Linux LAMP sunucunuzda öğrenci ve hobi bilgilerini içeren bir veritabanı oluşturalım. Bu veritabanına veri eklemek ve PHP ile yönetim paneli oluşturmak için adımlar aşağıda verilmiştir:

**1. MySQL Veritabanı ve Tablo Oluşturma**

**1.1. Veritabanını Oluşturma**

Öncelikle, ogrenci\_hobi adında bir veritabanı oluşturun:

sql

CREATE DATABASE IF NOT EXISTS ogrenci\_hobi CHARACTER SET utf8mb4 COLLATE utf8mb4\_unicode\_ci;

USE ogrenci\_hobi;

**1.2. ogrenci Tablosunu Oluşturma**

ogrenci tablosunda öğrenci numarası, adı, soyadı ve GNO bilgilerini tutacağız:

sql

CREATE TABLE IF NOT EXISTS ogrenci (

ogrenci\_no INT AUTO\_INCREMENT PRIMARY KEY,

ad VARCHAR(50) NOT NULL,

soyad VARCHAR(50) NOT NULL,

gno DECIMAL(3,2) NOT NULL

);

**1.3. hobi Tablosunu Oluşturma**

hobi tablosunda hobi ID, öğrenci numarası, hobi adı ve hobi araçları bilgilerini tutacağız:

sql

CREATE TABLE IF NOT EXISTS hobi (

hobi\_id INT AUTO\_INCREMENT PRIMARY KEY,

ogrenci\_no INT NOT NULL,

hobi\_adı VARCHAR(100) NOT NULL,

hobi\_araclari VARCHAR(100),

FOREIGN KEY (ogrenci\_no) REFERENCES ogrenci(ogrenci\_no) ON DELETE CASCADE

);

**1.4. Öğrenci Verilerini Eklemek**

20 adet rastgele öğrenci verisi ekleyelim:

sql

INSERT INTO ogrenci (ad, soyad, gno) VALUES

('Ahmet', 'Yılmaz', 3.50),

('Mehmet', 'Kaya', 3.20),

('Ayşe', 'Demir', 3.80),

('Fatma', 'Şahin', 3.60),

('Ali', 'Çelik', 3.40),

('Elif', 'Koç', 3.70),

('Hasan', 'Aydın', 3.30),

('Hülya', 'Kurt', 3.90),

('Burak', 'Güneş', 3.10),

('Zeynep', 'Yıldız', 3.55),

('Murat', 'Öztürk', 3.25),

('Seda', 'Polat', 3.65),

('Kemal', 'Arslan', 3.35),

('Sevgi', 'Kılıç', 3.85),

('Orhan', 'Balcı', 3.15),

('Derya', 'Taş', 3.75),

('Cem', 'Kara', 3.05),

('Gül', 'Duman', 3.95),

('Turan', 'Erdoğan', 3.45),

('Bahar', 'Doğan', 3.60);

**1.5. Hobi Verilerini Eklemek**

Her öğrenciye 3 hobi ekleyerek toplamda 60 hobi ekleyeceğiz. Ancak kullanıcı talebine göre 30 hobi eklememiz istenmiş. Bunun için her öğrenciye 1 veya 2 hobi ekleyerek toplam 30 hobiye ulaşabiliriz. Ancak genellikle her öğrenciye 3 hobi eklemek isteniyor. Burada kullanıcı talebinde bir çelişki olabilir. Kullanıcının istediği 20 öğrenciye 3'er hobi ekleyerek toplamda 60 hobi eklememiz daha mantıklı olacaktır. Aşağıda 60 hobi eklemek için bir örnek verilmiştir:

