4th CS Hand in

Description of the solution

The solution is an extension of the third hand in. Login page is added and users have different user profiles according to their employee type – coach or administrator.

From the login page users will be directed to the welcome page (Welcome.aspx). Depending on the user profile (coach or administrator), the user can access different information via the menu bar at the welcome page. The coach and administrator files are opened when a user tries to login. A users email and password are check with information in the files.

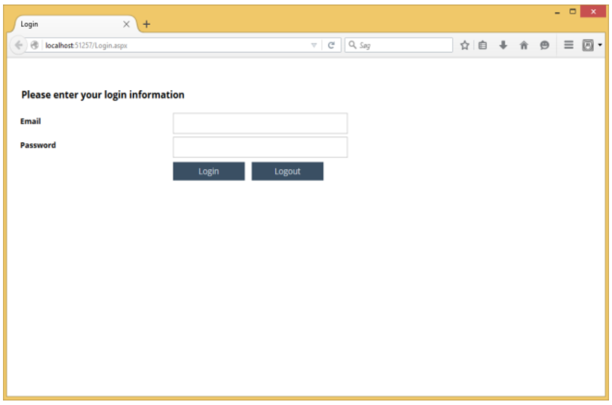
Administrators can through the ADMINSTRATOR menu item, which directs to the display administrators page (DisplayAdmin.aspx) get a list of administrators displayed and create or delete administrators. Furthermore the administrator that are logged in can update own user profile/information. From the display page administrators can also get a list of all employees (coaches and administrators) displayed. Administrators are allowed to go to the coaches menu to get coach information displayed and create or delete coaches, but administrators cannot update a coach user profile/information.

Coaches can through the COACH menu item, which directs to the display coaches page (DisplayCoach.aspx) get a list of coaches displayed and create or delete coaches. Furthermore the coach that are logged in can update own user profile/information. Coaches are not allowed to go to the administrators section of the solution and can therefore not see any information regarding administrators.

An error page will be displayed if a coach tries to access the administrator section or if an unknown user will try to access any page in solution (except welcome and login page).

Login Page

Login uses the users email address and password as parameters.



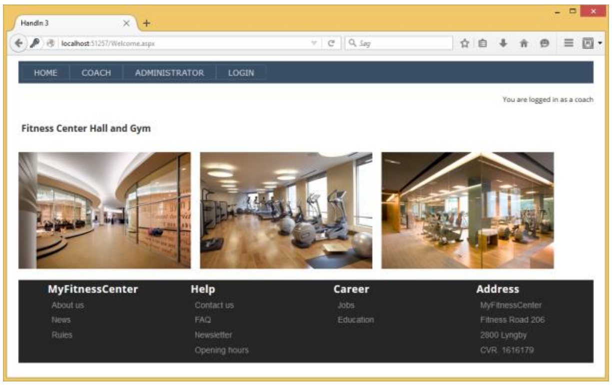
Please enter your login information

Login.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

using System.Collections;

namespace HandIn3CS
{
    public partial class Login : System.Web.UI.Page
    {
        ArrayList adminlist;
        ArrayList coachlist;
        int userlevel;
        int staffnumber;
        string role;

        protected void Page_Load(object sender, EventArgs e)
        {
            LabelMessage.Text = "";

            try
            {
                adminlist = FileUtility.ReadFile(Server.MapPath("~/App_Data/Admfile.ser"));
                Application["myadminlist"] = adminlist;

                coachlist = 
FileUtility.ReadFile(Server.MapPath("~/App_Data/Coachfile.ser"));
                Application["mycoachlist"] = coachlist;
            }
            catch (Exception ex)
            {
                LabelMessage.Text = ex.Message;
            }
        }

        protected void ButtonLogout_Click(object sender, EventArgs e)
        {
            userlevel = 0;
            Session["Level"] = userlevel;
            LabelMessage.Text = "You are logged out";
            Response.Redirect("Welcome.aspx");
        }

        protected void ButtonLogin_Click(object sender, EventArgs e)
        {

            foreach (Coach c in coachlist)
            {
                if ((c.Email == TextBoxEmail.Text) && (c.Password == TextBoxPassword.Text))
                {
                    userlevel = 2;
                    Session["Level"] = userlevel;
                    staffnumber = c.StaffNumber;
                    Session["StaffNumber"] = staffnumber;
                    role = c.Role;
                    Session["Role"] = role;
                    Response.Redirect("Welcome.aspx");
                }
                else
                {
                    foreach (Administrator a in adminlist)
                    {
                        if ((a.Email == TextBoxEmail.Text) && (a.Password ==
TextBoxPassword.Text))
                        {
                            userlevel = 1;
                            Session["Level"] = userlevel;
                            staffnumber = a.StaffNumber;
                            Session["StaffNumber"] = staffnumber;
                            role = a.Role;
                            Session["Role"] = role;
                            Response.Redirect("Welcome.aspx");
                        }
                        else
                        {
                            LabelMessage.Text = "You are not a registred user";
                        }
                    }
                }
            }
        }
    }
}
```

Welcome page

Welcome page with a menu bar with links to administration of coaches and Administrators. And login/out.



You are logged in as a coach

Fitness Center Hall and Gym

Welcome.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace HandIn3CS
{
    public partial class Welcome : System.Web.UI.Page
    {
        int userlevel;
        int staffnumber;
        string role;

        protected void Page_Load(object sender, EventArgs e)
        {
            try
            {
                userlevel = (int)Session["Level"];
                staffnumber = (int)Session["StaffNumber"];
                role = (string)Session["Role"];
            }
            catch (NullReferenceException nre)
            {
                userlevel = 0;
                Session["Level"] = userlevel;
                LabelUserLevel.Text = nre.Message;
            }
            finally
            {
                SetLevelOrError();
            }
        }

        private void SetLevelOrError()
        {
            if (userlevel == 1)
            {
                LabelUserLevel.Text = "You are logged in as an administrator ";
            }
            else if (userlevel == 2)
            {
                LabelUserLevel.Text = "You are logged in as a coach ";
            }

            else
            {
                LabelUserLevel.Text = "You are not logged in";
            }
        }
    }
}
```

Display Administrator

The page displays Administrators. The user also have the possibility to delete administrators. Buttons update and create will direct the user to either the update or create page. Save button will save the changes made by the user. An administrator who are logged in can only update own user information. At the page administrators also can get all employees listed.

