

LAMPIRAN

A. *Source code pada controller*

```
1. using System;
2. using System.Collections.Generic;
3. using System.Data;
4. using System.Data.Entity;
5. using System.Linq;
6. using System.Net;
7. using System.Web;
8. using System.Web.Mvc;
9. using StudentPresence.Models;
10. using Microsoft.AspNet.Identity;
11. using PagedList;
12.
13. namespace StudentPresence.Controllers
14. {
15.     [Authorize]
16.     public class Acd_Sched_RealController : Controller
17.     {
18.         private UMYEntities db = new UMYEntities();
19.
20.         // GET: Acd_Sched_Real
21.         public ActionResult Index(string sortOrder, string currentFilter, string
searchString, int? page)
22.         {
23.             ViewBag.CurrentSort = sortOrder;
24.             ViewBag.DateSortParm = sortOrder == "Date" ? "date_desc" :
"Date";
25.
26.
27.             if (searchString != null)
28.             {
29.                 page = 1;
30.             }
31.             else
32.             {
33.                 searchString = currentFilter;
34.             }
35.
36.             ViewBag.CurrentFilter = searchString;
37.
38.             var employee = User.Identity.GetUserName();
```

```

39.     var User_Id = (from n in db.Emp_Employee where
n.Email_Corporate == employee select n.Employee_Id);
40.
41.     var acd_Sched_Real = db.Acd_Sched_Real.Where(a =>
a.Employee_Id == User_Id.FirstOrDefault());
42.     if (!String.IsNullOrEmpty(searchString))
43.     {
44.         acd_Sched_Real = acd_Sched_Real.Where(s =>
s.Acd_Course.Course_Name.Contains(searchString));
45.     }
46.
47.     switch (sortOrder)
48.     {
49.         case "Date":
50.             acd_Sched_Real = acd_Sched_Real.OrderBy(s => s.Date);
51.             break;
52.         case "date_desc":
53.             acd_Sched_Real = acd_Sched_Real.OrderByDescending(s =>
s.Date);
54.             break;
55.         default:
56.             acd_Sched_Real = acd_Sched_Real.OrderByDescending(s =>
s.Date);
57.             break;
58.     }
59.
60.     int pageSize = 4;
61.     int pageNumber = (page ?? 1);
62.     return View(acd_Sched_Real.ToPagedList(pageNumber,
pageSize));
63. }
64.
65. // GET: Acd_Sched_Real/Details/5
66. public ActionResult Details(int? id)
67. {
68.     if (id == null)
69.     {
70.         return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
71.     }
72.     Acd_Sched_Real acd_Sched_Real = db.Acd_Sched_Real.Find(id);
73.     if (acd_Sched_Real == null)
74.     {
75.         return HttpNotFound();
76.     }
77.
78.     if (acd_Sched_Real.Time_End == null)

```

```

79.     {
80.         ViewBag.visible = "visibility:visible";
81.     }
82.     else
83.     {
84.         ViewBag.visible = "visibility:hidden";
85.     }
86.
87.     return View(acd_Sched_Real);
88. }
89. public string checkData(int count, int id)
90. {
91.     var check = db.Acd_Student_Sched.Where(m => m.Sched_Real_Id
    == id).Count();
92.
93.     if (check != count)
94.     {
95.         return "true";
96.     }
97.     else
98.     {
99.         return "false";
100.    }
101.
102. }
103. public ActionResult _Details(int? id)
104. {
105.
106.     if (id == null)
107.     {
108.         return new
    HttpStatusCodeResult(HttpStatusCode.BadRequest);
109.     }
110.     Acd_Sched_Real acd_Sched_Real =
    db.Acd_Sched_Real.Find(id);
111.     if (acd_Sched_Real == null)
112.     {
113.         return HttpNotFound();
114.     }
115.     return PartialView(acd_Sched_Real);
116. }
117.
118. // GET: Acd_Sched_Real/Create
119. public ActionResult Create()
120. {
121.     string employee=null;

```

```

122.     IQueryable<int> User_Id = null;
123.     List<sp_termyear_Result> getTermYear = null;
124.     List<sp_prodi_Result> getDepartment = null;
125.     List<sp_matakuliah_Result> getCourse = null;
126.     List<sp_classprog_Result> getClassProg = null;
127.     List<sp_kelas_Result> getClass = null;
128.     try
129.     {
130.         employee = User.Identity.GetUserName();
131.         User_Id = (from n in db.Emp_Employee where
                n.Email_Corporate == employee select n.Employee_Id);
132.
133.         getTermYear = db.sp_termyear(User_Id.Single()).ToList();
134.         getDepartment = db.sp_prodi(User_Id.Single(),
135.             getTermYear.Take(1).Single().Term_Year_Id).ToList();
136.         getCourse = db.sp_matakuliah(User_Id.Single(),
137.             getTermYear.Take(1).Single().Term_Year_Id,
138.             getDepartment.Take(1).Single().Department_Id).ToList();
139.         getClassProg = db.sp_classprog(User_Id.Single(),
140.             getTermYear.Take(1).Single().Term_Year_Id,
141.             getCourse.Take(1).Single().Course_Id).ToList();
142.         getClass = db.sp_kelas(User_Id.Single(),
143.             getTermYear.Take(1).Single().Term_Year_Id,
144.             getDepartment.Take(1).Single().Department_Id,
145.             getCourse.Take(1).Single().Course_Id).ToList();
146.     }
147.     catch
148.     {
149.         ViewBag.error = "Kemungkinan data Dosen belum di
                tambahkan di ketersediaan Dosen di Mata Kuliah";
150.
151.         return View();
152.     }
153.     ViewBag.Term_Year_Id = new SelectList(getTermYear,
        "Term_Year_Id", "Term_Year_Name");
154.     ViewBag.Department_Id = new SelectList(getDepartment,
        "Department_Id", "Department_Name");
155.     ViewBag.Course_Id = new SelectList(getCourse, "Course_Id",
        "Course_Name");
156.     ViewBag.Class_Prog_Id = new SelectList(getClassProg,
        "Class_Prog_Id", "Class_Program_Name");
157.     ViewBag.Class_Id = new SelectList(getClass, "Class_Id",
        "Class_Name");
158.     ViewBag.Room_Id = new SelectList(db.Mstr_Room, "Room_Id",
        "Room_Name");
159.     ViewBag.Employee_Id = new SelectList(User_Id);