sql

INSERT INTO hobi (ogrenci\_no, hobi\_adı, hobi\_araclari) VALUES

(1, 'Futbol', 'Top, Ayakkabı, Form'),

(1, 'Müzik', 'Piyano, Gitar, Nota'),

(1, 'Resim', 'Fırça, Boya, Tuval'),

(2, 'Basketbol', 'Top, Forma, Sepet'),

(2, 'Yüzme', 'Mayo, Gözlük, Havlu'),

(2, 'Satranç', 'Satranç Tahtası, Saat, Taşlar'),

(3, 'Koşu', 'Koşu Ayakkabısı, Spor Giyim, Saat'),

(3, 'Fotoğrafçılık', 'Kamera, Lens, Tripod'),

(3, 'Yazılım', 'Bilgisayar, Programlama Dilleri, IDE'),

(4, 'Kitap Okuma', 'Kitap, Okuma Lambası, Kitap Ayracı'),

(4, 'Bahçecilik', 'Aletler, Tohum, Saksı'),

(4, 'Yoga', 'Mat, Kıyafet, Blok'),

(5, 'Dans', 'Müzik, Dans Ayakkabısı, Aksesuar'),

(5, 'Sinema', 'Film, Bilet, Patlamış Mısır'),

(5, 'Seyahat', 'Çanta, Harita, Kamerа'),

(6, 'Pişirme', 'Tencere, Malzemeler, Kitap'),

(6, 'Yüzme', 'Mayo, Gözlük, Havlu'),

(6, 'Bisiklet', 'Bisiklet, Kask, Eldiven'),

(7, 'Müzik', 'Enstrüman, Nota, Stüdyo'),

(7, 'Trekking', 'Çadır, Uyku Tulumu, Sırt Çantası'),

(7, 'Programlama', 'Bilgisayar, Kitap, Yazılım'),

(8, 'Resim', 'Fırça, Boya, Tuval'),

(8, 'Kino', 'Film, Bilet, Patlamış Mısır'),

(8, 'Koşu', 'Koşu Ayakkabısı, Spor Giyim, Saat'),

(9, 'Futbol', 'Top, Ayakkabı, Form'),

(9, 'Satranç', 'Satranç Tahtası, Saat, Taşlar'),

(9, 'Yazılım', 'Bilgisayar, Programlama Dilleri, IDE'),

(10, 'Basketbol', 'Top, Forma, Sepet'),

(10, 'Yoga', 'Mat, Kıyafet, Blok'),

(10, 'Yüzme', 'Mayo, Gözlük, Havlu'),

(11, 'Fotoğrafçılık', 'Kamera, Lens, Tripod'),

(11, 'Kitap Okuma', 'Kitap, Okuma Lambası, Kitap Ayracı'),

(11, 'Bahçecilik', 'Aletler, Tohum, Saksı'),

(12, 'Dans', 'Müzik, Dans Ayakkabısı, Aksesuar'),

(12, 'Seyahat', 'Çanta, Harita, Kamera'),

(12, 'Pişirme', 'Tencere, Malzemeler, Kitap'),

(13, 'Bisiklet', 'Bisiklet, Kask, Eldiven'),

(13, 'Müzik', 'Enstrüman, Nota, Stüdyo'),

(13, 'Trekking', 'Çadır, Uyku Tulumu, Sırt Çantası'),

(14, 'Resim', 'Fırça, Boya, Tuval'),

(14, 'Kino', 'Film, Bilet, Patlamış Mısır'),

(14, 'Koşu', 'Koşu Ayakkabısı, Spor Giyim, Saat'),

(15, 'Futbol', 'Top, Ayakkabı, Form'),

(15, 'Satranç', 'Satranç Tahtası, Saat, Taşlar'),

(15, 'Yazılım', 'Bilgisayar, Programlama Dilleri, IDE'),

(16, 'Basketbol', 'Top, Forma, Sepet'),

(16, 'Yoga', 'Mat, Kıyafet, Blok'),

(16, 'Yüzme', 'Mayo, Gözlük, Havlu'),

(17, 'Fotoğrafçılık', 'Kamera, Lens, Tripod'),

(17, 'Kitap Okuma', 'Kitap, Okuma Lambası, Kitap Ayracı'),

(17, 'Bahçecilik', 'Aletler, Tohum, Saksı'),

(18, 'Dans', 'Müzik, Dans Ayakkabısı, Aksesuar'),

(18, 'Seyahat', 'Çanta, Harita, Kamera'),

(18, 'Pişirme', 'Tencere, Malzemeler, Kitap'),

(19, 'Bisiklet', 'Bisiklet, Kask, Eldiven'),

(19, 'Müzik', 'Enstrüman, Nota, Stüdyo'),

(19, 'Trekking', 'Çadır, Uyku Tulumu, Sırt Çantası'),

(20, 'Resim', 'Fırça, Boya, Tuval'),

(20, 'Kino', 'Film, Bilet, Patlamış Mısır'),

(20, 'Koşu', 'Koşu Ayakkabısı, Spor Giyim, Saat');