DisplayAdmin.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

using System.Collections;
using System.Data;

namespace HandIn3CS
{
    public partial class DisplayAdmin : System.Web.UI.Page
    {
        ArrayList adminlist;
        ArrayList coachlist;
        int userlevel;
        int staffnumber;
        string role;

        protected void Page_Load(object sender, EventArgs e)
        {
            Response.Cache.SetNoStore();

            try
            {
                userlevel = (int)Session["Level"];
                staffnumber = (int)Session["StaffNumber"];
                role = (string)Session["Role"];
            }
            catch (NullReferenceException nre)
            {
                userlevel = 0;
                Session["Level"] = userlevel;
                LabelMessage.Text = nre.Message;
            }
            finally
            {
                SetLevelOrError();
            }

            if (Application["myadminlist"] == null)
            {
                adminlist = new ArrayList();
                Application["myadminlist"] = adminlist;
                ButtonUpdate.Enabled = false;
                ButtonDelete.Enabled = false;
                ButtonSave.Enabled = false;
            }
            else
            {
                adminlist = (ArrayList)Application["myadminlist"];  // type casting
                ButtonUpdate.Enabled = true;
                ButtonDelete.Enabled = true;
                ButtonSave.Enabled = true;
            }

            if (!Page.IsPostBack)
            {
                if (adminlist.Count == 0)
                {
                    LabelMessage.Text = "No Administrators";
                }
                else
                {
                    PrintAdmin(adminlist);
                    ButtonHide.Visible = false;
                }
            }
        }

        protected void ButtonRead_Click(object sender, EventArgs e)
        {
            try
            {
                adminlist = FileUtility.ReadFile(Server.MapPath("~/App_Data/Admfile.ser"));
                Application["myadminlist"] = adminlist;

                LabelMessage.Text = "File read from App_Data";

                if (adminlist.Count == 0)
                {
                    LabelMessage.Text = "No Administrators in file";
                }
                else
                {
                    PrintAdmin(adminlist);
                }
            }
            catch (Exception ex)
            {
                LabelMessage.Text = "Admfile could not be read" + ex.Message;
            }
        }

        protected void ButtonCreate_Click(object sender, EventArgs e)
        {
            Response.Redirect("CreateAdmin.aspx");
        }

        protected void ButtonSave_Click(object sender, EventArgs e)
        {
            adminlist = (ArrayList)Application["myadminlist"];

            if (adminlist != null)
            {
                FileUtility.WriteFile(adminlist, Server.MapPath("~/App_Data/Admfile.ser"));
                LabelMessage.Text = "File written in App_Data folder";
            }
            else
            {
                LabelMessage.Text = "No Administrators to save";
            }
        }

        protected void ButtonDelete_Click(object sender, EventArgs e)
        {
            if (ListBoxDisplay.SelectedIndex == -1)
            {
                LabelMessage.Text = "Please select an administrator from the list";
            }
            else
            {
                for (int i = 0; i < adminlist.Count; i++)
                {
                    if (ListBoxDisplay.SelectedIndex == i)
                    {
                        adminlist.RemoveAt(i);
                        LabelMessage.Text = "Selected Administrator has been deleted";
                    }
                }
            }

            PrintAdmin(adminlist);
        }

        protected void ButtonUpdate_Click(object sender, EventArgs e)
        {
            Response.Redirect("UpdateAdmin.aspx");
        }

        private void PrintAdmin(ArrayList adminlist)
        {
            ListBoxDisplay.Items.Clear();

            foreach (Administrator a in adminlist)
            {
                ListBoxDisplay.Items.Add(a.ToString());
            }
        }

        private void SetLevelOrError()
        {
            if (userlevel == 1)
            {
                LabelUserLevel.Text = "You are logged in as an administrator ";
            }
            else
            {
                Response.Redirect("ErrorPage.aspx");
            }
        }

        protected void ButtonShowAll_Click(object sender, EventArgs e)
        {
            coachlist = (ArrayList)Application["mycoachlist"];

            ButtonShowAll.Visible = false;
            ButtonHide.Visible = true;

            ButtonCreate.Enabled = false;
            ButtonRead.Enabled = false;
            ButtonUpdate.Enabled = false;
            ButtonDelete.Enabled = false;
            ButtonSave.Enabled = false;


            ListBoxDisplay.Visible = false;
            ListBoxAll.Visible = true;
            ListBoxAll.Items.Clear();
            ListBoxAll.Items.Add("Administrators");

            foreach (Administrator a in adminlist)
            {
                ListBoxAll.Items.Add(a.ToString());
            }

            ListBoxAll.Items.Add("");
            ListBoxAll.Items.Add("Coaches");

            foreach (Coach c in coachlist)
            {
                ListBoxAll.Items.Add(c.ToString());
            }
        }

        protected void ButtonHide_Click(object sender, EventArgs e)
        {
            ListBoxDisplay.Visible = true;
            ListBoxAll.Visible = false;
            ButtonHide.Visible = false;
            ButtonShowAll.Visible = true;
            ButtonCreate.Enabled = true;
            ButtonRead.Enabled = true;
            ButtonDelete.Enabled = true;
            ButtonSave.Enabled = true;

