

تسجيل جديد ابيه ينقله على حسب الدول للصفحة المناسبة

خرج req_system.php ,

دكتور requests.php

```
// Insert into graduates or professors

if($role=='graduate'){

$stmt2 = $conn->prepare("INSERT INTO graduates (user_id) VALUES (?)");

$stmt2->bind_param("i",$user_id);

$stmt2->execute();

$stmt2->close();

} elseif($role=='professor'){

$stmt2 = $conn->prepare("INSERT INTO professors (user_id,specialization)
VALUES (?,?)");

$stmt2->bind_param("is",$user_id,$specialization);

$stmt2->execute();

$stmt2->close();

}

$_SESSION['message'] = "  Registration successful!";

header("Location: login.php");

exit;

}

$stmt->close();

} else {

$_SESSION['message'] = "  Please fill all required fields";

}
```

صفحة الطلب ابى اذا حالة الطلب درافت تظخر للخريج تحت الاجراء عند اكماله وظهور كلمة كومبليت يظهر زر تحميل

إضافة شرط في حال كان عنده اربع طلبات لا يسمح له بطلب خامس

```
<button class="btn" onclick="window.location.href='new_request.php'">  
+ New Recommendation Request  
</button>
```

```
<h3>My Requests</h3>
```

```
<table>  
<tr>  
  <th>#</th>  
  <th>Professor</th>  
  <th>Date</th>  
  <th>Status</th>  
  <th>Actions</th>  
</tr>  
<?php  
// الذي تم جلبه في بداية الملف $result تم إزالة الاستعلام المكرر هنا. يتم استخدام X
```

```
if ($result->num_rows > 0) {  
    while($row = $result->fetch_assoc()) {  
        // سينظهر اسم الأستاذ الآن 🚀  
        $professor_name = $row['professor_name'] ?? '—';
```

```

echo

"<tr>

<td>".$row['id']."'</td>
<td>".$professor_name."</td>
<td>".$row['created_at']."'</td>
<td
class='".($strtolower($row['status'])=="pending"?"pending":"accepted")."'>".$row['status']."'</td>

<td class='actions'>
<button class='edit' onclick=\"editRequest(\".$row['id'].\")\"></button>
Edit</button>

<button class='delete' onclick=\"deleteRequest(\".$row['id'].\", this)\"></button>
Delete</button>

</td>
</tr>";

}

} else {

echo "<tr><td colspan='5'>No requests found</td></tr>";

}
?>

</table>

```

</div>

عند مشكلة ان اذا ضغطت موافقه او رفض ماتتغير الا اذا حذفت الصفحة واذا تغيرت حالة الطلب غير عنهم مثل انه اكتمل او درافت يصير كنه جديد والطلب ماحذف من الصفحة نهائى يرجع اذا سوينت تحديث لها(ابي اذا ضغطت زر السلة للحذف مايصير يظهر نهائى بس انه ماحذف من القاعدة)

```
if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['request_id'],  
$_POST['action'])) {
```

```
    $request_id = intval($_POST['request_id']);
```

```
    $action = $_POST['action'];
```

```
    if ($action === 'accept' || $action === 'reject') {
```

```
        $newStatus = ($action === 'accept') ? 'accepted' : 'rejected';
```

تحديث حالة الطالب //

```
        $stmt = $conn->prepare("UPDATE requests SET status = ? WHERE id = ?");
```

```
        $stmt->bind_param("si", $newStatus, $request_id);
```

```
        $stmt->execute();
```

```
        $stmt->close();
```

واسم الطالب لإرسال الإشعار user_id جلب //

```
$res = $conn->prepare("SELECT u.id AS student_user_id, u.name AS  
student_name, r.purpose
```

```
    FROM requests r
```

```
    JOIN users u ON r.user_id = u.id
```

```
    WHERE r.id = ?");
```

```
$res->bind_param("i", $request_id);
```

```
$res->execute();
```

```
$row = $res->get_result()->fetch_assoc();
```

```
$student_user_id = $row['student_user_id'];
```

```
$student_name = $row['student_name'];
```

```
$purpose = $row['purpose'];
```

```
$res->close();
```

