

清华大学力学计算与仿真实验室网站

开发成果汇报

指导教师：胡振中

汇报人：王珩玮

实验室网站开发

- 目标
- 功能实现
- 更多？
- 效果展示
- 总结

目标

为清华大学力学计算与仿真实验室制作实验室网站。

功能需求

账户功能

实验室相关
信息展示

信息发布

科研管理

实验室管理

教学管理

功能实现

为清华大学力学计算与仿真实验室制作实验室网站。

功能需求



模块设计

功能实现

技术思路	账户模块	文章模块	实验室管理模块	课程模块
基于ASP.NET，使用MVC架构，采用Razor视图引擎 每项功能分别设计 模型设计 Model Controller 模型设计 数据库 服务器端 前端 View 视图设计				

功能实现

技术思路

账户模块

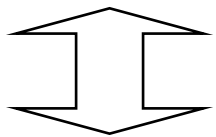
文章模块

实验室管理模块

课程模块

- Model与数据库交互采用System.Data.SqlClient.Sqlconnection/Command

Model



数据库

```
public DataTable ExecuteDataTable(string sql, CommandType commandType, SqlParameter[] parameters)
{
    DataTable data = new DataTable(); //实例化DataTable, 用于装载查询结果集
    using (SqlConnection connection = new SqlConnection(_connectionString))
    {
        using (SqlCommand command = new SqlCommand(sql, connection))
        {
            command.CommandType = commandType; //设置command的CommandType为指定的CommandType
            //如果同时传入了参数, 则添加这些参数
            if (parameters != null)
            {
                foreach (SqlParameter parameter in parameters)
                {
                    command.Parameters.Add(parameter);
                }
            }
            //通过包含查询SQL的SqlCommand实例来实例化SqlDataAdapter
            SqlDataAdapter adapter = new SqlDataAdapter(command);

            adapter.Fill(data); //填充DataTable
        }
    }
    return data;
}
```

```
public static class AppendDBService
{
    public static void deleteRowFromTable(string colname, string valuecol)
    {
        public static void updateValueInTable(string valuecol, string value)
        public static void updateValueInTable(string[] valuecol, string[] value)
        public static void insertRowToTable(string colname, string value)
        public static void insertRowToTable(string[] colnames, string[] values)
        public static DataTable getAllFromTable(string tablename)
        public static DataRowCollection getRowsFromTable(string colname, string value)
        public static DataRow getRowFromTable(string colname, string value)
        public static DataTable getAllFromTable(string[] colnames, string[] values)
        public static DataTable getAllUserInformation()
        public static void DoEditUserInfo(EditUserModel model)
        public static void DoAddUserInfo(AddUserInfoModel model)
    }
}
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

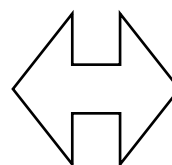
课程模块

- Controller可访问Model命名空间中的Class，从而实现Class的实例化以及对数据库的操作。
- 之后ViewResult中添加Model参数将Model传递给View
- View引用Model对应的Class即可接受Model信息

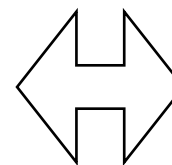
```
public ActionResult AddCourse()
{
    AddCourseModel model = new AddCourseModel();
    return View(model);
}

[HttpPost]
public ActionResult AddCourse(AddCourseModel model)
{
    AppendDBService.AddCourse(model);
    return RedirectToAction("CourseManagement", "Management", null);
}
```

Model



Controller



View

```
@model LaboratorySiteNew.Models.AddCourseModel
```

```
<input type="hidden" id = "experimentids" name = "experimentids" value = @Model.experimentids >
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

• 数据表设计

- 用户信息分为：账户信息、实际信息
- 同时包括三种账户类型：管理员、教师、学生

+ dbo.ACCOUNT_STUDENT
+ dbo.ACCOUNT_TEACHER

+ dbo.aspnet_Applications
+ **dbo.aspnet_Membership**
+ dbo.aspnet_Paths
+ dbo.aspnet_PersonalizationAllUsers
+ dbo.aspnet_PersonalizationPerUser
+ dbo.aspnet_Profile
+ **dbo.aspnet_Roles**
+ dbo.aspnet_SchemaVersions
+ **dbo.aspnet_Users**
+ **dbo.aspnet_UsersInRoles**
+ dbo.aspnet_WebEvent_Events

LINJIARUI-PC\LO....ACCOUNT_STUDENT			
列名	数据类型	允许 Null 值	
username	nvarchar(50)	☐	
schoolyear	int	☒	
class	int	☒	
studentnumber	nvarchar(50)	☒	
name	nvarchar(50)	☒	
sex	int	☒	
birthday	datetime	☒	
institute	int	☒	

LINJIARUI-PC\LOC....aspnet_Membership			
列名	数据类型	允许 Null 值	
ApplicationId	uniqueidentifier	☐	
 UserId	uniqueidentifier	☐	
Password	nvarchar(128)	☐	
PasswordFormat	int	☐	
PasswordSalt	nvarchar(128)	☐	
MobilePIN	nvarchar(16)	☒	
Email	nvarchar(256)	☒	
LoweredEmail	nvarchar(256)	☒	
PasswordQuestion	nvarchar(256)	☒	
PasswordAnswer	nvarchar(128)	☒	
IsApproved	bit	☐	
IsLockedOut	bit	☐	
CreateDate	datetime	☐	