        }
    }
}
```

Create Administrator

The page will create an administrator and after clicking, submit, the user will return to the display administrator's page. The solution uses auto numbering of new administrators.

CreateAdmin.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

using System.Collections;

namespace HandIn3CS
```

```
{
    public partial class CreateAdmin : System.Web.UI.Page
    {
        ArrayList adminlist;
        int userlevel;
        int staffnumber;
        string role;

        protected void Page_Load(object sender, EventArgs e)
        {
            Response.Cache.SetNoStore();

            try
            {
                userlevel = (int)Session["Level"];
                staffnumber = (int)Session["StaffNumber"];
                role = (string)Session["Role"];
            }
            catch (NullReferenceException nre)
            {
                userlevel = 0;
                Session["Level"] = userlevel;
                LabelMessage.Text = nre.Message;
            }
            finally
            {
                SetLevelOrError();
            }

            adminlist = (ArrayList)Application["myadminlist"];

            int maxstaffnumber = 0;

            if (adminlist == null)
            {
                adminlist = new ArrayList();
                maxstaffnumber = maxstaffnumber + 1;
                TextBoxStaffNumber.Text = maxstaffnumber.ToString();
            }
            else
            {
                foreach (Administrator a in adminlist)
                {
                    if (a.StaffNumber > maxstaffnumber)
                    {
                        maxstaffnumber = a.StaffNumber;
                    }
                }
                maxstaffnumber = maxstaffnumber + 1;
                TextBoxStaffNumber.Text = maxstaffnumber.ToString();
            }
        }

        protected void ButtonCreate_Click(object sender, EventArgs e)
        {
            //create admin
            Administrator a = new Administrator((Convert.ToInt32(TextBoxStaffNumber.Text)),
TextBoxName.Text, TextBoxGender.Text, TextBoxPassword.Text, TextBoxEmail.Text,
"Administrator", (Convert.ToInt32(TextBoxLevel.Text)));
```

```
            string email = a.Email;

            if (a.ChangeEmail(email) == "ok")
            {
                //add to coachlist
                adminlist.Add(a);
                Application.Set("myadminlist", adminlist);


                LabelMessage.Text = "New administrator added";

                TextBoxStaffNumber.Text = "";
                TextBoxName.Text = "";
                TextBoxGender.Text = "";
                TextBoxPassword.Text = "";
                TextBoxEmail.Text = "";
                TextBoxLevel.Text = "";

                Response.Redirect("DisplayAdmin.aspx");
            }
            else
            {
                LabelMessage.Text = "Error in email format";
            }
        }

        protected void ButtonCancel_Click(object sender, EventArgs e)
        {
            Response.Redirect("DisplayAdmin.aspx");
        }

        private void SetLevelOrError()
        {
            if (userlevel == 1)
            {
                LabelUserLevel.Text = "You are logged in as an administrator ";
            }
            else
            {
                Response.Redirect("ErrorPage.aspx");
            }
        }
    }
}
```

Update Administrators

The page updates administrators. Administrators can only update own user information.

UpdateAdmin.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

using System.Collections;
using System.Data;

namespace HandIn3CS
{
    public partial class UpdateAdmin : System.Web.UI.Page
    {
        ArrayList adminlist;
        int userlevel;
        int staffnumber;
        string role;

        protected void Page_Load(object sender, EventArgs e)
        {
            Response.Cache.SetNoStore();

            try
            {
                userlevel = (int)Session["Level"];
                staffnumber = (int)Session["StaffNumber"];
                role = (string)Session["Role"];
            }
            catch (NullReferenceException nre)
            {
                userlevel = 0;
                Session["Level"] = userlevel;
                LabelMessage.Text = nre.Message;
            }
            finally
            {
                SetLevelOrError();
            }

            if (Application["myadminlist"] == null)
            {
                adminlist = new ArrayList();
                Application["myadminlist"] = adminlist;
            }
            else
            {
                adminlist = (ArrayList)Application["myadminlist"];  // type casting
            }

            if (!Page.IsPostBack)
            {
                if (adminlist.Count == 0)
                {
                    LabelMessage.Text = "No Administrators";
                    ButtonUpdate.Enabled = false;
                }
                else
                {
                    ButtonUpdate.Enabled = false;
                    FillTextBoxes(adminlist);
                }
            }
        }

        protected void ButtonUpdate_Click(object sender, EventArgs e)
        {
            LabelMessage.Text = "";

            foreach (Administrator a in adminlist)
            {
                if (a.StaffNumber == staffnumber)
                {
                    string email = TextBoxEmail.Text;

                    if (a.ChangeEmail(email) == "ok")
                    {
                        a.StaffNumber = Convert.ToInt32(TextBoxStaffNumber.Text);
                        a.Name = TextBoxName.Text;
                        a.Gender = TextBoxGender.Text;
                        a.Password = TextBoxPassword.Text;
                        a.Email = TextBoxEmail.Text;
                        a.Level = Convert.ToInt32(TextBoxLevel.Text);
                        a.Role = "Administrator";
                        LabelMessage.Text = "Administrator updated";
                        Response.Redirect("DisplayAdmin.aspx");
                    }
                    else
                    {
                        LabelMessage.Text = "Error in email format";
                    }
                }
            }
        }

        protected void ButtonCancel_Click(object sender, EventArgs e)
        {
            Response.Redirect("DisplayAdmin.aspx");
        }

        private void SetLevelOrError()
        {
            if (userlevel == 1)
            {
                LabelUserLevel.Text = "You are logged in as an administrator ";
            }
            else
            {
                Response.Redirect("ErrorPage.aspx");
            }
        }

