penderma, dan pentadbir. Sistem ini menggunakan bahasa pengaturcaraan PHP untuk membina logik sistem manakala HTML, CSS, dan JavaScript digunakan untuk membina paparan antara muka. Pembangunan dijalankan menggunakan perisian *Visual Studio Code* sebagai tempat menulis kod dan XAMPP sebagai pelayan setempat. Dalam XAMPP, Apache digunakan sebagai pelayan web, MySQL sebagai pangkalan data, dan phpMyAdmin digunakan untuk mengurus data. Sistem ini direka agar boleh digunakan melalui komputer atau telefon pintar

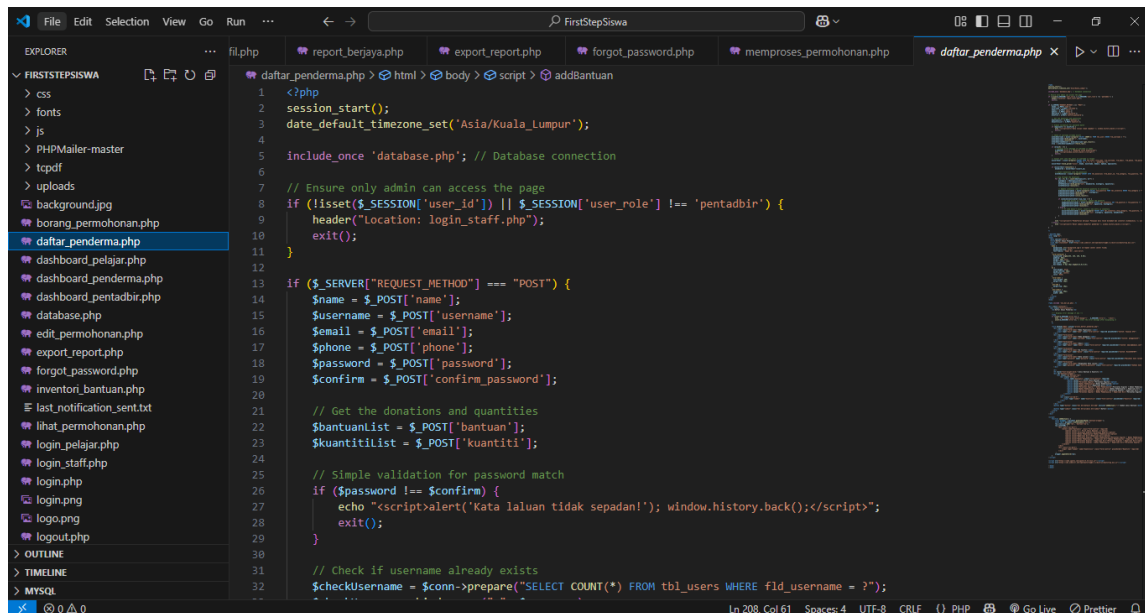
menggunakan pelayar web seperti *Google Chrome*. Rajah 4.1 menunjukkan perisian XAMPP manakala Rajha 4.2 menunjukkan perisian phpMyAdmin.



Rajah 4.1 Perisian XAMPP



Rajah 4.2 Perisian phpMyAdmin



Rajah 4.3 Perisian *Visual Studio Code*

4.2.1 Segmen Kod Kritikal

Segmen kod kritikal ialah bahagian kod yang penting dan perlu berfungsi dengan betul supaya sistem dapat berjalan lancar dan selamat. Dalam pembangunan sistem web, segmen ini biasanya melibatkan proses seperti sambungan ke pangkalan data, penyimpanan data penting serta kemaskini maklumat pengguna. Sekiranya berlaku kesilapan pada kod ini, ia boleh menjejaskan keseluruhan sistem dan menyebabkan kehilangan data. Oleh itu, aspek keselamatan dan ketepatan kod amat dititikberatkan dalam pembangunan aplikasi web masa kini (Tharani & Nagarajan, 2021).