**Not:** Yukarıdaki örnekte her öğrenciye 3 hobi eklenmiştir, toplamda 60 hobi bulunmaktadır. Eğer sadece 30 hobi eklemek isterseniz, her öğrenciye 1-2 hobi ekleyebilirsiniz.

**2. PHP ile Yönetim Paneli Oluşturma**

Yönetim paneli için aşağıdaki adımları takip edebilirsiniz:

**2.1. Proje Yapısı**

Öncelikle, web projenizin dosya yapısını oluşturun:

css

/var/www/html/ogrenci\_hobi/

│

├── css/

│ └── styles.css

├── js/

│ └── scripts.js

├── index.php

├── ogrenci\_ekle.php

├── ogrenci\_sil.php

├── hobi\_ekle.php

├── hobi\_sil.php

└── db.php

**2.2. Veritabanı Bağlantısı (db.php)**

Tüm PHP sayfalarında kullanılacak veritabanı bağlantısını sağlayan db.php dosyasını oluşturun:

php

<?php

$servername = "localhost";

$username = "root"; // MySQL kullanıcı adınızı girin

$password = ""; // MySQL şifrenizi girin

$dbname = "ogrenci\_hobi";

// Bağlantıyı oluştur

$conn = new mysqli($servername, $username, $password, $dbname);

// Bağlantıyı kontrol et

if ($conn->connect\_error) {

die("Bağlantı hatası: " . $conn->connect\_error);

}

?>

**2.3. Stil Sayfası (css/styles.css)**

Basit ve şık bir görünüm için CSS kullanabilirsiniz. Aşağıda temel bir CSS örneği bulunmaktadır:

css

/\* css/styles.css \*/

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

margin: 0;

padding: 0;

}

.container {

width: 80%;

margin: auto;

overflow: hidden;

padding: 20px;

background: #fff;

margin-top: 30px;

box-shadow: 0 0 10px rgba(0,0,0,0.1);

}

header {

background: #50b3a2;

color: #fff;

padding: 10px 0;

text-align: center;

}

nav a {

margin: 0 15px;

text-decoration: none;

color: #50b3a2;

font-weight: bold;

}

nav a:hover {

color: #333;

}

form label {

display: block;

margin: 10px 0 5px;

}

form input, form select {

width: 100%;

padding: 8px;

box-sizing: border-box;

}

form button {

padding: 10px 15px;

background: #50b3a2;

border: none;

color: #fff;

cursor: pointer;

}

form button:hover {

background: #3d8b7d;

}

table {

width: 100%;

border-collapse: collapse;

margin-top: 20px;

}

table, th, td {

border: 1px solid #ddd;

}

th, td {

padding: 12px;

text-align: left;

}

tr:nth-child(even) {

background-color: #f9f9f9;

}

.alert {

padding: 10px;

background-color: #4CAF50;

color: white;

margin-bottom: 15px;

}

.alert.error {background-color: #f44336;}

**2.4. JavaScript Dosyası (js/scripts.js)**

Basit doğrulamalar ve etkileşimler için JavaScript kullanabilirsiniz:

javascript

// js/scripts.js

// Örnek: Form doğrulama

function validateForm() {

let ad = document.forms["myForm"]["ad"].value;

let soyad = document.forms["myForm"]["soyad"].value;

if (ad == "" || soyad == "") {

alert("Ad ve Soyad alanları boş bırakılamaz.");

return false;

}

return true;

