

Lampiran

Buku_TamuController

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using Sanggar_Anak_Alam_Yogyakarta.Models;
using PagedList;
using System.IO;
using Sanggar_Anak_Alam_Yogyakarta.Laporan;

namespace Sanggar_Anak_Alam_Yogyakarta.Controllers
{
    public class Buku_TamuController : Controller
    {
        private SalamEntities db = new SalamEntities();

        // GET: Buku_Tamu
        public ActionResult Index(string SearchString, string currentFilter, int? page)
        {
            if (SearchString != null)
            {
                page = 1;
            }
            else
            {
                SearchString = currentFilter;
            }

            ViewBag.CurrentFilter = SearchString;

            var buku_tamu = from m in db.Buku_Tamu
                            select m;

            //Search
            if (!String.IsNullOrEmpty(SearchString))
            {
                buku_tamu = buku_tamu.Where(s => s.Tamu_Nama.Contains(SearchString));
            }

            //int pageSize = 3;
            int pageNumber = (page ?? 1);
            //return
            View(buku_tamu.ToPagedList(pageNumber, pageSize));

            return View(buku_tamu.OrderBy(x => x.Tamu_Nama).ToPagedList(page ?? 1, 5));
        }

        public ActionResult LaporanBT(DateTime? fromDate, DateTime? toDate, int? Page)
        {
            var bukutamu = db.Buku_Tamu;
            if (!fromDate.HasValue)
                fromDate = DateTime.Now.Date;
            if (!toDate.HasValue) toDate = fromDate.GetValueOrDefault(DateTime.Now.Date).Date.AddDays(1);
            if (toDate < fromDate) toDate = fromDate.GetValueOrDefault(DateTime.Now.Date).Date.AddDays(1);
            ViewBag.fromDate = fromDate;
            ViewBag.toDate = toDate;
            {
                int page = Page ?? 1;
                var sPPss = db.Buku_Tamu.Where(c => c.Tanggal >= fromDate && c.Tanggal < toDate).ToList();

                return View(sPPss.OrderBy(x => x.Tamu_id).ToPagedList(page, 10));
            }

            public ActionResult Laporan(DateTime? fromDate, DateTime? toDate, int? Page)
            {
                //Laporan SPP();
                if (!fromDate.HasValue)
                    fromDate = DateTime.Now.Date;
                if (!toDate.HasValue) toDate = fromDate.GetValueOrDefault(DateTime.Now.Date).Date.AddDays(1);
                if (toDate < fromDate) toDate = fromDate.GetValueOrDefault(DateTime.Now.Date).Date.AddDays(1);
                ViewBag.fromDate = fromDate;
                ViewBag.toDate = toDate;
                List<Laporan_Tamu> sppList = new List<Laporan_Tamu>();
                var sPPss = db.Buku_Tamu.Where(c => c.Tanggal >= fromDate && c.Tanggal < toDate).ToList();

                foreach (var item in sppss)
                {

```

```

        sppList.Add(new
Laporan_Tamu
        {
            Tamu_id =
            item.Tamu_id,
            Tamu_Nama =
            item.Tamu_Nama,
            Tujuan = item.Tujuan,
            Alamat = item.Alat,
            No_Hp = item.No_Hp,
            Nama_Instansi
            =item.Nama_Instansi,
            Bertemu_Dengan=item.Bertemu_Dengan,
            Keterangan
            =item.Keterangan,
            Tanggal = item.Tanggal
            == null ? DateTime.Now :
            Convert.ToDateTime(item.Tanggal)
        });

        L_BukuTamu rb = new
L_BukuTamu();
        rb.Load();
        rb.SetDataSource(sppList);
        Stream CetakLaporan =
rb.ExportToStream(CrystalDecisions.Shared.
ExportFormatType.PortableDocFormat);

        return File(CetakLaporan,
"application/pdf");
    }
    // GET: Buku_Tamu/Details/5
    public ActionResult Details(int?
id)
    {
        if (id == null)
        {
            return new
HttpStatusCodeResult(HttpStatusCode.BadReq
uest);
        }
        Buku_Tamu buku_Tamu =
db.Buku_Tamu.Find(id);
        if (buku_Tamu == null)
        {
            return HttpNotFound();
        }
        return View(buku_Tamu);
    }

    // GET: Buku_Tamu/Create
    public ActionResult Create()
    {
        return View();
    }