        private void FillTextBoxes(ArrayList adminlist)
        {
            foreach (Administrator a in adminlist)
            {
                if (a.StaffNumber == staffnumber)
                {
                    TextBoxStaffNumber.Text = a.StaffNumber.ToString();
                    TextBoxName.Text = a.Name;
                    TextBoxGender.Text = a.Gender;
                    TextBoxPassword.Text = a.Password;
                    TextBoxEmail.Text = a.Email;
                    TextBoxLevel.Text = a.Level.ToString();
                    LabelMessage.Text = "";
                    ButtonUpdate.Enabled = true;
                }
            }
        }
    }
}
```

Display Coaches

The page displays coaches. The user also have the possibility to delete coaches. Buttons update and create will direct the user to either the update or create page. Save button will save the changes made by the user. A coach can only update own user information.

DisplayCoach.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

using System.Collections;
using System.Data;

namespace HandIn3CS
{
    public partial class DisplayCoach : System.Web.UI.Page
    {
        ArrayList coachlist;
        int index;
        int userlevel;
        int staffnumber;
        string role;

        protected void Page_Load(object sender, EventArgs e)
        {
            Response.Cache.SetNoStore();

            try
            {
                userlevel = (int)Session["Level"];
                staffnumber = (int)Session["StaffNumber"];
                role = (string)Session["Role"];
            }
            catch (NullReferenceException nre)
            {
                userlevel = 0;
                Session["Level"] = userlevel;
                LabelMessage.Text = nre.Message;
            }
            finally
            {
                SetLevelOrError();
            }

            if (Application["mycoachlist"] == null)
            {
                coachlist = new ArrayList();
                Application["mycoachlist"] = coachlist;
                ButtonUpdate.Enabled = false;
                ButtonDelete.Enabled = false;
                ButtonSave.Enabled = false;
            }
            else
            {
                coachlist = (ArrayList)Application["mycoachlist"];  // type casting
                ButtonUpdate.Enabled = true;
                ButtonDelete.Enabled = true;
                ButtonSave.Enabled = true;
            }

            if (!Page.IsPostBack)
            {
                if (coachlist.Count == 0)
                {
                    LabelMessage.Text = "No coaches";
                }
                else
                {
                    PrintCoach(coachlist);
                    if (role == "Administrator")
                    {
                        // administrators can only update at administrator page
                        ButtonUpdate.Visible = false;
                    }
                }
            }
        }

        protected void ButtonRead_Click(object sender, EventArgs e)
        {
            try
            {
                coachlist = 
FileUtility.ReadFile(Server.MapPath("~/App_Data/Coachfile.ser"));
                Application["mycoachlist"] = coachlist;

                LabelMessage.Text = "File read from App_Data";

                if (coachlist.Count == 0)
                {
                    LabelMessage.Text = "No Coaches in file";
                }
                else
                {
                    PrintCoach(coachlist);
                }
            }
            catch (Exception ex)
            {
                LabelMessage.Text = "Coachfile could not be read" + ex.Message;
            }
        }

        protected void ListBoxDisplay_SelectedIndexChanged(object sender, EventArgs e)
        {
            try
            {
                Application.Set("mycoachlist", coachlist);
                index = ListBoxDisplay.SelectedIndex;
                Session["Index"] = index;
            }
            catch
            {
                //TextBoxInfo.Text = "Choose an employee";
            }
        }

        protected void ButtonCreate_Click(object sender, EventArgs e)
        {
            Response.Redirect("CreateCoach.aspx");
        }

        protected void ButtonSave_Click(object sender, EventArgs e)
        {
            coachlist = (ArrayList)Application["mycoachlist"];

            if (coachlist != null)
            {
                FileUtility.WriteFile(coachlist,
Server.MapPath("~/App_Data/Coachfile.ser"));
                LabelMessage.Text = "File written in App_Data folder";
            }
            else
            {
                LabelMessage.Text = "No coaches to save";
            }
        }

        protected void ButtonDelete_Click(object sender, EventArgs e)
        {
            if (ListBoxDisplay.SelectedIndex == -1)
            {
                LabelMessage.Text = "Please select a coach from the list";
            }
            else
            {
                for (int i = 0; i < coachlist.Count; i++)
                {
                    if (ListBoxDisplay.SelectedIndex == i)
                    {
                        coachlist.RemoveAt(i);
                        LabelMessage.Text = "Selected coach has been deleted";
                    }
                }
            }

            PrintCoach(coachlist);
        }

        protected void ButtonUpdate_Click(object sender, EventArgs e)
        {
            Response.Redirect("UpdateCoach.aspx");
        }

        private void PrintCoach(ArrayList coachlist)
        {
            ListBoxDisplay.Items.Clear();

            foreach (Coach c in coachlist)
            {
                ListBoxDisplay.Items.Add(c.ToString());
            }
        }

        private void SetLevelOrError()
        {
            if (userlevel == 1)
            {
                LabelUserLevel.Text = "You are logged in as an administrator ";
            }
            else if (userlevel == 2)
            {
                LabelUserLevel.Text = "You are logged in as a coach ";
            }
            else
            {
                Response.Redirect("ErrorPage.aspx");
            }
        }
    }
}
```

Create Coach

The page will create a coach and after clicking, submit, the user will return to the display administrator's page.

CreateCoach.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

using System.Collections;

namespace HandIn3CS
{
    public partial class CreateCoach : System.Web.UI.Page
    {
        ArrayList coachlist;
        int userlevel;
        int staffnumber;
        string role;

        protected void Page_Load(object sender, EventArgs e)
        {
            Response.Cache.SetNoStore();

            try
            {
                userlevel = (int)Session["Level"];
                staffnumber = (int)Session["StaffNumber"];
                role = (string)Session["Role"];
            }
            catch (NullReferenceException nre)
            {
                userlevel = 0;
                Session["Level"] = userlevel;
                LabelMessage.Text = nre.Message;
            }
            finally
            {
                SetLevelOrError();
            }

            coachlist = (ArrayList)Application["mycoachlist"];

            int maxstaffnumber = 100;

            if (coachlist == null)
            {
                coachlist = new ArrayList();
                maxstaffnumber = maxstaffnumber + 1;
                TextBoxStaffNumber.Text = maxstaffnumber.ToString();
            }
            else
            {
                foreach (Coach c in coachlist)
                {
                    if (c.StaffNumber > maxstaffnumber)
                    {
                        maxstaffnumber = c.StaffNumber;
                    }
                }
                maxstaffnumber = maxstaffnumber + 1;
                TextBoxStaffNumber.Text = maxstaffnumber.ToString();
            }
        }

        protected void ButtonCreate_Click(object sender, EventArgs e)
        {


            //create coach
            Coach c = new Coach((Convert.ToInt32(TextBoxStaffNumber.Text)),
TextBoxName.Text, TextBoxGender.Text, TextBoxPassword.Text, TextBoxEmail.Text, "Coach",
TextBoxEvent.Text);
            string email = c.Email;

            if (email.Trim().EndsWith("@fitness.dk"))
            {
                //add to coachlist
                coachlist.Add(c);
                Application.Set("mycoachlist", coachlist);

                TextBoxStaffNumber.Text = "";
                TextBoxName.Text = "";
                TextBoxGender.Text = "";
                TextBoxPassword.Text = "";
                TextBoxEmail.Text = "";
                TextBoxEvent.Text = "";