}

**2.5. Ana Menü Sayfası (index.php)**

Yönetim panelinin ana menüsünü oluşturun:

php

<?php

// index.php

?>

<!DOCTYPE html>

<html lang="tr">

<head>

<meta charset="UTF-8">

<title>Öğrenci ve Hobi Yönetim Paneli</title>

<link rel="stylesheet" href="css/styles.css">

</head>

<body>

<header>

<h1>Öğrenci ve Hobi Yönetim Paneli</h1>

</header>

<div class="container">

<nav>

<a href="ogrenci\_ekle.php">Öğrenci Ekle</a>

<a href="ogrenci\_sil.php">Öğrenci Sil</a>

<a href="hobi\_ekle.php">Hobi Ekle</a>

<a href="hobi\_sil.php">Hobi Sil</a>

</nav>

</div>

</body>

</html>

**2.6. Öğrenci Ekleme Sayfası (ogrenci\_ekle.php)**

Öğrenci eklemek için bir form oluşturun:

php

<?php

// ogrenci\_ekle.php

include 'db.php';

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$ad = $\_POST['ad'];

$soyad = $\_POST['soyad'];

$gno = $\_POST['gno'];

if (empty($ad) || empty($soyad) || empty($gno)) {

$error = "Tüm alanlar doldurulmalıdır.";

} else {

$stmt = $conn->prepare("INSERT INTO ogrenci (ad, soyad, gno) VALUES (?, ?, ?)");

$stmt->bind\_param("ssd", $ad, $soyad, $gno);

if ($stmt->execute()) {

$success = "Öğrenci başarıyla eklendi.";

} else {

$error = "Hata: " . $conn->error;

}

$stmt->close();

}

}

$conn->close();

?>

<!DOCTYPE html>

<html lang="tr">

<head>

<meta charset="UTF-8">

<title>Öğrenci Ekle</title>

<link rel="stylesheet" href="css/styles.css">

<script src="js/scripts.js"></script>

</head>

<body>

<header>

<h1>Öğrenci Ekle</h1>

</header>

<div class="container">

<nav>

<a href="index.php">Ana Sayfa</a>

<a href="ogrenci\_sil.php">Öğrenci Sil</a>

<a href="hobi\_ekle.php">Hobi Ekle</a>

<a href="hobi\_sil.php">Hobi Sil</a>

</nav>

<?php if(isset($error)): ?>

<div class="alert error"><?php echo $error; ?></div>

<?php endif; ?>

<?php if(isset($success)): ?>

<div class="alert"><?php echo $success; ?></div>

<?php endif; ?>

<form name="myForm" action="ogrenci\_ekle.php" method="post" onsubmit="return validateForm()">

<label for="ad">Ad:</label>

<input type="text" id="ad" name="ad" required>

<label for="soyad">Soyad:</label>

<input type="text" id="soyad" name="soyad" required>

<label for="gno">GNO:</label>

<input type="number" step="0.01" id="gno" name="gno" required>

<button type="submit">Ekle</button>

</form>

</div>

</body>

</html>

**2.7. Öğrenci Silme Sayfası (ogrenci\_sil.php)**

Öğrenci silmek için bir liste ve silme işlemi:

php

<?php

// ogrenci\_sil.php

include 'db.php';

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$ogrenci\_no = $\_POST['ogrenci\_no'];

if (empty($ogrenci\_no)) {

$error = "Silinecek öğrenci seçilmelidir.";

} else {

$stmt = $conn->prepare("DELETE FROM ogrenci WHERE ogrenci\_no = ?");

$stmt->bind\_param("i", $ogrenci\_no);

if ($stmt->execute()) {

$success = "Öğrenci başarıyla silindi.";

} else {

$error = "Hata: " . $conn->error;

}

$stmt->close();

}

}

// Öğrencileri listele

$result = $conn->query("SELECT ogrenci\_no, ad, soyad FROM ogrenci ORDER BY ogrenci\_no ASC");

$conn->close();