    // POST: Buku_Tamu/Create
    // To protect from overposting
attacks, please enable the specific
properties you want to bind to, for

```

```

        // more details see
http://go.microsoft.com/fwlink/?LinkId=317598.
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult
Create([Bind(Include =
"Tamu_id,Tamu_Nama,Alamat,Tanggal,No_Hp,Na
ma_Instansi,Tujuan,Bertemu_Dengan,Keterang
an")] Buku_Tamu buku_Tamu)
        {
            if (ModelState.IsValid)
            {
                db.Buku_Tamu.Add(buku_Tamu);
                db.SaveChanges();
                return
RedirectToAction("Index");
            }
            return View(buku_Tamu);
        }

        // GET: Buku_Tamu/Edit/5
        public ActionResult Edit(int? id)
        {
            if (id == null)
            {
                return new
HttpStatusCodeResult(HttpStatusCode.BadReq
uest);
            }
            Buku_Tamu buku_Tamu =
db.Buku_Tamu.Find(id);
            if (buku_Tamu == null)
            {
                return HttpNotFound();
            }
            return View(buku_Tamu);
        }

        // POST: Buku_Tamu/Edit/5
        // To protect from overposting
attacks, please enable the specific
properties you want to bind to, for
        // more details see
http://go.microsoft.com/fwlink/?LinkId=317598.
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult
Edit([Bind(Include =
"Tamu_id,Tamu_Nama,Alamat,Tanggal,No_Hp,Na
ma_Instansi,Tujuan,Bertemu_Dengan,Keterang
an")] Buku_Tamu buku_Tamu)
        {
            if (ModelState.IsValid)
            {
                db.Entry(buku_Tamu).State
= EntityState.Modified;
                db.SaveChanges();
                return
RedirectToAction("Index");
            }
            return View(buku_Tamu);
        }
    }

```

```

// GET: Buku_Tamu/Delete/5
public ActionResult Delete(int?
id)
{
    if (id == null)
    {
        return new
 HttpStatusCodeResult(HttpStatusCode.BadReq
uest);
    }
    Buku_Tamu buku_Tamu =
db.Buku_Tamu.Find(id);
    if (buku_Tamu == null)
    {
        return HttpNotFound();
    }
    return View(buku_Tamu);
}

// POST: Buku_Tamu/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]

```

FasilitatorController

```

using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using Sanggar_Anak_Alam_Yogyakarta.Models;
using System.IO;
using PagedList;
using PagedList.Mvc;

namespace Sanggar_Anak_Alam_Yogyakarta.Controller
s
{
    public class FasilitatorsController
    : Controller
    {
        private SalamEntities db = new
SalamEntities();

        // GET: Fasilitators
        public ActionResult
Index(string SearchString, int? Page,
string Status_Kawin_Fk, string
Agama_FK)
        {
            int page = Page ?? 1;

            ViewBag.Status_Kawin_Fk =
new SelectList(db.Status, "Status_Id",
"Status1");

```

```

        public ActionResult
DeleteConfirmed(int id)
        {
            Buku_Tamu buku_Tamu =
db.Buku_Tamu.Find(id);

            db.Buku_Tamu.Remove(buku_Tamu);
            db.SaveChanges();
            return
RedirectToAction("Index");
        }

        protected override void
Dispose(bool disposing)
        {
            if (disposing)
            {
                db.Dispose();
            }
            base.Dispose(disposing);
        }
    }
}

        ViewBag.Agama_FK = new
SelectList(db.Agama, "Agama_id",
"Agama1");
        var fasilitator =
db.Fasilitator.Include(f =>
f.Agama).Include(f =>
f.Jabatan).Include(f =>
f.Jenis_Kelamin).Include(f =>
f.Status);

        //Search
        if
(!String.IsNullOrEmpty(SearchString))
        {
            fasilitator =
fasilitator.Where(s =>
s.Fasilitator_Nama.Contains(SearchStrin
g));
        }

        //Filter
        if
(!string.IsNullOrEmpty(Agama_FK))
        {
            int sa =
Convert.ToInt32(Agama_FK);
            List<Fasilitator> plist
= db.Fasilitator.Where(x => x.Agama_Fk
== sa).ToList();
            return
View(plist.OrderBy(x =>
x.Fasilitator_Id).ToPagedList(page,
5));
        }
        if
(!string.IsNullOrEmpty(Status_Kawin_Fk)
)
        {

```