                Response.Redirect("DisplayCoach.aspx");
            }
            else
            {
                LabelMessage.Text = "Error in email format";
            }
        }

        protected void ButtonCancel_Click(object sender, EventArgs e)
        {
            Response.Redirect("DisplayCoach.aspx");
        }

        private void SetLevelOrError()
        {
            if (userlevel == 1)
            {
                LabelUserLevel.Text = "You are logged in as an administrator ";
            }
            else if (userlevel == 2)
            {
                LabelUserLevel.Text = "You are logged in as a coach ";
            }
            else
            {
                Response.Redirect("ErrorPage.aspx");
            }
        }
    }
}
```

Update Coach

The page updates coaches. Coaches can only update own user information.

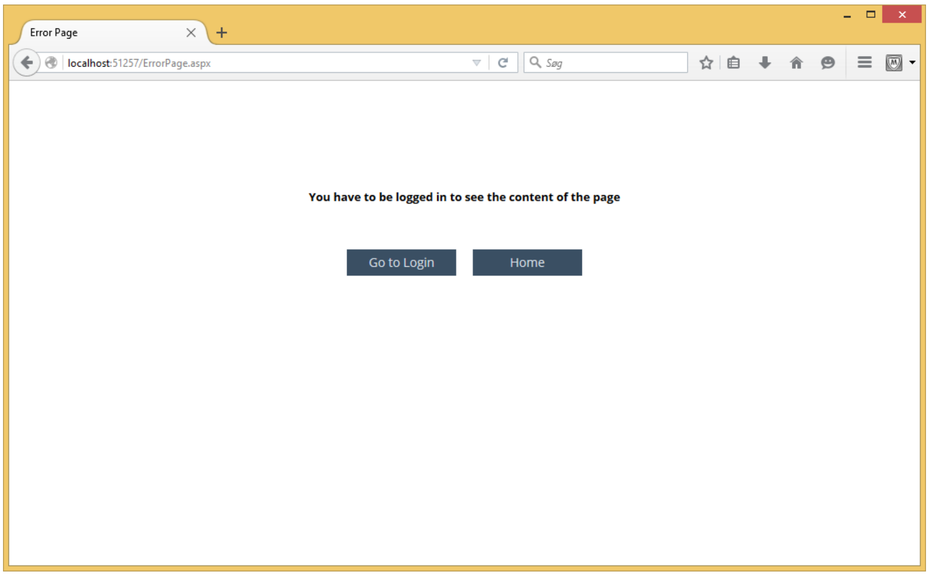
UpdateCoach.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Collections;

namespace HandIn3CS
{
    public partial class UpdateCoach : System.Web.UI.Page
    {
        ArrayList coachlist;
        int userlevel;
        int staffnumber;
        string role;

        protected void Page_Load(object sender, EventArgs e)
        {
            try
            {
                userlevel = (int)Session["Level"];
                staffnumber = (int)Session["StaffNumber"];
                role = (string)Session["Role"];
            }
            catch (NullReferenceException nre)
            {
                userlevel = 0;
                Session["Level"] = userlevel;
                LabelMessage.Text = nre.Message;
            }
            finally
            {
                SetLevelOrError();
            }

            if (Application["mycoachlist"] == null)
            {
                coachlist = new ArrayList();
                Application["mycoachlist"] = coachlist;
            }
            else
            {
                coachlist = (ArrayList)Application["mycoachlist"];  // type casting
            }

            if (!Page.IsPostBack)
            {
                if (coachlist.Count == 0)
                {
                    LabelMessage.Text = "No Coaches";
                    ButtonUpdate.Enabled = false;
                }
                else
                {
                    ButtonUpdate.Enabled = false;
                    FillTextBoxes(coachlist);
                }
            }
        }

        protected void ButtonUpdate_Click(object sender, EventArgs e)
        {
            LabelMessage.Text = "";

            foreach (Coach c in coachlist)
            {
                if (c.StaffNumber == staffnumber)
                {
                    string email = TextBoxEmail.Text;

                    if (email.Trim().EndsWith("@fitness.dk"))
                    {
                        c.StaffNumber = Convert.ToInt32(TextBoxStaffNumber.Text);
                        c.Name = TextBoxName.Text;
                        c.Gender = TextBoxGender.Text;
                        c.Password = TextBoxPassword.Text;
                        c.Email = TextBoxEmail.Text;
                        c.EventType = TextBoxEvent.Text;
                        c.Role = "Coach";
                        LabelMessage.Text = "Coach updated";
                        Response.Redirect("DisplayCoach.aspx");
                    }
                    else
                    {
                        LabelMessage.Text = "Error in email format";
                    }
                }
            }
        }

        protected void ButtonCancel_Click(object sender, EventArgs e)
        {
            Response.Redirect("DisplayCoach.aspx");
        }

        private void SetLevelOrError()
        {
            if (userlevel == 2)
            {
                LabelUserLevel.Text = "You are logged in as an coach ";
            }
            else
            {
                Response.Redirect("ErrorPage.aspx");
            }
        }

        private void FillTextBoxes(ArrayList adminlist)
        {
            foreach (Coach c in adminlist)
            {
                if (c.StaffNumber == staffnumber)
                {
                    TextBoxStaffNumber.Text = c.StaffNumber.ToString();
                    TextBoxName.Text = c.Name;
                    TextBoxGender.Text = c.Gender;
                    TextBoxPassword.Text = c.Password;
                    TextBoxEmail.Text = c.Email;
                    TextBoxEvent.Text = c.EventType.ToString();
                    LabelMessage.Text = "";
                    ButtonUpdate.Enabled = true;
                }
            }
        }
    }
```

```
}
```

Error Page

If a user without being logged in tries to access any of the subpages, the user will be directed to the error page.



You have to be logged in to see the content of the page

Error.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace HandIn3CS
{
    public partial class ErrorPage : System.Web.UI.Page
    {
        static string prevPage = String.Empty;

        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                //prevPage = Request.UrlReferrer.ToString();
            }
        }

        protected void ButtonReturn_Click(object sender, EventArgs e)
        {
            Response.Redirect("Login.aspx");
        }
        protected void ButtonBack_Click(object sender, EventArgs e)
        {
            Response.Redirect("Welcome.aspx");
        }
    }
}
```

Master Page

```
<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site.master.cs"
Inherits="HandIn3CS.Site" %>