إرسال إشعار للطالب //

```
$message = "Your request " . $purpose . " has been " . $newStatus . " by the professor.;"
```

```
$notif = $conn->prepare("INSERT INTO notifications (user_id, message, created_at) VALUES (?, ?, NOW())");
```

```
$notif->bind_param("is", $student_user_id, $message);
```

```
$notif->execute();
```

```
$notif->close();
```

// ★ (إضافة الوحيدة المطلوبة) إضافة سجل في track_request

```
$profUserId = $_SESSION['user_id'];
```

```
$statusTrack = ($action === 'accept') ? 'Professor Approval' : 'Professor Rejection';
```

```
$noteTrack = ($action === 'accept') ? 'Approved by Professor' : 'Rejected by Professor';
```

```
$track = $conn->prepare("INSERT INTO track_request (request_id, user_id, status, note) VALUES (?, ?, ?, ?)");
```

```
$track->bind_param("iiss", $request_id, $profUserId, $statusTrack, $noteTrack);
```

```
$track->execute();
```

```
$track->close();
```

```
echo $newStatus;
```

```
exit;
```

```
}
```

```
}
```

```
<h2>Incoming Recommendation Requests</h2>
```

```
<section class="cards">

<?php

if ($list_res->num_rows === 0) {

    echo "<div style='grid-column:1/-1;text-align:center;color:#666;'>No requests
found.</div>";

} else {

    while ($r = $list_res->fetch_assoc()):

        $rid = (int)$r['id'];

        $graduate_name = htmlspecialchars($r['graduate_name']);

        $date = htmlspecialchars($r['created_at']);

        $type = htmlspecialchars($r['type']);

        $purpose = htmlspecialchars($r['purpose']);

        $status = $r['status'] ?? "";

    ?>

    <div class="card">

        <h3><?= $graduate_name ?></h3>

        <p><strong>Date:</strong> <?= $date ?></p>

        <p><strong>Type:</strong> <?= $type ?></p>

        <p><strong>Purpose:</strong> <?= $purpose ?></p>

    <div class="card-actions" id="request-<?= $rid ?>">

        <?php if (strtolower($status) === 'accepted'): ?>

            <div class="status-box accepted">Accepted</div>
```

```

<a href="recommendation-writing.php?id=<?= $rid ?>" class="btn-accept"
style="margin-left:10px;"><img alt="Pencil icon" style="vertical-align: middle; height: 1em;"/> </a>

<?php elseif ($status === 'rejected'): ?>

<div class="status-box rejected">Rejected</div>

<button type="button" class="btn-delet" style="margin-left:10px;" onclick="deleteCard(<?= $rid ?>)"><img alt="Delete icon" style="vertical-align: middle; height: 1em;"/> </button>

<?php else: ?>

<button type="button" class="btn-accept" onclick="updateStatus(<?= $rid ?>, 'accept')">Accept</button>

<button type="button" class="btn-reject" onclick="updateStatus(<?= $rid ?>, 'reject')">Reject</button>

<?php endif; ?>

</div>

</div>

<?php endwhile; } ?>

</section>

</div>

```

```

<script>

const toggleBtn = document.getElementById("toggleBtn");

const sidebar = document.getElementById("sidebar");

toggleBtn.addEventListener("click", () => {

  sidebar.classList.toggle("collapsed");

});


```

تحديث حالة الطلب بدون إعادة تحميل الصفحة //

```
function updateStatus(requestId, action) {
```

```

const formData = new FormData();

formData.append('request_id', requestId);
formData.append('action', action);

fetch('', { method: 'POST', body: formData })

.then(response => response.text())

.then(status => {

  const container = document.getElementById('request-' + requestId);

  if (status === 'accepted') {

    container.innerHTML = '<div class="status-box accepted">Accepted</div><a href="recommendation-writing.php?id=' + requestId + '" class="btn-accept" style="margin-left:10px;">📝 </a>';

  } else if (status === 'rejected') {

    container.innerHTML = '<div class="status-box rejected">Rejected</div><button type="button" class="btn-reject" style="margin-left:10px;" onclick="deleteCard(' + requestId + ')">ลบ</button>';

  }

})

.catch(error => console.error('Error:', error));

}

// حذف الکارد عند الضغط على زر "Delete"

function deleteCard(requestId) {

  const card = document.getElementById('request-' + requestId).parentElement;

  card.remove();

}

</script>

