

```

<!DOCTYPE html>
<html lang="id">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>SpengGa Sinergi - SMP Negeri 3 Madiun</title>

  <!-- Tailwind CSS (Via CDN) -->
  <script src="https://cdn.tailwindcss.com"></script>

  <!-- Lucide Icons (Via CDN) -->
  <script src="https://unpkg.com/lucide@latest"></script>

  <!-- PapaParse (Untuk Baca CSV) -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/PapaParse/
5.3.0/papaparse.min.js"></script>

  <!-- Firebase SDK (Compat Version) -->
  <script src="https://www.gstatic.com/firebasejs/10.7.1/firebase-
app-compat.js"></script>
  <script src="https://www.gstatic.com/firebasejs/10.7.1/firebase-
firestore-compat.js"></script>

  <!-- Custom Font -->
  <link href="https://fonts.googleapis.com/css2?
family=Inter:wght@400;600;800;900&display=swap" rel="stylesheet">

  <style>
    body { font-family: 'Inter', sans-serif; }
    @keyframes fadeIn { from { opacity: 0; transform:
translateY(10px); } to { opacity: 1; transform: translateY(0); } }
    .animate-fade-in { animation: fadeIn 0.4s ease-out
forwards; }
    ::selection { background-color: #facc15; color: black; }
    /* Loader Overlay */
    #loader { position: fixed; inset: 0; background: white; z-
index: 9999; display: flex; flex-direction: column; justify-content:
center; align-items: center; transition: opacity 0.5s; }
    .hidden-input { display: none; }
    .rank-1 { background: linear-gradient(135deg, #FFF7D1 0%,
#FFD700 100%); border-color: #FFD700; color: #854d0e; }
    .rank-2 { background: linear-gradient(135deg, #F3F4F6 0%,
#C0C0C0 100%); border-color: #C0C0C0; color: #374151; }
    .rank-3 { background: linear-gradient(135deg, #FFF1E6 0%,
#CD7F32 100%); border-color: #CD7F32; color: #7c2d12; }
  </style>
</head>
<body class="bg-white text-gray-900 flex flex-col min-h-screen">

  <!-- LOADER -->
  <div id="loader">
    <div class="animate-spin rounded-full h-16 w-16 border-t-4
border-b-4 border-yellow-400 mb-4"></div>

```

```

        <p class="text-blue-900 font-bold animate-pulse" id="loader-
text">Menghubungkan ke Database...</p>
    </div>

```

```

    <!-- NAVIGASI -->
    <nav class="bg-white border-b border-gray-200 sticky top-0 z-40
shadow-sm">
        <div class="max-w-7xl mx-auto px-4 sm:px-6 lg:px-8">
            <div class="flex justify-between h-20 items-center">
                <div class="flex items-center gap-3 cursor-pointer
group" onclick="switchView('home')">
                    
                    <div>
                        <h1 class="text-2xl font-black text-blue-900
tracking-tight leading-none group-hover:text-yellow-600 transition-
colors">
                            SpenGa <span class="text-
yellow-500">Sinergi</span>
                        </h1>
                        <p class="text-xs text-gray-500 font-bold
tracking-widest mt-0.5">SMP NEGERI 3 MADIUN</p>
                    </div>
                </div>
                <div class="hidden lg:flex items-center space-x-1"
id="desktop-menu"></div>
                <div class="lg:hidden">
                    <button onclick="toggleMobileMenu()"
class="text-black p-2 bg-yellow-400 rounded-lg">
                        <i data-lucide="menu" class="w-6 h-6"></i>
                    </button>
                </div>
            </div>
        </div>
        <div id="mobile-menu" class="hidden lg:hidden border-t
border-gray-100 bg-white p-4 space-y-2 shadow-xl"></div>
    </nav>

```

```

    <!-- MAIN CONTENT -->
    <main class="max-w-7xl mx-auto px-4 sm:px-6 lg:px-8 py-8 flex-
grow w-full">

```

```

        <!-- 1. HOME VIEW -->
        <section id="view-home" class="animate-fade-in block">
            <div class="bg-gradient-to-br from-blue-950 to-black
rounded-3xl p-8 md:p-12 text-white relative overflow-hidden
shadow-2xl border-b-8 border-yellow-400 mb-12">
                <div class="absolute top-0 right-0 w-96 h-96 bg-
blue-900 rounded-full mix-blend-screen filter blur-3xl opacity-30
-translate-y-1/2 translate-x-1/3"></div>
                <div class="relative z-10 max-w-2xl">
                    <h1 class="text-5xl md:text-6xl font-black mb-6

```

```

tracking-tight text-yellow-400 drop-shadow-lg">SpenGa Sinergi</h1>
      <p class="text-blue-100 text-lg md:text-xl mb-8
leading-relaxed">
          Platform asesmen digital SMP Negeri 3
Madiun. <br>
          Membangun transparansi, integritas, dan
prestasi dalam satu sinergi.
      </p>
      <button onclick="openLoginModal()" class="bg-
yellow-400 hover:bg-yellow-500 text-black font-extrabold py-4 px-8
rounded-full inline-flex items-center gap-2 transition-transform
hover:scale-105 shadow-[0_0_20px_rgba(250,204,21,0.5)]">
          <i data-lucide="log-in" class="w-5 h-5"></i>
Masuk Portal
      </button>
    </div>
  </div>
  <div class="text-center mb-10">
    <div class="inline-flex items-center justify-center
gap-3 mb-2">
      <i data-lucide="trophy" class="text-yellow-500
w-10 h-10"></i>
      <h2 class="text-3xl font-black text-blue-900
uppercase tracking-wide">Student of The Month</h2>
    </div>
    <p class="text-gray-500">Siswa paling rajin dan
berprestasi akademik bulan ini</p>
  </div>

  <!-- FIXED ID HERE -->
  <div id="sotm-container" class="grid lg:grid-cols-3
gap-8 px-4">
    <div class="col-span-3 text-center text-gray-400
py-8">Memuat data prestasi...</div>
  </div>
</section>

  <!-- 2. DATA GURU VIEW -->
  <section id="view-teachers" class="animate-fade-in hidden
max-w-4xl mx-auto">
    <div class="bg-blue-900 text-white p-8 rounded-2xl
shadow-lg flex items-center gap-4 mb-6">
      <div class="bg-yellow-400 p-3 rounded-full text-
blue-900"><i data-lucide="users" class="w-8 h-8"></i></div>
      <div><h2 class="text-2xl font-bold">Data Guru
Pengajar</h2><p class="text-blue-200">Daftar tenaga pendidik SMP
Negeri 3 Madiun</p></div>
    </div>
    <div id="teachers-grid" class="grid md:grid-cols-2
gap-4"></div>
  </section>

  <!-- 3. MAPEL VIEW -->
  <section id="view-subjects" class="animate-fade-in hidden

```

```

max-w-4xl mx-auto">
    <div class="bg-black text-white p-8 rounded-2xl shadow-
lg flex items-center gap-4 border-b-4 border-yellow-400 mb-6">
        <div class="bg-white p-3 rounded-full text-black"><i
data-lucide="book-open" class="w-8 h-8"></i></div>
        <div><h2 class="text-2xl font-bold">Mata Pelajaran</
h2><p class="text-gray-400">Kurikulum Merdeka Belajar</p></div>
    </div>
    <div id="subjects-grid" class="grid grid-cols-2 md:grid-
cols-3 gap-4"></div>
</section>

```

```

<!-- 4. DASHBOARD GURU -->
<section id="view-teacher-dashboard" class="animate-fade-in
hidden space-y-8">

```

```

    <div class="flex flex-col md:flex-row justify-between
items-start md:items-center gap-4">

```

```

        <div><h2 class="text-2xl font-black text-
blue-900">Dashboard Guru</h2><p class="text-gray-500">Kelola nilai
siswa secara manual atau kolektif.</p></div>

```

```

        <div class="flex flex-wrap gap-3">
            <button onclick="exportToCSV()" class="flex
items-center gap-2 bg-blue-600 hover:bg-blue-700 text-white px-5
py-2.5 rounded-xl font-bold shadow-md"><i data-lucide="download"
class="w-5 h-5"></i> Download Rekap</button>
            <button onclick="openImportModal()" class="flex
items-center gap-2 bg-green-600 hover:bg-green-700 text-white px-5
py-2.5 rounded-xl font-bold shadow-md"><i data-lucide="file-
spreadsheet" class="w-5 h-5"></i> Import Kolektif</button>

```

```

        </div>

```

```

    </div>

```

```

<!-- FORM INPUT -->

```

```

    <div class="bg-white rounded-xl shadow-md border border-
gray-100 overflow-hidden p-8 border-t-8 border-yellow-400">

```

```

        <h3 class="font-bold text-xl mb-6 text-gray-800 flex
items-center gap-2" id="form-title">Input Nilai Manual</h3>

