

/db/chart.json

```
Pertemuan-3 > db > {} chartjson > ...
1  {
2    "students": [
3      { "id": 1, "faculty": "Teknik", "count": 320 },
4      { "id": 2, "faculty": "Ekonomi", "count": 210 },
5      { "id": 3, "faculty": "Kedokteran", "count": 150 },
6      { "id": 4, "faculty": "Hukum", "count": 180 },
7      { "id": 5, "faculty": "Ilmu Komputer", "count": 260 }
8    ],
9    "genderRatio": [
10     { "id": 1, "gender": "Laki-laki", "count": 600 },
11     { "id": 2, "gender": "Perempuan", "count": 700 }
12   ],
13   "registrations": [
14     { "id": 1, "year": 2020, "total": 500 },
15     { "id": 2, "year": 2021, "total": 580 },
16     { "id": 3, "year": 2022, "total": 640 },
17     { "id": 4, "year": 2023, "total": 710 }
18   ],
19   "gradeDistribution": [
20     { "subject": "TI", "A": 120, "B": 80, "C": 40 },
21     { "subject": "SI", "A": 100, "B": 90, "C": 50 },
22     { "subject": "MI", "A": 110, "B": 85, "C": 30 }
23   ],
24   "lecturerRanks": [
25     { "rank": "Asisten Ahli", "count": 30 },
26     { "rank": "Lektor", "count": 50 },
27     { "rank": "Lektor Kepala", "count": 40 },
28     { "rank": "Guru Besar", "count": 10 }
29   ]
30 }
```

/src/Pages/Layouts/Utils/Apis/ChartApi.js

```
Pertemuan-3 > src > Pages > Layouts > Utils > Apis > JS ChartApijs > ...
1  // ChartApi.jsx
2  import axios from "@/Pages/Layouts/Utils/AxiosInstance";
3
4  export const getAllChartData = () => axios.get("/chart");
```

/src/Pages/Layouts/Utils/Hooks/useChart.jsx

```
Pertemuan-3 > src > Pages > Layouts > Utils > Hooks > useChartjsx > ...
1  import { useQuery } from "@tanstack/react-query";
2  import { getAllChartData } from "@/Pages/Layouts/Utils/Apis/ChartApi";
3
4  export const useChartData = () =>
5  {
6    useQuery({
7      queryKey: ["chart", "all"],
8      queryFn: getAllChartData,
9      select: (res) => res.data,
10    });
11  }
```

/src/Pages/Admin/Dashboard.jsx

```
Pertemuan-3 > src > Pages > Admin > Dashboard.jsx > ...
1 // Dashboard.jsx
2 import React from "react";
3 import {
4   BarChart,
5   Bar,
6   PieChart,
7   Pie,
8   Cell,
9   LineChart,
10  Line,
11  RadarChart,
12  Radar,
13  AreaChart,
14  Area,
15  XAxis,
16  YAxis,
17  PolarGrid,
18  PolarAngleAxis,
19  PolarRadiusAxis,
20  CartesianGrid,
21  Tooltip,
22  Legend,
23  ResponsiveContainer,
24 } from "recharts";
25
26 import { useChartData } from "@Pages/Layouts/Utils/Hooks/useChart";
27
28 const COLORS = ["#8884d8", "#82ca9d", "#ffc658", "#ff8042", "#8dd1e1"];
29
30 const Dashboard = () => {
31   const { data = {}, isLoading } = useChartData();
32
33   const {
34     students = [],
35     genderRatio = [],
36     registrations = [],
37     gradeDistribution = [],
38     lecturerRanks = [],
39   } = data;
40 }
```

```
41 if (isLoading) {
42   return (
43     <div className="p-6 text-center text-gray-500">
44       Loading chart data...
45     </div>
46   );
47 }
48
49 return (
50   <div className="p-6">
51     <h1 className="text-2xl font-bold mb-6">
52       Selamat Datang di Dashboard
53     </h1>
54
55     /* GRID LAYOUT */
56     <div className="grid grid-cols-1 md:grid-cols-2 gap-6">
57
58       /* ===== A. BAR CHART ===== */
59       <div className="bg-white p-4 rounded shadow">
60         <h2 className="font-semibold mb-4">
61           Jumlah Mahasiswa per Fakultas
62         </h2>
63
64         <ResponsiveContainer width="100%" height={300}>
65           <BarChart data={students}>
66             <CartesianGrid strokeDasharray="3 3" />
67             <XAxis dataKey="faculty" />
68             <YAxis />
69             <Tooltip />
70             <Legend />
71             <Bar dataKey="count" fill="#8884d8" />
72           </BarChart>
73         </ResponsiveContainer>
74       </div>
75
76       /* ===== B. PIE CHART ===== */
77       <div className="bg-white p-4 rounded shadow">
78         <h2 className="font-semibold mb-4">
79           Rasio Gender Mahasiswa
80         </h2>
81 
```

```

81
82
83     <ResponsiveContainer width="100%" height={300}>
84         <PieChart>
85             <Pie
86                 data={genderRatio}
87                 dataKey="count"
88                 nameKey="gender"
89                 cx="50%"
90                 cy="50%"
91                 outerRadius={80}
92                 label
93             >
94                 {genderRatio.map((_, index) => (
95                     <Cell
96                         key={`cell-${index}`}
97                         fill={COLORS[index % COLORS.length]}
98                     />
99                 ))}
100             </Pie>
101             <Tooltip />
102             <Legend />
103         </PieChart>
104     </ResponsiveContainer>
105 </div>
106
107     { /* ===== C. LINE CHART ===== */}
108     <div className="bg-white p-4 rounded shadow">
109         <h2 className="font-semibold mb-4">
110             Registrasi Mahasiswa per Tahun
111         </h2>