```

هنا صار ماتحفظ المسودة اول التوصية اللي أرسلتها تتغير الحالة لكن لما ارجع للطلب تكون المربع للنص فاضي

```
$request_id = isset($_GET['id']) ? intval($_GET['id']) : 0;
```

```
if ($request_id <= 0) die("Invalid request ID.");
```

جلب بيانات الطلب والخرج //

```
$sql = "SELECT
```

```
    g.graduate_id,  
    r.id AS request_id,  
    r.user_id,  
    r.major,  
    r.purpose,  
    r.type AS recommendation_type,  
    r.status,  
    r.course,  
    r.grades_file,  
    r.file_name,  
    g.gpa,  
    g.graduation_year,  
    g.cv_path,  
    u.name,  
    u.email,  
    u.department,  
    u.National_ID,  
    u.university
```

```
FROM requests r
```

```
INNER JOIN users u ON r.user_id = u.id  
INNER JOIN graduates g ON u.id = g.user_id  
WHERE r.id = ?";
```

```
$stmt = $conn->prepare($sql);  
$stmt->bind_param("i", $request_id);  
$stmt->execute();  
$result = $stmt->get_result();  
  
if ($result && $result->num_rows > 0) {  
    $graduate = $result->fetch_assoc();  
    $graduate_id = $graduate['graduate_id'];  
    $student_user_id = $graduate['user_id'];  
}  
else {  
    die("✖ Request not found or graduate not found.");  
}
```

جلب مسودة سابقة (إن وجدت) //

```
$recommendation = null;  
$rec_query = $conn->prepare("SELECT * FROM recommendations WHERE  
graduate_id = ? AND professor_id = 1");  
$rec_query->bind_param("i", $graduate_id);  
$rec_query->execute();  
$rec_result = $rec_query->get_result();  
if ($rec_result->num_rows > 0) {  
    $recommendation = $rec_result->fetch_assoc();
```

}

\$message_alert = ""; // متغير الرسالة

حفظ التوصية عند الإرسال أو المسودة //

if (\$_SERVER['REQUEST_METHOD'] === 'POST') {

\$content = \$_POST['recommendation_text'];

\$status = \$_POST['action']; // draft أو sent

\$professor_id = \$_SESSION['user_id']; // لا حفظ session

\$request_id = \$graduate['request_id'];

\$student_user_id = \$graduate['user_id'];

جاء النص من الفورم //

\$content = \$_POST['recommendation_text'];

// إزالة التعليقات الخاصة ب Word/VML **1**

\$clean_content = preg_replace('/<!--\[if.*?<!\[endif\]-->/is', "", \$content);</p>

// إزالة أكواد VML مثل <v:shape> و <v:shapetype> **2**

\$clean_content = preg_replace('/<v:.*?</v:.*?>/is', "", \$clean_content);

// إزالة أكواد Office خاصة ب <o:p>

\$clean_content = preg_replace('/<o:p>\s*</o:p>/is', "", \$clean_content);

// إزالة السمات المزعجة داخل Word span (mso-*) **4**

```
$clean_content = preg_replace('/<span[^>]*mso-[^>]*>/is', '<span>',  
$clean_content);
```

إزالة أي فقرات فارغة أو زائدة // 5

```
$clean_content = preg_replace('/<p[^>]*>\s*</p>/is', "", $clean_content);
```

تحويل النص ل UTF-8 // 6

```
$content_utf8 = mb_convert_encoding($clean_content, 'UTF-8', 'auto');
```

```
$pdf = new TCPDF();
```

```
$pdf->AddPage();
```

```
$pdf->SetFont('helvetica', '', 14);
```