```

```

        <form onsubmit="handleGradeSubmit(event)"
class="grid md:grid-cols-2 gap-6">

```

```

            <input type="hidden" id="edit-id" value="">

```

```

            <!-- SELECT CLASS FIRST -->

```

```

            <div class="md:col-span-2 grid md:grid-cols-2
gap-6">

```

```

                <div>

```

```

                    <label class="block text-sm font-bold
text-gray-700 mb-2">Langkah 1: Pilih Kelas</label>

```

```

                    <select id="input-class-filter"
class="w-full border-2 border-gray-200 rounded-xl p-3 outline-none
focus:border-yellow-400" onchange="filterStudentsByClass()"
required><option value="">-- Pilih Kelas --</option></select>

```

```

                </div>

```

```

                <div>

```

```

                    <label class="block text-sm font-bold

```

```

text-gray-700 mb-2">Langkah 2: Cari Nama Siswa</label>
      <input list="student-options" id="input-
student-search" class="w-full border-2 border-gray-200 rounded-xl
p-3 outline-none focus:border-yellow-400" placeholder="Ketik nama
siswa..." disabled required autocomplete="off">
      <datalist id="student-options"></
datalist>
    </div>
  </div>

  <div><label class="block text-sm font-bold text-
gray-700 mb-2">Mata Pelajaran</label><select id="input-subject"
class="w-full border-2 border-gray-200 rounded-xl p-3 outline-
none"></select></div>

  <!-- New Chapter Field -->
  <div><label class="block text-sm font-bold text-
gray-700 mb-2">Judul Bab</label><input type="text" id="input-
chapter" class="w-full border-2 border-gray-200 rounded-xl p-3
outline-none focus:border-yellow-400" placeholder="Contoh: Bab 1 -
Bilangan" required></div>

  <div><label class="block text-sm font-bold text-
gray-700 mb-2">Jenis Asesmen</label><select id="input-type"
class="w-full border-2 border-gray-200 rounded-xl p-3 outline-
none"></select></div>
  <div><label class="block text-sm font-bold text-
gray-700 mb-2">Sub Bab / Topik Materi</label><input type="text"
id="input-topic" class="w-full border-2 border-gray-200 rounded-xl
p-3 outline-none" placeholder="Contoh: Operasi Hitung" required></
div>

  <div><label class="block text-sm font-bold text-
gray-700 mb-2">Nilai (0-100)</label><input type="number" id="input-
score" min="0" max="100" class="w-full border-2 border-gray-200
rounded-xl p-3 outline-none" placeholder="0" required
oninput="updateStatusPreview()"><p class="text-xs mt-2 font-bold
text-blue-900" id="status-preview">Status: -</p></div>

  <div class="md:col-span-2">
    <div class="flex justify-between items-
center mb-2"><label class="block text-sm font-bold text-
gray-700">Catatan Guru</label><button type="button"
onclick="autoGenerateAI()" class="text-xs bg-purple-100 text-
purple-700 px-3 py-1.5 rounded-lg hover:bg-purple-200 flex items-
center gap-1 font-bold"><i data-lucide="sparkles" class="w-3 h-3"></
i> Auto-Generate AI</button></div>
    <textarea id="input-note" class="w-full
border-2 border-gray-200 rounded-xl p-3 outline-none" rows="3"
placeholder="Isi manual atau gunakan AI..."></textarea>
  </div>
  <div class="md:col-span-2 flex gap-3 mt-4">
    <button type="button" onclick="resetForm()"
id="btn-cancel" class="hidden flex-1 bg-gray-200 text-gray-800

```

```

py-3.5 rounded-xl font-bold">Batal</button>
        <button type="submit" class="flex-1 bg-
yellow-400 text-black py-3.5 rounded-xl font-extrabold hover:bg-
yellow-500 shadow-lg flex justify-center items-center gap-2"><i
data-lucide="save" class="w-5 h-5"></i> Simpan Nilai</button>
        </div>
    </form>
</div>

    <!-- REKAM JEJAK NILAI (Accordion Per Kelas) -->
    <div id="grades-accordion-container" class="space-y-4">
        <div class="text-center p-12 text-gray-400 bg-
gray-50 rounded-xl border border-dashed border-gray-200">Memuat data
nilai...</div>
        </div>
    </section>

    <!-- 5. STUDENT DASHBOARD -->
    <section id="view-student-dashboard" class="animate-fade-in
hidden space-y-8">
        <div class="bg-black text-white rounded-3xl p-8 shadow-
xl border-l-8 border-yellow-400 relative overflow-hidden">
            <div class="relative z-10 flex flex-col md:flex-row
items-center gap-8">
                <div class="w-24 h-24 bg-white rounded-full flex
items-center justify-center text-black font-black text-4xl
shadow-2xl border-4 border-yellow-400" id="student-initial"></div>
                <div class="text-center md:text-left flex-1"><h2
class="text-3xl font-black tracking-tight" id="student-name-
display">Nama Siswa</h2><p class="text-yellow-400 font-bold text-lg"
id="student-class-display">Kelas </p></div>
                <div class="bg-white/10 p-6 rounded-2xl
backdrop-blur-md border border-white/10 text-center"><p class="text-
xs text-gray-300 uppercase tracking-widest font-bold">Total Tugas</
p><p class="text-4xl font-black text-yellow-400 mt-1" id="student-
total-tasks">0</p></div>
            </div>
        </div>
        <div class="grid md:grid-cols-4 gap-6">
            <div class="md:col-span-1 space-y-4">
                <div class="bg-white rounded-xl shadow-md border
border-gray-100 p-5"><h3 class="font-bold text-blue-900 mb-4 flex
items-center gap-2"><i data-lucide="book-open" class="w-4 h-4"></i>
Rekap Mapel</h3><ul class="space-y-3 text-sm" id="student-subjects-
list"></ul></div>
            </div>
            <div class="md:col-span-3">
                <h3 class="font-black text-2xl text-blue-900
mb-6 border-l-4 border-yellow-400 pl-4">Laporan Hasil Belajar</h3>
                <div class="space-y-4" id="student-grades-
list"></div>
            </div>
        </div>
    </section>

```

```

        <!-- 6. ADMIN DASHBOARD -->
        <section id="view-admin-dashboard" class="animate-fade-in
hidden space-y-8">
            <div class="bg-gradient-to-r from-gray-900 to-blue-900
text-white p-8 rounded-3xl shadow-xl flex justify-between items-
center">
                <div><h2 class="text-3xl font-black text-
yellow-400">Admin Dashboard</h2><p class="text-blue-200">Panel
Manajemen Data Master & Database</p></div>
                <button onclick="exportToCSV()" class="bg-white/10
hover:bg-white/20 text-white px-4 py-2 rounded-lg font-bold text-sm
flex items-center gap-2 backdrop-blur-sm"><i data-lucide="download"
class="w-4 h-4"></i> Backup Data Nilai</button>
            </div>
            <div class="flex gap-4 border-b border-gray-200">
                <button onclick="switchAdminTab('teachers')"
id="tab-admin-teachers" class="px-6 py-3 font-bold text-sm border-
b-4 border-yellow-400 text-blue-900 bg-yellow-50 rounded-t-lg
transition-all">Data Guru</button>
                <button onclick="switchAdminTab('students')"
id="tab-admin-students" class="px-6 py-3 font-bold text-sm text-
gray-500 hover:bg-gray-50 rounded-t-lg transition-all">Data Siswa</
button>
                <button onclick="switchAdminTab('subjects')"
id="tab-admin-subjects" class="px-6 py-3 font-bold text-sm text-
gray-500 hover:bg-gray-50 rounded-t-lg transition-all">Data Mapel</
button>
            </div>
            <div id="admin-panel-teachers" class="space-y-4">
                <div class="flex justify-between items-center"><h3
class="font-bold text-xl text-gray-800">Daftar Guru</h3><button
onclick="openAdminModal('teacher')" class="bg-blue-900 hover:bg-
blue-800 text-white px-4 py-2 rounded-lg font-bold text-sm flex
items-center gap-2"><i data-lucide="plus" class="w-4 h-4"></i>
Tambah Guru</button></div>
                <div class="bg-white rounded-xl shadow border
border-gray-200 overflow-hidden"><table class="w-full text-left
text-sm"><thead class="bg-gray-100 text-gray-600 font-bold"><tr><th
class="p-4">Nama Lengkap</th><th class="p-4">Mata Pelajaran</th><th
class="p-4">Username</th><th class="p-4">Password</th><th class="p-4
text-right">Aksi</th></tr></thead><tbody id="admin-table-teachers"
class="divide-y divide-gray-100"></tbody></table></div>
            </div>
            <div id="admin-panel-students" class="space-y-4 hidden">
                <div class="flex flex-col md:flex-row justify-
between items-center gap-4">
                    <h3 class="font-bold text-xl text-
gray-800">Daftar Siswa</h3>
                    <div class="flex gap-2">
                        <input type="file" id="csv-student-import"
accept=".csv" class="hidden-input"
onchange="handleStudentCSVUpload(this)">
                        <button onclick="triggerStudentImport()"