<!DOCTYPE html>

<html>
<head runat="server">
    <title>HandIn 3</title>
    <link href="~/app_Themes/ThemeMyFitness/MyFitnessStyleSheet.css" rel="stylesheet"
type="text/css" />
    <link href="Content/bootstrap.min.css" rel="stylesheet" />
    <asp:ContentPlaceHolder ID="head" runat="server">
    </asp:ContentPlaceHolder>
</head>
<body>
    <form id="form1" runat="server">
            <div class="myMenuBar">
                <asp:Menu ID="NavigationMenu" runat="server" CssClass="menu"
EnableViewState="false" IncludeStyleBlock="false" Orientation="Horizontal">
                    <Items>
                        <asp:MenuItem NavigateUrl="~/Welcome.aspx" Text="HOME"
Value="HOME"></asp:MenuItem>
                        <asp:MenuItem Text="COACH" NavigateUrl="~/DisplayCoach.aspx"
Value="COACH">
                        </asp:MenuItem>
                        <asp:MenuItem Text="ADMINISTRATOR" Value="ADMINISTRATOR"
NavigateUrl="~/DisplayAdmin.aspx"></asp:MenuItem>
                        <asp:MenuItem NavigateUrl="~/Login.aspx" Text="LOGIN"
Value="LOGIN"></asp:MenuItem>
                    </Items>
                </asp:Menu>
            </div>
            <br />
        <asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">
        </asp:ContentPlaceHolder>
        <br />
        <div class="footer">
        <div class="row">
            <div class="col-sm-6 col-md-3">
                <h4>MyFitnessCenter</h4>
                <ul>
                    <li><a>About us</a></li>
                    <li><a>News</a></li>
                    <li><a>Rules</a></li>
                </ul>
            </div>
```

```
                <div class="col-sm-6 col-md-3">
                    <h4>Help</h4>
                    <ul>
                        <li><a>Contact us</a></li>
                        <li><a>FAQ</a></li>
                        <li><a>Newsletter</a></li>
                        <li><a>Opening hours</a></li>
                    </ul>
                </div>
                <div class="col-sm-6 col-md-3">
                    <h4>Career</h4>
                    <ul>
                        <li><a>Jobs</a></li>
                        <li><a>Education</a></li>
                    </ul>
                </div>
                <div class="col-sm-6 col-md-3">
                    <h4>Address</h4>
                        <ul>
                            <li><a>MyFitnessCenter</a></li>
                            <li><a>Fitness Road 206</a></li>
                            <li><a>2800 Lyngby</a></li>
                            <li><a>CVR. 1616179</a></li>
                        </ul>
                </div>
            </div>
            </div>
        </form>
        <script src="Scripts/bootstrap.js" ></script>
        <script src="Scripts/jquery-2.1.4.js" ></script>
</body>
</html>
```

Themes

Both CSS and Skin files are used

MyFitnessStyleSheet.css

```
body
{
    margin-left:20px; margin-top:10px; margin-right:20px; margin-bottom:10px; line-height:
1.4;
}

.headertext
{
    font-family:'Open Sans', sans-serif; font-size:24px; text-align:center; font-weight:
bold;
}

.toptext
{
    font-family:'Open Sans', sans-serif; font-size:16px; font-weight: bold; margin-left:
5px;
}

.footertext
{
    font-family: 'Open Sans', sans-serif; font-size: 14px; text-align: center;
}

.myTable
{
    text-align: left; width: 80%;
}

.myTableContCol1
{
    width: 150px; height: 35px;
}

.myTableContCol2
{
    width: 200px; height: 35px;
}

.myUserlevel
{
    font-family: 'Open Sans'; font-size: 12px; text-align: right;
}

.myMenuBar
{
    background-color: #3a4f63; width: 100%; font-family: Verdana; font-size: 14px; text-
align: left;
}

.menu
{
    background-color: #3a4f63;
}

.menu ul
{
    list-style: none;
    margin: 3px 3px 3px 3px;
    padding: 3px 3px 3px 3px;
    width: auto;
}

.menu ul li a
{
    background-color: #3a4f63;
    border: 1px #4e667d solid;
    color: #dde4ec;
    display: block;
    line-height: 1.25em;
    padding: 3px 20px;
    text-decoration: none;
    white-space: nowrap;
}
```

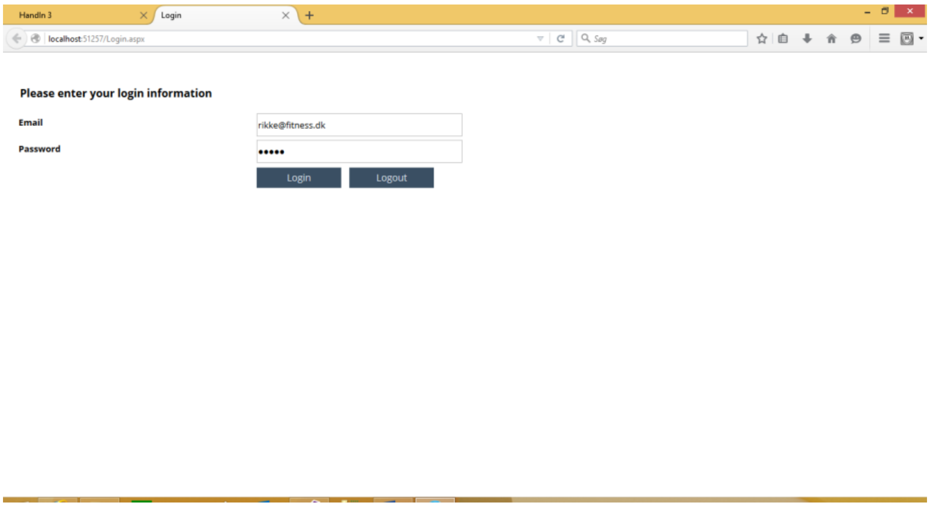
```
.menu ul li a:visited
{
    background-color: #3a4f63;
    border: 1px #4e667d solid;
    color: #dde4ec;
    display: block;
    line-height: 1.25em;
    padding: 3px 20px;
    text-decoration: none;
    white-space: nowrap;
}

.menu ul li a:hover
{
    background-color: #bfcbd6; color: #465c71; text-decoration: none;
}

.menu ul li a:active
{
    background-color: #465c71; color: #cfdbe6; text-decoration: none;
}

.footer a, .footer p
{
    color: #999;
}

.footer
{
    background-color: #262626;
}

.footer h4
{
    color: #fff;
    font-size: 16px;
    letter-spacing: 0.4px;
    font-family: 'Open Sans', sans-serif;
    font-size: 18px;
    font-weight: 700;
    padding-top: 3px;
    padding-right: 3px;
    padding-bottom: 3px;
    padding-left: 3px;
    margin-top: 2px;
    margin-right: 0px;
    margin-bottom: 2px;
    margin-left: 30px;
}