كتابة النص مع الحفاظ على بعض التنسيقات //

```
$pdf->writeHTML($content_utf8, true, false, true, false, "");
```

كتابة النص داخل PDF //

```
$pdf->MultiCell(0, 10, $content_utf8);
```

تحديد مجلد الرفع الكامل //

```
$upload_dir = __DIR__. '/uploads';
```

إذا المجلد ما ظهر، يتم إنشاؤه تلقائياً بصلاحيات الكتابة //

```
if (!is_dir($upload_dir)) {  
    mkdir($upload_dir, 0777, true);  
}
```

تحديد اسم ومسار الملف الكامل //

```
$pdf_file = $upload_dir . '/recommendation_' . time() . '.pdf';
```

حفظ الملف داخل المجلد المحدد //

```
$pdf->Output($pdf_file, 'F');
```

جلب معرف الدكتور الفعلى من جدول professors //

```
$get_prof = $conn->prepare("SELECT professor_id FROM professors WHERE user_id = ?");
```

```
$get_prof->bind_param("i", $_SESSION['user_id']);
```

```
$get_prof->execute();
```

```
$prof_result = $get_prof->get_result();
```

```
$professor_id = $prof_result->fetch_assoc()['professor_id'];
```

تحقق إذا كانت التوصية موجودة مسبقا //

```
$check = $conn->prepare("SELECT recommendation_id FROM recommendations WHERE graduate_id = ? AND professor_id = ?");
```

```
$check->bind_param("ii", $graduate_id, $professor_id);
```

```
$check->execute();
```

```
$exists = $check->get_result();
```

إذا موجودة → تحديث، إذا لا → إدخال جديد //

```
if ($exists->num_rows > 0) {
```

```
    $update = $conn->prepare("UPDATE recommendations SET content = ?, date_created = NOW(), request_id = ? WHERE graduate_id = ? AND professor_id = ?");
```

```

$update->bind_param("siii", $content, $request_id, $graduate_id, $professor_id);
$update->execute();
} else {
    $insert = $conn->prepare("INSERT INTO recommendations (graduate_id,
professor_id, content, date_created, request_id) VALUES (?, ?, ?, ?, NOW(), ?)");
    $insert->bind_param("iisi", $graduate_id, $professor_id, $content, $request_id);
    $insert->execute();
}

```

تحديث حالة الطالب //

```

if ($status === 'draft') {
    $req_update = $conn->prepare("UPDATE requests SET status = 'draft' WHERE id
= ?");

```

```

$req_update->bind_param("i", $request_id);
$req_update->execute();

```

```

$message_alert = "  The recommendation has been saved as a draft.";

```

```

} elseif ($status === 'completed') {

```

```

$req_update = $conn->prepare("UPDATE requests SET status = 'completed'
WHERE id = ?");

```

```

$req_update->bind_param("i", $request_id);
$req_update->execute();

```

إرسال إشعار للطالب //

```

$message = "Your recommendation has been sent by the professor.";

```

```

$notif = $conn->prepare("INSERT INTO notifications (user_id, message,
created_at) VALUES (?, ?, NOW())");

```

```

$notif->bind_param("is", $student_user_id, $message);

$notif->execute();

}

}

$message_alert = " ✅ The recommendation has been sent successfully!";

}

 هنا ارسال الاشعار يضبط لكن التتبع مايضبط مره
// ارسال اشعار للطلاب //



$message = "Your request " . $purpose . " has been " . $newStatus . " by the
professor.';

$notif = $conn->prepare("INSERT INTO notifications (user_id, message,
created_at) VALUES (?, ?, NOW())");

$notif->bind_param("is", $student_user_id, $message);

$notif->execute();

$notif->close();

// ⭐️ (إضافة الوحيدة المطلوبة) / إضافة سجل في track_request
$profUserId = $_SESSION['user_id'];

$statusTrack = ($action === 'accept') ? 'Professor Approval' : 'Professor
Rejection';

$noteTrack = ($action === 'accept') ? 'Approved by Professor' : 'Rejected by
Professor';

$track = $conn->prepare("INSERT INTO track_request (request_id, user_id,
status, note) VALUES (?, ?, ?, ?)");

$track->bind_param("iiss", $request_id, $profUserId, $statusTrack, $noteTrack);

$track->execute();