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- **Model设计**

- 功能包括：登录、登出、修改密码、浏览用户列表、添加用户、编辑用户、删除用户

```
public class ChangePasswordModel{...}

public class LogOnModel
{
    [Required]
    [Display(Name = "用户名")]
    public string UserName { get; set; }

    [Required]
    [DataType(DataType.Password)]
    [Display(Name = "密码")]
    public string Password { get; set; }

    [Display(Name = "记住我?")]
    public bool RememberMe { get; set; }
}

public class RegisterModel{...}
```

```
public class AddUserModel{...}

public class EditUserModel
{
    [属性]

    private void initialArray(){...}

    public EditUserModel(){...}

    public void setValues(string Email, string School,
        string Name, int Sex, string Birthday, int ID)
    {
        ...
    }

    public EditUserModel(string username){...}

    public void CreateViewDataForUserData(DataTable dt)
    {
        ViewData["UserInformation"] = dt;
    }

    public ActionResult UserManagement()
    {
        CreateViewDataForUserData(AppendDBService.getAllUsers());
        CreateViewDataForBreadcrumb("用户");

        return View(ViewData);
    }
}
```

```
public static DataTable getAllUserInformation(){...}

public static void EditUserInfo(EditUserModel model)
{
    string userid = AppendDBService.getRowFromTable("UserName", model.UserName);
    if (model.role == AccountRoles.Manager)
    {
        AppendDBService.updateValueInTable("Email", model.Email, userid);
    }
    else if (model.role == AccountRoles.Teacher)
    {
        AppendDBService.updateValueInTable("Email", model.Email, userid);
        string[] valuecolnames = { "institute", "discription", "name", "sex", "birthday", "schoolyear", "class", "studentid" };
        string[] values = { model.Institute.ToString(), model.Description.ToString(), model.Sex.ToString(), model.Birthday.ToString(), model.SchoolYear.ToString(), model.Class.ToString(), model.StudentID.ToString() };
        AppendDBService.updateValueInTable(valuecolnames, values, userid);
    }
    else if (model.role == AccountRoles.Student)
    {
        AppendDBService.updateValueInTable("Email", model.Email, userid);
        string[] valuecolnames = { "schoolyear", "class", "studentid" };
        string[] values = { model.SchoolYear.ToString(), model.Class.ToString(), model.StudentID.ToString() };
        AppendDBService.updateValueInTable(valuecolnames, values, userid);
    }
}
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- **Controller设计**

- 功能包括：登录、登出、修改密码、浏览用户列表、添加用户、编辑用户、删除用户

```
public class AccountController : Controller
{
    protected override void Initialize(Requ

    public ActionResult LogOn()...

    [HttpPost]
    public ActionResult LogOn(LogOnModel mo

    public ActionResult LogOff()...

    public ActionResult Register()...

    [HttpPost]
    public ActionResult Register(RegisterMo

    [Authorize]
    public ActionResult ChangePassword()...

    [Authorize]
    [HttpPost]
    public ActionResult ChangePassword(Chan
```

```
public class ManagementController : Controller
{
    public ActionResult AddUser()...

    [HttpPost]
    public ActionResult AddUser(AddUserModel model)...

    public ActionResult DeleteUser(string username)...

    public ActionResult EditUser(string username)...

    [HttpPost]
    public ActionResult EditUser(EditUserModel model)...
```

功能实现

技术思路

账户模块

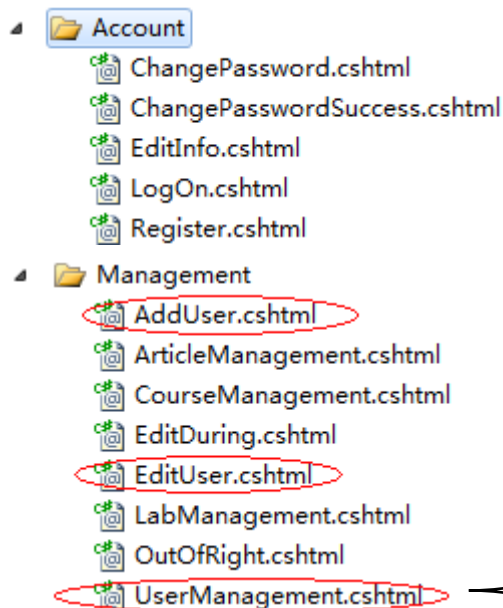
文章模块

实验室管理模块

课程模块

- View设计

- 功能包括：登录、登出、修改密码、浏览用户列表、添加用户、编辑用户、删除用户