```

```

class="bg-green-600 hover:bg-green-700 text-white px-4 py-2 rounded-
lg font-bold text-sm flex items-center gap-2"><i data-lucide="file-
spreadsheet" class="w-4 h-4"></i> Import dari Excel/CSV</button>
      <button onclick="downloadStudentTemplate()"
class="bg-gray-200 hover:bg-gray-300 text-gray-800 px-3 py-2
rounded-lg font-bold text-xs" title="Download Template CSV"><i data-
lucide="download" class="w-4 h-4"></i></button>
      <button onclick="openAdminModal('student')"
class="bg-blue-900 hover:bg-blue-800 text-white px-4 py-2 rounded-lg
font-bold text-sm flex items-center gap-2"><i data-lucide="plus"
class="w-4 h-4"></i> Tambah Siswa</button>

```

```

    </div>
  </div>
  <p class="text-xs text-gray-500 mt-1 bg-yellow-50
p-2 rounded border border-yellow-100">Format CSV: Nama, NISN, Kelas.
Password default = NamaDepan123 (Contoh: Budi123)</p>
  <div class="relative">
    <input type="text" id="search-student"
oninput="renderAdminStudents()" class="w-full border border-gray-300
rounded-lg p-3 pl-10 focus:outline-none focus:ring-2 focus:ring-
blue-500 transition shadow-sm" placeholder="Cari nama siswa..."><i
data-lucide="search" class="absolute left-3 top-3.5 text-gray-400
w-5 h-5"></i>

```

```

  </div>
  <div class="bg-white rounded-xl shadow border
border-gray-200 overflow-hidden"><table class="w-full text-left
text-sm"><thead class="bg-gray-100 text-gray-600 font-bold"><tr><th
class="p-4">NIS</th><th class="p-4">Nama Siswa</th><th
class="p-4">Kelas</th><th class="p-4">Username</th><th
class="p-4">Password</th><th class="p-4 text-right">Aksi</th></tr></thead><tbody id="admin-table-students" class="divide-y divide-
gray-100"></tbody></table></div>

```

```

    </div>
    <div id="admin-panel-subjects" class="space-y-4 hidden">
      <div class="flex justify-between items-center"><h3
class="font-bold text-xl text-gray-800">Daftar Mata Pelajaran</h3><button onclick="openAdminModal('subject')" class="bg-blue-900
hover:bg-blue-800 text-white px-4 py-2 rounded-lg font-bold text-sm
flex items-center gap-2"><i data-lucide="plus" class="w-4 h-4"></i>
Tambah Mapel</button></div>

```

```

      <div class="bg-white rounded-xl shadow border
border-gray-200 overflow-hidden"><table class="w-full text-left
text-sm"><thead class="bg-gray-100 text-gray-600 font-bold"><tr><th
class="p-4">Nama Mata Pelajaran</th><th class="p-4 text-
right">Aksi</th></tr></thead><tbody id="admin-table-subjects"
class="divide-y divide-gray-100"></tbody></table></div>

```

```

    </div>
  </section>
</main>

```

```

<!-- FOOTER -->
<footer class="bg-white mt-auto py-10 border-t-8 border-
yellow-400">
  <div class="max-w-7xl mx-auto px-4 text-center">

```



```

        <div class="flex items-center justify-center gap-3
mb-4">
            
            <span class="text-2xl font-black tracking-
tight"><span class="text-blue-900">SpenGa</span> <span class="text-
yellow-500">Sinergi</span></span>
        </div>
        <p class="text-gray-800 text-sm font-medium">©2026 SMP
Negeri 3 Madiun <br> All rights reserved</p>
        <p class="text-xs text-gray-400 mt-2">Connected to
Firebase Firestore</p>
    </div>
</footer>

<!-- MODALS -->
<div id="login-modal" class="fixed inset-0 bg-black bg-
opacity-60 z-50 flex items-center justify-center p-4 backdrop-blur-
sm hidden">
    <div class="bg-white rounded-3xl w-full max-w-md overflow-
hidden shadow-2xl border-4 border-yellow-400">
        <div class="bg-blue-900 p-6 text-white text-center
relative"><button onclick="closeLoginModal()" class="absolute top-4
right-4 text-white/70 hover:text-white"><i data-lucide="x"
class="w-6 h-6"></i></button><h2 class="text-2xl font-bold">Portal
Masuk</h2></div>
        <div class="flex border-b border-gray-200">
            <button onclick="setLoginRole('student')" id="tab-
student" class="flex-1 py-4 font-bold text-sm text-blue-900 border-
b-4 border-yellow-400 bg-yellow-50 transition-colors">SISWA / ORTU</
button>
            <button onclick="setLoginRole('teacher')" id="tab-
teacher" class="flex-1 py-4 font-bold text-sm text-gray-400
hover:text-gray-600 transition-colors">STAF / GURU</button>
        </div>
        <form onsubmit="handleLoginSubmit(event)" class="p-6
space-y-4">
            <div id="login-student-wrapper" class="space-
y-2"><label class="text-sm font-bold text-gray-700">NISN Siswa</
label><input type="text" id="login-student-nis" class="w-full
border-2 border-gray-200 rounded-xl p-3 outline-none"
placeholder="Masukkan NISN Siswa..."></div>
            <div id="login-teacher-wrapper" class="hidden space-
y-4"><div class="space-y-2"><label class="text-sm font-bold text-
gray-700">Username</label><input type="text" id="login-username"
class="w-full border-2 border-gray-200 rounded-xl p-3 outline-none"
placeholder="Masukkan username..."></div></div>
            <div class="space-y-2"><label class="text-sm font-
bold text-gray-700">Password</label><div class="relative"><input
type="password" id="login-password" class="w-full border-2 border-
gray-200 rounded-xl p-3 pr-10 outline-none" placeholder="Masukkan
kata sandi..." required><i data-lucide="lock" class="absolute
right-3 top-3 text-gray-400 w-5 h-5"></i></div></div>

```

```

        <button type="submit" class="w-full bg-yellow-400
hover:bg-yellow-500 text-black font-extrabold py-3.5 rounded-xl
shadow-lg">MASUK SEKARANG</button>
        <div id="login-hint" class="text-center text-xs
text-gray-400 mt-4">Hint: Siswa/Ortu = siswa123 (Default)</div>
    </form>
</div>
</div>

<!-- IMPORT MODAL -->
<div id="import-modal" class="fixed inset-0 bg-black bg-
opacity-60 z-50 flex items-center justify-center p-4 backdrop-blur-
sm hidden">
    <div class="bg-white rounded-3xl w-full max-w-2xl p-6
shadow-2xl relative animate-fade-in">
        <button onclick="closeImportModal()" class="absolute
top-4 right-4 text-gray-400 hover:text-red-500"><i data-
lucide="x"></i></button>
        <h2 class="text-2xl font-bold text-blue-900 mb-4 flex
items-center gap-2"><i data-lucide="file-spreadsheet"></i> Import
Nilai Kolektif</h2>
        <div class="grid md:grid-cols-2 gap-4 mb-4 bg-gray-50
p-4 rounded-xl border border-gray-200">
            <div class="md:col-span-2"><label class="block text-
sm font-bold text-gray-700 mb-1">Pilih Mapel (Wajib)</label><select
id="import-subject" class="w-full border border-gray-300 rounded p-2
outline-none focus:border-blue-500"></select></div>
            <div class="md:col-span-2"><label class="block text-
sm font-bold text-gray-700 mb-1">Judul Bab (Wajib)</label><input
type="text" id="import-chapter" class="w-full border border-gray-300
rounded p-2 outline-none" placeholder="Contoh: Bab 1 - Bilangan"></
div>
            <div><label class="block text-sm font-bold text-
gray-700 mb-1">Jenis Asesmen (Wajib)</label><input type="text"
id="import-type" class="w-full border border-gray-300 rounded p-2
outline-none" placeholder="Contoh: Sumatif 1"></div>
            <div><label class="block text-sm font-bold text-
gray-700 mb-1">Sub Bab / Topik</label><input type="text" id="import-
topic" class="w-full border border-gray-300 rounded p-2 outline-
none" placeholder="Contoh: Algoritma"></div>
            <div class="md:col-span-2"><label class="block text-
sm font-bold text-gray-700 mb-1">Catatan Umum (Opsional)</
label><input type="text" id="import-note" class="w-full border
border-gray-300 rounded p-2 outline-none" placeholder="Biarkan
kosong untuk generate catatan otomatis berdasarkan nilai"></div>
        </div>
        <div class="grid md:grid-cols-2 gap-6 mb-6">
            <div class="bg-yellow-50 p-4 rounded-xl border
border-yellow-200"><h3 class="font-bold text-yellow-800 mb-2">1.
Download Template</h3><button onclick="downloadGradeTemplate()"
class="bg-yellow-400 hover:bg-yellow-500 text-black px-4 py-2
rounded-lg font-bold text-sm flex items-center gap-2"><i data-
lucide="download" class="w-4 h-4"></i> Download Template Nilai</
button></div>