.footer li
{
    list-style-type: none;
    padding-top: 3px;
    padding-right: 3px;
    padding-bottom: 3px;
    padding-left: 0px;
    text-align: left;
    font-size: 14px;
}
```

```css
.text-center
{
    text-align: center;
}
.row
{
    margin-left: 3px; margin-right: 3px;
}

.GridViewStyle
{
    border:1px solid #BFCBD6; border-collapse:collapse; font-family:'Open Sans', sans-serif;
table-layout :auto; font-size: 14px;
}

/*Header*/
.HeaderStyle
{
    border:1px solid #ddd; background-color: #3a4f63; color: #dde4ec;
}

.HeaderStyle th
{
    padding:3px 3px 3px 3px; color: #cfdbe6; text-align: center;
}

/*Row*/
.RowStyle
{
    text-align:center; background-color:#fff;
}

.RowStyle td
{
    padding:3px 3px 3px 3px; border-color: #BFCBD6;
}

.RowStyle:hover
{
    cursor:pointer; background-color:#f69542;
}

.AlternatingRowStyle
{
    text-align: center; background-color:#bfcbd6;
}

.AlternatingRowStyle td
{
    padding:3px 3px 3px 3px; border-color: #BFCBD6;
}

.AlternatingRowStyle:hover
{
    cursor: pointer; background-color: #f69542;
}
```

Screen Dumps

Some selected screen dumps from the login, display, update, delete and create functionally.

Login Page

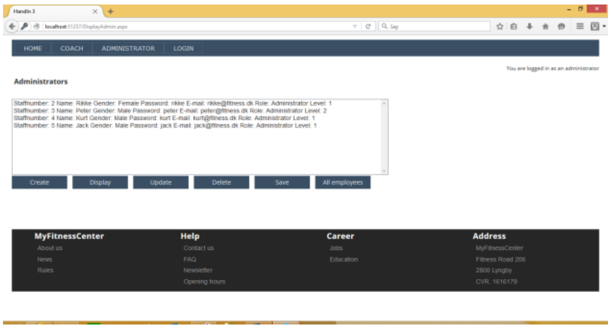


Please enter your login information

Email: rikke@fitness.dk

Password: •••••

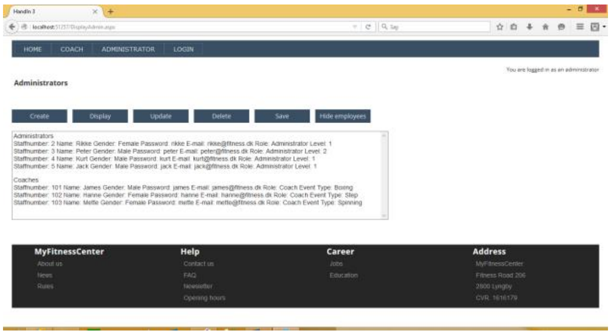
Display Administrator



You are logged in as an administrator

Administrators

```
Staffnumber: 2 Name: Rikke Gender: Female Password: rikke E-mail: rikke@fitness.dk Role: Administrator Level: 1
Staffnumber: 3 Name: Peter Gender: Male Password: peter E-mail: peter@fitness.dk Role: Administrator Level: 2
Staffnumber: 4 Name: Kurt Gender: Male Password: kurt E-mail: kurt@fitness.dk Role: Administrator Level: 1
Staffnumber: 5 Name: Jack Gender: Male Password: jack E-mail: jack@fitness.dk Role: Administrator Level: 1
```

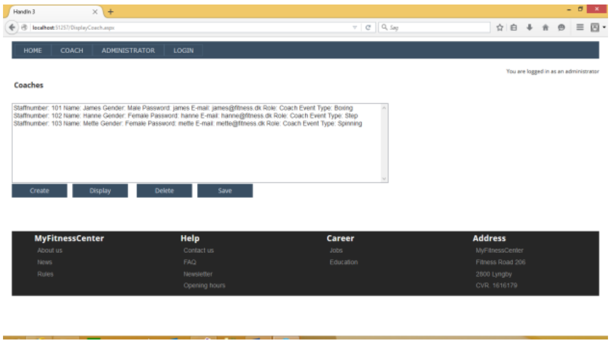


You are logged in as an administrator

Administrators

```
Administrators
Staffnumber: 2 Name: Rikke Gender: Female Password: rikke E-mail: rikke@fitness.dk Role: Administrator Level: 1
Staffnumber: 3 Name: Peter Gender: Male Password: peter E-mail: peter@fitness.dk Role: Administrator Level: 2
Staffnumber: 4 Name: Kurt Gender: Male Password: kurt E-mail: kurt@fitness.dk Role: Administrator Level: 1
Staffnumber: 5 Name: Jack Gender: Male Password: jack E-mail: jack@fitness.dk Role: Administrator Level: 1

Coaches
Staffnumber: 101 Name: James Gender: Male Password: james E-mail: james@fitness.dk Role: Coach Event Type: Boxing
Staffnumber: 102 Name: Hanne Gender: Female Password: hanne E-mail: hanne@fitness.dk Role: Coach Event Type: Step
Staffnumber: 103 Name: Mette Gender: Female Password: mette E-mail: mette@fitness.dk Role: Coach Event Type: Spinning
```

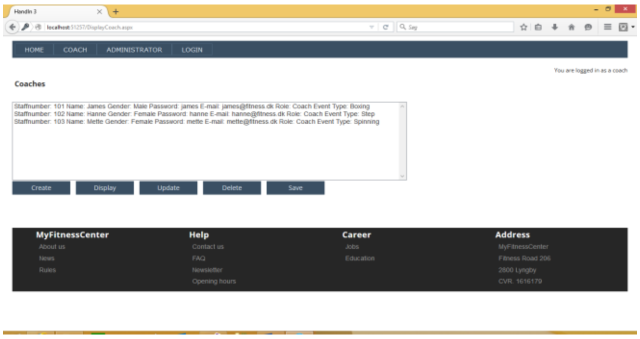
Display Coaches



You are logged in as an administrator

Coaches

