

```

using Microsoft.VisualStudio.TestTools.UnitTesting;
using CupPlaner.Controllers;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Web.Mvc;

namespace CupPlaner.Controllers.Tests
{
    [TestClass()]
    public class DivisionControllerTests
    {
        DivisionController controller = new DivisionController();

        [TestMethod()]
        public void CreateTest()
        {
            //Create a new division
            dynamic jsonResult = ((JsonResult)controller.Create("U20 Drenge",
ID.TournamentId, 60, FieldSize.EightMan)).Data;
            ID.DivisionId = jsonResult.id;
            Assert.AreEqual("success", jsonResult.status);

            //Create a new division, using null values
            jsonResult = ((JsonResult)controller.Create(null, ID.TournamentId, 60,
FieldSize.EightMan)).Data;
            Assert.AreEqual("error", jsonResult.status);

            //Create a new division, but to a non-existing tournament
            jsonResult = ((JsonResult)controller.Create("U20 Drenge", 999999, 60,
FieldSize.EightMan)).Data;
            Assert.AreEqual("error", jsonResult.status);
        }

        [TestMethod()]
        public void DetailsTest()
        {
            //Find the created division
            dynamic jsonResult = ((JsonResult)controller.Details(ID.DivisionId)).Data;
            Assert.AreEqual("success", jsonResult.status);
            Assert.AreEqual(ID.DivisionId, jsonResult.Id);
            Assert.AreEqual("U20 Drenge", jsonResult.Name);
            Assert.AreEqual(FieldSize.EightMan, jsonResult.FieldSize);
            Assert.AreEqual(60, jsonResult.MatchDuration);

            //Find a division that does not exist
            jsonResult = ((JsonResult)controller.Details(999999)).Data;
            Assert.AreEqual("error", jsonResult.status);
        }

        [TestMethod()]
        public void EditTest()
        {
            //Edit the created division
            dynamic jsonResult = ((JsonResult)controller.Edit(ID.DivisionId, "U21
Drenge", ID.TournamentId, 11, 75)).Data;
            Assert.AreEqual("success", jsonResult.status);
        }
    }
}

```

```

        //Check to see if edits have been saved
        jsonResult = ((JsonResult)controller.Details(ID.DivisionId)).Data;
        Assert.AreEqual("U21 Drenge", jsonResult.Name);
        Assert.AreEqual(FieldSize.ElevenMan, jsonResult.FieldSize);
        Assert.AreEqual(75, jsonResult.MatchDuration);

        //Edit a division using null values
        jsonResult = ((JsonResult)controller.Edit(ID.DivisionId, null,
ID.TournamentId, 11, 75)).Data;
        Assert.AreEqual("error", jsonResult.status);

        //Edit a division that does not exist
        jsonResult = ((JsonResult)controller.Edit(999999, "U21 Drenge",
ID.TournamentId, 11, 75)).Data;
        Assert.AreEqual("error", jsonResult.status);
    }

    [TestMethod()]
    public void DeleteTest()
    {
        //Delete the created division
        dynamic jsonResult = ((JsonResult)controller.Delete(ID.DivisionId)).Data;
        Assert.AreEqual("success", jsonResult.status);

        //Delete a division that does not exist
        jsonResult = ((JsonResult)controller.Delete(999999)).Data;
        Assert.AreEqual("error", jsonResult.status);
    }
}

```