```

<@using System.Data;
<div class="wrapper row2">
  <div class="rnd">
    <div id="container" class="clear">
      <ul class="breadcrumb">...</ul>
      <div id="left_column">
        <article class="span9">
          <div class="row">
            <div class="span5">
              <h3>用户信息统计表</h3>
            </div>
            <div class="span4">...</div>
          </div>
          <table class="datatable table table-striped">
            <article>
              <!-- /Page grid cell (12 blocks - full) -->
            </div>
          </div>
        </div>
      </div>
    </div>
  <!-- jQuery DataTable -->
  <script src="@Url.Content("~/js/plugins/dataTables/jquery.data"
</script>...</script>

```

```
<table class="datatable table-striped table-bordered" id="example-2">
  <thead>...</thead>
  <tbody>
    @foreach (DataRow item in dt.Rows)
    {
      Html.ViewContext.Writer.WriteLine("<tr>");
      for (int i = 0; i < dt.Columns.Count; i++)
      {
        Html.ViewContext.Writer.WriteLine("<td>" + item[i] + "</td>");
      }
      Html.ViewContext.Writer.WriteLine("<td>" +
        Html.ActionLink("修改", "EditUser", "Management", new { user
        Html.ViewContext.Writer.WriteLine(
          Html.ActionLink("删除", "DeleteUser", "Management", new { user
          + "</td>");
        Html.ViewContext.Writer.WriteLine("</tr>");
      }
    }
  </tbody>
</table>
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- View设计

[主页](#) / [管理](#) / [用户](#)

用户信息统计表

添加

列过滤

10

条记录/页

Search:

用户名	账户类型	姓名	性别	研究所	班级	操作
Administrator	管理员					修改 删除
Whw13	学生	王珩玮	男	防灾所	结91	修改 删除

Showing 1 to 2 of 2 entries

← Previous

1

Next →

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- 数据表设计

- smallimage&largeimage : (缩略图&完整) 图片信息
- detail : html格式文档
- brief : 去除标签过后的文档内容 (用于预览)
- type : 储存文章类型
- isfeature : 是否显示在主页的特辑框中

LINJIARUI-PC\LOCA...Site - dbo.ARTICLE			
	列名	数据类型	允许 Null 值
	id	uniqueidentifier	☐
	title	nvarchar(50)	☐
	smallimage	image	☒
	largeimage	image	☒
	detail	nvarchar(MAX)	☒
	brief	nvarchar(MAX)	☒
	type	nvarchar(50)	☒
	author	nvarchar(50)	☒
	datetime	datetime	☒
	route	nvarchar(50)	☒
	lasteditor	nvarchar(50)	☒
	isfeature	int	☒

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- **Model设计**

- 功能包括：写新文章、删除、修改、浏览文章列表、浏览指定文章

```
public class ArticleManagementModel : BasicArticleModel
{
    public List<BriefArticle> articles = new List<BriefArticle>();

    public void init(string role)
    {
        InitArray(role);
        string[] colnames = { "id", "title", "datetime", "author", "lasteditor", "brief", "type" };
        DataTable articleTable = AppendDBService.getAllFromTable(colnames, "ARTICLE");
        foreach (DataRow r in articleTable.Rows)
        {
            BriefArticle article = new BriefArticle();
            article.id = r["id"].ToString();
            article.author = r["author"].ToString();
            article.brief = r["brief"].ToString();
            article.datetime = r["datetime"].ToString();
            article.lasteditor = r["lasteditor"].ToString();
            article.title = r["title"].ToString();
            article.type = this.articletypeArray.Find(m => (m.type.ToString().Equals(r["type"].ToString())));
            articles.Add(article);
        }
    }
}
```

```
public class DetailArticleModel[...]
public class AddArticleModel : BasicArticleModel[...]
public class EditArticleModel : BasicArticleModel
{
    [属性]

    public EditArticleModel(DataRow data, string role)
    {
        InitArray(role);
        id = data["id"].ToString();
        title = data["title"].ToString();
        detail = data["detail"].ToString();
        isfeature = false;
        type = (ArticleTypeEnum.Parse(typeof(ArticleTypeEnum), data
    }
}

public class TypeArticleModel : BasicArticleModel[...]
public class BasicArticleModel[...]
public static string AddArticle(string title, string de
public static void EditArticle(string id, string title,
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- **Model设计**

- 重点技术：将图片二进制化，并在数据库中存取

```
BinaryFormatter bf = new BinaryFormatter();  
MemoryStream s1 = new MemoryStream();  
Image largeimg = Image.FromStream(image.InputStream);  
bf.Serialize(s1, (object)largeimg);  
largeimage = s1.ToArray();
```

```
MemoryStream ms = new MemoryStream(img, 0, img.Length);  
BinaryFormatter bf = new BinaryFormatter();  
object obj = bf.Deserialize(ms);  
ms.Close();  
MemoryStream files = new MemoryStream();  
Bitmap image = (Bitmap)obj;  
image.Save(files, System.Drawing.Imaging.ImageFormat.Jpeg);  
return File(files.ToArray(), "image/jpeg");
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- **Controller设计**