Dalam sistem First Step Siswa, terdapat beberapa segmen kod kritikal yang memainkan peranan utama. Rajah 4.4 menunjukkan segmen kod bagi sambungan pelayan web ke pangkalan data. Sambungan kod pelayan ke pangkalan data adalah penting supaya semua maklumat seperti data pelajar, permohonan dan bantuan dapat diakses dan disimpan oleh sistem.



Rajah 4.4 Segmen Kod *database.php*

Rajah 4.5 dan Rajah 4.6 menunjukkan segmen kod yang digunakan dalam fungsi log masuk pengguna. Ia penting bagi memastikan hanya pengguna yang sah seperti pelajar, penderma atau pentadbir boleh masuk ke sistem. Ia juga menentukan papan muka yang betul mengikut peranan pengguna. Jika kod ini gagal, pengguna tidak boleh log masuk atau mungkin dapat akses bahagian yang tidak sepatutnya. Kod ini akan menyemak nama pengguna dan kata laluan yang dihantar. Jika maklumat sah dan peranan pengguna sesuai, sistem akan simpan maklumat pengguna ke dalam `$_SESSION` dan terus arahkan ke papan muka yang betul. Jika tidak sah, mesej ralat akan dipaparkan. Logik ini penting kerana ia menjadi pengawal utama untuk akses pengguna kepada fungsi-fungsi sensitif seperti pengurusan permohonan dan tajaan.

```
if (isset($_POST['login'])) {
    $username = $_POST['username'];
    $password = $_POST['password'];

    try {
        $stmt = $conn->prepare("SELECT * FROM tbl_users WHERE fld_username = :username");
        $stmt->bindParam(':username', $username, PDO::PARAM_STR);
        $stmt->execute();
        $user = $stmt->fetch(PDO::FETCH_ASSOC);

        if ($user && $user['fld_role'] === 'pelajar' && $password === $user['fld_password']) {
            $_SESSION['user_id'] = $user['fld_user_id'];
            $_SESSION['username'] = $user['fld_username'];
            $_SESSION['full_name'] = $user['fld_name'];
            $_SESSION['user_role'] = $user['fld_role'];
            $_SESSION['user_matric'] = $user['fld_matric_no'];

            $_SESSION['user_email'] = $user['fld_email'];
            $_SESSION['user_phone'] = $user['fld_phone'];

            header("Location: dashboard_pelajar.php");
            exit();
        } else {
            $error = "Nama pengguna atau kata laluan tidak sah untuk pelajar.";
        }
    } catch (PDOException $e) {
        $error = "Ralat: " . $e->getMessage();
    }
}
```

Rajah 4.5 Segmen Kod Log Masuk Pelajar

```
if (isset($_POST['login'])) {
    $username = $_POST['username'];
    $password = $_POST['password'];

    try {
        $stmt = $conn->prepare("SELECT * FROM tbl_users WHERE fld_username = :username");
        $stmt->bindParam(':username', $username, PDO::PARAM_STR);
        $stmt->execute();

        $user = $stmt->fetch(PDO::FETCH_ASSOC);

        if ($user && ($user['fld_role'] === 'pentadbir' || $user['fld_role'] === 'penderma') && $password === $user['fld_password']) {
            $_SESSION['user_id'] = $user['fld_user_id'];
            $_SESSION['username'] = $user['fld_username'];
            $_SESSION['full_name'] = $user['fld_name'];
            $_SESSION['user_role'] = $user['fld_role'];

            $redirect = $user['fld_role'] === 'pentadbir' ? 'dashboard_pentadbir.php' : 'dashboard_penderma.php';
            header("Location: $redirect");
            exit();
        } else {
            $error = "Nama pengguna atau kata laluan tidak sah untuk pentadbir atau penderma.";
        }
    } catch (PDOException $e) {
        $error = "Ralat: " . $e->getMessage();
    }
}
```