```

```

$track->close();

echo $newStatus;

exit;

هذا صفحة التتبع

<?php

session_start();

error_reporting(E_ALL);

ini_set('display_errors', 1);

/* track_request.php

- Timeline for each request with step timestamps

- LTR layout, Poppins, sidebar same style as other pages

*/



/* ----- DB connection ----- */

if (file_exists(__DIR__. '/db.php')) {

    require_once __DIR__. '/db.php'; // expects $conn (mysqli)

} else {

    include 'index.php';

    التحقق من تسجيل الدخول //


    if (!isset($_SESSION['user_id']) || $_SESSION['role'] != 'graduate') {

        header("Location: login.php");

        exit;

    }
}

```

```

}

$user_id = $_SESSION['user_id'];

/*
track_request.php
- Detailed timeline view for user's requests (LTR / English)
- Includes optional demo-data button to populate sample requests + track entries
for testing
- IMPORTANT: demo-data is for local testing only; remove it in production
*/
/* ----- DB connection (uses db.php if present) -----
if (file_exists(__DIR__ . '/db.php')) {
    (require_once(__DIR__ . '/db.php')); // must define $conn (mysqli)
} else {
    // fallback: change these if your MySQL credentials differ
    $host = "localhost";
    $user = "root";
    $pass = "";
    $dbname = "agdb";
    $conn = new mysqli($host, $user, $pass, $dbname);
    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
    }
}

```

```

----- Auth check -----
if (!isset($_SESSION['user_id'])) {
    // For quick local testing you can uncomment the following two lines:
    // $_SESSION['user_id'] = 1;
    // $_SESSION['role'] = 'student';
    die('Please log in first (or set $_SESSION["user_id"] for local testing).');
}

$user_id = intval($_SESSION['user_id']);

*/
/* ----- Ensure tables exist (safe operations) ----- */
/* We create minimal tables only if missing so the page can run locally for testing.
   If you already have tables in your app, these CREATE TABLE IF NOT EXISTS will do
   nothing.
*/
$create_requests_sql = "
CREATE TABLE IF NOT EXISTS requests (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    title VARCHAR(255) DEFAULT '',
    purpose TEXT DEFAULT NULL,
    status VARCHAR(100) DEFAULT 'Created',
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
";
$conn->query($create_requests_sql);

```

```

$create_track_sql = "
CREATE TABLE IF NOT EXISTS track_request (
    id INT AUTO_INCREMENT PRIMARY KEY,
    request_id INT NOT NULL,
    user_id INT NOT NULL,
    status VARCHAR(100) NOT NULL,
    note VARCHAR(255) DEFAULT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    INDEX (request_id),
    INDEX (user_id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
";

$conn->query($create_track_sql);

/* ----- Demo data insertion (optional) -----

$demo_msg = "";
if($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['insert_demo'])) {
    // create one demo request for this user + track entries
    $now = date('Y-m-d H:i:s');

    $stmt = $conn->prepare("INSERT INTO requests (user_id, purpose, status,
    created_at) VALUES (?, ?, ?, ?)");
    $title = "Demo Recommendation Request";
    $purpose = "Recommendation for Master's application";
    $status = "Created";
    $stmt->bind_param("isss", $user_id, $purpose, $status, $now);
}

```

```

if ($stmt->execute()) {

    $newReqId = $stmt->insert_id;

    $stmt->close();

    // insert track entries for that request (chronological)

    $tracks = [
        ['Created', 'Student submitted request', date('Y-m-d H:i:s', strtotime('-3 days'))],
        ['Under Review', 'Office checking documents', date('Y-m-d H:i:s', strtotime('-2 days'))],
        ['Professor Approval', 'Prof. A approved', date('Y-m-d H:i:s', strtotime('-1 day'))],
        ['Recommendation Sent', 'Letter uploaded', date('Y-m-d H:i:s')]
    ];

    $ins = $conn->prepare("INSERT INTO track_request (request_id, status,
created_at) VALUES (?, ?, ?)");

    foreach ($tracks as $tr) {

        $ins->bind_param("iss", $newReqId, $tr[0], $tr[2]);

        $ins->execute();

    }

    $ins->close();

    $demo_msg = "Demo request & track entries created (request id: $newReqId).
Refresh the page to see it./";

} else {

    $demo_msg = "Demo creation failed: ". $stmt->error;

}

}*/



/* ----- Fetch user's requests (newest first) ----- */

$requests = [];