- 功能包括：写新文章、删除、修改、浏览文章列表、浏览指定文章

```
public class ArticleController : Controller
{
    public ActionResult ViewArticle(string id)...

    public ActionResult AddArticle()...

    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult AddArticle(AddArticleModel model)...

    public ActionResult DeleteArticle(string id, string viewname, string controllername)...

    public ActionResult EditArticle(string id)...

    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult EditArticle(AddArticleModel model)...

    public ActionResult ShowArticleByType(string type)...
}
```



```
public ActionResult ArticleManagement()
{
    ArticleManagementModel model = new ArticleManagementModel();
    model.init(Roles.GetRolesForUser()[0]);
    return View(model);
}

public ActionResult getImageForBriefArticle(string id)...

public ActionResult getImageForFeatureArticle(string id)...
```


功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- **View设计**

- 功能包括：写新文章、删除、修改、浏览文章列表、浏览指定文章

- Article

- AddArticle.cshtml
 - EditArticle.cshtml
 - ShowArticleByType.cshtml
 - ViewArticle.cshtml

- Management

- AddUser.cshtml
 - ArticleManagement.cshtml**
 - CourseManagement.cshtml
 - EditDuring.cshtml
 - EditUser.cshtml
 - LabManagement.cshtml
 - OutOfRight.cshtml

```
@model LaboratorySiteNew.Models.ArticleMa
<script src=@Url.Content("~/js/ArticleMar
<div class="wrapper row3">
  <div class="rnd">
    <div id="container" class="clear">
      <a class="btn" style="margin
      <div class="btn-group btn-mir
      <h2><a><b>文章列表</b></a></f
      <table width="906px">
        @
        <thead class="shownInr
        <tbody id=@article.id
          <tr>...</tr>
          <tr>...</tr>
          <tr>...</tr>
          <tr>...</tr>
        </tbody>
      </table></div>
    </div>
  </div>
</div>
```

```
public ActionResult getImageForBriefArticle(string id)...
```

```
<tr>
  <td align="center" colspan="2">
    <div class="thumbnail">
      <img src=@Url.Action("getImageForBriefArticle", "Management", new { id = article.id }) />
    </div>
  </td>
  <td colspan="5" rowspan="4">
    <div class="content" style="height:275px;width:655px;padding:20px"><p>@article.brief</p></div>
  </td>
</tr>
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- View设计

发布新文章

类型

文章列表

[实验室管理]

实验室简介

隐藏详细 编辑 删除



清华大学“力学计算与仿真实验室”的前身是土木工程系下的“CAD/CAE教学实验室”和“数字防灾与虚拟工程实验室”。在20世纪80年代后，随着应用电子计算机进行工程计算和数值仿真的兴起。清华大学于2002年和2005年分别投入“985”和“211”经费共272万，建立了“CAD/CAE教学实验室”和“数字防灾与虚拟工程实验室”，主要服务于本科生、研究生教学及科研。2007年，土木工程系将此两个实验室联合起来，成立了“力学计算与仿真实验室”，作为清华大学“力学实验教学基地”的组成实验室之一。截至2012年9月，“力学计算与仿真实验室”服务于土木系、建管系和水利系共12位教师、近500名本科生和约100名研究生、以及11门本科生课程和3门研究生课程，拥有设备138台件和丰富的软件环境，每年完成约21000机时的上机教学工作量。近5年来，本实验室已培育出国家级精品课1门、北京市教学名师1名、北京市“教育先锋”先进个人1名、华夏建设科学技术一等奖2项、中建总公司科学技术奖一等奖1项、清华大学教学成果奖一、二等奖各1次、茅以升青年科技奖1项、国家自然科学基金优秀青年基金获得者1名、教育部新世纪人才...(省略更多内容)

作者:

Administrator

最后编辑:

Administrator

最后修改时间:

2014/4/17 13:41:16

[其他]

实验室网站进展

显示详细 编辑 删除

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- 数据表设计

- 表DAYPLAN中每一行代表一天
- 表DURINGPLAN中每一行代表一大节课程
- 表ACADEMICDATE存储学期信息

LINJIARUI-PC\LOC...ite - dbo.DAYPLAN		
列名	数据类型	允许 Null 值
datenumbr	int	☐
during1	uniqueidentifier	☒
during2	uniqueidentifier	☒
during3	uniqueidentifier	☒
during4	uniqueidentifier	☒
during5	uniqueidentifier	☒
during6	uniqueidentifier	☒

LINJIARUI-PC\LOC...- dbo.DURINGPLAN		
列名	数据类型	允许 Null 值
id	uniqueidentifier	☐
starttime	datetime	☒
endtime	datetime	☒
coursename	nvarchar(50)	☒
usemachine	int	☒
classnames	nvarchar(50)	☒

NJIARUI-PC\LO...bo.ACADEMICDATE	
startdate	
2014-02-24 00:00:00.000	

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

• Model设计

```
public class LabManagementModel
{
    public string usermachine { set; get; }
    public string machinenumber { set; get; }
    public int todayNo { set; get; }
    public int toduringNo { set; get; }

    public List<WeekPlan> plans = new List<WeekPlan>();
    public List<string> coursenames = new List<string>();

    private int getToduringNo()...