Rajah 4.6 Segmen Kod Log Masuk Penderma dan Pentadbir

Rajah 4.7 menunjukkan segmen kod pendaftaran akaun pelajar baru. Dalam segmen ini, sistem akan menyemak kesahihan data seperti kesamaan kata laluan dan kewujudan nama pengguna sebelum menyimpan maklumat ke dalam pangkalan data. Fungsi ini memastikan setiap pelajar mendaftar dengan maklumat yang tepat dan unik. Penggunaan *prepared statement* juga membantu menjaga keselamatan data daripada serangan *SQL Injection*.

```
$check = $conn->prepare("SELECT COUNT(*) FROM tbl_users WHERE fld_username = :username");
$check->bindParam(':username', $username);
$check->execute();

if ($check->fetchColumn() > 0) {
    $_SESSION['error'] = "Nama pengguna telah didaftarkan. Sila pilih yang lain.";
} else {
    try {
        $stmt = $conn->prepare("INSERT INTO tbl_users (fld_name, fld_email, fld_phone, fld_username, fld_matric_no, fld_password, fld_role)
                                VALUES (:name, :email, :phone, :username, :matric_no, :password, :role)");
        $stmt->bindParam(':name', $name);
        $stmt->bindParam(':email', $email);
        $stmt->bindParam(':phone', $phone);
        $stmt->bindParam(':username', $username);
        $stmt->bindParam(':matric_no', $matric_no);
        $stmt->bindParam(':password', $password);
        $stmt->bindParam(':role', $role);

        $stmt->execute();
        $_SESSION['success'] = "Daftar Akaun Berjaya!";
        echo "<script>alert('Daftar Akaun Berjaya!'); window.location.href='login_pelajar.php';</script>";
        exit();
    } catch (PDOException $e) {
        $_SESSION['error'] = "Ralat Sebenar: " . $e->getMessage();
    }
}
```

Rajah 4.7 Segmen Kod Daftar Akaun Baru oleh Pelajar

Rajah 4.8 menunjukkan segmen kod yang digunakan dalam fail proses_permohonan.php. Kod ini berperanan penting kerana menguruskan dua fungsi utama, iaitu simpanan draf permohonan dan penghantaran permohonan bantuan. Bahagian ini termasuk semakan data, pengurusan fail dokumen (*PDF*) dan simpanan data ke dalam pangkalan data jadual *tbl_requests*. Jika kod ini tidak berfungsi dengan betul, permohonan pelajar tidak akan direkodkan dan sistem tidak dapat menjalankan proses penyaluran bantuan.

```

proses_permohonan.php x
proses_permohonan.php > ...
51  if (isset($_POST['submit'])) {
52      $user_id = $_POST['fld_user_id'];
53      $name = mysqli_real_escape_string($conn, $_POST['fld_name']);
54      $matric_no = mysqli_real_escape_string($conn, $_POST['fld_matric_no']);
55      $phone = mysqli_real_escape_string($conn, $_POST['fld_phone']);
56      $email = mysqli_real_escape_string($conn, $_POST['fld_email']);
57      $category = $_POST['fld_category'] ?? '';
58      $status = 'sedang diproses';
59      $msg = "Permohonan berjaya dihantar dan sedang diproses.";
60
61      // Check if files are in the draft (use the stored paths from session)
62      $income_path = $_SESSION['draft']['fld_income_slip'] ?? '';
63      $support_path = $_SESSION['draft']['fld_supporting_doc'] ?? '';
64
65      // If files are not in the session, upload new ones
66      if ($_FILES['fld_income_slip']['error'] == 0) {
67          $income_slip_name = $_FILES['fld_income_slip']['name'];
68          $income_slip_tmp = $_FILES['fld_income_slip']['tmp_name'];
69          $income_path = "uploads/income_slip/" . basename($income_slip_name);
70          if (!file_exists("uploads/income_slip")) mkdir("uploads/income_slip", 0777, true);
71          move_uploaded_file($income_slip_tmp, $income_path);
72      }
73
74      // Handle supporting document upload if present
75      if ($_FILES['fld_supporting_doc']['error'] == 0) {
76          $support_doc_name = $_FILES['fld_supporting_doc']['name'];
77          $support_doc_tmp = $_FILES['fld_supporting_doc']['tmp_name'];
78          $support_path = "uploads/support_doc/" . basename($support_doc_name);
79          if (!file_exists("uploads/support_doc")) mkdir("uploads/support_doc", 0777, true);
80          move_uploaded_file($support_doc_tmp, $support_path);
81      }
82