```

```

112
113         <ResponsiveContainer width="100%" height={300}>
114             <LineChart data={registrations}>
115                 <CartesianGrid strokeDasharray="3 3" />
116                 <XAxis dataKey="year" />
117                 <YAxis />
118                 <Tooltip />
119                 <Legend />
120                 <Line
121                     type="monotone"
122                     dataKey="total"
123                     stroke="■ #82ca9d"
124                     strokeWidth={2}
125                 />
126             </LineChart>
127         </ResponsiveContainer>
128     </div>

```

```

128
129      {/* ===== D. RADAR CHART ===== */}
130      <div className="bg-white p-4 rounded shadow">
131        <h2 className="font-semibold mb-4">
132          Nilai Mahasiswa per Jurusan
133        </h2>
134
135        <ResponsiveContainer width="100%" height={300}>
136          <RadarChart
137            cx="50%"
138            cy="50%"
139            outerRadius="80%"
140            data={gradeDistribution}
141          >
142            <PolarGrid />
143            <PolarAngleAxis dataKey="subject" />
144            <PolarRadiusAxis />
145            <Radar
146              name="A"
147              dataKey="A"
148              stroke="#8884d8"
149              fill="#8884d8"
150              fillOpacity={0.6}
151            />
152            <Radar
153              name="B"
154              dataKey="B"
155              stroke="#82ca9d"
156              fill="#82ca9d"
157              fillOpacity={0.4}
158            />
159            <Radar
160              name="C"
161              dataKey="C"
162              stroke="#ffc658"
163              fill="#ffc658"
164              fillOpacity={0.3}
165            />
166            <Legend />
167          </RadarChart>
168        </ResponsiveContainer>
169      </div>

```

```

170
171      {/* ===== E. AREA CHART (FULL WIDTH) ===== */}
172      <div className="bg-white p-4 rounded shadow md:col-span-2">
173        <h2 className="font-semibold mb-4">
174          Jumlah Dosen Berdasarkan Pangkat
175        </h2>
176
177        <ResponsiveContainer width="100%" height={300}>
178          <AreaChart data={lecturerRanks}>
179            <defs>
180              <linearGradient
181                id="colorLecturer"
182                x1="0"
183                y1="0"
184                x2="0"
185                y2="1"
186              >
187                <stop
188                  offset="5%"
189                  stopColor="#8884d8"
190                  stopOpacity={0.8}
191                />
192                <stop
193                  offset="95%"
194                  stopColor="#8884d8"
195                  stopOpacity={0}
196                />
197              </linearGradient>
198            </defs>
199            <XAxis dataKey="rank" />
200            <YAxis />
201            <CartesianGrid strokeDasharray="3 3" />
202            <Tooltip />
203            <Area