    public LabManagementModel()...
}

public class EditDuringModel...
```

```
public class WeekPlan
{
    public int weekNo { get; set; }
    public DateTime startdate { get; set; }
    public DateTime enddate { get; set; }
    public DayPlan[] dayplans = new DayPlan[7];
}

public class DayPlan
{
    public int dayNo { get; set; }
    public DuringPlan[] durings = new DuringPlan[6];
}

public class DuringPlan
{
    public int starthour { get; set; }
    public int startminute { get; set; }
    public int endhour { get; set; }
    public int endminute { get; set; }
    public string coursename { get; set; }
    public int usemachine { get; set; }
    public string classname { get; set; }
}
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

• Controller设计

```
public ActionResult LabManagement ()
{
    LabManagementModel model = new LabManagementModel();
    return View(model);
}

public ActionResult EditDuring(int weekNo, int dayNo, int duringNo)
{
    EditDuringModel model = new EditDuringModel(weekNo * 7 + dayNo, duringNo);
    return View(model);
}

[HttpPost]
public ActionResult EditDuring(EditDuringModel model)
{
    AppendDBService.UpdateDuringCondition(model);
    return RedirectToAction("LabManagement");
}
```

功能实现

技术思路

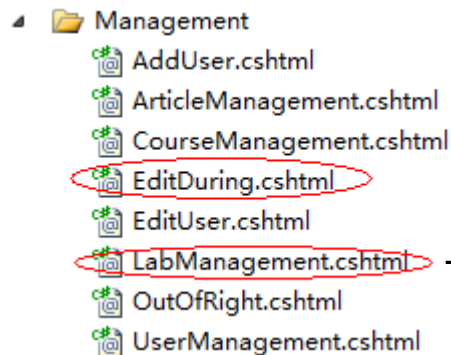
账户模块

文章模块

实验室管理模块

课程模块

• View设计



```
@model LaboratorySiteNew.Models.LabManagementModel

<script src=@Url.Content("~/js/LabManagement.js")></script>

<div class="wrapper row3">
    <div class="rnd">
        <div id="container" class="clear">
            <div class="row">...</div>
            @for (int wNo = 0; wNo < 18; wNo++)
            {
                <table class="week week@(wNo)">
                    <thead>
                        <tr>
                            <th></th>
                            <th>星期一</th>
                            <th>星期二</th>
                            <th>星期三</th>
                            <th>星期四</th>
                            <th>星期五</th>
                            <th>星期六</th>
                            <th>星期日</th>
                        </tr>
                    </thead>
                    <tbody>...</tbody>
                </table>
            }
        </div>
    </div>
</div>
```

```
@for (int i = 0; i < 6; i++)
{
    <tr>
        <td style="width:60px">第@(i + 1)大节</td>
        @for (int j = 0; j < 7; j++)
        {
            <td style="height:120px; width:95px">
                @if (Model.todayNo == wNo * 7 + j && Model.toduringNo == i)
                {
                    <a class="btn during_btn btn-danger" style="height:100%; width:95px" href = @Url.Action("EditDuring", "Mar")>
                        @if (Model.plans[wNo] != null && Model.plans[wNo].dayplans[j] != null && Model.plans[wNo].dayplans[j].durings[i] != null)
                        {
                            LaboratorySiteNew.Models.DuringPlan during = Model.plans[wNo].dayplans[j].durings[i];
                            <p>@{(during.starthour.ToString("D2"))}:@{(during.startminute.ToString("D2"))}-@{(during.endhour.ToString("D2"))}</p>
                            <p>@{(during.coursename)}</p>
                            <p>@{(during.classname)}</p>
                        }
                    </a>
                }
                else
                {
                    <a class="btn during_btn" style="height:100%; width:95px" href = @Url.Action("EditDuring", "Management")>
                        @if (Model.plans[wNo] != null && Model.plans[wNo].dayplans[j] != null && Model.plans[wNo].dayplans[j].durings[i] != null)
                        {
                            LaboratorySiteNew.Models.DuringPlan during = Model.plans[wNo].dayplans[j].durings[i];
                            <p>@{(during.starthour.ToString("D2"))}:@{(during.startminute.ToString("D2"))}-@{(during.endhour.ToString("D2"))}</p>
                            <p>@{(during.coursename)}</p>
                            <p>@{(during.classname)}</p>
                        }
                    </a>
                }
            </td>
        }
    </tr>
}
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- View设计



功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

• 数据表设计

- + dbo.COURSE
- + dbo.REL_COURSE_EXPERIMENT
- + dbo.EXPERIMENT
- + dbo.LABCONDITION

LINJIARUI-PC\LOCA...Site - dbo.COURSE			
	列名	数据类型	允许 Null 值
	id	uniqueidentifier	☐
	name	nvarchar(50)	☒
	teachername	nvarchar(50)	☒
	description	nvarchar(MAX)	☒

LINJIARUI-PC\LOC...- dbo.EXPERIMENT			
	列名	数据类型	允许 Null 值
	id	uniqueidentifier	☐
	name	nvarchar(50)	☒

LINJIARUI-PC\LO...OURSE_EXPERIMENT			
	列名	数据类型	允许 Null 值
	courseid	uniqueidentifier	☐
	experimentid	uniqueidentifier	☐

LINJIARUI-PC\LOC...dbo.LABCONDITION			
	列名	数据类型	允许 Null 值
	usemachine	int	☒
	machinenumber	int	☒

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

• Model设计

