

## LAMPIRAN

### Source Code Aplikasi

#### 1. Source Code Tampilan Halaman Web

```
@model
PagedList.IPagedList<PerpusMVCWebAPI.Models.CariBuku_Result>
    @using PagedList.Mvc;
<link href="~/Views/bibliografis/Index.cshtml" rel="stylesheet"
type="text/css" />
@{
    ViewBag.Title = "biblio";
<style>
    .button{
        position:relative;
        left:-4px;
        padding:3px 5px;
        border:0px solid #008080;
        background-color: #6B8E23;
        color:#fafafa;
        font-weight:bold;
    }
    button:hover {
        background: #008080;
        color:#fafafa;
    }
    .button:active {
        background:#9ACD32;
    }
    th{
        background:#022;
```

```

        color:#fff;
    }
    .table tr:nth-child(2n+1) {
        background: #aaa;
    }
</style>
}
<br />
<br/>
<h3>Bibliografi</h3>

@using (Html.BeginForm("Index", "bibliografis", FormMethod.Get))
{
    <p>
        @Html.TextBox("SearchString", ViewBag.CurrentFilter as string)
        <input class="button" type="submit" value="Search" />
    </p>
}
<table class="table">
    <tr>
        <th>
            Judul
        </th>
        <th>
            Penulis
        </th>
        <th>
            Penerbit
        </th>
        <th>
            Tempat

```

```
</th>
<th>
    Tahun
</th>
<th>
    Subject
</th>
<th>
    Summary
</th>
<th>
    Score
</th>
</tr>

@foreach (var item in Model) {
<tr>
<td>
    @Html.DisplayFor(modelItem => item.judul)
</td>
<td>
    @Html.DisplayFor(modelItem => item.penulis)
</td>
<td>
    @Html.DisplayFor(modelItem => item.penerbit)
</td>
<td>
    @Html.DisplayFor(modelItem => item.tempat)
</td>
<td>
    @Html.DisplayFor(modelItem => item.tahun)

```

```

</td>
<td>
    @Html.DisplayFor(modelItem => item.subject)
</td>
<td>
    @Html.DisplayFor(modelItem => item.summary)
</td>
<td>
    @Html.DisplayFor(modelItem => item.score)
</td>
</tr>
}
</table>
<br/>
Page @(Model.PageCount <Model.PageNumber ? 0 : Model.PageNumber)
of @Model.PageCount
@Html.PagedListPager(Model, page => Url.Action("Index", new { page,
sortOrder = ViewBag.CurrentSort, CurrentFilter =
ViewBag.CurrentFilter}))

```

## 2. *Source Code* Pencarian Judul di halaman web

```

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 PagedList;
using PerpusMVCWebAPI.Models;

namespace PerpusMVCWebAPI.Controllers
{
    public class bibliografisController : Controller
    {
        private DbPelatihanPerpusEntities3 db = new
        DbPelatihanPerpusEntities3();

        // GET: bibliografis
        public ActionResult Index(string sortOrder, string currentFilter, string
        searchString, int? page)
        {
            ViewBag.CurrentSort = sortOrder;
            ViewBag.judulSortParm = String.IsNullOrEmpty(sortOrder) ?
            "judul_desc" : "";
            ViewBag.penulisSortParm = sortOrder == "penulis" ? "penulis_desc" :
            "penulis";
            List<CariBuku_Result> biblio = new List<CariBuku_Result>();

            {
                if (searchString != null)
                {
                    page = 1;
                }
                else
                {
                    searchString = currentFilter;
                }
                ViewBag.CurrentFilter = searchString;
            }
        }
    }
}

```

```

if (!String.IsNullOrEmpty(searchString))
{
    string S = searchString.ToString();
    string[] words = S.Split(' ');
    foreach (var item in words)
    {
        var listbook = from b in db.CariBuku(item)
            select b;
        biblio.AddRange(listbook);
        //foreach(var detail in listbook)
    }
}
biblio=biblio.OrderBy(k => k.bibliografiid).ToList();
int index = 0;
while (index<biblio.Count()-1)
{
    if (biblio[index].bibliografiid == biblio[index + 1].bibliografiid)
        biblio.RemoveAt(index);
    else
        index++;
}
var biblio_akhir = biblio;//biblio.Distinct().ToList();
switch (sortOrder)
{
    case "judul_desc":
        biblio_akhir = biblio_akhir.OrderByDescending(b =>
b.judul).ToList();
        break;
    case "penulis":
        biblio_akhir = biblio_akhir.OrderBy(b => b.penulis).ToList();