```

        int sa =
Convert.ToInt32(Status_Kawin_Fk);
        List<Fasilitator> plist
= db.Fasilitator.Where(x =>
x.Status_Kawin_Fk == sa).ToList();
        return
View(plist.OrderBy(x =>
x.Fasilitator_Id).ToPagedList(page,
5));
    }
    if
(!string.IsNullOrEmpty(Agama_Fk) &&
(!string.IsNullOrEmpty(Status_Kawin_Fk)
))
    {
        int sa =
Convert.ToInt32(Agama_Fk);
        int s =
Convert.ToInt32(Status_Kawin_Fk);
        List<Fasilitator> plist
= db.Fasilitator.Where(x => x.Agama_Fk
== s).ToList();
        return
View(plist.OrderBy(x =>
x.Fasilitator_Id).ToPagedList(page,
5));
    }
    else
        return
View(fasilitator.OrderBy(x =>
x.Fasilitator_Id).ToPagedList(page,
5));/* OrderBy(x =>
x.Barang_id).ToPagedList(page, 5)*/
    }
    // return
View(fasilitator.ToList());
    //}

// GET: Fasilitators/Details/5
public ActionResult
Details(int? id)
{
    if (id == null)
    {
        return new
 HttpStatusCodeResult(HttpStatusCode.Bad
Request);
    }
    Fasilitator fasilitator =
db.Fasilitator.Find(id);
    if (fasilitator == null)
    {
        return HttpNotFound();
    }
    return View(fasilitator);
}

// GET: Fasilitators/Create
public ActionResult Create()
{

```

```

        ViewBag.Agama_Fk = new
SelectList(db.Agama, "Agama_id",
"Agama1");
        ViewBag.Jabatan_Fk = new
SelectList(db.Jabatan, "Jabatan_Id",
"Jabatan1");
        ViewBag.Jenis_Kelamin_Fk =
new SelectList(db.Jenis_Kelamin,
"Jenis_Kelamin_Id", "Jenis_Kelamin1");
        ViewBag.Status_Kawin_Fk =
new SelectList(db.Status, "Status_id",
"Status1");
        return View();
    }

// POST: Fasilitators/Create
// To protect from overposting
attacks, please enable the specific
properties you want to bind to, for
// more details see
http://go.microsoft.com/fwlink/?LinkId=317598.
    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult
Create([Bind(Include =
"Fasilitator_Id,Fasilitator_Nama,NIK,Te
mpat_lahir,Tanggal_Lahir,Jenis_Kelamin
_Fk,Agama_Fk,Status_Kawin_Fk,Ijazah_Tera
khir,Jabatan_Fk,No_Hp,Alamat,Foto")])
Fasilitator fasilitator,
HttpPostedFileBase Foto1)
    {
        if (ModelState.IsValid)
        {
            /*CODING UNTUK
MENAMBAHKAN FOTO */
            var db = new
SalamEntities();
            if (Foto1 != null)
            {
                fasilitator.Foto =
new byte[Foto1.ContentLength];
                Foto1.InputStream.Read(fasilitator.Foto
, 0, Foto1.ContentLength);
            }
            db.Fasilitator.Add(fasilitator);
            db.SaveChanges();
            return
RedirectToAction("Index");
        }

        ViewBag.Agama_Fk = new
SelectList(db.Agama, "Agama_id",
"Agama1", fasilitator.Agama_Fk);
        ViewBag.Jabatan_Fk = new
SelectList(db.Jabatan, "Jabatan_Id",
"Jabatan1", fasilitator.Jabatan_Fk);
        ViewBag.Jenis_Kelamin_Fk =
new SelectList(db.Jenis_Kelamin,

```

```

"Jenis_Kelamin_Id", "Jenis_Kelamin1",
fasilitator.Jenis_Kelamin_Fk);
    ViewBag.Status_Kawin_Fk =
new SelectList(db.Status, "Status_id",
"Status1",
fasilitator.Status_Kawin_Fk);
    return View(fasilitator);
}