```

```

203              <Area
204                type="monotone"
205                dataKey="count"
206                stroke="#8884d8"
207                fillOpacity={1}
208                fill="url(#colorLecturer)"
209              />
210            </AreaChart>
211          </ResponsiveContainer>
212        </div>
213
214      </div>
215    </div>
216  );
217 };
218
219 export default Dashboard;
220

```

/src/Pages/Admin/RencanaStudi.jsx

```
Pertemuan-3 > src > Pages > Admin > RencanaStudi.jsx > RencanaStudi
1  import { useEffect, useState } from "react";
2  import Card from "@Pages/Layouts/Components/Card";
3  import Heading from "@Pages/Layouts/Components/Heading";
4  import Button from "@Pages/Layouts/Components/Button";
5  import { useAuthStateContext } from "@Pages/Layouts/Utils/Context/AuthContext";
6  import { toastError, toastSuccess } from "@Pages/Layouts/Utils/Helpers/ToastHelpers";
7  import { confirmDelete } from "@Pages/Layouts/Utils/Helpers/SwalHelpers";
8  import TableRencanaStudi from "../TableRencanaStudi";
9  import ModalRencanaStudi from "../ModalRencanaStudi";
10 import { getAllMahasiswa } from "@Pages/Layouts/Utils/Apis/MahasiswaApi";
11
12 // Kelas
13 import {
14   getAllKelas,
15   storeKelas,
16   updateKelas,
17   deleteKelas,
18 } from "@Pages/Layouts/Utils/Apis/KelasApi";
19
20 // Dosen
21 import { getAllDosen } from "@Pages/Layouts/Utils/Apis/DosenApi";
22
23 // Mata Kuliah
24 import { getAllMataKuliah } from "@Pages/Layouts/Utils/Apis/MataKuliahApi";
25
26 const RencanaStudi = () => {
27   const { user } = useAuthStateContext();
28
29   // ===== STATE DATA =====
30   const [kelas, setKelas] = useState([]);
31   const [dosen, setDosen] = useState([]);
32   const [mahasiswa, setMahasiswa] = useState([]);
33   const [mataKuliah, setMataKuliah] = useState([]);
34
35   // ===== STATE UI =====
36   const [selectedMhs, setSelectedMhs] = useState({});
37   const [selectedDsn, setSelectedDsn] = useState({});
38   const [isModalOpen, setIsModalOpen] = useState(false);
39   const [form, setForm] = useState({ mata_kuliah_id: "", dosen_id: "" });
40
```

```
41 // ===== FETCH DATA =====
42 useEffect(() => {
43   fetchData();
44 }, []);
45
46 const fetchData = async () => {
47   try {
48     const [resKelas, resDosen, resMahasiswa, resMataKuliah] =
49       await Promise.all([
50         getAllKelas(),
51         getAllDosen(),
52         getAllMahasiswa(),
53         getAllMataKuliah(),
54       ]);
55
56     setKelas(resKelas.data);
57     setDosen(resDosen.data);
58     setMahasiswa(resMahasiswa.data);
59     setMataKuliah(resMataKuliah.data);
60   } catch (error) {
61     console.error(error);
62     toastError("Gagal mengambil data rencana studi");
63   }
64 };
65
66 // ===== UTILITIES SKS =====
67 const getMaxSks = (id) =>
68   mahasiswa.find((m) => m.id === id)?.max_sks || 0;
69
70 const getDosenMaxSks = (id) =>
71   dosen.find((d) => d.id === id)?.max_sks || 0;
72
73 // ===== OPEN ADD MODAL =====
74 const openAddModal = () => {
75   setForm({ mata_kuliah_id: "", dosen_id: "" });
76   setIsModalOpen(true);
77 };
78
```

```

78
79 // ===== SUBMIT TAMBAH KELAS =====
80 const handleSubmit = async (e) => {
81   e.preventDefault();
82
83   if (!form.mata_kuliah_id || !form.dosen_id) {
84     toastError("Form tidak lengkap");
85     return;
86   }
87
88   const dosenSudahDipakai = kelas.find(
89     k => k.dosen_id === form.dosen_id
90   );
91
92   if (dosenSudahDipakai) {
93     toastError("Dosen ini sudah mengampu mata kuliah lain");
94     return;
95   }
96
97   await storeKelas({
98     ...form,
99     mahasiswa_ids: [],
100   });
101
102   setIsModalOpen(false);
103   toastSuccess("Kelas ditambahkan");
104   fetchData();
105 };
106
107 // ===== HANDLE CHANGE FORM =====
108 const handleChange = (e) => {
109   setForm({
110     ...form,
111     [e.target.name]: e.target.value,
112   });
113 };
114