?>

<!DOCTYPE html>

<html lang="tr">

<head>

<meta charset="UTF-8">

<title>Öğrenci Sil</title>

<link rel="stylesheet" href="css/styles.css">

</head>

<body>

<header>

<h1>Öğrenci Sil</h1>

</header>

<div class="container">

<nav>

<a href="index.php">Ana Sayfa</a>

<a href="ogrenci\_ekle.php">Öğrenci Ekle</a>

<a href="hobi\_ekle.php">Hobi Ekle</a>

<a href="hobi\_sil.php">Hobi Sil</a>

</nav>

<?php if(isset($error)): ?>

<div class="alert error"><?php echo $error; ?></div>

<?php endif; ?>

<?php if(isset($success)): ?>

<div class="alert"><?php echo $success; ?></div>

<?php endif; ?>

<form action="ogrenci\_sil.php" method="post">

<label for="ogrenci\_no">Öğrenci Seç:</label>

<select id="ogrenci\_no" name="ogrenci\_no" required>

<option value="">-- Seçiniz --</option>

<?php

if ($result->num\_rows > 0) {

while($row = $result->fetch\_assoc()) {

echo "<option value='" . $row['ogrenci\_no'] . "'>" . $row['ad'] . " " . $row['soyad'] . "</option>";

}

}

?>

</select>

<button type="submit">Sil</button>

</form>

</div>

</body>

</html>

**2.8. Hobi Ekleme Sayfası (hobi\_ekle.php)**

Öğrenciye hobi eklemek için bir form:

php

<?php

// hobi\_ekle.php

include 'db.php';

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$ogrenci\_no = $\_POST['ogrenci\_no'];

$hobi\_adı = $\_POST['hobi\_adı'];

$hobi\_araclari = $\_POST['hobi\_araclari'];

if (empty($ogrenci\_no) || empty($hobi\_adı)) {

$error = "Öğrenci ve Hobi Adı alanları doldurulmalıdır.";

} else {

$stmt = $conn->prepare("INSERT INTO hobi (ogrenci\_no, hobi\_adı, hobi\_araclari) VALUES (?, ?, ?)");

$stmt->bind\_param("iss", $ogrenci\_no, $hobi\_adı, $hobi\_araclari);

if ($stmt->execute()) {

$success = "Hobi başarıyla eklendi.";

} else {

$error = "Hata: " . $conn->error;

}

$stmt->close();

}

}

// Öğrencileri listele

$result = $conn->query("SELECT ogrenci\_no, ad, soyad FROM ogrenci ORDER BY ogrenci\_no ASC");

$conn->close();

?>

<!DOCTYPE html>

<html lang="tr">

<head>

<meta charset="UTF-8">

<title>Hobi Ekle</title>

<link rel="stylesheet" href="css/styles.css">

<script src="js/scripts.js"></script>

</head>

<body>

<header>

<h1>Hobi Ekle</h1>

</header>

<div class="container">

<nav>

<a href="index.php">Ana Sayfa</a>

<a href="ogrenci\_ekle.php">Öğrenci Ekle</a>

<a href="ogrenci\_sil.php">Öğrenci Sil</a>

<a href="hobi\_sil.php">Hobi Sil</a>

</nav>

<?php if(isset($error)): ?>

<div class="alert error"><?php echo $error; ?></div>

<?php endif; ?>

<?php if(isset($success)): ?>

<div class="alert"><?php echo $success; ?></div>

<?php endif; ?>

<form name="myForm" action="hobi\_ekle.php" method="post" onsubmit="return validateForm()">

<label for="ogrenci\_no">Öğrenci Seç:</label>

<select id="ogrenci\_no" name="ogrenci\_no" required>

<option value="">-- Seçiniz --</option>

<?php

// Veritabanından öğrenci listesini doldur

include 'db.php';

$students = $conn->query("SELECT ogrenci\_no, ad, soyad FROM ogrenci ORDER BY ogrenci\_no ASC");

if ($students->num\_rows > 0) {

while($row = $students->fetch\_assoc()) {

echo "<option value='" . $row['ogrenci\_no'] . "'>" . $row['ad'] . " " . $row['soyad'] . "</option>";

}

}

$conn->close();

?>

</select>

<label for="hobi\_adı">Hobi Adı:</label>

<input type="text" id="hobi\_adı" name="hobi\_adı" required>

<label for="hobi\_araclari">Hobi Araçları:</label>

<input type="text" id="hobi\_araclari" name="hobi\_araclari">

<button type="submit">Ekle</button>

</form>

</div>

</body>

</html>

**2.9. Hobi Silme Sayfas (hobi\_sil.php)**

Hobiyi silmek için bir liste ve silme işlemi:

php

<?php

// hobi\_sil.php

include 'db.php';