// GET: Fasilitators/Edit/5
public ActionResult Edit(int?
id)
{
    if (id == null)
    {
        return new
 HttpStatusCodeResult(HttpStatusCode.Bad
Request);
    }
    Fasilitator fasilitator =
db.Fasilitator.Find(id);
    if (fasilitator == null)
    {
        return HttpNotFound();
    }
    ViewBag.Agama_Fk = new
SelectList(db.Agama, "Agama_id",
"Agama1", fasilitator.Agama_Fk);
    ViewBag.Jabatan_Fk = new
SelectList(db.Jabatan, "Jabatan_Id",
"Jabatan1", fasilitator.Jabatan_Fk);
    ViewBag.Jenis_Kelamin_Fk =
new SelectList(db.Jenis_Kelamin,
"Jenis_Kelamin_Id", "Jenis_Kelamin1",
fasilitator.Jenis_Kelamin_Fk);
    ViewBag.Status_Kawin_Fk =
new SelectList(db.Status, "Status_id",
"Status1",
fasilitator.Status_Kawin_Fk);
    return View(fasilitator);
}

// POST: Fasilitators/Edit/5
// To protect from overposting
attacks, please enable the specific
properties you want to bind to, for
// more details see
http://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult
Edit([Bind(Include =
"Fasilitator_Id,Fasilitator_Nama,NIK,Tempat_lahir,Tanggal_Lahir,Jenis_Kelamin_Fk,Agama_Fk,Status_Kawin_Fk,Ijazah_Terakhir,Jabatan_Fk,No_Hp,Alamat,Foto")])
Fasilitator fasilitator,
HttpPostedFileBase Foto1)
{
    if (ModelState.IsValid)
    {
        if (Foto1 != null)
        {
            fasilitator.Foto =
new byte[Foto1.ContentLength];
            Foto1.InputStream.Read(fasilitator.Foto
, 0, Foto1.ContentLength);

            db.Entry(fasilitator).State =
EntityType.Modified;
        }
        else
        {
            db.Fasilitator.Attach(fasilitator);

            db.Entry(fasilitator).Property(x =>
x.Fasilitator_Nama).IsModified = true;

            db.Entry(fasilitator).Property(x =>
x.NIK).IsModified = true;

            db.Entry(fasilitator).Property(x =>
x.Tempat_lahir).IsModified = true;

            db.Entry(fasilitator).Property(x =>
x.Tanggal_Lahir).IsModified = true;

            db.Entry(fasilitator).Property(x =>
x.Jenis_Kelamin_Fk).IsModified = true;

            db.Entry(fasilitator).Property(x =>
x.Agama_Fk).IsModified = true;

            db.Entry(fasilitator).Property(x =>
x.Status_Kawin_Fk).IsModified = true;

            db.Entry(fasilitator).Property(x =>
x.Ijazah_Terakhir).IsModified = true;

            db.Entry(fasilitator).Property(x =>
x.Jabatan_Fk).IsModified = true;

            db.Entry(fasilitator).Property(x =>
x.No_Hp).IsModified = true;

            db.Entry(fasilitator).Property(x =>
x.Alat).IsModified = true;

            //db.Entry(fasilitator).State =
EntityType.Modified;
        }

        db.SaveChanges();
        return
RedirectToAction("Index");
    }
    ViewBag.Agama_Fk = new
SelectList(db.Agama, "Agama_id",
"Agama1", fasilitator.Agama_Fk);

```

```

        ViewBag.Jabatan_Fk = new
SelectList(db.Jabatan, "Jabatan_Id",
"Jabatan1", fasilitator.Jabatan_Fk);
        ViewBag.Jenis_Kelamin_Fk =
new SelectList(db.Jenis_Kelamin,
"Jenis_Kelamin_Id", "Jenis_Kelamin1",
fasilitator.Jenis_Kelamin_Fk);
        ViewBag.Status_Kawin_Fk =
new SelectList(db.Status, "Status_id",
"Status1",
fasilitator.Status_Kawin_Fk);
        return View(fasilitator);
    }

    // GET: Fasilitators/Delete/5
public ActionResult Delete(int?
id)
    {
        if (id == null)
        {
            return new
 HttpStatusCodeResult(HttpStatusCode.Bad
Request);
        }
        Fasilitator fasilitator =
db.Fasilitator.Find(id);
        if (fasilitator == null)
        {
            return HttpNotFound();
        }
        return View(fasilitator);
    }

    // POST: Fasilitators/Delete/5
[HttpPost,
ActionName("Delete")]
[ValidateAntiForgeryToken]
public ActionResult
DeleteConfirmed(int id)
    {
        Fasilitator fasilitator =
db.Fasilitator.Find(id);

db.Fasilitator.Remove(fasilitator);
        db.SaveChanges();
        return
RedirectToAction("Index");
    }

    protected override void
Dispose(bool disposing)
    {
        if (disposing)
        {
            db.Dispose();
        }
        base.Dispose(disposing);
    }
}

```

Jenis Transaksi Controller