```

```
        <div class="bg-blue-50 p-4 rounded-xl border border-
blue-200"><h3 class="font-bold text-blue-900 mb-2">2. Contoh
Format</h3><p class="text-xs text-gray-600 font-mono bg-white p-2
rounded border border-gray-200 overflow-x-auto whitespace-
nowrap">NIS | Nama_Siswa | Kelas | Nilai</p></div>
```

```
        <div class="space-y-3">
            <label class="font-bold text-gray-700">Upload CSV /
Paste:</label>
            <input type="file" id="csv-file-input"
accept=".csv, .txt" class="block w-full text-sm text-gray-500
file:mr-4 file:py-2 file:px-4 file:rounded-full file:border-0
file:text-sm file:font-semibold file:bg-blue-50 file:text-blue-700
hover:file:bg-blue-100" onchange="handleFileUpload(this)">
            <textarea id="import-data" class="w-full h-40
border-2 border-gray-300 rounded-xl p-3 outline-none font-mono text-
sm" placeholder="Paste data di sini (Header: NIS, Nama, Kelas,
Nilai)..."></textarea>
            <div class="flex justify-end gap-3"><button
onclick="closeImportModal()" class="px-5 py-2.5 rounded-xl font-bold
text-gray-600 hover:bg-gray-100">Batal</button><button
onclick="processGradeImport()" class="bg-blue-900 hover:bg-blue-800
text-white px-6 py-2.5 rounded-xl font-bold shadow-lg flex items-
center gap-2"><i data-lucide="clipboard-paste" class="w-4 h-4"></i>
Proses Data</button></div>
        </div>
    </div>
</div>
```

```
    <!-- NEW EDIT MODAL -->
    <div id="edit-modal" class="fixed inset-0 bg-black bg-opacity-60
z-50 flex items-center justify-center p-4 backdrop-blur-sm hidden">
        <div class="bg-white rounded-3xl w-full max-w-md p-6
shadow-2xl relative">
            <button onclick="closeEditModal()" class="absolute top-4
right-4 text-gray-400 hover:text-red-500"><i data-lucide="x"></i></
button>
            <h2 class="text-xl font-bold text-blue-900 mb-4">Edit
Data Nilai</h2>
            <form onsubmit="handleEditSubmit(event)" class="space-
y-4">
                <input type="hidden" id="edit-modal-id">
                <div><label class="block text-sm font-bold text-
gray-700">Nama Siswa</label><input type="text" id="edit-student-
name" class="w-full border bg-gray-100 rounded p-2" disabled></div>
                <div><label class="block text-sm font-bold text-
gray-700">Judul Bab</label><input type="text" id="edit-chapter"
class="w-full border rounded p-2" required></div>
                <div><label class="block text-sm font-bold text-
gray-700">Jenis Asesmen</label><input type="text" id="edit-type"
class="w-full border rounded p-2" required></div>
                <div><label class="block text-sm font-bold text-
gray-700">Sub Bab / Topik Materi</label><input type="text" id="edit-
topic" class="w-full border rounded p-2" required></div>
```

```

        <div><label class="block text-sm font-bold text-
gray-700">Nilai</label><input type="number" id="edit-score"
class="w-full border rounded p-2" required></div>
        <div class="flex justify-end gap-2 pt-2"><button
type="button" onclick="closeEditModal()" class="px-4 py-2 text-sm
font-bold text-gray-500 hover:bg-gray-100 rounded-lg">Batal</
button><button type="submit" class="bg-blue-900 hover:bg-blue-800
text-white px-4 py-2 rounded-lg text-sm font-bold shadow">Simpan
Perubahan</button></div>
    </form>
</div>
</div>

<!-- ADMIN MODAL -->
<div id="admin-modal" class="fixed inset-0 bg-black bg-
opacity-60 z-50 flex items-center justify-center p-4 backdrop-blur-
sm hidden">
    <div class="bg-white rounded-3xl w-full max-w-md p-6
shadow-2xl relative">
        <button onclick="closeAdminModal()" class="absolute
top-4 right-4 text-gray-400 hover:text-red-500"><i data-
lucide="x"></i></button>
        <h2 id="admin-modal-title" class="text-xl font-bold
text-blue-900 mb-4">Tambah Data</h2>
        <form onsubmit="handleAdminSubmit(event)" class="space-
y-4">
            <input type="hidden" id="admin-edit-id"><input
type="hidden" id="admin-data-type">
            <div id="admin-fields" class="space-y-3"></div>
            <div class="flex justify-end gap-2 pt-2"><button
type="button" onclick="closeAdminModal()" class="px-4 py-2 text-sm
font-bold text-gray-500 hover:bg-gray-100 rounded-lg">Batal</
button><button type="submit" class="bg-yellow-400 hover:bg-
yellow-500 text-black px-4 py-2 rounded-lg text-sm font-bold
shadow">Simpan Data</button></div>
        </form>
    </div>
</div>

<!-- JAVASCRIPT LOGIC -->
<script>
    const firebaseConfig = { apiKey:
"AIzaSyA7THW53XHZPQtspRs_R1kIfiJ8dmFs7Dw", authDomain: "spenga-
sinergi.firebaseio.com", projectId: "spenga-sinergi",
storageBucket: "spenga-sinergi.firebaseio.com",
messagingSenderId: "236285816278", appId:
"1:236285816278:web:bbb85b97963b0ca768877" };
    firebase.initializeApp(firebaseConfig);
    const db = firebase.firestore();

    let GRADES = [], GLOBAL_GRADES = [], TEACHERS = [], SUBJECTS
= [], STUDENTS = [], USERS = [];
    let currentUser = null, loginRole = 'student', activeView =
'home';

```

```

        let unsubscribeGrades = null;
        const ASSESSMENT_TYPES = ["Ulangan Harian", "Tugas 1",
        "Tugas 2", "Tugas 3", "Tugas 4", "Tugas 5", "Unjuk Keahlian", "SAS",
        "Proyek"];

```

```

        document.addEventListener('DOMContentLoaded', () =>
        { initFirebaseListeners(); lucide.createIcons();
        switchView('home'); });

```

```

        function initFirebaseListeners() {
            db.collection('users').where('username', '==',
            'admin').get().then(snap => { if (snap.empty)
            db.collection('users').add({ username: 'admin', password:
            'admin123', role: 'admin', name: 'Super Admin' }); });
            db.collection('users').onSnapshot(snap => { USERS =
            snap.docs.map(doc => ({ id: doc.id, ...doc.data() })); });
            db.collection('teachers').onSnapshot(snap => { TEACHERS
            = snap.docs.map(doc => ({ id: doc.id, ...doc.data() }));
            renderPublicData(); if(activeView === 'admin_dashboard')
            renderAdminTeachers(); });
            db.collection('students').onSnapshot(snap => { STUDENTS
            = snap.docs.map(doc => ({ id: doc.id, ...doc.data() }));
            fillFormSelects(); if(activeView === 'admin_dashboard')
            renderAdminStudents(); if(activeView === 'home')
            renderStudentOfTheMonth(); });
            db.collection('subjects').onSnapshot(snap => { SUBJECTS
            = snap.docs.map(doc => ({ id: doc.id, ...doc.data() }));
            renderPublicData(); fillFormSelects(); if(activeView ===
            'admin_dashboard') renderAdminSubjects(); });
            db.collection('grades').onSnapshot(snap => {
                GLOBAL_GRADES = snap.docs.map(doc => ({ id:
            doc.id, ...doc.data() }));
                document.getElementById('loader').style.opacity =
            '0'; setTimeout(() =>
            document.getElementById('loader').style.display = 'none', 500);
                if(activeView === 'home') renderStudentOfTheMonth();
            });
        }

```

```

        function setupRealtimeGrades() {
            if (unsubscribeGrades) { unsubscribeGrades();
            unsubscribeGrades = null; }
            if (!currentUser) { GRADES = []; return; }
            let query = db.collection('grades');
            if (currentUser.role === 'teacher') query =
            query.where('teacherId', '==', currentUser.username);
            else if (currentUser.role === 'student') query =
            query.where('studentId', '==', currentUser.id);
            unsubscribeGrades = query.onSnapshot(snap => {
                GRADES = snap.docs.map(doc => ({ id:
            doc.id, ...doc.data() }));
                if(activeView === 'teacher_dashboard')
            renderGradeTable();
                if(activeView === 'student_dashboard')