```
public class CourseManagementModel
{
    public List<Course> courses = new List<Course>();

    public void init(string role, string username)
    {
        if (role == "Student") return;
        DataTable coursetable = AppendDBService.getAllFromTable("COURSE");
        foreach (DataRow row in coursetable.Rows)
        {
            if (role == "Teacher" && !row["teachername"].ToString().Contains(username)) continue;
            Course course = new Course();
            course.id = row["id"].ToString();
            course.name = row["name"].ToString();
            //course.description = row["description"].ToString();
            course.teachername = row["teachername"].ToString();
            courses.Add(course);
        }
    }
}
```

```
public class EditCourseModel...

public class AddCourseModel...

public class EditExperimentModel
{
    public string id { set; get; }

    [Display(Name="实验名称")]
    public string name { set; get; }

    public string courseid { set; get; }

    public EditExperimentModel()
    {}

    public EditExperimentModel(string courseid, string experimentid)
    {
        this.courseid = courseid;
        this.id = experimentid;
        DataRow r = AppendDBService.getRowFromTable("id", id, "EXPERIMENT");
        id = r["id"].ToString();
        name = r["name"].ToString();
    }
}

public class AddExperimentModel...

public class ViewCourseModel...
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

• Controller设计

```
public ActionResult EditCourse(string id)...

[HttpPost]
public ActionResult EditCourse(EditCourseModel model)...

public ActionResult AddCourse()...

[HttpPost]
public ActionResult AddCourse(AddCourseModel model)...

public ActionResult AddExperiment(string courseid)...

public ActionResult DelCourse(string id, string actionname, string controlname)...

[HttpPost]
public ActionResult AddExperiment(AddExperimentModel model)...

public ActionResult EditExperiment(string courseid, string experimentid)...

[HttpPost]
public ActionResult EditExperiment(EditExperimentModel model)...

public ActionResult DelExperiment(string courseid, string experimentid)...

public ActionResult ViewCourse(string id)...

public ActionResult OpenExperimentView(string id)...

[HttpPost]
public ActionResult ReceiveExperimentForm()...
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- **网上实验功能实现机制**

- 实验文件：dll文件（封装了实验类）、两个html/cshtml页面（实验UI以及结果展示UI）
- 实验新建及编辑流程：在课程编辑、上传或新建界面可进行实验的选择、新建、编辑删除功能
- 对实验的处理机制：
 - 在Views文件夹下设置一子文件夹Experiment存储实验视图文件
 - 实验视图文件被存放在Experiment/实验id文件夹中
 - dll文件存放于服务器中另一文件夹中
 - 当浏览实验时，先推送实验UI视图
 - 当接受到HttpPost请求提交的Form数据后，动态加载实验类，实例化、载入数据、运行之后使用ViewData将实验结果传递给结果视图



The screenshot shows a web form titled "课程实验" (Course Experiments). It contains a text input field with the placeholder "测试实验;" (Test Experiment;). Below the input field is a dropdown menu currently showing "图形学实验" (Graphics Experiment). To the right of the dropdown are two buttons: "向课程添加" (Add to Course) and "取消添加" (Cancel Add). Below these are three buttons: "新增实验" (Add New Experiment), "修改选中实验" (Modify Selected Experiment), and "删除选中实验" (Delete Selected Experiment). At the bottom left of the form is a "提交" (Submit) button.

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

• 网上实验功能实现机制

- **对实验的处理机制：**
- 在Views文件夹下设置一子文件夹
Experiment存储实验视图文件
- 实验视图文件被存放在Experiment/实验id文件夹中
- dll文件存放于服务器中另一文件夹
- 当浏览实验时，先推送实验UI视图
- 当接受到HttpPost请求提交的Form数据后，动态加载实验类，实例化、载入数据、运行之后使用ViewData将实验结果传递给结果视图

```
[HttpPost]
public ActionResult ReceiveExperimentForm()
{
    string expid = Request.Form["ID"];
    try
    {
        byte[] buffer = System.IO.File.ReadAllBytes(HttpRuntime.AppDomainAppPath + "App_Data\\Experiment\\" + expid + "\\");
        Assembly ass = Assembly.Load(buffer);
        Type tp = ass.GetType("WebExperiment.ExperimentModel");
        Object model = Activator.CreateInstance(tp);
        PropertyInfo[] properties = tp.GetProperties();
        foreach (PropertyInfo p in properties)
        {
            if (Request.Form[p.Name] != null)
            {
                p.SetValue(model, TypeDescriptor.GetConverter(p.PropertyType).ConvertFrom(Request.Form[p.Name]), null);
            }
        }

        MethodInfo method = tp.GetMethod("DoExperiment"); //获取方法
        method.Invoke(model, null); //Invoke调用方法

        foreach (PropertyInfo p in properties)
        {
            ViewData[p.Name] = p.GetValue(model, null);
        }
    }
    catch (Exception)
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

• View设计

Course

Experiment

- AddCourse.cshtml
- AddExperiment.cshtml
- EditCourse.cshtml
- EditExperiment.cshtml
- ViewCourse.cshtml