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$hobi\_id = $\_POST['hobi\_id'];

if (empty($hobi\_id)) {

$error = "Silinecek hobi seçilmelidir.";

} else {

$stmt = $conn->prepare("DELETE FROM hobi WHERE hobi\_id = ?");

$stmt->bind\_param("i", $hobi\_id);

if ($stmt->execute()) {

$success = "Hobi başarıyla silindi.";

} else {

$error = "Hata: " . $conn->error;

}

$stmt->close();

}

}

// Hobileri listele

$result = $conn->query("SELECT hobi\_id, hobi\_adı, hobi\_araclari, ogrenci\_no FROM hobi ORDER BY hobi\_id ASC");

$conn->close();

?>

<!DOCTYPE html>

<html lang="tr">

<head>

<meta charset="UTF-8">

<title>Hobi Sil</title>

<link rel="stylesheet" href="css/styles.css">

</head>

<body>

<header>

<h1>Hobi Sil</h1>

</header>

<div class="container">

<nav>

<a href="index.php">Ana Sayfa</a>

<a href="ogrenci\_ekle.php">Öğrenci Ekle</a>

<a href="ogrenci\_sil.php">Öğrenci Sil</a>

<a href="hobi\_ekle.php">Hobi Ekle</a>

</nav>

<?php if(isset($error)): ?>

<div class="alert error"><?php echo $error; ?></div>

<?php endif; ?>

<?php if(isset($success)): ?>

<div class="alert"><?php echo $success; ?></div>

<?php endif; ?>

<form action="hobi\_sil.php" method="post">

<label for="hobi\_id">Hobi Seç:</label>

<select id="hobi\_id" name="hobi\_id" required>

<option value="">-- Seçiniz --</option>

<?php

if ($result->num\_rows > 0) {

while($row = $result->fetch\_assoc()) {

// Öğrenci adını almak için ayrı sorgu

include 'db.php';

$ogrenci = $conn->query("SELECT ad, soyad FROM ogrenci WHERE ogrenci\_no = " . $row['ogrenci\_no'])->fetch\_assoc();

echo "<option value='" . $row['hobi\_id'] . "'>[" . $row['hobi\_id'] . "] " . $row['hobi\_adı'] . " - " . $ogrenci['ad'] . " " . $ogrenci['soyad'] . "</option>";

$conn->close();

}

}

?>

</select>

<button type="submit">Sil</button>

</form>

</div>

</body>

</html>

**Not:** hobi\_sil.php sayfasında her hobiye ait öğrencinin adını göstermek için her döngüde ayrı bir veritabanı bağlantısı açılmıştır. Bu, performans açısından verimsiz olabilir. Daha iyi bir performans için JOIN kullanarak tek bir sorgu ile gerekli verileri çekebilirsiniz.

**2.10. Hobi Listeleme ve Yönetme (Opsiyonel)**

Ekstra bir özellik olarak, tüm hobileri listeleyebilir ve yönetebilirsiniz. Ancak, kullanıcı talebinde bulunmadığı için bu adım isteğe bağlıdır.

**3. Güvenlik ve İyileştirmeler**

Yukarıdaki örnekler temel işlevselliği sağlar, ancak güvenlik ve performans açısından bazı iyileştirmeler yapılabilir:

* **Hazırlanmış İfadeler (Prepared Statements):** SQL enjeksiyon saldırılarına karşı korunmak için her SQL sorgusunda hazırlanan ifadeler kullanılmıştır.
* **Form Doğrulamaları:** Hem istemci tarafında (JavaScript) hem de sunucu tarafında (PHP) form doğrulamaları eklenmiştir.
* **CSS ve JavaScript:** Basit ve temiz bir görünüm için temel CSS ve JavaScript kullanılmıştır. Daha gelişmiş tasarımlar için Bootstrap veya başka CSS framework'leri kullanılabilir.
* **Hata Yönetimi:** Hatalar kullanıcıya gösterilmiştir, ancak üretim ortamında daha güvenli hata yönetimi yöntemleri kullanılabilir.