```

```

160.
161.     //For Code TOKEN (Generate Number)
162.     int txtPassLength = 8;
163.
164.     string allowedChars = "";
165.     allowedChars = "a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z,";
166.     allowedChars +=
    "A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z,";
167.     allowedChars += "1,2,3,4,5,6,7,8,9,0";
168.     char[] sep = { ',' };
169.     string[] arr = allowedChars.Split(sep);
170.     string Token = "";
171.     string temp = "";
172.     int duplikasi = 0;
173.     do
174.     {
175.         Token = "";
176.         temp = "";
177.         Random rand = new Random();
178.         for (int i = 0; i < txtPassLength; i++)
179.         {
180.             temp = arr[i];
181.             temp = arr[rand.Next(0, arr.Length)];
182.             Token += temp;
183.         }
184.         var query = db.Acd_Sched_Real.Where(w => w.Date ==
    DateTime.Now && w.Token == Token).Count();
185.         duplikasi = query;
186.     } while (duplikasi > 0);
187.
188.     ViewBag.Token = Token.ToString();
189.
190.     //DATETIME NOW
191.     var autofill = new Acd_Sched_Real
192.     {
193.         Time_Start = DateTime.Now,
194.         Date = DateTime.Now,
195.         Max_Minutes = 30
196.     };
197.     return View(autofill);
198. }
199.
200. public ActionResult Course(short termYearId, short departmentId)
201. {
202.     var employee = User.Identity.GetUserName();

```

```

203.         var User_Id = (from n in db.Emp_Employee where
                n.Email_Corporate == employee select n.Employee_Id);
204.
205.         var getCourse = db.sp_matakuliah(User_Id.Single(), termYearId,
                departmentId).ToList();
206.         ViewBag.Course_Id = new SelectList(getCourse, "Course_Id",
                "Course_Name");
207.         return PartialView();
208.     }
209.     public ActionResult Class(short termYearId, short departmentId, int
                courseId)
210.     {
211.         var employee = User.Identity.GetUserName();
212.         var User_Id = (from n in db.Emp_Employee where
                n.Email_Corporate == employee select n.Employee_Id);
213.
214.         var getClass = db.sp_kelas(User_Id.Single(), termYearId,
                departmentId, courseId).ToList();
215.         ViewBag.Class_Id = new SelectList(getClass, "Class_Id",
                "Class_Name");
216.         return PartialView();
217.     }
218.     public ActionResult ClassProg(short termYearId, int courseId)
219.     {
220.         var employee = User.Identity.GetUserName();
221.         var User_Id = (from n in db.Emp_Employee where
                n.Email_Corporate == employee select n.Employee_Id);
222.
223.         var getClassProg = db.sp_classprog(User_Id.Single(), termYearId,
                courseId).ToList();
224.         ViewBag.Class_Prog_Id = new SelectList(getClassProg,
                "Class_Prog_Id", "Class_Program_Name");
225.         return PartialView();
226.     }
227.     public ActionResult MeetingOrder(short termYearId, short
                classProgId, int courseId, short classId)
228.     {
229.         var employee = User.Identity.GetUserName();
230.         var User_Id = (from n in db.Emp_Employee where
                n.Email_Corporate == employee select n.Employee_Id);
231.
232.         var getMeetingOrder = db.sp_Meeting_Order(User_Id.Single(),
                termYearId, classProgId, courseId, classId);
233.         ViewBag.Meeting_Order = new SelectList(getMeetingOrder);
234.         return PartialView();
235.

```

```

236.     }
237.     // POST: Acd_Sched_Real/Create
238.     [HttpPost]
239.     [ValidateAntiForgeryToken]
240.     public ActionResult Create([Bind(Include =
        "Sched_Real_Id,Term_Year_Id,Class_Prog_Id,Course_Id,Class_Id,Meetin
        g_Order,Room_Id,Employee_Id,Date,Time_Start,Time_End,Token,Cours
        e_Content,Closed_By,Max_Minutes,Description,Created_By,Created_Dat
        e,Modified_By,Modified_Date")] Acd_Sched_Real acd_Sched_Real)
241.     {
242.         if (ModelState.IsValid)
243.         {
244.             db.Acd_Sched_Real.Add(acd_Sched_Real);
245.             db.SaveChanges();
246.             return RedirectToAction("Details", new { id =
                acd_Sched_Real.Sched_Real_Id });
247.         }
248.
249.         ViewBag.Course_Id = new SelectList(db.Acd_Course,
        "Course_Id", "Course_Code", acd_Sched_Real.Course_Id);
250.         ViewBag.Employee_Id = new SelectList(db.Emp_Employee,
        "Employee_Id", "Nik", acd_Sched_Real.Employee_Id);
251.         ViewBag.Class_Id = new SelectList(db.Mstr_Class, "Class_Id",
        "Class_Name", acd_Sched_Real.Class_Id);
252.         ViewBag.Class_Prog_Id = new
        SelectList(db.Mstr_Class_Program, "Class_Prog_Id", "Class_Prog_Code",
        acd_Sched_Real.Class_Prog_Id);
253.         ViewBag.Room_Id = new SelectList(db.Mstr_Room, "Room_Id",
        "Room_Code", acd_Sched_Real.Room_Id);
254.         ViewBag.Term_Year_Id = new SelectList(db.Mstr_Term_Year,
        "Term_Year_Id", "Term_Year_Name", acd_Sched_Real.Term_Year_Id);
255.         return View(acd_Sched_Real);
256.     }
257.
258.     // GET: Acd_Sched_Real/Edit/5
259.     public ActionResult EditPopup(int? id)
260.     {
261.         if (id == null)
262.         {
263.             return new
                HttpStatusCodeResult(HttpStatusCode.BadRequest);
264.         }
265.         Acd_Sched_Real acd_Sched_Real =
                db.Acd_Sched_Real.Find(id);
266.         if (acd_Sched_Real == null)
267.         {