```

```

if ($stmt = $conn->prepare("SELECT id, COALESCE(purpose, '') AS purpose,
COALESCE(purpose,'') AS purpose, status AS current_status, created_at FROM
requests WHERE user_id = ? ORDER BY created_at DESC")) {

$stmt->bind_param("i", $user_id);

$stmt->execute();

$res = $stmt->get_result();

$requests = $res->fetch_all(MYSQLI_ASSOC);

$stmt->close();

}

if (!$requests) $requests = [];

/* ----- Helpers ----- */

function safe($s){ return htmlspecialchars($s ?? '', ENT_QUOTES, 'UTF-8'); }

function fmt_dt($dt){ return safe($dt); }

/* Ordered steps to show in timeline (adjust labels if your app uses different words)
*/
$steps_order = [
    'Created' => 'Created',
    'Under Review' => 'Under Review',
    'Professor Approval' => 'Professor Approval',
    'Recommendation Sent' => 'Recommendation Sent'
];

/* match a track status text to a step key (customize keywords if needed) */
function match_step($trackStatus, $stepKey) {
    $t = strtolower($trackStatus);

```

```

$k = strtolower($stepKey);

if ($k === 'created') {

    return (strpos($t, 'create') !== false || strpos($t, 'submitted') !== false);

}

if ($k === 'under review') {

    return (strpos($t, 'under') !== false || strpos($t, 'review') !== false || strpos($t,
'pending') !== false);

}

if ($k === 'professor approval') {

    return (strpos($t, 'prof') !== false || strpos($t, 'approve') !== false || strpos($t,
'accepted') !== false);

}

if ($k === 'recommendation sent') {

    return (strpos($t, 'sent') !== false || strpos($t, 'uploaded') !== false || strpos($t,
'recommend') !== false);

}

return stripos($trackStatus, $stepKey) !== false;

}

?>

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<title>Track Requests</title>

<link
href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;700&display=
swap" rel="stylesheet">