```

```

renderStudentDashboard();
    });
}

function switchView(viewName) {
    activeView = viewName;
    ['view-home', 'view-teachers', 'view-subjects', 'view-teacher-dashboard', 'view-student-dashboard', 'view-admin-dashboard'].forEach(id => document.getElementById(id).classList.add('hidden'));
    const viewMap = { 'home': 'view-home', 'teachers_list': 'view-teachers', 'subjects_list': 'view-subjects', 'teacher_dashboard': 'view-teacher-dashboard', 'student_dashboard': 'view-student-dashboard', 'admin_dashboard': 'view-admin-dashboard' };

    document.getElementById(viewMap[viewName]).classList.remove('hidden');

    renderNav(viewName); document.getElementById('mobile-menu').classList.add('hidden');
    if (viewName === 'teacher_dashboard') { fillFormSelects(); renderGradeTable(); }
    if (viewName === 'student_dashboard') renderStudentDashboard();
    if (viewName === 'home') renderStudentOfTheMonth();
    if (viewName === 'admin_dashboard') switchAdminTab('teachers');
}

function toggleMobileMenu() { document.getElementById('mobile-menu').classList.toggle('hidden'); }

function renderNav(active) {
    const container = document.getElementById('desktop-menu'); const mobileContainer = document.getElementById('mobile-menu'); let html = '', mobileHtml = '';
    const btnClass = (isActive) => isActive ? "bg-yellow-400 text-black shadow-md px-5 py-2.5 rounded-full font-bold text-sm transition-all" : "text-gray-600 hover:bg-gray-100 px-5 py-2.5 rounded-full font-bold text-sm transition-all";
    html += `<button onclick="switchView('home')" class="${btnClass(active === 'home')}">Beranda</button>`; mobileHtml += `<button onclick="switchView('home')" class="block w-full text-left px-4 py-3 rounded-xl font-bold text-gray-700 hover:bg-gray-50">Beranda</button>`;
    if (!currentUser) {
        html += `<button onclick="switchView('teachers_list')" class="${btnClass(active === 'teachers_list')}">Data Guru</button>`; html += `<button onclick="switchView('subjects_list')" class="${btnClass(active === 'subjects_list')}">Mapel</button>`; html += `<button onclick="openLoginModal()" class="ml-4 bg-blue-900 hover:bg-blue-800 text-white px-7 py-2.5 rounded-full font-bold shadow-lg flex items-center gap-2">Masuk</button>`;
        mobileHtml += `<button`

```

```

onclick="switchView('teachers_list')" class="block w-full text-left
px-4 py-3 rounded-xl font-bold text-gray-700 hover:bg-gray-50">Data
Guru</button>`; mobileHtml += `<button
onclick="switchView('subjects_list')" class="block w-full text-left
px-4 py-3 rounded-xl font-bold text-gray-700 hover:bg-gray-50">Mata
Pelajaran</button>`; mobileHtml += `<button
onclick="openLoginModal()" class="block w-full text-left px-4 py-3
rounded-xl bg-yellow-400 text-black font-bold">Masuk Portal</
button>`;
    } else {
        if (currentUser.role === 'admin') { html += `<button
onclick="switchView('admin_dashboard')" class="${btnClass(active ===
'admin_dashboard')}">Admin Dashboard</button>`; mobileHtml +=
`<button onclick="switchView('admin_dashboard')" class="block w-full
text-left px-4 py-3 rounded-xl font-bold text-gray-700 hover:bg-
gray-50">Admin Dashboard</button>`; }
        else if (currentUser.role === 'teacher') { html +=
`<button onclick="switchView('teacher_dashboard')" class="$
{btnClass(active === 'teacher_dashboard')}">Dashboard Guru</
button>`; mobileHtml += `<button
onclick="switchView('teacher_dashboard')" class="block w-full text-
left px-4 py-3 rounded-xl font-bold text-gray-700 hover:bg-
gray-50">Dashboard Guru</button>`; }
        else { html += `<button
onclick="switchView('student_dashboard')" class="${btnClass(active
=== 'student_dashboard')}">Raport Saya</button>`; mobileHtml +=
`<button onclick="switchView('student_dashboard')" class="block w-
full text-left px-4 py-3 rounded-xl font-bold text-gray-700
hover:bg-gray-50">Raport Saya</button>`; }
        const roleDisplay = currentUser.role === 'admin' ?
'Administrator' : (currentUser.role === 'teacher' ? 'Guru' :
'Siswa');
        html += `<div class="flex items-center gap-4 ml-4
pl-4 border-l-2 border-gray-200"><div class="text-right"><p
class="text-sm font-black text-blue-900">${currentUser.name}</p><p
class="text-xs text-gray-500 font-semibold uppercase">${roleDisplay}
</p></div><button onclick="handleLogout()" class="w-10 h-10 rounded-
full bg-red-50 text-red-500 hover:bg-red-500 hover:text-white flex
items-center justify-center transition-all"><i data-lucide="log-out"
class="w-4 h-4"></i></button></div>`;
        mobileHtml += `<div class="px-4 py-3 text-sm font-
bold text-blue-900 bg-blue-50 rounded-xl mb-2">Halo, $
{currentUser.name}</div><button onclick="handleLogout()"
class="block w-full text-left px-4 py-3 rounded-xl text-red-600 bg-
red-50 font-bold">Keluar</button>`;
    }
    container.innerHTML = html; mobileContainer.innerHTML =
mobileHtml; lucide.createIcons();
}

function renderPublicData() {
    document.getElementById('teachers-grid').innerHTML =
TEACHERS.map(t => `<div class="bg-white p-6 rounded-xl shadow-sm
border border-gray-100 flex items-center gap-4 hover:shadow-md

```

```

transition"><div class="w-12 h-12 bg-gray-100 rounded-full flex
items-center justify-center text-gray-500 font-bold">$
{t.name.charAt(0)}</div><div><h3 class="font-bold text-gray-800">$
{t.name}</h3><p class="text-sm text-yellow-600 font-medium">$
{t.mapel}</p></div></div>`).join('');
    document.getElementById('subjects-grid').innerHTML =
SUBJECTS.map(s => `<div class="bg-white p-6 rounded-xl shadow-sm
border-l-4 border-yellow-400 hover:bg-yellow-50 transition cursor-
default"><h3 class="font-bold text-gray-800 text-center">${s.name}</
h3></div>`).join('');
    }

    // --- AUTH ---
    function openLoginModal() { document.getElementById('login-
modal').classList.remove('hidden'); document.getElementById('login-
student-nis').value = ''; setLoginRole('student'); }
    function closeLoginModal() { document.getElementById('login-
modal').classList.add('hidden'); }
    function setLoginRole(role) {
        loginRole = role;
        const tabS = document.getElementById('tab-student'),
        tabT = document.getElementById('tab-teacher'), wrapS =
document.getElementById('login-student-wrapper'), wrapT =
document.getElementById('login-teacher-wrapper'), hint =
document.getElementById('login-hint');
        if (role === 'student') { tabS.className = "flex-1 py-4
font-bold text-sm text-blue-900 border-b-4 border-yellow-400 bg-
yellow-50 transition-colors"; tabT.className = "flex-1 py-4 font-
bold text-sm text-gray-400 hover:text-gray-600 transition-colors";
wrapS.classList.remove('hidden'); wrapT.classList.add('hidden');
hint.innerText = "Hint: Siswa/Ortu = siswa123 (Default)"; }
        else { tabT.className = "flex-1 py-4 font-bold text-sm
text-blue-900 border-b-4 border-yellow-400 bg-yellow-50 transition-
colors"; tabS.className = "flex-1 py-4 font-bold text-sm text-
gray-400 hover:text-gray-600 transition-colors";
wrapS.classList.add('hidden'); wrapT.classList.remove('hidden');
hint.innerText = "Hint: Login Guru/Admin"; }
    }
    function handleLoginSubmit(e) {
        e.preventDefault();
        const password = document.getElementById('login-
password').value;
        if (loginRole === 'teacher') {
            const username = document.getElementById('login-
username').value;
            const user = USERS.find(u => u.username === username
&& u.password === password);
            if (user) { currentUser = user;
setupRealtimeGrades(); switchView(user.role === 'admin' ?
'admin_dashboard' : 'teacher_dashboard'); closeLoginModal(); } else
{ alert("Username atau Password Salah!"); }
        } else {
            const nisInput = document.getElementById('login-
student-nis').value.trim();