```

```

268.         return HttpNotFound();
269.     }
270.     var employee = User.Identity.GetUserName();
271.     var User_Id = (from n in db.Emp_Employee where
n.Email_Corporate == employee select n.Employee_Id);
272.
273.     var getTermYear = db.sp_termyear(User_Id.Single()).ToList();
274.     var getDepartment = db.sp_prodi(User_Id.Single(),
275.         getTermYear.Take(1).Single().Term_Year_Id).ToList();
276.     var getCourse = db.sp_matakuliah(User_Id.Single(),
277.         getTermYear.Take(1).Single().Term_Year_Id,
278.         getDepartment.Take(1).Single().Department_Id).ToList();
279.     var getClassProg = db.sp_classprog(User_Id.Single(),
280.         getTermYear.Take(1).Single().Term_Year_Id,
281.         getCourse.Take(1).Single().Course_Id).ToList();
282.     var getClass = db.sp_kelas(User_Id.Single(),
283.         getTermYear.Take(1).Single().Term_Year_Id,
284.         getDepartment.Take(1).Single().Department_Id,
285.         getCourse.Take(1).Single().Course_Id).ToList();
286.
287.     ViewBag.Term_Year_Id = new SelectList(getTermYear,
"Term_Year_Id", "Term_Year_Name", acd_Sched_Real.Term_Year_Id);
288.     ViewBag.Department_Id = new SelectList(getDepartment,
"Department_Id", "Department_Name");
289.     ViewBag.Course_Id = new SelectList(getCourse, "Course_Id",
"Course_Name", acd_Sched_Real.Course_Id);
290.     ViewBag.Class_Prog_Id = new SelectList(getClassProg,
"Class_Prog_Id", "Class_Program_Name",
acd_Sched_Real.Class_Prog_Id);
291.     ViewBag.Class_Id = new SelectList(getClass, "Class_Id",
"Class_Name", acd_Sched_Real.Class_Id);
292.     ViewBag.Room_Id = new SelectList(db.Mstr_Room, "Room_Id",
"Room_Name", acd_Sched_Real.Room_Id);
293.     ViewBag.Employee_Id = new SelectList(User_Id);
294.
295.     acd_Sched_Real.Time_End = DateTime.Now;
296.
297.     return View(acd_Sched_Real);
298. }
299.
300. [HttpPost]
301. [ValidateAntiForgeryToken]
302. public ActionResult EditPopup(Acd_Sched_Real acd_Sched_Real)
303. {
304.     if (ModelState.IsValid)
305.     {

```



```

306.         db.Entry(acd_Sched_Real).State = EntityState.Modified;
307.         db.Entry(acd_Sched_Real).Property(x =>
           x.Term_Year_Id).IsModified = false;
308.         db.Entry(acd_Sched_Real).Property(x =>
           x.Class_Prog_Id).IsModified = false;
309.         db.Entry(acd_Sched_Real).Property(x =>
           x.Course_Id).IsModified = false;
310.         db.Entry(acd_Sched_Real).Property(x =>
           x.Class_Id).IsModified = false;
311.         db.Entry(acd_Sched_Real).Property(x =>
           x.Meeting_Order).IsModified = false;
312.         db.Entry(acd_Sched_Real).Property(x =>
           x.Room_Id).IsModified = false;
313.         db.Entry(acd_Sched_Real).Property(x =>
           x.Employee_Id).IsModified = false;
314.         db.Entry(acd_Sched_Real).Property(x => x.Date).IsModified =
           false;
315.         db.Entry(acd_Sched_Real).Property(x =>
           x.Time_Start).IsModified = false;
316.
317.         db.Entry(acd_Sched_Real).Property(x => x.Token).IsModified
           = false;
318.         db.Entry(acd_Sched_Real).Property(x =>
           x.Max_Minutes).IsModified = false;
319.         db.SaveChanges();
320.         return RedirectToAction("Details", new { id =
           acd_Sched_Real.Sched_Real_Id });
321.     }
322.     return View(acd_Sched_Real);
323. }
324. // GET: Acd_Sched_Real/Edit/5
325. public ActionResult Edit(int? id)
326. {
327.     if (id == null)
328.     {
329.         return new
           HttpStatusCodeResult(HttpStatusCode.BadRequest);
330.     }
331.     Acd_Sched_Real acd_Sched_Real =
           db.Acd_Sched_Real.Find(id);
332.     if (acd_Sched_Real == null)
333.     {
334.         return HttpNotFound();
335.     }
336.     var employee = User.Identity.GetUserName();

```

```

337.     var User_Id = (from n in db.Emp_Employee where
n.Email_Corporate == employee select n.Employee_Id);
338.
339.     var getTermYear = db.sp_termyear(User_Id.Single()).ToList();
340.     var getDepartment = db.sp_prodi(User_Id.Single(),
341.     getTermYear.Take(1).Single().Term_Year_Id).ToList();
342.     var getCourse = db.sp_matakuliah(User_Id.Single(),
343.     getTermYear.Take(1).Single().Term_Year_Id,
344.     getDepartment.Take(1).Single().Department_Id).ToList();
345.     var getClassProg = db.sp_classprog(User_Id.Single(),
346.     getTermYear.Take(1).Single().Term_Year_Id,
347.     getCourse.Take(1).Single().Course_Id).ToList();
348.     var getClass = db.sp_kelas(User_Id.Single(),
349.     getTermYear.Take(1).Single().Term_Year_Id,
350.     getDepartment.Take(1).Single().Department_Id,
351.     getCourse.Take(1).Single().Course_Id).ToList();
352.
353.     ViewBag.Term_Year_Id = new SelectList(getTermYear,
"Term_Year_Id", "Term_Year_Name", acd_Sched_Real.Term_Year_Id);
354.     ViewBag.Department_Id = new SelectList(getDepartment,
"Department_Id", "Department_Name");
355.     ViewBag.Course_Id = new SelectList(getCourse, "Course_Id",
"Course_Name", acd_Sched_Real.Course_Id);
356.     ViewBag.Class_Prog_Id = new SelectList(getClassProg,
"Class_Prog_Id", "Class_Program_Name",
acd_Sched_Real.Class_Prog_Id);
357.     ViewBag.Class_Id = new SelectList(getClass, "Class_Id",
"Class_Name", acd_Sched_Real.Class_Id);
358.     ViewBag.Room_Id = new SelectList(db.Mstr_Room, "Room_Id",
"Room_Name", acd_Sched_Real.Room_Id);
359.     ViewBag.Employee_Id = new SelectList(User_Id);
360.     return View(acd_Sched_Real);
361. }
362.
363. // POST: Acd_Sched_Real/Edit/5
364. [HttpPost]
365. [ValidateAntiForgeryToken]
366. public ActionResult Edit([Bind(Include =
"Sched_Real_Id,Term_Year_Id,Class_Prog_Id,Course_Id,Class_Id,Meetin
g_Order,Room_Id,Employee_Id,Date,Time_Start,Time_End,Token,Cours
e_Content,Closed_By,Max_Minutes,Description,Created_By,Created_Dat
e,Modified_By,Modified_Date")] Acd_Sched_Real acd_Sched_Real)
367. {
368.     if (ModelState.IsValid)
369.     {
370.         db.Entry(acd_Sched_Real).State = EntityState.Modified;