```

Rajah 4.8 Segmen Kod Proses Permohonan Oleh Pelajar

Rajah 4.9 menunjukkan segmen kod untuk memaparkan status permohonan bantuan pelajar secara individu. Segmen kod ini menghubungkan beberapa jadual dalam pangkalan data untuk mendapatkan maklumat permohonan dan penderma yang berkaitan. Fungsi ini penting untuk memberikan maklumat tepat dan terkini kepada pelajar mengenai status bantuan mereka. Selain itu, sistem memastikan hanya pelajar yang log masuk dapat mengakses data mereka bagi menjaga privasi.

```

<?php
session_start();
include_once 'database.php';

// Ensure logged-in student can access the page
if (!isset($_SESSION['user_id']) || $_SESSION['user_role'] !== 'pelajar') {
    header("Location: login_pelajar.php");
    exit();
}

$user_id = $_SESSION['user_id']; // Student's user ID

// Query to get the most recent request and donor details (if any)
$query = "
    SELECT r.*, u.fld_name AS donor_name
    FROM tbl_requests r
    LEFT JOIN tbl_sponsorships s ON r.fld_request_id = s.fld_request_id
    LEFT JOIN tbl_users u ON s.fld_sponsor_id = u.fld_user_id
    WHERE r.fld_user_id = '$user_id'
    ORDER BY r.fld_request_id DESC
    LIMIT 1";
$result = mysqli_query($conn, $query);
$data = mysqli_fetch_assoc($result);
?>

```

Rajah 4.9 Segmen Kod Status Permohonan Pelajar

Rajah 4.10 menunjukkan segmen kod yang memaparkan senarai permohonan bantuan pelajar yang telah diluluskan kepada penderma. Fungsi ini membolehkan penderma melihat maklumat seperti nama pelajar, nombor telefon, emel dan jenis bantuan. Fungsi carian juga disediakan untuk menapis permohonan berdasarkan jenis bantuan. Segmen kod ini penting kerana ia membantu penderma memilih permohonan yang sesuai untuk ditaja. Sistem turut memastikan hanya penderma sahaja boleh mengakses halaman ini bagi menjaga keselamatan data.

```

<?php
include_once 'database.php';
session_start();
date_default_timezone_set('Asia/Kuala_Lumpur');

if (!isset($_SESSION['user_id']) || $_SESSION['user_role'] !== 'penderma') {
    header("Location: login_staff.php");
    exit();
}

$search = $_GET['search'] ?? '';

$sql = "SELECT * FROM tbl_requests WHERE fld_status = 'diluluskan'";
if ($search) {
    $sql .= " AND fld_category LIKE '%" . mysqli_real_escape_string($conn, $search) . "%'";
}
$sql .= " ORDER BY fld_request_id DESC";
$result = mysqli_query($conn, $sql);
?>