```

using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using
Sanggar_Anak_Alam_Yogyakarta.Models;
using System.IO;
using PagedList;
using
CrystalDecisions.CrystalReports.Engine;

namespace
Sanggar_Anak_Alam_Yogyakarta.Controller
s
{
    public class
Transaksi_DonasiController : Controller
    {
        private SalamEntities db = new
SalamEntities();

        // GET: Transaksi_Donasi
public ActionResult
Index(string searchString, string
currentFilter, int? page)
        {
            if (searchString != null)
            {
                page = 1;
            }
            else
            {
                searchString =
currentFilter;
            }

            ViewBag.CurrentFilter =
searchString;
            ViewBag.Transaksi_Jenis_Fk
= new SelectList(db.Jenis_Transaksi,
"Jenis_Transaksi_Id",
"Jenis_Transaksi_Nama");
            long uangpemasukan = 0;
            long uangpengeluaran = 0;
            var donatur =
db.Transaksi_Donasi.ToList();

            foreach (var total in
donatur)
            {
                if
(total.Transaksi_Jenis_Fk == 1)
                {
                    uangpemasukan +=
total.Transaksi_Jumlah;
                }
            }
        }
    }
}

```

```

    }
    else
    {
        uangpengeluaran +=
total.Transaksi_Jumlah;
    }
}
long uangsaldo =
uangpemasukan - uangpengeluaran;
ViewBag.Pemasukan =
uangpemasukan.ToString("c");
ViewBag.Pengeluaran =
uangpengeluaran.ToString("c");
ViewBag.Saldo =
uangsaldo.ToString("c");

int pageNumber = (page ??
1);
return
View(donatur.OrderBy(x =>
x.Jenis_Transaksi.Jenis_Transaksi_Nama)
.ToPagedList(page ?? 1, 5));

}
public ActionResult
printbyid(int? Transaksi_Jenis_Fk)
{
    var cari_nama_transaksi =
db.Jenis_Transaksi.Where(x=>
x.Jenis_Transaksi_Id ==
Transaksi_Jenis_Fk).FirstOrDefault();
    var nama_transaksi =
cari_nama_transaksi.Jenis_Transaksi_Nam
a;
    ReportDocument rd = new
ReportDocument();

rd.Load(Path.Combine(Server.MapPath("~/
Laporan/Donasibyid.rpt")));

rd.SetDataSource(db.Transaksi_Donasi.Se
lect(c => new VM_L_Donasi()
{
    Transaksi_Id =
c.Transaksi_Id,
    Jenis_Transaksi_Nama =
c.Jenis_Transaksi.Jenis_Transaksi_Nama,
    Transaksi_Tanggal =
c.Transaksi_Tanggal,
    Transaksi_Ket =
c.Transaksi_Ket,
    Transaksi_Jumlah =
c.Transaksi_Jumlah.ToString(),

}).Where(p =>
p.Jenis_Transaksi_Nama ==
nama_transaksi).ToList());
    Response.Buffer = false;
    Response.ClearContent();
    Response.ClearHeaders();
    Stream stream =
rd.ExportToStream

```

```

(CrystalDecisions.Shared.ExportFormatTy
pe.PortableDocFormat);
    return File(stream,
"application/pdf");
}
// GET:
Transaksi_Donasi/Details/5
public ActionResult
Details(int? id)
{
    if (id == null)
    {
        return new
 HttpStatusCodeResult(HttpStatusCode.Bad
Request);
    }
    Transaksi_Donasi
transaksi_Donasi =
db.Transaksi_Donasi.Find(id);
    if (transaksi_Donasi ==
null)
    {
        return HttpNotFound();
    }
    return
View(transaksi_Donasi);
}

// GET: Transaksi_Donasi/Create
public ActionResult Create()
{
    ViewBag.Transaksi_Jenis_Fk
= new SelectList(db.Jenis_Transaksi,
"Jenis_Transaksi_Id",
"Jenis_Transaksi_Nama");
    return View();
}