```

```

371.         db.Entry(acd_Sched_Real).Property(x =>
           x.Class_Prog_Id).IsModified = false;
372.         db.Entry(acd_Sched_Real).Property(x =>
           x.Meeting_Order).IsModified = false;
373.         db.Entry(acd_Sched_Real).Property(x =>
           x.Employee_Id).IsModified = false;
374.         db.Entry(acd_Sched_Real).Property(x => x.Date).IsModified =
           false;
375.         db.Entry(acd_Sched_Real).Property(x =>
           x.Time_Start).IsModified = false;
376.         db.Entry(acd_Sched_Real).Property(x =>
           x.Time_End).IsModified = false;
377.         db.Entry(acd_Sched_Real).Property(x => x.Token).IsModified
           = false;
378.
379.         db.SaveChanges();
380.         return RedirectToAction("Index");
381.     }
382.     ViewBag.Course_Id = new SelectList(db.Acd_Course,
           "Course_Id", "Course_Code", acd_Sched_Real.Course_Id);
383.     ViewBag.Employee_Id = new SelectList(db.Emp_Employee,
           "Employee_Id", "Nik", acd_Sched_Real.Employee_Id);
384.     ViewBag.Class_Id = new SelectList(db.Mstr_Class, "Class_Id",
           "Class_Name", acd_Sched_Real.Class_Id);
385.     ViewBag.Class_Prog_Id = new
           SelectList(db.Mstr_Class_Program, "Class_Prog_Id", "Class_Prog_Code",
           acd_Sched_Real.Class_Prog_Id);
386.     ViewBag.Room_Id = new SelectList(db.Mstr_Room, "Room_Id",
           "Room_Code", acd_Sched_Real.Room_Id);
387.     ViewBag.Term_Year_Id = new SelectList(db.Mstr_Term_Year,
           "Term_Year_Id", "Term_Year_Name", acd_Sched_Real.Term_Year_Id);
388.     return View(acd_Sched_Real);
389. }
390.
391. // GET: Acd_Sched_Real/Delete/5
392. public ActionResult Delete(int? id)
393. {
394.     if (id == null)
395.     {
396.         return new
           HttpStatusCodeResult(HttpStatusCode.BadRequest);
397.     }
398.     Acd_Sched_Real acd_Sched_Real =
           db.Acd_Sched_Real.Find(id);
399.     if (acd_Sched_Real == null)
400.     {

```

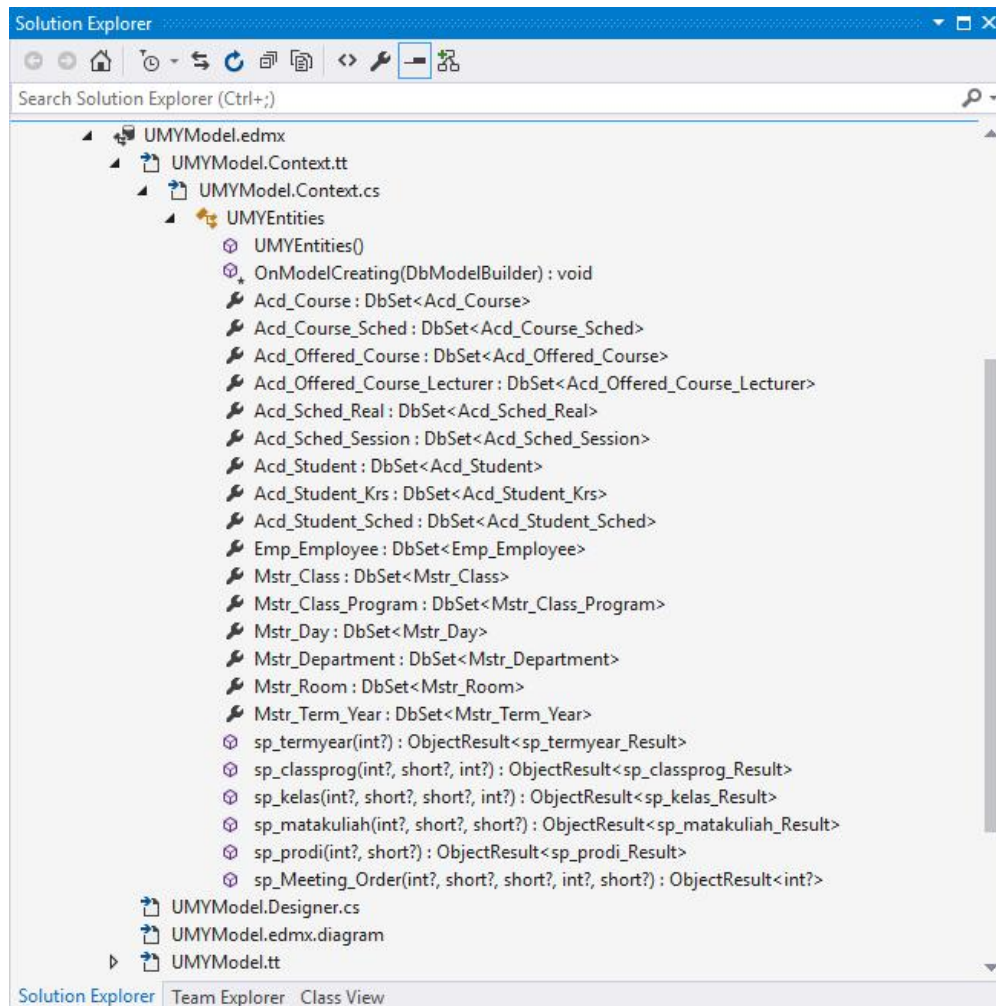
```

401.         return HttpNotFound();
402.     }
403.     return View(acd_Sched_Real);
404. }
405.
406. // POST: Acd_Sched_Real/Delete/5
407. [HttpPost, ActionName("Delete")]
408. [ValidateAntiForgeryToken]
409. public ActionResult DeleteConfirmed(int id)
410. {
411.     Acd_Sched_Real acd_Sched_Real =
    db.Acd_Sched_Real.Find(id);
412.     db.Acd_Sched_Real.Remove(acd_Sched_Real);
413.     db.SaveChanges();
414.     return RedirectToAction("Index");
415. }
416.
417. protected override void Dispose(bool disposing)
418. {
419.     if (disposing)
420.     {
421.         db.Dispose();
422.     }
423.     base.Dispose(disposing);
424. }
425. }
426. }

```

B. *Source code pada model*

Aplikasi presensi *online* ini mempunyai 16 tabel dan 6 store procedure dalam *database*. Daftar tabel dapat dilihat pada Screenshot 1.



Screenshot 1. Daftar table dan store procedure dalam database.