```



```

        const s = STUDENTS.find(x => x.nis === nisInput ||
x.username === nisInput);
        if (!s) return alert("NISN tidak ditemukan!");
        if (s && (s.password === password || (!s.password &&
password === 'siswa123')))) { currentUser = { role:
'student', ...s }; setupRealtimeGrades();
switchView('student_dashboard'); closeLoginModal(); } else
{ alert("Password Salah!"); }
    }
    function handleLogout() { currentUser = null;
if(unsubscribeGrades) { unsubscribeGrades(); unsubscribeGrades =
null; } GRADES = []; switchView('home'); }

    // --- GURU ---
    function fillFormSelects() {
        const classes = [...new Set(STUDENTS.map(s =>
s.class))].sort();
        document.getElementById('input-class-filter').innerHTML
= '<option value="">-- Pilih Kelas --</option>' + classes.map(c =>
`<option value="${c}">${c}</option>`).join('');
        document.getElementById('input-subject').innerHTML =
SUBJECTS.map(s => `<option value="${s.name}">${s.name}</
option>`).join('');
        document.getElementById('input-type').innerHTML =
ASSESSMENT_TYPES.map(t => `<option value="${t}">${t}</
option>`).join('');
        document.getElementById('import-subject').innerHTML =
'<option value="">-- Pilih Mapel --</option>' + SUBJECTS.map(s =>
`<option value="${s.name}">${s.name}</option>`).join('');
    }
    function filterStudentsByClass() {
        const cls = document.getElementById('input-class-
filter').value, input = document.getElementById('input-student-
search'), datalist = document.getElementById('student-options');
        input.value = ''; datalist.innerHTML = '';
        if (!cls) { input.disabled = true; return; }
        input.disabled = false;
        const filtered = STUDENTS.filter(s => s.class ===
cls).sort((a, b) => a.name.localeCompare(b.name));
        datalist.innerHTML = filtered.map(s => `<option value="$
{s.nis} - ${s.name}">`).join('');
    }
    function updateStatusPreview() { const score =
document.getElementById('input-score').value;
document.getElementById('status-preview').innerText = `Status: $
{score >= 75 ? 'Tuntas' : 'Belum Tuntas'}`; }
    function autoGenerateAI() {
        const inputValue = document.getElementById('input-
student-search').value, score =
parseFloat(document.getElementById('input-score').value);
        if(!inputValue || isNaN(score)) return alert("Isi data
dulu.");
        const nisan = inputValue.split(' - ')[0], s =

```

```

STUDENTS.find(x => x.nis === nsn);
    if (!s) return alert("Siswa tidak valid.");
    document.getElementById('input-note').value = score >=
90 ? `Luar biasa! ${s.name} sangat menguasai materi.` : score >=
75 ? `Bagus. ${s.name} tuntas.` : `Perlu remedial.`;
    }
    function handleGradeSubmit(e) {
        e.preventDefault();
        const id = document.getElementById('edit-id').value,
inputValue = document.getElementById('input-student-search').value,
nisn = inputValue.split(' - ')[0], s = STUDENTS.find(x => x.nis ===
nisn);
        if (!s) return alert("Pilih siswa dari daftar.");
        const data = {
            studentId: s.id, teacherId: currentUser.username,
            subject: document.getElementById('input-
subject').value,
            type: document.getElementById('input-type').value,
            topic: document.getElementById('input-topic').value,
            chapter: document.getElementById('input-
chapter').value || "Bab Umum", // NEW FIELD
            score: parseFloat(document.getElementById('input-
score').value),
            note: document.getElementById('input-note').value,
            date: new Date().toISOString()
        };
        if(id) db.collection('grades').doc(id).update(data);
    else db.collection('grades').add(data);
        resetForm();
    }
    function resetForm() {
        document.getElementById('form-title').innerText = "Input
Nilai Manual"; document.getElementById('edit-id').value = "";
        document.getElementById('input-class-filter').value = "";
        document.getElementById('input-student-search').value =
        ""; document.getElementById('input-student-search').disabled = true;
        document.getElementById('input-topic').value = "";
        document.getElementById('input-chapter').value = "";
        document.getElementById('input-score').value = "";
        document.getElementById('input-note').value = "";
        document.getElementById('btn-cancel').classList.add('hidden');
    }
    function editGrade(id) {
        const g = GRADES.find(x => x.id == id), s =
STUDENTS.find(x => x.id == g.studentId);
        if (s) {
            document.getElementById('edit-id').value = g.id;
            document.getElementById('input-class-filter').value = s.class;
            filterStudentsByClass();
            document.getElementById('input-student-
search').value = `${s.nis} - ${s.name}`;
            document.getElementById('input-subject').value = g.subject;
            document.getElementById('input-type').value =
g.type; document.getElementById('input-topic').value = g.topic;

```

```

document.getElementById('input-score').value = g.score;
    document.getElementById('input-note').value =
g.note; document.getElementById('btn-
cancel').classList.remove('hidden'); window.scrollTo({top:0,
behavior:'smooth'});
    }
    }
    function deleteGrade(id) { if(confirm("Hapus nilai?"))
db.collection('grades').doc(id).delete(); }

    // --- NEW: EDIT FEATURE ---
    function openEditGradeModal(id) {
        const g = GRADES.find(x => x.id == id); const s =
STUDENTS.find(st => st.id === g.studentId);
        if (!g || !s) return;
        document.getElementById('edit-
modal').classList.remove('hidden');
        document.getElementById('edit-modal-id').value = g.id;
        document.getElementById('edit-student-name').value =
s.name;
        document.getElementById('edit-chapter').value =
g.chapter || "Bab Umum";
        document.getElementById('edit-type').value = g.type;
        document.getElementById('edit-topic').value = g.topic;
        document.getElementById('edit-score').value = g.score;
    }
    function closeEditModal() { document.getElementById('edit-
modal').classList.add('hidden'); }
    function handleEditSubmit(e) {
        e.preventDefault();
        const id = document.getElementById('edit-modal-
id').value;
        db.collection('grades').doc(id).update({
            chapter: document.getElementById('edit-
chapter').value,
            type: document.getElementById('edit-type').value,
            topic: document.getElementById('edit-topic').value,
            score: parseFloat(document.getElementById('edit-
score').value)
        }).then(() => { closeEditModal(); });
    }

    // --- NEW: RENDER 3-LEVEL NESTED ---
    function renderGradeTable() {
        const container = document.getElementById('grades-
accordion-container');
        const enrichedGrades = GRADES.map(g => {
            const s = STUDENTS.find(st => st.id ===
g.studentId);
            return { ...g, studentName: s ? s.name : 'Unknown',
studentClass: s ? s.class : 'Unknown', studentNis: s ? s.nis : '-',
chapter: g.chapter || "Bab Umum" };
        });
    }

```

```

// Group: Class -> Chapter
const grouped = {};
enrichedGrades.forEach(g => {
  if (!grouped[g.studentClass])
grouped[g.studentClass] = {};
  if (!grouped[g.studentClass][g.chapter])
grouped[g.studentClass][g.chapter] = [];
  grouped[g.studentClass][g.chapter].push(g);
});

let html = '';
const sortedClasses = Object.keys(grouped).sort();

if (sortedClasses.length === 0) html = `<div
class="text-center p-12 text-gray-400 bg-gray-50 rounded-xl border-2
border-dashed border-gray-200">Belum ada data nilai.</div>`;
else {
  sortedClasses.forEach(cls => {
    const accordionId = `acc-cls-${cls.replace(/\s+/
g, '-')}`;

    const classChapters = grouped[cls];

    html += `
      <div class="bg-white rounded-xl shadow-sm border
border-gray-200 overflow-hidden mb-4 hover:shadow-md transition-
all">
        <div class="p-4 flex flex-col sm:flex-row
justify-between items-center bg-blue-50 border-b border-blue-100">
          <button onclick="toggleAccordion('${
{accordionId}}')" class="flex-1 flex items-center gap-3 text-left w-
full focus:outline-none group">
            <div class="bg-blue-600 text-white
p-2 rounded-lg group-hover:bg-blue-700 transition-colors"><i data-
lucide="users" class="w-6 h-6"></i></div>
            <div><h3 class="font-bold text-xl
text-blue-900">Kelas ${cls}</h3></div>
            <i id="icon-${{accordionId}}" data-
lucide="chevron-down" class="w-5 h-5 text-blue-400 ml-2 transition-
transform"></i>
          </button>
          <button onclick="downloadRekapKelas('${
{cls}}')" class="bg-green-600 hover:bg-green-700 text-white px-4 py-2
rounded-lg font-bold text-sm flex items-center gap-2 shadow-sm
transition-all active:scale-95 whitespace-nowrap">
            <i data-lucide="download" class="w-4
h-4"></i> Unduh Leger Kelas
          </button>
        </div>

        <div id="${{accordionId}}" class="hidden p-6
space-y-6">
          `;

    // Level 2: Chapters

```

```

Object.keys(classChapters).sort().forEach(chap
=> {
    const grades =
classChapters[chap].sort((a,b) =>
a.studentName.localeCompare(b.studentName));

    html += `