// POST:
Transaksi_Donasi/Create
// To protect from overposting
attacks, please enable the specific
properties you want to bind to, for
// more details see
http://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult
Create([Bind(Include =
"Transaksi_Id,Transaksi_Jenis_Fk,Transa
ksi_Tanggal,Transaksi_Ket,Transaksi_Jum
lah")] Transaksi_Donasi
transaksi_Donasi)
{
    if (ModelState.IsValid)
    {
        db.Transaksi_Donasi.Add(transaksi_Donas
i);
        db.SaveChanges();
    }
}

```

```

        return
RedirectToAction("Index");
    }

    ViewBag.Transaksi_Jenis_Fk
= new SelectList(db.Jenis_Transaksi,
"Jenis_Transaksi_Id",
"Jenis_Transaksi_Nama",
transaksi_Donasi.Transaksi_Jenis_Fk);
    return
View(transaksi_Donasi);
}

// GET: Transaksi_Donasi/Edit/5
public ActionResult Edit(int?
id)
{
    if (id == null)
    {
        return new
 HttpStatusCodeResult(HttpStatusCode.Bad
Request);
    }
    Transaksi_Donasi
transaksi_Donasi =
db.Transaksi_Donasi.Find(id);
    if (transaksi_Donasi ==
null)
    {
        return HttpNotFound();
    }
    ViewBag.Transaksi_Jenis_Fk
= new SelectList(db.Jenis_Transaksi,
"Jenis_Transaksi_Id",
"Jenis_Transaksi_Nama",
transaksi_Donasi.Transaksi_Jenis_Fk);
    return
View(transaksi_Donasi);
}

// POST:
Transaksi_Donasi/Edit/5
// To protect from overposting
attacks, please enable the specific
properties you want to bind to, for
// more details see
http://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult
Edit([Bind(Include =
"Transaksi_Id,Transaksi_Jenis_Fk,Transa
ksi_Tanggal,Transaksi_Ket,Transaksi_Jum
lah")] Transaksi_Donasi
transaksi_Donasi)
{
    if (ModelState.IsValid)
    {
        db.Entry(transaksi_Donasi).State =
EntityState.Modified;

```

```

        db.SaveChanges();
        return
RedirectToAction("Index");
    }
    ViewBag.Transaksi_Jenis_Fk
= new SelectList(db.Jenis_Transaksi,
"Jenis_Transaksi_Id",
"Jenis_Transaksi_Nama",
transaksi_Donasi.Transaksi_Jenis_Fk);
    return
View(transaksi_Donasi);
}

// GET:
Transaksi_Donasi/Delete/5
public ActionResult Delete(int?
id)
{
    if (id == null)
    {
        return new
 HttpStatusCodeResult(HttpStatusCode.Bad
Request);
    }
    Transaksi_Donasi
transaksi_Donasi =
db.Transaksi_Donasi.Find(id);
    if (transaksi_Donasi ==
null)
    {
        return HttpNotFound();
    }
    return
View(transaksi_Donasi);
}

// POST:
Transaksi_Donasi/Delete/5
[HttpPost,
ActionName("Delete")]
[ValidateAntiForgeryToken]
public ActionResult
DeleteConfirmed(int id)
{
    Transaksi_Donasi
transaksi_Donasi =
db.Transaksi_Donasi.Find(id);

db.Transaksi_Donasi.Remove(transaksi_Do
nasi);
    db.SaveChanges();
    return
RedirectToAction("Index");
}

protected override void
Dispose(bool disposing)
{
    if (disposing)
    {
        db.Dispose();
    }
}

```

```

        base.Dispose(disposing);
    }
}
SPP Controller

using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using Sanggar_Anak_Alam_Yogyakarta.Models;
using System.IO;
using Sanggar_Anak_Alam_Yogyakarta.Laporan;
using PagedList;
using PagedList.Mvc;
using CrystalDecisions.CrystalReports.Engine;

namespace Sanggar_Anak_Alam_Yogyakarta.Controller
{
    public class SPPsController : Controller
    {
        private SalamEntities db = new SalamEntities();

        // GET: SPPs
        public ActionResult Index(string searchString, string currentFilter, int? page)
        {
            if (searchString != null)
            {
                page = 1;
            }
            else
            {
                SearchString = currentFilter;
            }
            ViewBag.CurrentFilter = SearchString;
            var sPP = db.SPP.Include(s => s.Bulan1).Include(s => s.Data_Siswa).Include(s => s.Semester);
            //Search
            if (!String.IsNullOrEmpty(SearchString))
            {
                sPP = sPP.Where(s => s.Data_Siswa>Nama_Lengkap.Contains(SearchString));
            }

```