C. Source code (html) pada view

Aplikasi presensi *online* ini mempunyai lima halaman pengguna, yaitu *index*, *create*, *edit*, *detail*, *delete* dan *login*. Berikut source code yang ada pada halaman-halaman tersebut:

1. Halaman *index*

```

1 @model PagedList.IPagedList<StudentPresence.Models.Acd_Sched_Real>
2 @using PagedList.Mvc;
3 <link href="~/Content/PagedList.css" rel="stylesheet" type="text/css"
  />

```

```

4
5  @{
6     ViewBag.Title = "Session";
7  }
8
9  <h3>Daftar Sesi Perkuliahan</h3>
10
11 <p>
12     @Html.ActionLink("Buat sesi baru", "Create")
13 </p>
14 @using (Html.BeginForm("Index", "Acd_Sched_Real", FormMethod.Get))
15 {
16     <p>
17         Cari Mata Kuliah: @Html.TextBox("SearchString",
18         ViewBag.CurrentFilter as string)
19         <input type="submit" value="Search" /></p>
20 }
21 <table class="table">
22     <tr align="center">
23         <th align="center">
24             @Html.ActionLink("Tanggal", "Index", new { sortOrder =
25             ViewBag.DateSortParm, currentFilter = ViewBag.CurrentFilter })
26         </th>
27         <th align="center">
28             Tahun Ajaran
29         </th>
30         <th align="center">
31             Kelas Program
32         </th>
33         <th align="center">
34             Mata Kuliah
35         </th>
36         <th align="center">
37             Kelas
38         </th>
39         <th align="center">
40             Pertemuan Ke-
41         </th>
42         <th align="center">
43             Ruang
44         </th>
45         <th align="center">
46             Waktu Mulai
47         </th>
48         <th align="center">
49             Waktu Selesai
50         </th>
51         <th align="center">
52             Token
53         </th>
54         <th align="center">
55             Maks. Keterlambatan
56         </th>
57     </tr>
58 @foreach (var item in Model) {

```

```

59     <tr>
60         <td>
61             @Html.DisplayFor(modelItem => item.Date)
62         </td>
63         <td>
64             @Html.DisplayFor(modelItem =>
item.Mstr_Term_Year.Term_Year_Name)
65         </td>
66         <td>
67             @Html.DisplayFor(modelItem =>
item.Mstr_Class_Program.Class_Program_Name)
68         </td>
69         <td>
70             @Html.DisplayFor(modelItem => item.Acd_Course.Course_Name)
71         </td>
72         <td>
73             @Html.DisplayFor(modelItem => item.Mstr_Class.Class_Name)
74         </td>
75         <td>
76             @Html.DisplayFor(modelItem => item.Meeting_Order)
77         </td>
78         <td>
79             @Html.DisplayFor(modelItem => item.Mstr_Room.Room_Name)
80         </td>
81         <td>
82             @Html.DisplayFor(modelItem => item.Time_Start)
83         </td>
84         <td>
85             @Html.DisplayFor(modelItem => item.Time_End)
86         </td>
87         <td>
88             @Html.DisplayFor(modelItem => item.Token)
89         </td>
90         <td>
91             @Html.DisplayFor(modelItem => item.Max_Minutes) <text>
menit</text>
92         </td>
93         <td>
94             @Html.ActionLink("Edit", "Edit", new {
id=item.Sched_Real_Id })
95             @Html.ActionLink("Details", "Details", new {
id=item.Sched_Real_Id })
96             @Html.ActionLink("Delete", "Delete", new {
id=item.Sched_Real_Id })
97         </td>
98     </tr>
99 }
100
101 </table>
102 <br />
103 Halaman @(Model.PageCount < Model.PageNumber ? 0 : Model.PageNumber)
dari @Model.PageCount
104
105 @Html.PagedListPager(Model, page => Url.Action("Index",

```

```

106     new { page, sortOrder = ViewBag.CurrentSort, currentFilter =
        ViewBag.CurrentFilter })))

```

2. Halaman create

```

1  @model StudentPresence.Models.Acd_Sched_Real
2
3  @{
4      ViewBag.Title = "Create";
5  }
6
7  <h2>Create</h2>
8
9  @if (ViewBag.error == null)
10 {
11
12
13 using (Html.BeginForm())
14 {
15     @Html.AntiForgeryToken()
16
17     <h4>Sesi Perkuliahan</h4>
18     @Html.ValidationSummary(true, "", new { @class = "text-danger" })
19     <table class="table">
20         <tr>
21             <td>
22                 <div class="form-horizontal">
23                     <div id="TahunAjaran" class="form-group">
24                         @Html.LabelFor(model => model.Term_Year_Id,
25 "Tahun Ajaran", htmlAttributes: new { @class = "control-label col-md-
26 2" })
27                         <div class="col-md-10">
28                             @Html.DropDownList("Term_Year_Id", null,
29 htmlAttributes: new { @class = "form-control disabled = disabled" })
30                             @Html.ValidationMessageFor(model =>
31 model.Term_Year_Id, "", new { @class = "text-danger" })
32                         </div>
33                     </div>
34                 </td>
35             </tr>
36             <tr>
37                 <td>
38                     <div id="ProgramStudi" class="form-group">
39                         @Html.Label("Program Studi", htmlAttributes:
40 new { @class = "control-label col-md-2" })
41                         <div class="col-md-10">
42                             @Html.DropDownList("Department_Id", null,
43 htmlAttributes: new { @class = "form-control" })
44                             @Html.ValidationMessageFor(model =>
45 model.Class_Prog_Id, "", new { @class = "text-danger" })
46                         </div>
47                     </div>
48                 </td>
49             </tr>
50             <tr>
51                 <td>
52                     <div id="MataKuliah" class="form-group">