```

Rajah 4.10 Segmen Kod Senarai Permohonan Bantuan Diluluskan untuk Penderma

Rajah 4.11 dan 4.12 menunjukkan segmen kod yang memaparkan senarai permohonan yang telah ditaja oleh penderma. Dalam modul ini, penderma dapat melihat maklumat pelajar yang telah mereka taja termasuk nama, nombor telefon, email, jenis bantuan dan tarikh penajaan. Fungsi carian dan *pagination* turut disediakan untuk memudahkan pencarian dan pengurusan data. Fungsi ini penting untuk memudahkan penderma memantau bantuan yang telah disalurkan serta memastikan rekod penajaan sentiasa terurus.

```
<?php
include_once 'database.php';
session_start();
date_default_timezone_set('Asia/Kuala_Lumpur');

if (!isset($_SESSION['user_id']) || $_SESSION['user_role'] !== 'penderma') {
    header("Location: login_staff.php");
    exit();
}

$sponsor_id = $_SESSION['user_id'];
$search = $_GET['search'] ?? '';
$itemsPerPage = 5;
$page = isset($_GET['page']) && is_numeric($_GET['page']) ? intval($_GET['page']) : 1;
$offset = ($page - 1) * $itemsPerPage;

$countSql = "SELECT COUNT(*) as total FROM tbl_requests r INNER JOIN tbl_sponsorships s ON r.fld_request_id = s.fld_request_id
    WHERE s.fld_sponsor_id = '$sponsor_id' AND r.fld_status = 'berjaya'";

if ($search) {
    $safe = mysqli_real_escape_string($conn, $search);
    $countSql .= " AND r.fld_category LIKE '%$safe%'";
}

$countResult = mysqli_query($conn, $countSql);
$totalRows = mysqli_fetch_assoc($countResult)['total'];
$totalPages = ceil($totalRows / $itemsPerPage);

$sql = "SELECT r.*, s.fld_created_at FROM tbl_requests r INNER JOIN tbl_sponsorships s ON r.fld_request_id = s.fld_request_id
    WHERE s.fld_sponsor_id = '$sponsor_id' AND r.fld_status = 'berjaya'";
```

Rajah 4.11 Segmen Kod Penajaan Permohonan

```
if ($search) {
    $safe = mysqli_real_escape_string($conn, $search);
    $sql .= " AND r.fld_category LIKE '%$safe%'";
}

$sql .= " ORDER BY s.fld_created_at DESC LIMIT $itemsPerPage OFFSET $offset";
$result = mysqli_query($conn, $sql);

?>
```

4.12 Segmen kod Penajaan Permohonan

Rajah 4.13 menunjukkan segmen kod memproses permohonan bantuan. Dalam fungsi ini, sistem akan menyenaraikan semua permohonan pelajar serta mengesan permohonan yang tertangguh iaitu yang belum diproses selepas 3 atau 7 hari dari tarikh permohonan. Jika ada permohonan seperti itu, sistem akan menghantar emel notifikasi kepada pentadbir untuk tindakan segera. Fungsi ini sangat penting supaya tiada permohonan yang terlepas pandang dan bantuan dapat disalurkan dengan cepat.

```
$notifSql = "SELECT fld_name, fld_matric_no, fld_category, fld_request_date FROM tbl_requests WHERE fld_status = 'sedang diproses'";
$notifResult = mysqli_query($conn, $notifSql);

while ($rowNotif = mysqli_fetch_assoc($notifResult)) {
    $daysSince = floor((strtotime(date('Y-m-d')) - strtotime($rowNotif['fld_request_date'])) / (60 * 60 * 24));
    $statusLabel = '';

    if ($daysSince >= 7) {
        $statusLabel = "<span style='color:red;'>Lambat (7+ hari)</span>";
        $sendEmail = true;
    } elseif ($daysSince >= 3) {
        $statusLabel = "<span style='color:orange;'>Tertangguh (3+ hari)</span>";
        $sendEmail = true;
    }

    if ($statusLabel) {
        $emailBody .= "<tr style='text-align: center;'>
            <td style='word-break: break-word; white-space: normal;'>" . htmlspecialchars($rowNotif['fld_name']) . "</td>
            <td style='word-break: break-word; white-space: normal;'>" . htmlspecialchars($rowNotif['fld_matric_no']) . "</td>
            <td style='word-break: break-word; white-space: normal;'>" . htmlspecialchars($rowNotif['fld_category']) . "</td>
            <td>$daysSince hari</td>
            <td>" . date('Y-m-d', strtotime($rowNotif['fld_request_date'])) . "</td>
            <td>" . ($daysSince >= 7
                ? "<span style='color:red;'>Lambat (7+ hari)</span>"
                : "<span style='color:orange;'>Tertangguh (3+ hari)</span>" . "</td>
        </tr>";
        $notificationMessages[] = "<strong>Permohonan oleh {$rowNotif['fld_name']} masih belum diproses selama $daysSince hari.</strong>";
    }
}
```

Rajah 4.13 Segmen kod Penjejakan Permohonan Tertunggak dan Notifikasi Emel kepada Pentadbir