```

            int total = 0;
            foreach (var item in sPP)
            {
                total += Convert.ToInt32(item.Nominal);
                ViewBag.total = total;
            }
            int pageNumber = (page ?? 1);
            return View(sPP.OrderBy(x => x.SPP_Id).ToPagedList(page ?? 1, 5));
        }
        public ActionResult LaporanSPP(DateTime? fromDate, DateTime? toDate, int? Page)
        {
            var sPP = db.SPP.Include(s => s.Bulan1).Include(s => s.Data_Siswa).Include(s => s.Semester);
            if (!fromDate.HasValue)
                fromDate = DateTime.Now.Date;
            if (!toDate.HasValue)
                toDate = fromDate.GetValueOrDefault(DateTime.Now.Date).Date.AddDays(1);
            if (toDate < fromDate)
                toDate = fromDate.GetValueOrDefault(DateTime.Now.Date).Date.AddDays(1);
            ViewBag.fromDate = fromDate;
            ViewBag.toDate = toDate;
            {
                int page = Page ?? 1;
                var sPPss = db.SPP.Where(c => c.Tanggal >= fromDate && c.Tanggal < toDate).ToList();
                //(s => s.Bulan1).Include(s => s.Data_Siswa).Include(s => s.Semester).Include(s => s.Tanggal);
                int total = 0;
                foreach (var item in sPPss)
                {
                    total += Convert.ToInt32(item.Nominal);
                    ViewBag.total = total;
                }
                return View(sPPss.OrderBy(x => x.SPP_Id).ToPagedList(page, 10));
            }
        }
        public ActionResult Laporan(DateTime? fromDate, DateTime? toDate, int? Page)
        {
            //Laporan SPP();

```

```

        if (!fromDate.HasValue)
fromDate = DateTime.Now.Date;
        if (!toDate.HasValue)
toDate =
fromDate.GetValueOrDefault(DateTime.Now
.Date).Date.AddDays(1);
        if (toDate < fromDate)
toDate =
fromDate.GetValueOrDefault(DateTime.Now
.Date).Date.AddDays(1);
        ViewBag.fromDate =
fromDate;
        ViewBag.toDate = toDate;
        List<Laporan_SPP> sppList =
new List<Laporan_SPP>();
        var sppss = db.SPP.Where(c
=> c.Tanggal >= fromDate && c.Tanggal <
toDate).ToList();

        foreach (var item in sppss)
        {
            sppList.Add(new
Laporan_SPP
            {
                SPP_Id =
item.SPP_Id,
                Nama_Siswa_Fk =
item.Data_Siswa>NamaLengkap,
                Semester_Fk =
item.Semester.Semester1,
                Bulan =
item.Bulan1.Bulan1,
                Nominal =
item.Nominal,
                Tanggal =
item.Tanggal == null ? DateTime.Now :
Convert.ToDateTime(item.Tanggal)
            });
        }

        L_SPP rb = new L_SPP();
        rb.Load();
        rb.SetDataSource(sppList);
        Stream CetakLaporan =
rb.ExportToStream(CrystalDecisions.Shar
ed.ExportFormatType.PortableDocFormat);

        return File(CetakLaporan,
"application/pdf");
    }

    public ActionResult
printbyid(int? id)
    {
        ReportDocument rd = new
ReportDocument();

        rd.Load(Path.Combine(Server.MapPath("~/
Laporan/Sppbyid.rpt")));

```

```

rd.SetDataSource(db.SPP.Select(p => new
    {
        SPP_Id = p.SPP_Id,
        Nama_Siswa_Fk =
p.Data_Siswa>NamaLengkap,
        Semester_Fk =
p.Semester.Semester1,
        Bulan =
p.Bulan1.Bulan1,
        Nominal = p.Nominal,
        Tanggal =
p.Tanggal.Value,
    }).Where(p => p.SPP_Id ==
id).ToList());
        Response.Buffer = false;
        Response.ClearContent();
        Response.ClearHeaders();
        Stream stream =
rd.ExportToStream

(CrystalDecisions.Shared.ExportFormatTy
pe.PortableDocFormat);
        return File(stream,
"application/pdf");
    }