```



```

41         @Html.LabelFor(model => model.Course_Id,
    "Mata Kuliah", htmlAttributes: new { @class = "control-label col-md-
    2" })
42         <div class="col-md-10">
43             @Html.DropDownList("Course_Id", null,
    htmlAttributes: new { @class = "form-control" })
44             @Html.ValidationMessageFor(model =>
    model.Course_Id, "", new { @class = "text-danger" })
45         </div>
46     </div>
47
48     <div id="Kelas" class="form-group">
49         @Html.LabelFor(model => model.Class_Id,
    "Kelas", htmlAttributes: new { @class = "control-label col-md-2" })
50         <div class="col-md-10">
51             @Html.DropDownList("Class_Id", null,
    htmlAttributes: new { @class = "form-control" })
52             @Html.ValidationMessageFor(model =>
    model.Class_Id, "", new { @class = "text-danger" })
53         </div>
54     </div>
55     <div class="form-group">
56         <div class="col-md-offset-2 col-md-10">
57             <input id="btn-create" type="submit"
    value="Create" class="btn btn-default" />
58         </div>
59     </div>
60 </td>
61 <td>
62     <div class="form-horizontal">
63         <div id="Ruangan" class="form-group">
64             @Html.LabelFor(model => model.Room_Id,
    "Ruangan", htmlAttributes: new { @class = "control-label col-md-2" })
65             <div class="col-md-10">
66                 @Html.DropDownList("Room_Id", null,
    htmlAttributes: new { @class = "form-control" })
67                 @Html.ValidationMessageFor(model =>
    model.Room_Id, "", new { @class = "text-danger" })
68             </div>
69         </div>
70     </div>
71
72     <div class="form-group">
73         @Html.LabelFor(model => model.Max_Minutes,
    "Maks. Keterlambatan", htmlAttributes: new { @class = "control-label
    col-md-2" })
74         <div class="col-md-10">
75             @Html.EditorFor(model =>
    model.Max_Minutes, new { htmlAttributes = new { @class = "form-
    control" } })
76             @Html.ValidationMessageFor(model =>
    model.Max_Minutes, "", new { @class = "text-danger" })
77         </div>
78     </div>
79
80     <div class="form-group">

```

```

81         @Html.LabelFor(model => model.Course_Content,
"Isi Perkuliahan", htmlAttributes: new { @class = "control-label col-
md-2" })
82         <div class="col-md-10">
83             @Html.EditorFor(model =>
model.Course_Content, new { htmlAttributes = new { @class = "form-
control" } })
84             @Html.ValidationMessageFor(model =>
model.Course_Content, "", new { @class = "text-danger" })
85         </div>
86     </div>
87
88     <div class="form-group">
89         @Html.LabelFor(model => model.Description,
"Deskripsi", htmlAttributes: new { @class = "control-label col-md-2"
})
90         <div class="col-md-10">
91             @Html.EditorFor(model =>
model.Description, new { htmlAttributes = new { @class = "form-
control" } })
92             @Html.ValidationMessageFor(model =>
model.Description, "", new { @class = "text-danger" })
93         </div>
94     </div>
95 </div>
96 </td>
97 </tr>
98 </table>
99 <div>
100     @Html.ActionLink("Lihat semua sesi", "Index")
101 </div>
102 // FOR HIDE KONTEN !!
103 <div class="form-horizontal">
104     <div id="KelasProgram" class="form-group">
105         @Html.LabelFor(model => model.Class_Prog_Id, "Kelas
Program", htmlAttributes: new { style = "display:none;" })
106         <div class="col-md-10">
107             @Html.DropDownList("Class_Prog_Id", null,
htmlAttributes: new { style = "display:none;" })
108             @Html.ValidationMessageFor(model =>
model.Class_Prog_Id, "", new { style = "display:none;" })
109         </div>
110     </div>
111     <div id="PertemuanKe" class="form-group">
112         @Html.LabelFor(model => model.Meeting_Order, "Pertemuan
Ke", htmlAttributes: new { style = "display:none;" })
113         <div class="col-md-10">
114             @Html.EditorFor(model => model.Meeting_Order, new {
htmlAttributes = new { style = "display:none;" } })
115             @Html.ValidationMessageFor(model =>
model.Meeting_Order, "", new { style = "display:none;" })
116         </div>
117     </div>
118     <div class="form-group">
119         @Html.LabelFor(model => model.Employee_Id, "Employee_Id",
htmlAttributes: new { style = "display:none;" })
120         <div class="col-md-10">

```

```

121         @Html.DropDownList("Employee_Id", null,
htmlAttributes: new { style = "display:none;" })
122         @Html.ValidationMessageFor(model =>
model.Employee_Id, "", new { style = "display:none;" })
123     </div>
124 </div>
125 <div class="form-group">
126     @Html.LabelFor(model => model.Date, htmlAttributes: new {
style = "display:none;" })
127     <div class="col-md-10">
128         @Html.EditorFor(model => model.Date, new {
htmlAttributes = new { style = "display:none;" } })
129         @Html.ValidationMessageFor(model => model.Date, "",
new { style = "display:none;" })
130     </div>
131 </div>
132 <div class="form-group">
133     @Html.LabelFor(model => model.Time_Start, htmlAttributes:
new { style = "display:none;" })
134     <div class="col-md-10">
135         @Html.EditorFor(model => model.Time_Start, new {
htmlAttributes = new { style = "display:none;" } })
136         @Html.ValidationMessageFor(model => model.Time_Start,
"", new { style = "display:none;" })
137     </div>
138 </div>
139 <div class="form-group">
140     @Html.LabelFor(model => model.Time_End, htmlAttributes:
new { style = "display:none;" })
141     <div class="col-md-10">
142         @Html.EditorFor(model => model.Time_End, new {
htmlAttributes = new { style = "display:none;" } })
143         @Html.ValidationMessageFor(model => model.Time_End,
"", new { style = "display:none;" })
144     </div>
145 </div>
146 <div class="form-group">
147     @Html.LabelFor(model => model.Token, htmlAttributes: new
{ style = "display:none;" })
148     <div class="col-md-10">
149         @Html.Editor("Token", new { htmlAttributes = new {
style = "display:none;" } })
150         @Html.ValidationMessageFor(model => model.Token, "",
new { style = "display:none;" })
151     </div>
152 </div>
153 </div>
154
155 }
156 }
157 else
158 {
159     <div class="form-group">
160         <div class="col-md-10 alert-danger">
161             @ViewBag.error
162         </div>
163     </div>