Rajah 4.14 menunjukkan halaman perincian permohonan oleh pentadbir. Dalam halaman ini, pentadbir boleh melihat semua maklumat lengkap permohonan pelajar termasuk nama, nombor matrik, jenis bantuan dan fail dokumen yang dihantar. Segmen kod ini juga menyediakan butang untuk meluluskan atau menolak permohonan melalui kotak pengesahan. Jika pentadbir menekan butang "Sahkan", status permohonan akan dihantar ke fail proses_status_permohonan.php untuk dikemaskini.

```

function sahkanPermohonan(id) {
    Swal.fire({
        title: 'Adakah anda ingin meluluskan permohonan pelajar ini?',
        icon: 'question',
        showCancelButton: true,
        showCloseButton: true, // ✅ ADD THIS LINE
        confirmButtonText: 'Ya, Luluskan',
        cancelButtonText: 'Tidak, Tolak',
        confirmButtonColor: '#28a745',
        cancelButtonColor: '#dc3545'
    }).then((result) => {
        if (result.isConfirmed || result.dismiss === Swal.DismissReason.cancel) {
            let status = result.isConfirmed ? 'diluluskan' : 'ditolak';
            fetch('proses_status_permohonan.php?id=${id}&status=${status}')
                .then(response => response.text())
                .then(() => {
                    Swal.fire({
                        title: 'PERMOHONAN PELAJAR TELAH DIKEMASKINI',
                        html: `<a href='memproses_permohonan.php'>TEKAN DISINI UNTUK KEMBALI</a>`,
                        icon: 'success',
                        timer: 3000,
                        showConfirmButton: false
                    }).then(() => {
                        window.location.href = 'memproses_permohonan.php';
                    });
                });
        }
    });
}

```

Rajah 4.14 Segmen Kod Perincian Halaman Permohonan

Rajah 4.15 menunjukkan segmen kod bagi modul Inventori Bantuan di mana pentadbir boleh memantau senarai stok bantuan serta menambah atau mengurangkan kuantiti stok secara langsung. Kod ini menggunakan permintaan *Fetch API* untuk kemaskini data tanpa muat semula halaman dan memberikan notifikasi kejayaan atau ralat kepada pentadbir. Segmen ini penting kerana memastikan pengurusan stok bantuan berjalan lancar dan data inventori sentiasa tepat.

```

// Handle inventory updates (increase or decrease quantity)
if (isset($_GET['action']) && isset($_GET['id'])) {
    $item_id = $_GET['id'];
    $action = $_GET['action'];

    // Check the current quantity in the inventory
    $checkQuery = "SELECT fld_quantity FROM tbl_inventory WHERE fld_item_id = ?";
    $checkStmt = $conn->prepare($checkQuery);
    $checkStmt->bind_param("i", $item_id);
    $checkStmt->execute();
    $result = $checkStmt->get_result();
    $inventory = $result->fetch_assoc();

    // Update the inventory based on the action
    if ($inventory) {
        $currentQuantity = $inventory['fld_quantity'];
        $newQuantity = ($action === 'tambah') ? $currentQuantity + 1 : ($currentQuantity > 0 ? $currentQuantity - 1 : $currentQuantity);

        // Update inventory
        $updateQuery = "UPDATE tbl_inventory SET fld_quantity = ? WHERE fld_item_id = ?";
        $updateStmt = $conn->prepare($updateQuery);
        $updateStmt->bind_param("ii", $newQuantity, $item_id);
        $updateStmt->execute();

        echo json_encode(["status" => "success", "message" => "Stok telah dikemaskini!"]);
    } else {
        echo json_encode(["status" => "error", "message" => "Stok tidak wujud!"]);
    }
}

exit(); // Stop further script execution
}

```

Rajah 4.15 Segmen Kod Inventori Bantuan

Seterusnya, Rajah 4.16 menunjukkan segmen kod yang digunakan untuk menghantar permintaan kemaskini stok dan memaparkan notifikasi.

```
function updateStock(id, action) {  
    fetch(`inventori_bantuan.php?action=${action}&id=${id}`)  
    .then(response => response.json())  
    .then(data => {  
        Swal.fire({  
            title: 'NOTIFIKASI',  
            text: data.message,  
            icon: data.status, // success or error  
            confirmButtonText: 'OK'  
        }).then(() => location.reload());  
    });  
}
```

