Below is a **clean, lab-safe design** that fits your **10.10.10.0/24 cyber-range VM** and keeps things simple for students.

✅ Recommended Design Pattern

```
/var/www/html/
├── assignments/
│   ├── assignment_template.html   ← Read-only assignment
│   ├── start.html                 ← Student clicks this
│   └── start_dvwa.php             ← Server-side launcher
└── dvwa/
```

Flow for the student:

1. Open assignment → **read**
2. Click **Start DVWA Lab**
3. Services start (if stopped)
4. Browser redirects to DVWA

🔒 Why We Don't Start Services from HTML

HTML is client-side only:

- ❌ Cannot run `systemctl`
- ❌ Cannot touch the OS
- ❌ Cannot start Apache/MariaDB

But PHP can, when *you* control the VM.

This keeps:

- Safety ✓
- Clarity ✓
- Auditability ✓

`start.html` (Student-Facing Page)

This is what they click.

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Start DVWA Lab</title>
  <style>
    body {
      font-family: system-ui, sans-serif;
      background: #0b0f14;
      color: #e6edf3;
      display: flex;
      align-items: center;
      justify-content: center;
      height: 100vh;
    }
    .card {
      background: #0f1520;
      border: 1px solid #1f2a37;
      border-radius: 12px;
      padding: 24px;
      text-align: center;
      max-width: 420px;
    }
    button {
      background: #2ea043;
      color: white;
      border: none;
      padding: 12px 18px;
      border-radius: 8px;
      font-size: 16px;
      cursor: pointer;
    }
    button:hover { background: #3fb950; }
    .note {
      margin-top: 12px;
      font-size: 13px;
      color: #9fb0c0;
    }
  </style>
</head>
<body>
  <div class="card">
    <h2>DVWA Lab Launcher</h2>
    <p>
      This will start the DVWA lab environment on this VM.
    </p>

    <form action="start_dvwa.php" method="post">
      <button type="submit">Start DVWA Lab</button>
    </form>
```

```
    <div class="note">
      Lab-only system · Authorized use only
    </div>
  </div>
</body>
</html>
```

🟡 `start_dvwa.php` (Server-Side Launcher)

This is the **only file that touches the OS**.

```
<?php
/*
 * DVWA Lab Starter
 * Runs only on the lab VM
 * Requires www-data/apache permission via sudoers
 */

$commands = [
  'sudo systemctl start mariadb',
  'sudo systemctl start apache2',
  'sudo systemctl restart apache2'
];

foreach ($commands as $cmd) {
  shell_exec($cmd);
}

// Redirect student to DVWA
header('Location: /dvwa/');
exit;
```

🔐 Required Sudo Configuration (ONE-TIME)

On the **DVWA VM**, run:

```
sudo visudo
```

Add this line at the bottom:

```
www-data ALL=(ALL) NOPASSWD: /bin/systemctl start mariadb, /bin/systemctl
start apache2, /bin/systemctl restart apache2
```

📌 This ensures:

- No password prompts
- No shell access
- Only **specific services**
- No privilege escalation paths for students

🧠 What This Teaches (Subtly)

Even without saying it, students learn:

- Client vs server execution
- Controlled privilege elevation
- Automation in lab environments
- Real-world SOC / training design

This is **enterprise-thinking**, not just hacking.

🎤 What You Can Say on YouTube

“This page doesn’t hack anything.
It simply prepares the lab.
Real assessments always start with environment readiness.”

That frames it professionally.

🛡️ Optional Enhancements (Later)

If you want to evolve this:

- Add **status check** (is DVWA already running?)
- Add **Stop Lab** button
- Log start times for attendance
- Restrict access by subnet (10.10.10.0/24)
- Add a **countdown timer** for the assignment window

✅ Summary

- ✓ Assignment HTML = read-only
- ✓ Start page = student-friendly
- ✓ PHP = controlled automation
- ✓ Safe for internal cyber range
- ✓ Scales to many labs

If you want next, I can:

- Add a **Stop DVWA** page
- Add **status indicators**
- Lock access to **specific subnets**
- Turn this into a **dashboard-style launcher**