```

3. Halaman edit

```

1  @model StudentPresence.Models.Acd_Sched_Real
2
3  @{
4      ViewBag.Title = "Edit";
5  }
6
7  <h2>Edit</h2>
8
9
10 @using (Html.BeginForm())
11 {
12     @Html.AntiForgeryToken()
13
14     <table class="table">
15         <tr>
16             <td>
17                 <div class="form-horizontal">
18                     @Html.HiddenFor(model => model.Sched_Real_Id)
19
20                     <div id="TahunAjaran" class="form-group">
21                         @Html.LabelFor(model => model.Term_Year_Id,
22 "Tahun Ajaran", htmlAttributes: new { @class = "control-label col-md-
23 2" })
24                         <div class="col-md-10">
25                             @Html.DropDownList("Term_Year_Id", null,
26 htmlAttributes: new { @class = "form-control disabled = disabled" })
27                             @Html.ValidationMessageFor(model =>
28 model.Term_Year_Id, "", new { @class = "text-danger" })
29                         </div>
30                     </div>
31
32                     <div id="ProgramStudi" class="form-group">
33                         @Html.Label("Program Studi", htmlAttributes:
34 new { @class = "control-label col-md-2" })
35                         <div class="col-md-10">
36                             @Html.DropDownList("Department_Id", null,
37 htmlAttributes: new { @class = "form-control" })
38                             @Html.ValidationMessageFor(model =>
39 model.Class_Prog_Id, "", new { @class = "text-danger" })
40                         </div>
41                     </div>
42
43                     <div id="MataKuliah" class="form-group">
44                         @Html.LabelFor(model => model.Course_Id,
45 "Mata Kuliah", htmlAttributes: new { @class = "control-label col-md-
46 2" })
47                         <div class="col-md-10">
48                             @Html.DropDownList("Course_Id", null,
49 htmlAttributes: new { @class = "form-control" })
50                             @Html.ValidationMessageFor(model =>
51 model.Course_Id, "", new { @class = "text-danger" })

```

```

41         </div>
42     </div>
43
44     <div id="Kelas" class="form-group">
45         @Html.LabelFor(model => model.Class_Id,
46 "Kelas", htmlAttributes: new { @class = "control-label col-md-2" })
47         <div class="col-md-10">
48             @Html.DropDownList("Class_Id", null,
49 htmlAttributes: new { @class = "form-control" })
50             @Html.ValidationMessageFor(model =>
51 model.Class_Id, "", new { @class = "text-danger" })
52         </div>
53     </div>
54     <div class="form-group">
55         <div class="col-md-offset-2 col-md-10">
56             <input type="submit" value="Save"
57 class="btn btn-default" />
58         </div>
59     </div>
60 </div>
61 </td>
62 <td>
63     <div class="form-horizontal">
64     <div id="Ruangan" class="form-group">
65         @Html.LabelFor(model => model.Room_Id,
66 "Ruangan", htmlAttributes: new { @class = "control-label col-md-2" })
67         <div class="col-md-10">
68             @Html.DropDownList("Room_Id", null,
69 htmlAttributes: new { @class = "form-control" })
70             @Html.ValidationMessageFor(model =>
71 model.Room_Id, "", new { @class = "text-danger" })
72         </div>
73     </div>
74     <div class="form-group">
75         @Html.LabelFor(model => model.Max_Minutes,
76 "Maks. Keterlambatan", htmlAttributes: new { @class = "control-label
77 col-md-2" })
78         <div class="col-md-10">
79             @Html.EditorFor(model =>
80 model.Max_Minutes, new { htmlAttributes = new { @class = "form-
81 control" } })
82             @Html.ValidationMessageFor(model =>
83 model.Max_Minutes, "", new { @class = "text-danger" })
84         </div>
85     </div>
86     <div class="form-group">
87         @Html.LabelFor(model => model.Course_Content,
88 "Isi Perkuliahan", htmlAttributes: new { @class = "control-label col-
89 md-2" })
90         <div class="col-md-10">
91             @Html.EditorFor(model =>
92 model.Course_Content, new { htmlAttributes = new { @class = "form-
93 control" } })
94             @Html.ValidationMessageFor(model =>
95 model.Course_Content, "", new { @class = "text-danger" })

```

```

81         </div>
82     </div>
83
84     <div class="form-group">
85         @Html.LabelFor(model => model.Description,
86 "Deskripsi", htmlAttributes: new { @class = "control-label col-md-2"
87 })
88         <div class="col-md-10">
89             @Html.EditorFor(model =>
90 model.Description, new { htmlAttributes = new { @class = "form-
91 control" } })
92             @Html.ValidationMessageFor(model =>
93 model.Description, "", new { @class = "text-danger" })
94         </div>
95     </div>
96 </td>
97 </tr>
98 </table>
99 }
100
101 <div>
102     @Html.ActionLink("Lihat semua sesi", "Index")
103 </div>

```

4. Halaman detail

```

1  @model StudentPresence.Models.Acd_Sched_Real
2
3  @{
4      ViewBag.Title = "Details";
5  }
6
7  <h2>Details</h2>
8
9  <div id="myFade">
10     <h4>Sesi Perkuliahan</h4>
11     <table class="table">
12         <tr>
13             <td>
14                 <dl class="dl-horizontal">
15                     <dt>
16                         Tanggal
17                     </dt>
18
19                     <dd>
20                         @Html.DisplayFor(model => model.Date)
21                     </dd>
22
23                     <dt>
24                         Tahun Ajaran
25                     </dt>
26

```

```

27         <dd>
28             @Html.DisplayFor(model =>
model.Mstr_Term_Year.Term_Year_Name)
29         </dd>
30
31         <dt>
32             Kelas Program
33         </dt>
34
35         <dd>
36             @Html.DisplayFor(model =>
model.Mstr_Class_Program.Class_Program_Name)
37         </dd>
38
39         <dt>
40             Mata Kuliah
41         </dt>
42
43         <dd>
44             @Html.DisplayFor(model =>
model.Acd_Course.Course_Name)
45         </dd>
46
47         <dt>
48             Kelas
49         </dt>
50
51         <dd>
52             @Html.DisplayFor(model =>
model.Mstr_Class.Class_Name)
53         </dd>
54
55         <dt>
56             Ruang
57         </dt>
58
59         <dd>
60             @Html.DisplayFor(model =>
model.Mstr_Room.Room_Name)
61         </dd>
62         <dt>
63             <div id="btnSaveAndStop" class="form-group">
64                 <div class="col-md-offset-2 col-md-10">
65                     <input type="submit" value="Save and
Stop" class="btn btn-default" />
66                 </div>
67             </div>
68         </dt>
69     </dl>
70 </td>
71 <td>
72     <dl class="dl-horizontal">
73         <dt>
74             Pertemuan Ke
75         </dt>
76
77         <dd>

```

```

78         @Html.DisplayFor(model =>
model.Meeting_Order)
79         </dd>
80
81         <dt>
82             Waktu Mulai
83         </dt>
84
85         <dd>
86             @Html.DisplayFor(model => model.Time_Start)
87         </dd>
88
89         <dt>
90             Waktu Selesai
91         </dt>
92
93         <dd>
94             @Html.DisplayFor(model => model.Time_End)
95         </dd>
96
97         <dt>
98             Maks. Keterlambatan
99         </dt>
100
101         <dd>
102             @Html.DisplayFor(model => model.Max_Minutes)
<text>menit</text>
103         </dd>
104
105         <dt>
106             Isi Perkuliahan
107         </dt>
108
109         <dd>
110             @Html.DisplayFor(model =>
model.Course_Content)
111         </dd>
112
113         <dt>
114             Deskripsi
115         </dt>
116
117         <dd>
118             @Html.DisplayFor(model => model.Description)
119         </dd>
120     </dl>
121 </td>
122 <td>
123     <strong>Token</strong>
124     <h2 style="color:green;font-family:'Times New
Roman';" >
125         <b>@Html.DisplayFor(model => model.Token)</b>
126     </h2>
127     <dl align="right">
128         <a class="btn btn-default" data-toggle="modal"
data-target=".bootstrapmodal"><span class="glyphicon glyphicon-eye-
open"></span> View Token </a>