```

```

115 // ===== TAMBAH MAHASISWA =====
116 const handleAddMahasiswa = async (kelasItem, mhsId) => {
117   if (!mhsId) return;
118
119   const matkul = mataKuliah.find(
120     (m) => m.id === kelasItem.mata_kuliah_id
121   );
122   const sks = matkul?.sks || 0;
123
124   const totalSksMahasiswa = kelas
125     .filter((k) => k.mahasiswa_ids.includes(mhsId))
126     .map(
127       (k) =>
128         mataKuliah.find((m) => m.id === k.mata_kuliah_id)?.sks || 0
129     )
130     .reduce((acc, curr) => acc + curr, 0);
131
132   const maxSks = getMaxSks(mhsId);
133
134   if (totalSksMahasiswa + sks > maxSks) {
135     toastError(`SKS melebihi batas maksimal (${maxSks})`);
136     return;
137   }
138
139   if (kelasItem.mahasiswa_ids.includes(mhsId)) {
140     toastError("Mahasiswa sudah terdaftar");
141     return;
142   }
143
144   const updated = {
145     ...kelasItem,
146     mahasiswa_ids: [...kelasItem.mahasiswa_ids, mhsId],
147   };
148
149   await updateKelas(kelasItem.id, updated);
150   toastSuccess("Mahasiswa berhasil ditambahkan");
151
152   setSelectedMhs((prev) => ({ ...prev, [kelasItem.id]: "" }));
153   fetchData();
154 };
155

```

```

156 // ===== HAPUS MAHASISWA =====
157 const handleDeleteMahasiswa = async (kelasItem, mhsId) => {
158   const updated = {
159     ...kelasItem,
160     mahasiswa_ids: kelasItem.mahasiswa_ids.filter((id) => id !== mhsId),
161   };
162
163   await updateKelas(kelasItem.id, updated);
164   toastSuccess("Mahasiswa dihapus");
165   fetchData();
166 };
167
168 // ===== GANTI DOSEN =====
169 const handleChangeDosen = async (kelasItem) => {
170   const dsnId = selectedDsn[kelasItem.id];
171   if (!dsnId) return;
172
173   const totalSksDosen = kelas
174     .filter((k) => k.dosen_id === dsnId)
175     .map(
176       (k) =>
177         mataKuliah.find((m) => m.id === k.mata_kuliah_id)?.sks || 0
178     )
179     .reduce((acc, curr) => acc + curr, 0);
180
181   const kelasSks =
182     mataKuliah.find((m) => m.id === kelasItem.mata_kuliah_id)?.sks || 0;
183
184   const maxSks = getDosenMaxSks(dsnId);
185
186   if (totalSksDosen + kelasSks > maxSks) {
187     toastError("Dosen melebihi batas maksimal SKS ({maxSks})");
188     return;
189   }
190
191   await updateKelas(kelasItem.id, { ...kelasItem, dosen_id: dsnId });
192   toastSuccess("Dosen diperbarui");
193   fetchData();
194 };
195

```

```

195
196 // ===== HAPUS KELAS =====
197 const handleDeleteKelas = async (kelasId) => {
198   confirmDelete(async () => {
199     await deleteKelas(kelasId);
200     toastSuccess("Kelas dihapus");
201     fetchData();
202   });
203 };
204
205 // ===== 1 Dosen 1 Matakuliah =====
206 const dosenSudahPunyaKelas = kelas.map(k => k.dosen_id);
207 const dosenAvailable = dosen.filter(
208   d => !dosenSudahPunyaKelas.includes(d.id)
209 );
210
211 // ===== RENDER =====
212 return (
213   <>
214     <Card>
215       <div className="flex justify-between items-center mb-4">
216         <Heading as="h2">Rencana Studi</Heading>
217
218         {user.permission?.includes("rencana-studi.page") && (
219           <Button onClick={openAddModal}>
220             + Tambah Kelas
221           </Button>
222         )}
223       </div>
224