```

```

        });
        html += `</tbody></table></div></div>`;
    });
    html += `</div></div>`;
});
}
container.innerHTML = html;
lucide.createIcons();
}

// Toggle Function
function toggleAccordion(id) {
    const content = document.getElementById(id);
    const icon = document.getElementById(`icon-${id}`);
    if (content.classList.contains('hidden'))
{ content.classList.remove('hidden'); icon.style.transform =
'rotate(180deg)'; }
        else { content.classList.add('hidden');
icon.style.transform = 'rotate(0deg)'; }
    }

// --- NEW: PIVOT TABLE DOWNLOAD ---
function downloadRekapKelas(className) {
    if (!currentUser) return;
    const classGrades = GRADES.filter(g => {
        const s = STUDENTS.find(st => st.id ===
g.studentId);
        return s && s.class === className;
    });

    if (classGrades.length === 0) return alert("Tidak ada
data nilai untuk kelas ini.");

    // 1. Prepare Columns (Unique Chapter + Type)
    const uniqueCols = new Set();
    const uniqueChapters = new Set();
    classGrades.forEach(g => {
        const chap = g.chapter || "Bab Umum";
        uniqueChapters.add(chap);
        uniqueCols.add(`${chap} - ${g.type}`);
    });
    const sortedCols = Array.from(uniqueCols).sort();
    const sortedChapters =
Array.from(uniqueChapters).sort();

    // 2. Map Student Data
    const studentMap = {};
    classGrades.forEach(g => {
        const chap = g.chapter || "Bab Umum";
        if (!studentMap[g.studentId]) {
            const s = STUDENTS.find(st => st.id ===
g.studentId);
            studentMap[g.studentId] = { NISN: s ? s.nis :
'-', Nama: s ? s.name : 'Unknown', Kelas: className, Scores: {},

```

```

ChapTotals: {} };
    }
    const colKey = `${chap} - ${g.type}`;
    // Handle multiple scores for same type by taking
latest or avg? Taking latest for simplicity in Pivot
    studentMap[g.studentId].Scores[colKey] = g.score;

    if (!studentMap[g.studentId].ChapTotals[chap])
studentMap[g.studentId].ChapTotals[chap] = { sum: 0, count: 0 };
    studentMap[g.studentId].ChapTotals[chap].sum +=
g.score;
    studentMap[g.studentId].ChapTotals[chap].count += 1;
});

// 3. Header
let csvContent = "data:text/csv;charset=utf-8,\uFEFF";
let headers = ["NISN", "Nama Siswa", "Kelas"];
sortedChapters.forEach(chap => {
    sortedCols.filter(c => c.startsWith(chap)).forEach(c
=> headers.push(c));
    headers.push(`Rata-rata ${chap}`);
});
headers.push("Rata-rata Total");
csvContent += headers.join(",") + "\n";

// 4. Rows
Object.values(studentMap).sort((a,b) =>
a>Nama.localeCompare(b>Nama)).forEach(s => {
    let row = [`${s.NISN}`, `${s>Nama}`, `${
{s.Kelas}`];
    let grandSum = 0, grandCount = 0;

    sortedChapters.forEach(chap => {
        sortedCols.filter(c =>
c.startsWith(chap)).forEach(c => {
            row.push(s.Scores[c] || "");
        });
        const chapData = s.ChapTotals[chap];
        if (chapData) {
            row.push((chapData.sum /
chapData.count).toFixed(2));
            grandSum += chapData.sum; grandCount +=
chapData.count;
        } else { row.push(""); }
    });

    row.push(grandCount > 0 ? (grandSum /
grandCount).toFixed(2) : "");
    csvContent += row.join(",") + "\n";
});

// Download
const fileName = `Leger_Nilai_${className}_${
currentUser.name}.csv`;

```

```

        const encodedUri = encodeURI(csvContent);
        const link = document.createElement("a");
        link.setAttribute("href", encodedUri);
        link.setAttribute("download", fileName);
        document.body.appendChild(link);
        link.click();
        document.body.removeChild(link);
    }

    // --- NEW: SOTM LOGIC (HOMEPAGE) ---
    function renderStudentOfTheMonth() {
        if (!GLOBAL_GRADES || GLOBAL_GRADES.length === 0)
        { document.getElementById('sotm-container').innerHTML = '<div
class="col-span-3 text-center text-gray-400 py-8">Belum ada data
prestasi.</div>'; return; }
        const studentStats = {};
        GLOBAL_GRADES.forEach(g => {
            const s = STUDENTS.find(st => st.id ===
g.studentId);
            if (s) {
                if (!studentStats[s.id]) studentStats[s.id] =
{ name: s.name, class: s.class, total: 0, count: 0 };
                studentStats[s.id].total += g.score;
                studentStats[s.id].count += 1;
            }
        });
        const flatStats = Object.values(studentStats).map(stat
=> ({ name: stat.name, class: stat.class, avg: stat.total /
stat.count, level: stat.class.charAt(0) }));
        const levels = ['7', '8', '9']; let html = '';
        levels.forEach(lvl => {
            const topStudents = flatStats.filter(s => s.level
=== lvl).sort((a, b) => b.avg - a.avg).slice(0, 3);
            html += `<div class="bg-white rounded-2xl shadow-xl
border border-gray-100 overflow-hidden flex flex-col"><div
class="bg-gradient-to-r from-blue-900 to-blue-800 p-4 text-
center"><h3 class="text-xl font-black text-white tracking-
widest">KELAS ${lvl}</h3></div><div class="p-6 flex-1 flex flex-col
gap-4">`;
            if (topStudents.length === 0) html += `<div
class="text-center text-gray-400 text-sm py-4">Belum ada data</
div>`;
            else {
                topStudents.forEach((s, idx) => {
                    const rankClass = idx === 0 ? 'rank-1' : idx
=== 1 ? 'rank-2' : 'rank-3';
                    html += `<div class="flex items-center gap-4
p-3 rounded-xl border-2 ${rankClass} bg-opacity-10"><div
class="flex-shrink-0 w-10 h-10 rounded-full flex items-center
justify-center font-bold text-lg bg-white shadow-sm">${idx + 1}</
div><div class="flex-1 min-w-0"><p class="font-bold truncate text-
gray-800">${s.name}</p><p class="text-xs text-gray-600">${s.class}</
p></div><div class="text-right"><span class="block font-black text-
lg">${s.avg.toFixed(1)}</span></div></div>`;
                }
            }
        });
    }

```



```

        });
    }
    html += `</div></div>`;
});
document.getElementById('sotm-container').innerHTML =
html;
}

// --- GLOBAL EXPORT (BACKUP) ---
function exportToCSV() {
    if (!currentUser) return;
    const dataToExport = GRADES.map(g => {
        const s = STUDENTS.find(st => st.id ===
g.studentId);
        return {
            Nama: s ? s.name : "Unknown",
            NISN: s ? s.nis : "-",
            Kelas: s ? s.class : "-",
            Mapel: g.subject,
            JenisAsesmen: g.type,
            Materi: g.topic,
            Nilai: g.score,
            Catatan: g.note ? g.note.replace(/,/g, " ") :
            "-";
        };
    });
    if (dataToExport.length === 0) { alert("Belum ada data
nilai untuk diunduh."); return; }
    let csvContent = "data:text/csv;charset=utf-8,\uFEFF";
    csvContent += "Nama Siswa,NISN,Kelas,Mata
Pelajaran,Jenis Asesmen,Materi/Topik,Nilai,Catatan Guru\n";
    dataToExport.forEach(row => { csvContent += `"$
{row.Nama}","${row.NISN},${row.Kelas},"${row.Mapel}","$
{row.JenisAsesmen}","${row.Materi}","${row.Nilai},"$
{row.Catatan}"\n`; });
    const date = new Date().toISOString().slice(0,10);
    const teacherName = currentUser.name.replace(/\s+/g,
' ');
    const fileName = `Rekap_Nilai_${teacherName}_${
date}.csv`;
    const encodedUri = encodeURIComponent(csvContent);
    const link = document.createElement("a");
    link.setAttribute("href", encodedUri); link.setAttribute("download",
fileName);
    document.body.appendChild(link); link.click();
    document.body.removeChild(link);
}

// --- IMPORT/EXPORT (Updated Process) ---
function processGradeImport() {
    const raw = document.getElementById('import-
data').value;
    const subject = document.getElementById('import-
subject').value;