    // GET: SPPs/Details/5
    public ActionResult
Details(int? id)
    {
        if (id == null)
        {
            return new
HttpStatusCodeResult(HttpStatusCode.Bad
Request);
        }
        SPP spp = db.SPP.Find(id);
        if (spp == null)
        {
            return HttpNotFound();
        }
        return View(spp);
    }

    // GET: SPPs/Create
    public ActionResult Create()
    {
        var barang =
db.Data_Siswa.Include(m =>
m>NamaLengkap).Include(m => m.NIS);
        ViewBag.Bulan = new
SelectList(db.Bulan, "Bulan_id",
"Bulan1");
        ViewBag>Nama_Siswa_Fk = new
SelectList(db.Data_Siswa, "Siswa_Id",
>NamaLengkap");
        ViewBag.Semester_Fk = new
SelectList(db.Semester, "Semester_Id",
>Semester1");
        var datasiswa =
db.Data_Siswa.Select(s => new
    {

```

```

        Siswa_id = s.Siswa_Id,
        Nama = s>Nama_Lengkap +
"/" + s.NIS
    }).ToList();
    ViewBag>Nama_Siswa_Fk = new
SelectList(datasiswa, "Siswa_Id",
>Nama");
    return View();
}

// POST: SPPs/Create

[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult
Create([Bind(Include =
"SPP_Id, Nama_Siswa_Fk, Semester_Fk, Bulan
, Tanggal, Nominal")] SPP sPP)
{
    if (ModelState.IsValid)
    {
        db.SPP.Add(sPP);
        db.SaveChanges();
        return
RedirectToAction("Index");
    }

    ViewBag.Bulan = new
SelectList(db.Bulan, "Bulan_id",
>Bulan1", sPP.Bulan);
    ViewBag>Nama_Siswa_Fk = new
SelectList(db>Data_Siswa, "Siswa_Id",
>NIS", sPP>Nama_Siswa_Fk);
    ViewBag.Semester_Fk = new
SelectList(db.Semester, "Semester_Id",
>Semester1", sPP.Semester_Fk);
    return View(sPP);
}

// GET: SPPs/Edit/5
public ActionResult Edit(int?
id)
{
    if (id == null)
    {
        return new
 HttpStatusCodeResult(HttpStatusCode.Bad
Request);
    }
    SPP sPP = db.SPP.Find(id);
    if (sPP == null)
    {
        return HttpNotFound();
    }
    ViewBag.Bulan = new
SelectList(db.Bulan, "Bulan_id",
>Bulan1", sPP.Bulan);
    ViewBag>Nama_Siswa_Fk = new
SelectList(db>Data_Siswa, "Siswa_Id",
>Nama_Lengkap", sPP>Nama_Siswa_Fk);

```

```

        ViewBag.Semester_Fk = new
SelectList(db.Semester, "Semester_Id",
>Semester1", sPP.Semester_Fk);
        var datasiswa =
db.Data_Siswa.Select(s => new
{
            Siswa_id = s.Siswa_Id,
            Nama = s>Nama_Lengkap +
"/" + s.NIS
        }).ToList();
        ViewBag>Nama_Siswa_Fk = new
SelectList(datasiswa, "Siswa_Id",
>Nama");
        return View(sPP);
}

// POST: SPPs/Edit/5

[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult
Edit([Bind(Include =
"SPP_Id, Nama_Siswa_Fk, Semester_Fk, Bulan
, Tanggal, Nominal")] SPP sPP)
{
    if (ModelState.IsValid)
    {
        db.Entry(sPP).State =
 EntityState.Modified;
        db.SaveChanges();
        return
RedirectToAction("Index");
    }
    ViewBag.Bulan = new
SelectList(db.Bulan, "Bulan_id",
>Bulan1", sPP.Bulan);
    ViewBag>Nama_Siswa_Fk = new
SelectList(db>Data_Siswa, "Siswa_Id",
>NIS", sPP>Nama_Siswa_Fk);
    ViewBag.Semester_Fk = new
SelectList(db.Semester, "Semester_Id",
>Semester1", sPP.Semester_Fk);
    return View(sPP);
}

// GET: SPPs/Delete/5
public ActionResult Delete(int?
id)
{
    if (id == null)
    {
        return new
 HttpStatusCodeResult(HttpStatusCode.Bad
Request);
    }
    SPP sPP = db.SPP.Find(id);
    if (sPP == null)
    {
        return HttpNotFound();
    }
    return View(sPP);
}

```