Management

- AddUser.cshtml
- ArticleManagement.cshtml
- CourseManagement.cshtml
- EditDuring.cshtml

```
@model LaboratorySiteNew.Models.AddCourseModel
@{
    ViewBag.Title = "添加课程";
}
<div class="wrapper row3">
    <div class="rnd">
        <div id="container" class="clear">
            <h2>添加课程</h2>

            @using (Html.BeginForm())
            {
                <div>
                    <fieldset>...</fieldset>
                </div>
            }
        </div>
    </div>
</div>

<script language="javascript">
    $(function () {
        $(".delexperiment").hide();
        $(".editexperiment").hide();
        if ($("#experimentlist")[0].selectedIndex > -1) {
            var i = $("#experimentlist")[0].selectedIndex;
            $(". " + ($("#experimentlist")[0][i].value)).show();
        }
    })

    function ExperimentChange() {
        $(".delexperiment").hide();
        $(".editexperiment").hide();
        if ($("#experimentlist")[0].selectedIndex > -1) {
            var i = $("#experimentlist")[0].selectedIndex;
        }
    }
</script>
```

```
<fieldset>
    @Html.LabelFor(m => m.name)
    @Html.TextBoxFor(m => m.name)

    @Html.LabelFor(m => m.teachername)
    @Html.TextBoxFor(m => m.teachername)

    @Html.LabelFor(m => m.description)
    @Html.TextAreaFor(m => m.description)

    @Html.LabelFor(m => m.experimentnames)
    @Html.TextBoxFor(m => m.experimentnames)

    <div class="editor-field">
        @Html.DropDownList("experimentlist", new SelectList(Model.experimentArray, "id", "name"), new { id = "ex" })
        <p class="btn" onclick="AddExperiment();">向课程添加</p>
        <p class="btn" onclick="DelExperiment();">取消添加</p>
    </div>

    <a class="btn" href="@Url.Action("AddExperiment", "Course", null)">新增实验</a>
    @foreach (LaboratorySiteNew.Models.ExperimentBasic b in Model.experimentArray)
    {
        <a class="btn editexperiment" @b.id" href="@Url.Action("EditExperiment", "Course", new { experimentid = b.id" )">编辑</a>
        @foreach (LaboratorySiteNew.Models.ExperimentBasic b in Model.experimentArray)
        {
            <a class="btn delexperiment" @b.id" href="@Url.Action("DelExperiment", "Course", new { experimentid = b.id" )">删除</a>
        }

        <input type="hidden" id = "experimentids" name = "experimentids" value = @Model.experimentids >
    }

    <p>
        <input type="submit" value="提交" />
    </p>
</fieldset>
```