```
Staffnumber: 101 Name: James Gender: Male Password: james E-mail: james@fitness.dk Role: Coach Event Type: Boxing
Staffnumber: 102 Name: Hanne Gender: Female Password: hanne E-mail: hanne@fitness.dk Role: Coach Event Type: Step
Staffnumber: 103 Name: Mette Gender: Female Password: mette E-mail: mette@fitness.dk Role: Coach Event Type: Spinning
```

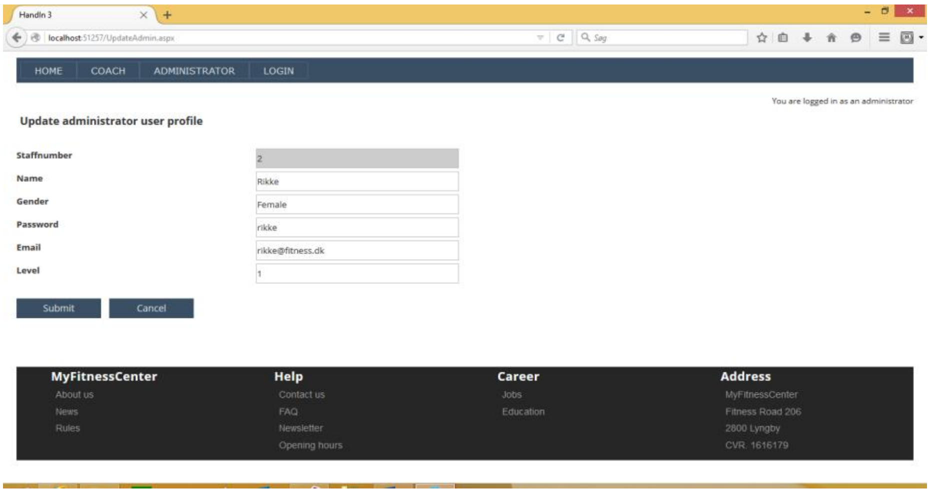


You are logged in as a coach

Coaches

```
Staffnumber: 101 Name: James Gender: Male Password: james E-mail: james@fitness.dk Role: Coach Event Type: Boxing
Staffnumber: 102 Name: Hanne Gender: Female Password: hanne E-mail: hanne@fitness.dk Role: Coach Event Type: Step
Staffnumber: 103 Name: Mette Gender: Female Password: mette E-mail: mette@fitness.dk Role: Coach Event Type: Spinning
```

Update Administrator



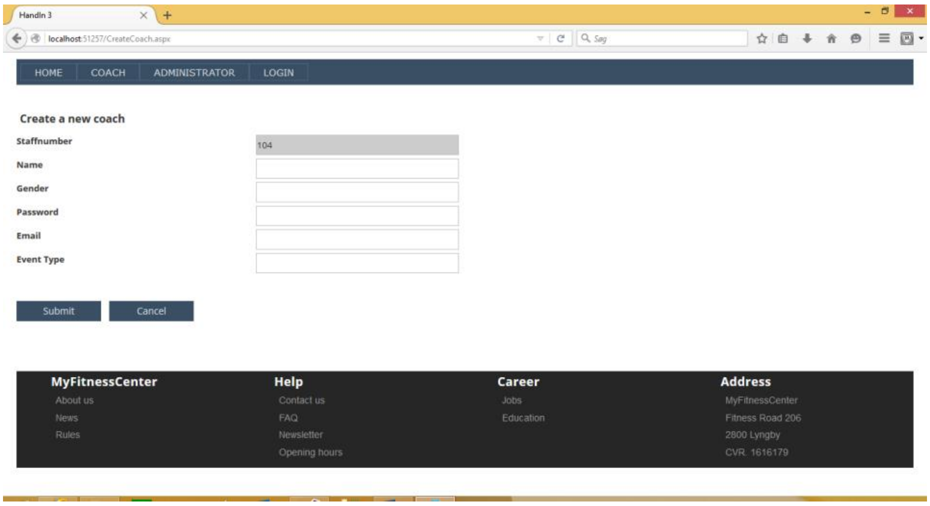
You are logged in as an administrator

Update administrator user profile

Staffnumber	2
Name	Rikke
Gender	Female
Password	rikke
Email	rikke@fitness.dk
Level	1

MyFitnessCenter	**Help**	**Career**	**Address**
About us	Contact us	Jobs	MyFitnessCenter
News	FAQ	Education	Fitness Road 206
Rules	Newsletter		2800 Lyngby
	Opening hours		CVR. 1616179

Create coaches

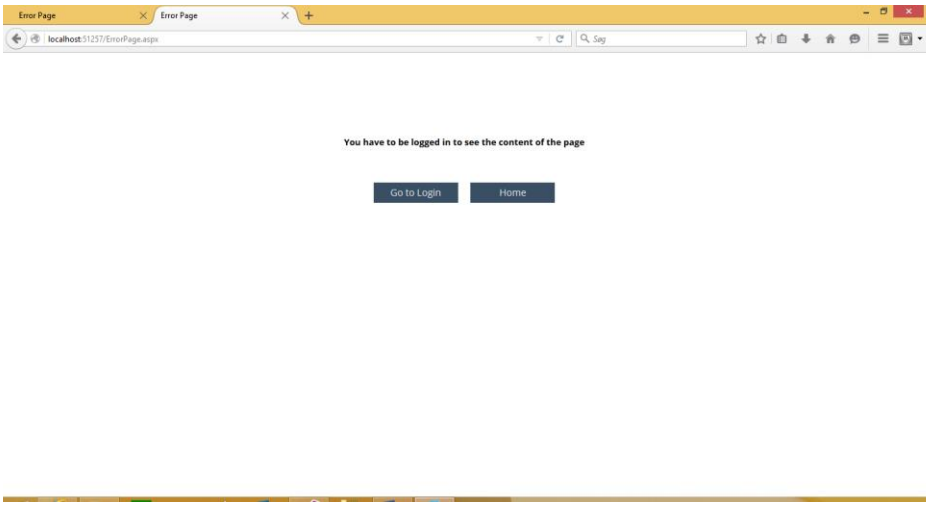


Create a new coach

Staffnumber	104
Name	
Gender	
Password	
Email	
Event Type	

MyFitnessCenter	**Help**	**Career**	**Address**
About us	Contact us	Jobs	MyFitnessCenter
News	FAQ	Education	Fitness Road 206
Rules	Newsletter		2800 Lyngby
	Opening hours		CVR. 1616179

Error page



You have to be logged in to see the content of the page