```

```

224
225     <TableRencanaStudi
226       kelas={kelas}
227       mahasiswa={mahasiswa}
228       dosen={dosen}
229       mataKuliah={mataKuliah}
230       selectedMhs={selectedMhs}
231       setSelectedMhs={setSelectedMhs}
232       selectedDsn={selectedDsn}
233       setSelectedDsn={setSelectedDsn}
234       handleAddMahasiswa={handleAddMahasiswa}
235       handleDeleteMahasiswa={handleDeleteMahasiswa}
236       handleChangeDosen={handleChangeDosen}
237       handleDeleteKelas={handleDeleteKelas}
238     />
239   </Card>
240
241   <ModalRencanaStudi
242     isOpen={isModalOpen}
243     onClose={() => setModalOpen(false)}
244     onChange={const handleSubmit: (e: any) => Promise<void>}
245     onSubmit={handleSubmit}
246     form={form}
247     dosen={dosenAvailable}
248     mataKuliah={mataKuliah}
249   />
250 </>
251 );
252 };
253
254 export default RencanaStudi;
255

```

/src/Pages/Admin/TableRencanaStudi.jsx

```

Pertemuan-3 > src > Pages > Admin > TableRencanaStudi.jsx > TableRencanaStudi
1  import Button from "@/Pages/Layouts/Components/Button";
2  import Select from "@/Pages/Layouts/Components/Select";
3
4  export default function TableRencanaStudi({
5    kelas,
6    mahasiswa,
7    dosen,
8    mataKuliah,
9    selectedMhs,
10   setSelectedMhs,
11   selectedDsn,
12   setSelectedDsn,
13   handleAddMahasiswa,
14   handleDeleteMahasiswa,
15   handleChangeDosen,
16   handleDeleteKelas
17 }) {
18
19   return (
20     <div className="space-y-6">
21       {kelas.map((kls) => {
22         const matkul = mataKuliah.find(m => m.id === kls.mata_kuliah_id);
23         const dosenPengampu = dosen.find(d => d.id === kls.dosen_id);
24         const mhsInClass = kls.mahasiswa_ids
25           .map(id => mahasiswa.find(m => m.id === id))
26           .filter(Boolean);
27
28         return (
29           <div key={kls.id} className="border rounded shadow bg-white">
30             {/* HEADER */}
31             <div className="flex justify-between items-center px-4 py-3 border-b bg-gray-50">
32               <div>
33                 <h3 className="text-lg font-semibold">{matkul?.name || "-"}</h3>
34                 <p className="text-sm text-gray-600">
35                   Dosen: <strong>{dosenPengampu?.name || "-"}</strong>
36                 </p>
37               </div>
38

```

```

38
39
40     <div className="flex items-center gap-2">
41       <Select
42         value={selectedDsn[kls.id] || ""}
43         onChange={(e) =>
44           setSelectedDsn({ ...selectedDsn, [kls.id]: e.target.value })
45         }
46         size="sm"
47         className="w-48"
48       >
49         <option value="">-- Ganti Dosen --</option>
50         {dosen.map(d => (
51           <option key={d.id} value={d.id}>{d.name}</option>
52         ))}
53       </Select>
54
55       <Button size="sm" onClick={() => handleChangeDosen(kls)}>
56         Simpan
57       </Button>
58
59       {mhsInClass.length === 0 && (
60         <Button
61           size="sm"
62           variant="danger"
63           onClick={() => handleDeleteKelas(kls.id)}
64         >
65           Hapus Kelas
66         </Button>
67       )}
68     </div>
69   </div>