```



```

129         <a style=@ViewBag.visible id="anchorEdit"
class="btn btn-default" data-id="@Model.Sched_Real_Id"><span
class="glyphicon glyphicon-stop"></span> Akhiri Sesi</a>
130
131         </dl>
132     </td>
133 </tr>
134 </table>
135 @{
136     // csharp code block
137     // everything in here is code, don't have to use @
138     int counter = 0;
139 }
140 <!--Baris Daftar Mahasiswa yang enroll-->
141 <div id="view_details" dat-id="@Model.Sched_Real_Id">
142     @Html.Partial("_Details")
143 </div>
144 </div>
145 <p>
146     @Html.ActionLink("Edit sesi", "Edit", new { id =
Model.Sched_Real_Id }) |
147     @Html.ActionLink("Lihat semua sesi", "Index")
148 </p>

```

5. Halaman delete

```

1 @model StudentPresence.Models.Acd_Sched_Real
2
3 @{
4     ViewBag.Title = "Delete";
5 }
6
7 <h3 class="text-danger">Apakah anda yakin untuk menghapus sesi
perkuliah ini?</h3>
8 <div>
9     <table class="table">
10         <tr>
11             <td>
12                 <dl class="dl-horizontal">
13                     <dt>
14                         Tanggal
15                     </dt>
16
17                     <dd>
18                         @Html.DisplayFor(model => model.Date)
19                     </dd>
20
21                     <dt>
22                         Tahun Ajaran
23                     </dt>
24
25                     <dd>

```

```

26         @Html.DisplayFor(model =>
model.Mstr_Term_Year.Term_Year_Name)
27         </dd>
28
29         <dt>
30             Kelas Program
31         </dt>
32
33         <dd>
34             @Html.DisplayFor(model =>
model.Mstr_Class_Program.Class_Program_Name)
35         </dd>
36
37         <dt>
38             Mata Kuliah
39         </dt>
40
41         <dd>
42             @Html.DisplayFor(model =>
model.Acd_Course.Course_Name)
43         </dd>
44
45         <dt>
46             Kelas
47         </dt>
48
49         <dd>
50             @Html.DisplayFor(model =>
model.Mstr_Class.Class_Name)
51         </dd>
52
53         <dt>
54             Ruangan
55         </dt>
56
57         <dd>
58             @Html.DisplayFor(model =>
model.Mstr_Room.Room_Name)
59         </dd>
60     </dl>
61 </td>
62 <td>
63     <dl class="dl-horizontal">
64         <dt>
65             Pertemuan Ke
66         </dt>
67
68         <dd>
69             @Html.DisplayFor(model =>
model.Meeting_Order)
70         </dd>
71
72         <dt>
73             Waktu Mulai
74         </dt>
75
76         <dd>

```

```

77         @Html.DisplayFor(model => model.Time_Start)
78     </dd>
79
80     <dt>
81         Waktu Selesai
82     </dt>
83
84     <dd>
85         @Html.DisplayFor(model => model.Time_End)
86     </dd>
87
88     <dt>
89         Maks. Keterlambatan
90     </dt>
91
92     <dd>
93         @Html.DisplayFor(model => model.Max_Minutes)
94     </dd>
95
96     <dt>
97         Isi Perkuliahan
98     </dt>
99
100    <dd>
101        @Html.DisplayFor(model =>
102        model.Course_Content)
103    </dd>
104
105    <dt>
106        Deskripsi
107    </dt>
108
109    <dd>
110        @Html.DisplayFor(model => model.Description)
111    </dd>
112    <dt></dt>
113    <dd><br/>
114    @using (Html.BeginForm())
115    {
116        @Html.AntiForgeryToken()
117
118        <div class="form-actions no-color ">
119            @Html.ActionLink("Kembali Ke Daftar
120            Sesi", "Index") |
121            <input type="submit" value="Yes"
122            class="btn btn-danger" />
123        </div>
124    }
125    </dd>
126    </dl>
127    </td>
128    </tr>
129    </table>
130 </div>

```

6. Halaman login

```
1 @using StudentPresence.Models
2 @model LoginViewModel
3 @{
4     ViewBag.Title = "Log in";
5 }
6
7 <h2>@ViewBag.Title.</h2>
8 <div class="row">
9     <div class="col-md-8">
10         <section id="loginForm">
11             @using (Html.BeginForm("Login", "Account", new {
12                 returnUrl = ViewBag.ReturnUrl }, FormMethod.Post, new { @class =
13                     "form-horizontal", role = "form" }))
14             {
15                 @Html.AntiForgeryToken()
16                 <h4>Gunakan email dari Universitas untuk log in.</h4>
17                 <hr />
18                 @Html.ValidationSummary(true, "", new { @class =
19                     "text-danger" })
20                 <div class="form-group">
21                     @Html.LabelFor(m => m.Email, new { @class = "col-
22                         md-2 control-label" })
23                     <div class="col-md-10">
24                         @Html.TextBoxFor(m => m.Email, new { @class =
25                             "form-control" })
26                         @Html.ValidationMessageFor(m => m.Email, "",
27                             new { @class = "text-danger" })
28                     </div>
29                 </div>
30                 <div class="form-group">
31                     @Html.LabelFor(m => m.Password, new { @class =
32                         "col-md-2 control-label" })
33                     <div class="col-md-10">
34                         @Html.PasswordFor(m => m.Password, new {
35                             @class = "form-control" })
36                         @Html.ValidationMessageFor(m => m.Password,
37                             "", new { @class = "text-danger" })
38                     </div>
39                 </div>
40                 <div class="form-group">
41                     <div class="col-md-offset-2 col-md-10">
42                         <input type="submit" value="Log in"
43                             class="btn btn-default" />
44                     </div>
45                 </div>
46             }
47         </section>
48     </div>
49     <div class="col-md-4">
50         <section id="socialLoginForm">
51         </section>
52     </div>
```

43 </div>