功能实现

技术思路

账户模块

文章模块

实验室管理模块

课程模块

- View设计

添加课程

课程名称

任课教师

课程简述

课程实验

图形学实验

向课程添加

取消添加

新增实验

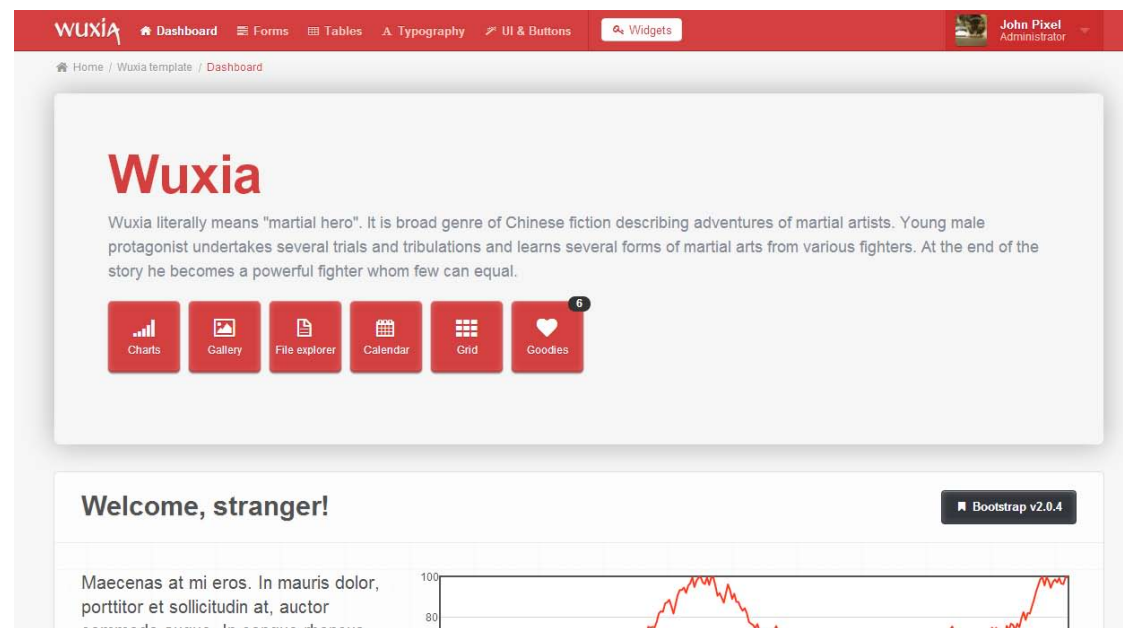
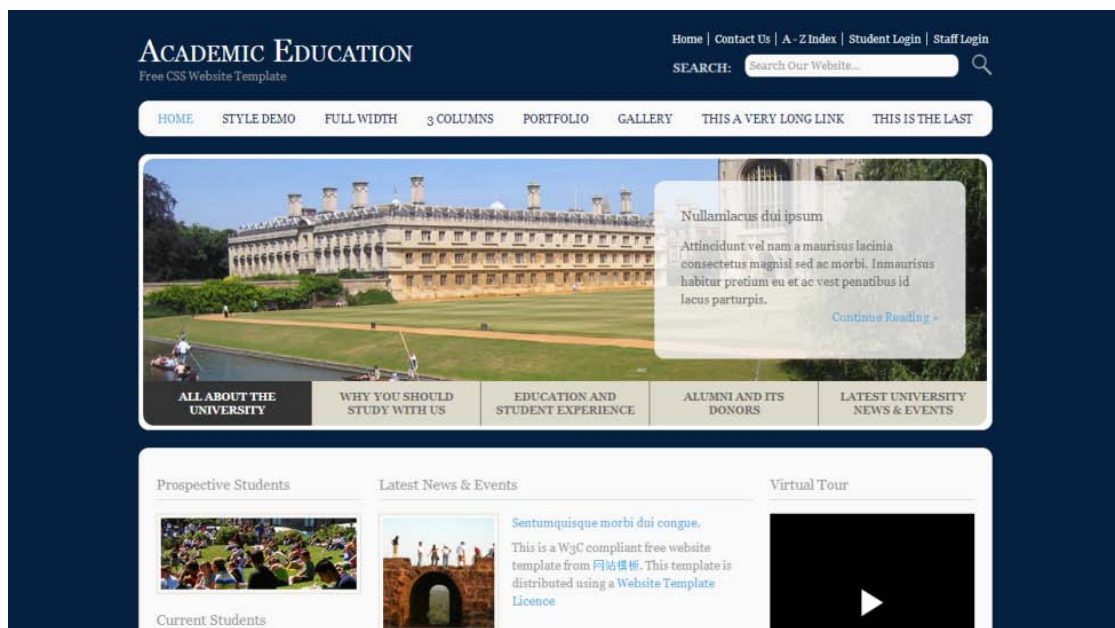
修改选中实验

删除选中实验

提交

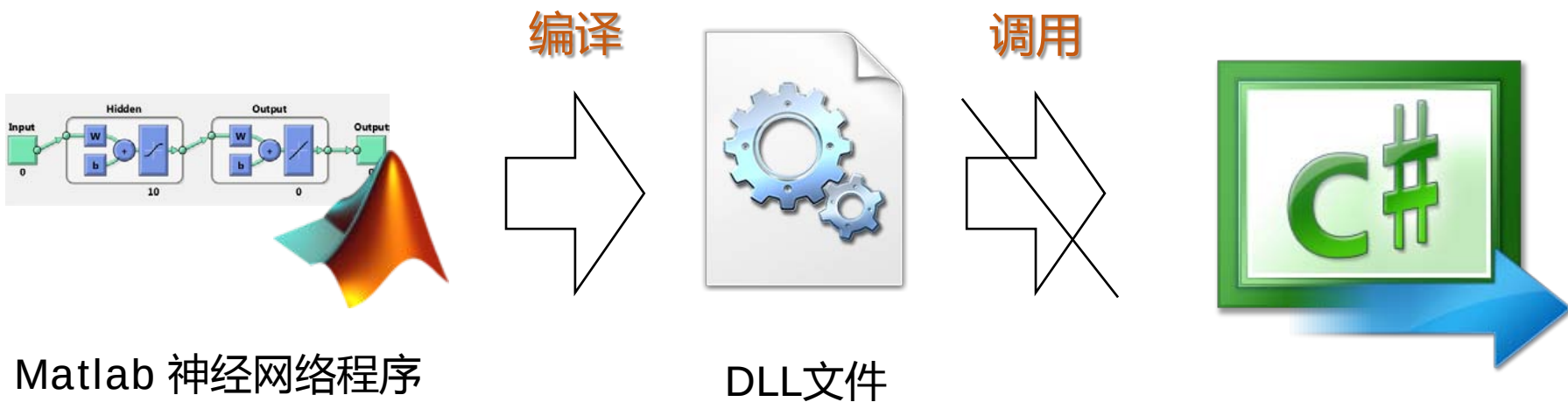
更多？

- 使用了两个网页模板



更多？

- 基于Matlab Compiler Runtime，能够运行Matlab生成的exe与库文件。但缺少神经网络初始化的相关文件，所以包含matlab预定义神经网络的程序都不能被调用。



效果展示

- [网站首页](#)

总结

- 不要随意地开发程序
- 不要轻视工作的复杂
- 不要期望能面面俱到
- 不要使用多个不熟悉的网页模板样式进行网站开发！

谢谢！