```

```

        const type = document.getElementById('import-
type').value;
        const topic = document.getElementById('import-
topic').value;
        const chapter = document.getElementById('import-
chapter').value; // NEW
        const noteInput = document.getElementById('import-
note').value;
        if(!raw) return alert("Data kosong");
        if(!subject || !type) return alert("Lengkapi data mapel/
asesmen.");

        Papa.parse(raw, {
            header: true, skipEmptyLines: true,
            complete: async function(results) {
                let batch = db.batch(); let count = 0;
                results.data.forEach((row) => {
                    const nisKey = Object.keys(row).find(k =>
k.trim().toUpperCase() === 'NIS' || k.trim().toUpperCase() ===
'NISN');
                    const nilaiKey = Object.keys(row).find(k =>
k.trim().toUpperCase() === 'NILAI');
                    if(nisKey && nilaiKey) {
                        const nisVal =
row[nisKey].toString().trim();
                        const s = STUDENTS.find(x => x.nis ===
nisVal);
                        if(s) {
                            const score =
parseFloat(row[nilaiKey]);
                            let finalNote = noteInput;
                            if (!finalNote) {
                                if (score >= 91) finalNote =
"Sangat Baik! Pertahankan prestasimu.";
                                else if (score >= 75) finalNote
= "Kompetensi Tercapai (Tuntas).";
                                else finalNote = "Perlu
Perbaikan / Remedial.";
                            }
                            const ref =
db.collection('grades').doc();
                            batch.set(ref, {
                                studentId: s.id,
                                teacherId: currentUser.username,
                                subject: subject, topic: topic
|| "Topik Umum", chapter: chapter || "Bab Umum", type: type,
                                score: score, note: finalNote,
                                date: new Date().toISOString()
                            });
                            count++;
                        }
                    }
                });
            if (count > 0) { await batch.commit();

```

```

alert(`Berhasil import ${count} data.`); closeImportModal(); }
    else { alert("Gagal. Pastikan NIS valid."); }
    }
  });
}
function openImportModal()
{ document.getElementById('import-
modal').classList.remove('hidden'); document.getElementById('import-
subject').innerHTML = '<option value="">-- Pilih Mapel --</option>'
+ SUBJECTS.map(s => `<option value="${s.name}">${s.name}</
option>`).join(''); }
function closeImportModal()
{ document.getElementById('import-modal').classList.add('hidden'); }
function downloadGradeTemplate() { const csv =
"NIS,Nilai\n12345,90\n"; const blob = new Blob([csv], { type: 'text/
csv' }); const a = document.createElement('a'); a.href =
window.URL.createObjectURL(blob); a.download =
"Template_Import_Nilai.csv"; a.click(); }
function handleFileUpload(input) { if(input.files[0])
{ const r = new FileReader(); r.onload = (e) =>
document.getElementById('import-data').value = e.target.result;
r.readAsText(input.files[0]); } }

// Admin functions unchanged
function switchAdminTab(tab) {
  ['teachers', 'students', 'subjects'].forEach(t =>
{ document.getElementById(`tab-admin-${t}`).className = "px-6 py-3
font-bold text-sm text-gray-500 hover:bg-gray-50 rounded-t-lg
transition-all"; document.getElementById(`admin-panel-${t}`
).classList.add('hidden'); });
  document.getElementById(`tab-admin-${tab}`).className =
"px-6 py-3 font-bold text-sm border-b-4 border-yellow-400 text-
blue-900 bg-yellow-50 rounded-t-lg transition-all";
  document.getElementById(`admin-panel-${tab}`
).classList.remove('hidden');
  if (tab === 'teachers') renderAdminTeachers(); if (tab
=== 'students') renderAdminStudents(); if (tab === 'subjects')
renderAdminSubjects();
}
function renderAdminTeachers()
{ document.getElementById('admin-table-teachers').innerHTML =
TEACHERS.map(t => `<tr class="hover:bg-gray-50 border-b border-
gray-100"><td class="p-4">${t.name}</td><td class="p-4">${t.mapel}</
td><td class="p-4 font-mono text-xs text-blue-600">${
t.username||'-'}</td><td class="p-4 font-mono text-xs text-
gray-400">${t.password||'*****'}</td><td class="p-4 text-
right"><button onclick="editAdminData('teacher','${t.id}')"
class="text-blue-600">Edit</button> <button
onclick="deleteAdminData('teachers','${t.id}')" class="text-
red-600">Hapus</button></td></tr>`).join(''); }
function renderAdminStudents() {
  const term = document.getElementById('search-
student')?.value.toLowerCase() || "";
  const filtered = STUDENTS.filter(s =>

```

```

s.name.toLowerCase().includes(term) ||
s.class.toLowerCase().includes(term));
    filtered.sort((a,b) => a.class.localeCompare(b.class) ||
a.name.localeCompare(b.name));
    document.getElementById('admin-table-
students').innerHTML = filtered.map(s => `<tr class="hover:bg-
gray-50 border-b border-gray-100"><td class="p-4 font-mono text-xs
text-gray-500">${s.nis||'-'}</td><td class="p-4 font-bold">${s.name}
</td><td class="p-4"><span class="bg-blue-100 text-blue-800 px-2
py-1 rounded text-xs font-bold">${s.class}</span></td><td class="p-4
font-mono text-xs text-blue-600">${s.username||'-'}</td><td
class="p-4 font-mono text-xs text-gray-400">${s.password||'*****'}
</td><td class="p-4 text-right"><button
onclick="editAdminData('student','${s.id}')" class="text-
blue-600">Edit</button> <button
onclick="deleteAdminData('students','${s.id}')" class="text-
red-600">Hapus</button></td></tr>`).join('');
    }
    function renderAdminSubjects()
{ document.getElementById('admin-table-subjects').innerHTML =
SUBJECTS.map(s => `<tr class="hover:bg-gray-50 border-b border-
gray-100"><td class="p-4 font-bold">${s.name}</td><td class="p-4
text-right"><button onclick="editAdminData('subject','${s.id}')"
class="text-blue-600">Edit</button> <button
onclick="deleteAdminData('subjects','${s.id}')" class="text-
red-600">Hapus</button></td></tr>`).join(''); }
    function openAdminModal(type) {
        document.getElementById('admin-
modal').classList.remove('hidden');
        document.getElementById('admin-data-type').value = type;
        document.getElementById('admin-edit-id').value = "";
        document.getElementById('admin-modal-title').innerText =
`Tambah ${type}`;
        const container = document.getElementById('admin-
fields');
        if (type === 'teacher') container.innerHTML = `<input
type="text" id="adm-t-name" class="w-full border p-2 rounded"
placeholder="Nama Guru" required><input type="text" id="adm-t-mapel"
class="w-full border p-2 rounded" placeholder="Mata Pelajaran"
required><hr><input type="text" id="adm-t-user" class="w-full border
p-2 rounded bg-yellow-50" placeholder="Username (Login)"
required><input type="text" id="adm-t-pass" class="w-full border p-2
rounded bg-yellow-50" placeholder="Password" required>`;
        else if (type === 'student') container.innerHTML =
`<input type="text" id="adm-s-id" class="w-full border p-2 rounded"
placeholder="NIS" required><input type="text" id="adm-s-name"
class="w-full border p-2 rounded" placeholder="Nama Siswa"
required><input type="text" id="adm-s-class" class="w-full border
p-2 rounded" placeholder="Kelas (ex: 7A)" required><hr><input
type="text" id="adm-s-user" class="w-full border p-2 rounded bg-
yellow-50" placeholder="Username (Opsional)"><input type="text"
id="adm-s-pass" class="w-full border p-2 rounded bg-yellow-50"
placeholder="Password (Opsional)">`;
        else if (type === 'subject') container.innerHTML =

```

```

`<input type="text" id="adm-sub-name" class="w-full border p-2
rounded" placeholder="Nama Mata Pelajaran" required>`;
    }
    function closeAdminModal() { document.getElementById('admin-
modal').classList.add('hidden'); }
    function handleAdminSubmit(e) {
        e.preventDefault();
        const type = document.getElementById('admin-data-
type').value;
        const editId = document.getElementById('admin-edit-
id').value;
        let col = '', data = {};
        if (type === 'teacher') {
            const uname = document.getElementById('adm-t-
user').value;
            const pass = document.getElementById('adm-t-
pass').value;
            col = 'teachers'; data = { name:
document.getElementById('adm-t-name').value, mapel:
document.getElementById('adm-t-mapel').value, username: uname,
password: pass };
            if (!editId) db.collection('users').add({ username:
uname, password: pass, role: 'teacher', name: data.name });
            } else if (type === 'student') {
                col = 'students'; data = { nis:
document.getElementById('adm-s-id').value, name:
document.getElementById('adm-s-name').value, class:
document.getElementById('adm-s-class').value, username:
document.getElementById('adm-s-user').value, password:
document.getElementById('adm-s-pass').value };
                } else if (type === 'subject') { col = 'subjects'; data
= { name: document.getElementById('adm-sub-name').value }; }
                if (editId) db.collection(col).doc(editId).update(data);
            else db.collection(col).add(data);
            closeAdminModal();
        }
        function deleteAdminData(col, id) { if(confirm("Hapus data
ini?")) db.collection(col).doc(id).delete(); }
        function editAdminData(type, id) { openAdminModal(type);
document.getElementById('admin-edit-id').value = id; /* populate
fields */ }
    }
</script>
</body>
</html>

```