```

```

69
70   /* TABLE MAHASISWA */
71   <table className="w-full text-sm">
72     <thead className="bg-blue-600 text-white">
73       <tr>
74         <th className="py-2 px-4 text-left">No</th>
75         <th className="py-2 px-4 text-left">Nama</th>
76         <th className="py-2 px-4 text-left">NIM</th>
77         <th className="py-2 px-4 text-center">Total SKS</th>
78         <th className="py-2 px-4 text-center">Max SKS</th>
79         <th className="py-2 px-4 text-center">Aksi</th>
80       </tr>
81     </thead>
82     <tbody>
83       {mhsInClass.length > 0 ? (
84         mhsInClass.map((m, i) => {
85           const totalSks = kelas
86             .filter(k => k.mahasiswa_ids.includes(m.id))
87             .map(k => mataKuliah.find(mk => mk.id === k.mata_kuliah_id)?.sks || 0)
88             .reduce((a, b) => a + b, 0);
89
90           return (
91             <tr key={m.id} className={i % 2 === 0 ? "bg-white" : "bg-gray-100"}>
92               <td className="py-2 px-4">{i + 1}</td>
93               <td className="py-2 px-4">{m.name}</td>
94               <td className="py-2 px-4">{m.nim}</td>
95               <td className="py-2 px-4 text-center">{totalSks}</td>
96               <td className="py-2 px-4 text-center">{m.max_sks}</td>
97               <td className="py-2 px-4 text-center">
98                 <Button
99                   size="sm"
100                   variant="danger"
101                   onClick={() => handleDeleteMahasiswa(kls, m.id)}
102                 >
103                   Hapus
104                 </Button>
105               </td>
106             </tr>
107           );
108         });
109       ) : (

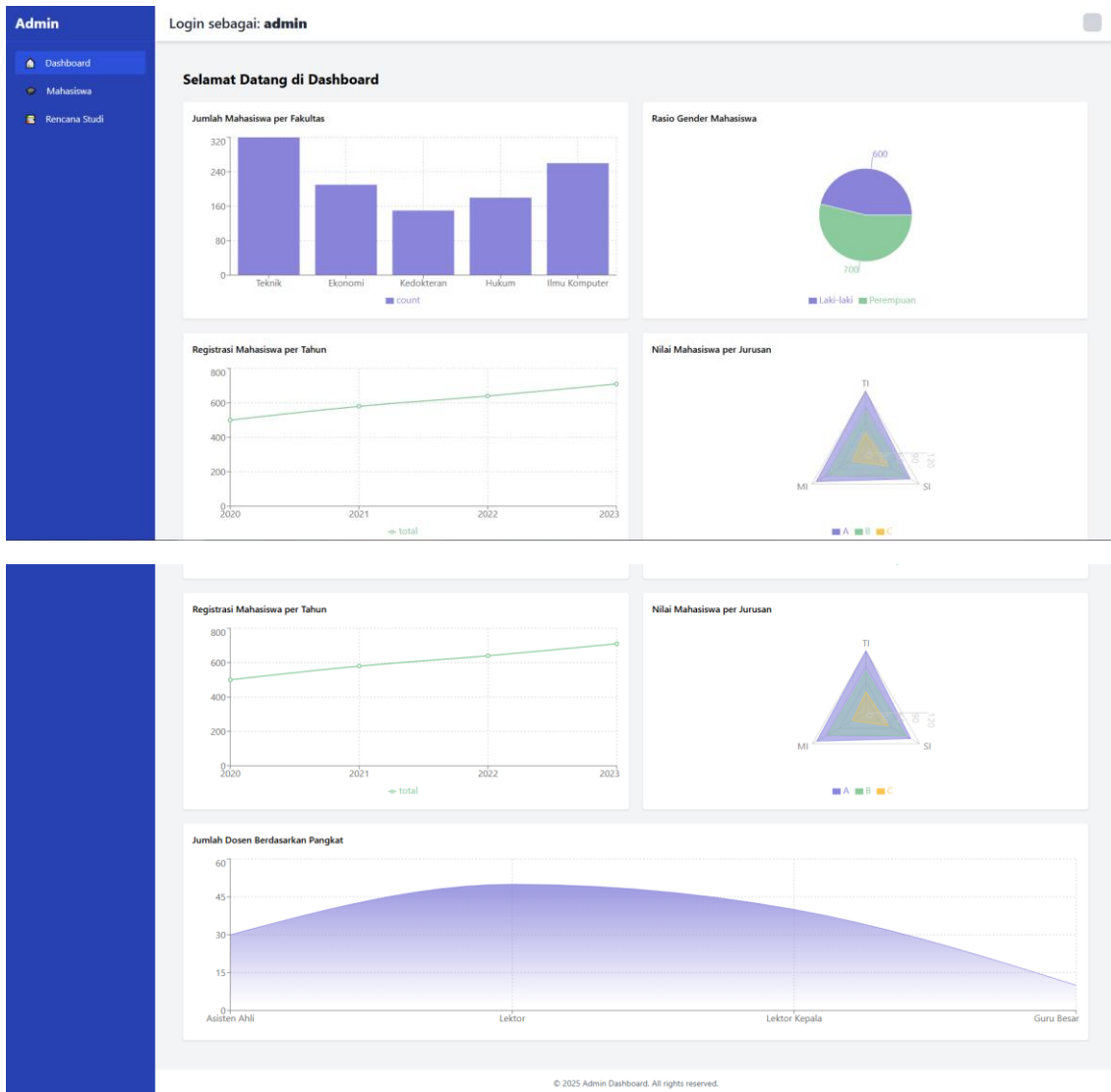
```

```

110 <tr>
111 <td colSpan="5" className="py-3 px-4 text-center italic text-gray-500">
112 Belum ada mahasiswa.
113 </td>
114 </tr>
115 </tbody>
116 </table>
117
118
119 /* TAMBAH MAHASISWA */
120 <div className="flex items-center gap-2 px-4 py-3 border-t bg-gray-50">
121 <Select
122 value={selectedMhs[kls.id] || ""}
123 onChange={(e) =>
124   setSelectedMhs(( ...selectedMhs, [kls.id]: e.target.value })
125 }
126 size="sm"
127 className="w-56"
128 >
129 <option value="">-- Pilih Mahasiswa --</option>
130 {mahasiswa.map((m) => (
131   <option key={m.id} value={m.id}>
132     {m.name} ({m.nim})
133   </option>
134 ))}
135 </Select>
136
137 <Button
138   size="sm"
139   onClick={() =>
140     handleAddMahasiswa(kls, selectedMhs[kls.id])
141   }
142 >
143   Tambah Mahasiswa
144 </Button>
145 </div>
146 </div>
147 );
148 ]]);
149 </div>
150 );
151 }