```

```

        break;
        case "penulis_desc":
            biblio_akhir = biblio_akhir.OrderByDescending(b =>
b.penulis).ToList();
            break;
        default:
            biblio_akhir = biblio_akhir.OrderBy(b => b.judul).ToList();
            break;
    }
    int pageSize = 5;
    int pageNumber = (page ?? 1);
    return View(biblio_akhir.ToPagedList(pageNumber, pageSize));
}
}
}

```

### 3. *Source Code* Pencarian Judul Dengan *Output XML*

```

using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Data.Entity.Infrastructure;
using System.Linq;
using System.Net;
using System.Net.Http;
using System.Web.Http;
using System.Web.Http.Description;
using PerpusMVCWebAPI.Models;

namespace PerpusMVCWebAPI.Controllers

```

```

{
    public class CariBukuController : ApiController
    {
        private DbPelatihanPerpusEntities3 db = new
        DbPelatihanPerpusEntities3();

        // GET: api/CariBuku
        public IEnumerable<CariBuku_Result> Get(string searchString)
        {
            List<CariBuku_Result> biblio = new List<CariBuku_Result>();
            {
                if (searchString != null)
                    if (!String.IsNullOrEmpty(searchString))
                    {
                        string S = searchString.ToString();
                        string[] words = S.Split(' ');
                        foreach (var item in words)
                        {
                            var listbook = from b in db.CariBuku(item)
                                select b;
                            biblio.AddRange(listbook);
                        }
                    }
                biblio = biblio.OrderBy(k => k.bibliografiid).ToList();
                int index = 0;
                while (index < biblio.Count() - 1)
                {
                    if (biblio[index].bibliografiid == biblio[index + 1].bibliografiid)
                        biblio.RemoveAt(index);
                    else
                        index++;
                }
            }
        }
    }
}

```



```
    }  
    return biblio.ToList();  
  }  
}
```

#### 4. *Source Code* Pencarian Kata Kunci Dengan *Output XML*

```
using System;  
  
using System.Collections.Generic;  
  
using System.Data;  
  
using System.Data.Entity;  
  
using System.Data.Entity.Infrastructure;  
  
using System.Linq;  
  
using System.Net;  
  
using System.Net.Http;  
  
using System.Web.Http;  
  
using System.Web.Http.Description;  
  
using PerpusMVCWebAPI.Models;  
  
namespace PerpusMVCWebAPI.Controllers  
{  
  
    public class KeyphraseDetailController : ApiController  
    {
```

```

private DbPelatihanPerpusEntities3 db = new
DbPelatihanPerpusEntities3();

// GET: api/KeyphraseDetail/5
[ResponseType(typeof(KeyphraseDetail_Result))]

public IEnumerable<KeyphraseDetail_Result> GetKeyPhrase(int? id)
{
    List<KeyphraseDetail_Result> kPhrase = new
List<KeyphraseDetail_Result>();

    if (id != null)
    {
        var listbook = from b in db.KeyphraseDetail(id)

            select b;

        kPhrase.AddRange(listbook);
    }

    kPhrase = kPhrase.OrderBy(k => k.keyphrase).ToList();

    int index = 0;

    while (index < kPhrase.Count() - 1)
    {
        if (kPhrase[index].keyphrase == kPhrase[index + 1].keyphrase)

            kPhrase.RemoveAt(index);

        else

```

```
        index++;  
  
    }  
  
    return kPhrase.ToList(); }
```

#### 5. *Source Code* Pencarian Dokumen Yang Mirip Dengan *Output XML*

```
using System;  
  
using System.Collections.Generic;  
  
using System.Data;  
  
using System.Data.Entity;  
  
using System.Data.Entity.Infrastructure;  
  
using System.Linq;  
  
using System.Net;  
  
using System.Net.Http;  
  
using System.Web.Http;  
  
using System.Web.Http.Description;  
  
using PerpusMVCWebAPI.Models;  
  
namespace PerpusMVCWebAPI.Controllers  
{  
  
    public class SimilarBukuController : ApiController  
  
    {
```

```

private DbPelatihanPerpusEntities3 db = new
DbPelatihanPerpusEntities3();

// GET: api/SimilarBuku/5

[ResponseType(typeof(SimilarBuku_Result))]

public IEnumerable<SimilarBuku_Result> GetSimilarBuku(int? id)

{

    List<SimilarBuku_Result> sbuku = new List<SimilarBuku_Result>();

    if (id != null)

    {

        var listbook = from b in db.SimilarBuku(id)

            select b;

        sbuku.AddRange(listbook);

    }

    sbuku = sbuku.OrderBy(k => k.bibliografiid).ToList();

    int index = 0;

    while (index < sbuku.Count() - 1)

    {

        if (sbuku[index].bibliografiid == sbuku[index + 1].bibliografiid)

            sbuku.RemoveAt(index);

        else

            index++;

    }

}