```

```
<link rel="stylesheet" href="https://netdna.bootstrapcdn.com/font-awesome/4.0.3/css/font-awesome.css">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.6.0/css/all.min.css">

<style>

/* ◆ General Layout */

body{
    margin: 0;
    font-family: "Poppins", sans-serif;
    background: #fdfaf6;
    display: flex;
}

/* ◆ Sidebar */

.sidebar{
    background-color: #c8e4eb;
    width: 230px;
    transition: width 0.3s;
    height: 100vh;
    padding-top: 20px;
    box-shadow: 2px 0 5px rgba(0,0,0,0.1);
    position: fixed;
    display: flex;
    flex-direction: column;
    justify-content: space-between;
}
```

```
.sidebar.collapsed {  
    width: 70px;  
}  
  
.sidebar .logo {  
    text-align: center;  
    margin-bottom: 30px;  
}  
  
.sidebar .logo img{  
    width: 80px;  
}  
  
.menu-item {  
    display: flex;  
    align-items: center;  
    padding: 12px 20px;  
    color: #333;  
    text-decoration: none;  
    transition: background 0.3s;  
}  
  
.menu-item:hover {  
    background: #bcd5db;  
}  
  
.menu-item i{  
    font-size: 20px;  
    margin-right: 10px;  
    width: 25px;  
    text-align: center;
```

```
}

.menu-text {
    font-size: 15px;
    white-space: nowrap;
}

.sidebar.collapsed .menu-text {
    display: none;
}

.bottom-section {
    margin-bottom: 20px;
}

/* ◆ Toggle Button */

.toggle-btn {
    position: absolute;
    top: 20px;
    right: -15px;
    background: #003366;
    color: #fff;
    border-radius: 50%;
    border: none;
    width: 30px;
    height: 30px;
    cursor: pointer;
}
```

```
/* ◆ Top Bar */

.top-bar{
    position: fixed;
    top: 0;
    right: 0;
    left: 230px;
    height: 60px;
    display: flex;
    justify-content: flex-end;
    align-items: center;
    padding: 0 20px;
    transition: left 0.3s;
    z-index: 10;
}

.sidebar.collapsed ~ .top-bar{
    left: 70px;
}

.top-icons{
    display: flex;
    align-items: center;
    gap: 20px;
}

.icon-btn{
    background: none;
    border: none;
    cursor: pointer;
}
```

```
    font-size: 20px;  
    color: #333;  
}  
  
.icon-btn:hover {  
    color: #003366;  
}  
  
  
/* ◆ Main Content */  
  
.main-content {  
    margin-left: 230px;  
    margin-top: 70px;  
    padding: 30px;  
    transition: margin-left 0.3s;  
    width: 100%;  
}  
  
.sidebar.collapsed + .top-bar + .main-content {  
    margin-left: 70px;  
}  
  
.request-card{ background:#fff; border-radius:10px; padding:18px; margin-bottom:18px; border:1px solid #eef3f6; box-shadow:0 2px 6px rgba(0,0,0,0.06);}  
  
.request-header{ display:flex; justify-content:space-between; align-items:center; gap:10px; flex-wrap:wrap;}  
  
.req-title{ font-weight:700; color:#003366; }  
  
.timeline{ margin-top:12px; }  
  
.step{ display:flex; align-items:flex-start; gap:12px; padding:10px 0; border-bottom:1px dashed #eef3f6;}  
  
.step:last-child{ border-bottom:none;}
```

```
.circle{ width:36px; height:36px; border-radius:50%; display:flex; align-items:center; justify-content:center; color:#fff; font-size:18px; flex-shrink:0; }

.green{ background:#7adba2; } .yellow{ background:#f3d37a; color:#333; } .red{
background:#f26b6b; }

.status-text{ font-size:14px; color:#222; font-weight:500; }

.status-time{ font-size:12px; color:#888; margin-top:4px; }

.current{ box-shadow:0 0 0 4px rgba(122,219,162,0.12); }

.no-requests{ text-align:center; padding:26px; background:#fff; border-radius:8px;
border:1px dashed #cf8dc; color:#777; }

.demo-box{ margin-bottom:16px; padding:12px; background:#fff; border-radius:8px;
border:1px solid #eef3f6; }

.btn{ background:#7adba2; color:#fff; padding:8px 12px; border-radius:6px;
border:none; cursor:pointer; }
```

```
/* ◆ Responsive */

@media (max-width: 768px) {

    .main-content {
        margin-left: 70px;
    }

    .sidebar {
        width: 70px;
    }

    .menu-text {
        display: none;
    }
}
```

</style>

</head>

```
<body>

<div class="sidebar" id="sidebar">

    <button class="toggle-btn" id="toggleBtn"><i class="fas fa-bars"></i></button>

    <div>

        <div class="logo">

            

        </div>

        <a href="req_system.php" class="menu-item"><i class="fas fa-home"></i><span class="menu-text">Home</span></a>

        <a href="track_request.php" class="menu-item"><i class="fas fa-clock"></i><span class="menu-text">Track Request</span></a>

        <a href="student_profile.php" class="menu-item"><i class="fas fa-user"></i><span class="menu-text">Profile</span></a>

    </div>

<div class="bottom-section">

    <a href="setting_s.php" class="menu-item"><i class="fas fa-gear"></i><span class="menu-text">Notification Settings</span></a>

</div>

</div>

<div class="top-bar">

    <div class="top-icons">

        <button class="icon-btn" title="Notifications"
        onclick="window.location.href='notifications.php'"><i class="fas fa-bell"></i></button>

    </div>