Rajah 4.16 Segmen Kod Kemaskini Stok dan Papar Notifikasi

Rajah 4.17 dan Rajah 4.18 menunjukkan kod pengurusan pendaftaran penderma baru oleh pentadbir. Kod ini bukan sahaja menyimpan maklumat akaun penderma, malah turut menyimpan jenis bantuan dan kuantiti yang akan disumbangkan ke dalam jadual tbl_donations. Selain itu, ia juga mengemaskini jadual inventori dengan menambah stok bantuan baru atau kemaskini stok sedia ada. Segmen ini sangat penting bagi memastikan rekod penderma dan stok bantuan sentiasa tepat dan teratur dalam sistem.

```
$checkUsername = $conn->prepare("SELECT COUNT(*) FROM tbl_users WHERE fld_username = ?");  
$checkUsername->bind_param("s", $username);  
$checkUsername->execute();  
$checkUsernameResult = $checkUsername->get_result();  
$row = $checkUsernameResult->fetch_row();  
  
if ($row[0] > 0) {  
    // Display error message if username is taken  
    $_SESSION['error'] = "Penderma telah didaftarkan";  
    echo "<script>window.history.back();</script>";  
    exit();  
}  
  
// Insert user into tbl_users if username is unique  
$insertUser = $conn->prepare("INSERT INTO tbl_users (fld_name, fld_username, fld_email, fld_phone, fld_password, fld_role)  
VALUES (?, ?, ?, ?, ?, 'penderma')");  
$insertUser->bind_param("sssss", $name, $username, $email, $phone, $password);  
  
if ($insertUser->execute()) {  
    $newUserId = $insertUser->insert_id;  
  
    // Insert donations into tbl_donations  
    $stmtDonation = $conn->prepare("INSERT INTO tbl_donations (fld_donor_id, fld_category, fld_quantity, fld_donation_date) VALUES (?, ?, ?, NOW())");
```

Rajah 4.17 Daftar Penderma Baru

```

// Loop through the donations
for ($i = 0; $i < count($bantuanList); $i++) {
    $category = $bantuanList[$i];
    $quantity = (int)$kuantitasList[$i];
    $stmtDonation->bind_param("isi", $newUserId, $category, $quantity);
    $stmtDonation->execute();

    // Update inventory (add the donation quantity to inventory)
    $checkInventoryStmt = $conn->prepare("SELECT fld_quantity FROM tbl_inventory WHERE fld_category = ?");
    $checkInventoryStmt->bind_param("s", $category);
    $checkInventoryStmt->execute();
    $checkInventoryStmt->store_result();

    if ($checkInventoryStmt->num_rows > 0) {
        // If item exists in inventory, update the quantity
        $updateInventoryStmt = $conn->prepare("UPDATE tbl_inventory SET fld_quantity = fld_quantity + ? WHERE fld_category = ?");
        $updateInventoryStmt->bind_param("is", $quantity, $category);
        $updateInventoryStmt->execute();
    } else {
        // If item doesn't exist, insert a new entry in inventory
        $insertInventoryStmt = $conn->prepare("INSERT INTO tbl_inventory (fld_category, fld_quantity, fld_donor_id, fld_last_updated) VALUES (?, ?, ?, ?)");
        $insertInventoryStmt->bind_param("sis", $category, $quantity, $newUserId);
        $insertInventoryStmt->execute();
    }
}

echo "<script>alert('Pendaftaran berjaya! Penajaan baru telah ditambah dan inventori dikemaskini.');

```

Rajah 4.18 Segmen Kod Daftar Penderma Baru

RUJUKAN

Tharani, T. R., & Nagarajan, G. (2021). The study on assessment of security web applications. International Journal of Engineering Research & Technology (IJERT), 10(11), 1-4.
<https://www.researchgate.net/publication/356888529>