```

Tampilan :



Menampilkan maksimal sks yang dapat diambil dan jumlah sks yang diambil mahasiswa:

Admin Login sebagai: **admin**

Rencana Studi + Tambah Kelas

Algoritma dan Pemrograman
Dosen: Blamet

-- Ganti Dosen -- Simpan

No	Nama	NIM	Total SKS	Max SKS	Aksi
1	Andi	M001	6	18	<button>Hapus</button>
2	Budi	M002	3	20	<button>Hapus</button>
3	Citra	M003	6	22	<button>Hapus</button>
4	Dina	M004	3	18	<button>Hapus</button>
5	Eko	M005	3	24	<button>Hapus</button>

-- Pilih Mahasiswa -- Tambah Mahasiswa

Basis Data
Dosen: Vina

-- Ganti Dosen -- Simpan

No	Nama	NIM	Total SKS	Max SKS	Aksi
1	Andi	M001	6	18	<button>Hapus</button>
2	Putri	M016	3	18	<button>Hapus</button>
3	Citra	M003	6	22	<button>Hapus</button>

-- Pilih Mahasiswa -- Tambah Mahasiswa

Saat Dosen sudah memilih 1 matakuliah maka tidak bisa ambil lagi:

Admin Login sebagai: **admin**

Rencana Studi [+ Tambah Kelas](#)

Algoritma dan Pemrograman
Dosen: Slamet [-- Ganti Dosen --](#) [Simpan](#)

No	Nama	NIM	Total SKS	Max SKS	Aksi
1	Andi			18	Hapus
2	Budi			20	Hapus
3	Citra			22	Hapus
4	Dina			18	Hapus
5	Eko			24	Hapus

[-- Pilih Mahasiswa --](#) [Tambah Mahasiswa](#)

Tambah Kelas Baru ×

Mata Kuliah
Algoritma dan Pemrograman

Dosen Pengampu
[-- Pilih --](#)
Tini
Udin
Wawan

Basis Data
Dosen: Vina [-- Ganti Dosen --](#) [Simpan](#)

No	Nama	NIM	Total SKS	Max SKS	Aksi
1	Andi	M001	6	18	Hapus
2	Putri	M016	3	18	Hapus
3	Citra	M003	6	22	Hapus

[-- Pilih Mahasiswa --](#) [Tambah Mahasiswa](#)