```

```

<button class="icon-btn" title="Logout"
onclick="window.location.href='logout.html'"><i class="fas fa-arrow-right-from-
bracket"></i></button>

</div>
</div>

<!-- Main -->
<div class="main-content">
<h2>Track Requests</h2>

<!-- Demo creation box (optional) -->
<div class="demo-box">
<form method="post" style="display:inline;">
<button name="insert_demo" class="btn" type="submit">Insert demo request +
timeline (for testing)</button>
</form>
<span style="margin-left:12px;color:#666;"><?php //echo safe($demo_msg);
?></span>
<div style="margin-top:8px;color:#666;font-size:13px;">(Use demo button only for
local testing; remove in production)</div>
</div>-->

<?php if (empty($requests)): ?>
<div class="no-requests">No requests yet.</div>
<?php else: ?>
<?php foreach ($requests as $req):>
    $reqId = intval($req['id']);

```

```

$reqTitle = $req['purpose'] ?: "Request #{$reqId}";
$reqCreated = $req['created_at'];
$current = $req['current_status'] ?? "";

// fetch track entries (ascending)

$tracks = [];

if ($s2 = $conn->prepare("SELECT status, created_at FROM track_request
WHERE request_id = ? ORDER BY created_at ASC")) {

    $s2->bind_param("i", $reqId);

    $s2->execute();

    $r2 = $s2->get_result();

    while ($row = $r2->fetch_assoc()) $tracks[] = $row;

    $s2->close();

}

// map earliest matching track for each step

$map = [];

foreach ($tracks as $tr) {

    foreach ($steps_order as $stepKey => $label) {

        if (!isset($map[$label]) && match_step($tr['status'], $stepKey)) {

            $map[$label] = ['time'=>$tr['created_at'] ?? '', 'raw'=>$tr['status']];

        }

    }

}

// determine current step

```

```

$currentStep = null;

foreach ($steps_order as $stepKey => $label) {
    if ($current && match_step($current, $stepKey)) { $currentStep = $label; break;
}

}

if (!$currentStep && !empty($tracks)) {

    $lastTrack = end($tracks);

    foreach ($steps_order as $stepKey => $label) {

        if (match_step($lastTrack['status'], $stepKey)) { $currentStep = $label; break;
}

    }

}

?>

<div class="request-card">

<div class="request-header">

<div>

    <div class="req-title"><?php echo safe($reqTitle); ?></div>

    <div class="req-meta">Created at: <?php echo fmt_dt($reqCreated); ?></div>

</div>

    <div class="req-meta">Current status: <strong><?php echo safe($current ?: 'N/A'); ?></strong></div>

</div>

<div class="timeline">

<?php foreach ($steps_order as $stepKey => $label):>

    $completed = isset($map[$label]);

    $is_current = ($currentStep === $label);

```

```
$colorClass = $completed ? 'green' : 'red';

if (!$completed && strpos($label, 'Under Review') !== false) $colorClass =
'yellow';

$circleClass = "circle {$colorClass}" . ($is_current ? ' current' : "");

?>

<div class="step">

<div class="<?php echo $circleClass; ?>"><?php echo $completed ? '✓' : '';
?></div>

<div>

<div class="status-text"><?php echo safe($label); ?></div>

<?php if ($completed): ?>

    <div class="status-time"><?php echo fmt_dt($map[$label]['time']);
?><?php if (!empty($map[$label]['note'])) echo ' • ' . safe($map[$label]['note']);
?></div>

    <?php else: ?>

        <div class="status-time">Not yet</div>

    <?php endif; ?>

</div>

</div>

<?php endforeach; ?>

</div>

</div>

<?php endforeach; ?>

<?php endif; ?>

<button class="btn" onclick="location.href='req_system.php'">Back to
Home</button>
```

```
</div>

<script>
// 🔺 Toggle sidebar

const toggleBtn = document.getElementById("toggleBtn");
const sidebar = document.getElementById("sidebar");
toggleBtn.addEventListener("click", () => {
  sidebar.classList.toggle("collapsed");
});
</script>
</body>
</html>
```