```

```
}  
  
return sbuku.ToList(); } } }
```

## 6. *Source Code* Pencarian Judul Dengan *Output* JSON

```
using System;  
using System.Collections.Generic;  
using System.Data;  
using System.Data.Entity;  
using System.Data.Entity.Infrastructure;  
using System.Linq;  
using System.Net;  
using System.Net.Http;  
using System.Web.Http;  
using System.Web.Http.Description;  
using PerpusMVCWebAPI.Models;  
using System.Web.Mvc;  
  
namespace PerpusMVCWebAPI.Controllers  
{  
    public class CariBukuJSONController : Controller  
    {  
        private DbPelatihanPerpusEntities3 db = new  
        DbPelatihanPerpusEntities3();  
  
        // GET: api/CariBukuJSON  
        public ActionResult Get(string searchString)  
        {  
            List<CariBuku_Result> biblio = new List<CariBuku_Result>();  
            {
```

```
        if (searchString != null)
            if (!String.IsNullOrEmpty(searchString))
            {
                string S = searchString.ToString();
                string[] words = S.Split(' ');
                foreach (var item in words)
                {
                    var listbook = from b in db.CariBuku(item)
                                    select b;
                    biblio.AddRange(listbook);
                }
            }
        biblio = biblio.OrderBy(k => k.bibliografiid).ToList();
        int index = 0;
        while (index < biblio.Count() - 1)
        {
            if (biblio[index].bibliografiid == biblio[index + 1].bibliografiid)
                biblio.RemoveAt(index);
            else
                index++;
        }
        return Json(biblio.ToList(), JsonRequestBehavior.AllowGet);
    }
}
}
```

7. *Source Code* Pencarian Kata Kunci Dengan *Output JSON*

```
using System;
using System.Collections.Generic;
```

```

using System.Data;
using System.Data.Entity;
using System.Data.Entity.Infrastructure;
using System.Linq;
using System.Net;
using System.Net.Http;
using System.Web.Http;
using System.Web.Http.Description;
using PerpusMVCWebAPI.Models;
using System.Web.Mvc;

namespace PerpusMVCWebAPI.Controllers
{
    public class KeyphraseDetailJSONController : Controller
    {
        private DbPelatihanPerpusEntities3 db = new
        DbPelatihanPerpusEntities3();

        // GET: api/KeyphraseDetailJSON/5
        [ResponseType(typeof(KeyphraseDetail_Result))]
        public ActionResult GetKeyPhrase(int? id)
        {
            List<KeyphraseDetail_Result> kPhrase = new
            List<KeyphraseDetail_Result>();
            if (id != null)
            {
                var listbook = from b in db.KeyphraseDetail(id)
                select b;
                kPhrase.AddRange(listbook);
            }
            kPhrase = kPhrase.OrderBy(k => k.keyphrase).ToList();
            int index = 0;
            while (index < kPhrase.Count() - 1)
            {
                if (kPhrase[index].keyphrase == kPhrase[index + 1].keyphrase)
                    kPhrase.RemoveAt(index);
                else
                    index++;
            }
            return Json(kPhrase.ToList(), JsonRequestBehavior.AllowGet);
        }
    }
}

```

## 8. *Source Code* Pencarian Dokumen yang mirip Dengan Output JSON

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Data.Entity.Infrastructure;
using System.Linq;
using System.Net;
using System.Net.Http;
using System.Web.Http;
using System.Web.Http.Description;
using PerpusMVCWebAPI.Models;
using System.Web.Mvc;

namespace PerpusMVCWebAPI.Controllers
{
    public class SimilarBukuJSONController : Controller
    {
        private DbPelatihanPerpusEntities3 db = new
        DbPelatihanPerpusEntities3();

        // GET: api/SimilarBukuJSON/5
        [ResponseType(typeof(SimilarBuku_Result))]
        public ActionResult GetSimilarBuku(int? id)
        {
            List<SimilarBuku_Result> sbuku = new List<SimilarBuku_Result>();
            if (id != null)
            {
                var listbook = from b in db.SimilarBuku(id)
                               select b;
                sbuku.AddRange(listbook);
            }
            sbuku = sbuku.OrderBy(k => k.bibliografiid).ToList();
            int index = 0;
            while (index < sbuku.Count() - 1)
            {
                if (sbuku[index].bibliografiid == sbuku[index + 1].bibliografiid)
                    sbuku.RemoveAt(index);
                else
                    index++;
            }
            return Json(sbuku.ToList(), JsonRequestBehavior.AllowGet);
        }
    }
}
```



