

HomeController.cs:

```
HomeController.cs
1 using Microsoft.AspNetCore.Mvc;
2
3 namespace SportsPro.Controllers
4 {
5     public class HomeController : Controller
6     {
7         // GET: /
8         public IActionResult Index()
9         {
10             return View();
11         }
12
13         // Attribute routing added for about page
14         // uses [action] token instead of hard coding
15         [Route("action")]
16         public IActionResult About()
17         {
18             return View();
19         }
20     }
21 }
```

ProductController.cs:

```
ProductController.cs
1 using System.Collections.Generic;
2 using Microsoft.AspNetCore.Mvc;
3 using SportsPro.Models;
4
5 namespace SportsPro.Controllers
6 {
7     public class ProductController : Controller
8     {
9         private IRepository<Product> data { get; set; }
10
11         // GET: /
12         public ProductController(IRepository<Product> rep)
13         {
14             data = rep;
15         }
16
17         [Route("[controller]")]
18         public IActionResult List()
19         {
20             var products = this.data.List(new QueryOptions<Product>
21             {
22                 OrderBy = p => p.ReleaseDate
23             });
24             return View(products);
25         }
26
27         [HttpGet]
28         public IActionResult Add()
29         {
30             ViewBag.Action = "Add";
31             return View("AddEdit", new Product());
32         }
33
34         [HttpGet]
35         public IActionResult Edit(int id)
36         {
37             ViewBag.Action = "Edit";
38             var product = data.Get(id);
39             return View("AddEdit", product);
40         }
41
42         [HttpPost]
43         public IActionResult Edit(Product product)
44         {
45             string message;
46             if (ModelState.IsValid)
47             {
48                 if (product.ProductID == 0)
49                 {
50                     data.Insert(product);
51                     message = product.Name + " was added";
52                 }
53                 else
54                 {
55                     data.Update(product);
56                     message = product.Name + " was updated";
57                 }
58                 TempData["message"] = message;
59                 return RedirectToAction("List");
60             }
61             else
62             {
63                 if (product.ProductID == 0)
64                 {
65                     ViewBag.Action = "Add";
66                 }
67                 else
68                 {
69                     ViewBag.Action = "Edit";
70                 }
71                 return View(product);
72             }
73         }
74
75         [HttpGet]
76         public IActionResult Delete(int id)
77         {
78             var product = data.Get(id);
79             return View(product);
80         }
81
82         [HttpPost]
83         public IActionResult Delete(Product product)
84         {
85             var p = data.Get(product.ProductID);
86             data.Delete(p);
87             TempData["message"] = $"{p.Name} was deleted";
88             return RedirectToAction("List", "Product");
89         }
90     }
91 }
```

```
ProductController.cs
13
14 string message;
15 if (ModelState.IsValid)
16 {
17     if (product.ProductID == 0)
18     {
19         data.Insert(product);
20         message = product.Name + " was added";
21     }
22     else
23     {
24         data.Update(product);
25         message = product.Name + " was updated";
26     }
27     TempData["message"] = message;
28     return RedirectToAction("List");
29 }
30 else
31 {
32     if (product.ProductID == 0)
33     {
34         ViewBag.Action = "Add";
35     }
36     else
37     {
38         ViewBag.Action = "Edit";
39     }
40     return View(product);
41 }
42
43 [HttpGet]
44 public IActionResult Delete(int id)
45 {
46     var product = data.Get(id);
47     return View(product);
48 }
49
50 [HttpPost]
51 public IActionResult Delete(Product product)
52 {
53     var p = data.Get(product.ProductID);
54     data.Delete(p);
55     TempData["message"] = $"{p.Name} was deleted";
56     return RedirectToAction("List", "Product");
57 }
58 }
```

TechnicianController.cs:

```
1 using Microsoft.AspNetCore.Mvc;
2 using SportsPro.Models;
3 namespace SportsPro.Controllers
4 {
5     [reference]
6     public class TechnicianController : Controller
7     {
8         [reference]
9         private IRepository<Technician> data { get; set; }
10
11         [reference]
12         public TechnicianController(IRepository<Technician> rep)
13         {
14             data = rep;
15         }
16
17         [Route("[controller]s")]
18         [reference]
19         public IActionResult List()
20         {
21             ViewBag.Action = "List";
22             var techs = this.data.List(new QueryOptions<Technician>
23             {
24                 OrderBy = t => t.Name
25             });
26             return View(techs);
27         }
28
29         [HttpGet]
30         [reference]
31         public IActionResult Add()
32         {
33             ViewBag.Action = "Add";
34             return View("AddEdit", new Technician());
35         }
36
37         [HttpGet]
38         [reference]
39         public IActionResult Edit(int id)
40         {
41             ViewBag.Action = "Edit";
42             var technician = data.Get(id);
43             return View("AddEdit", technician);
44         }
45     }
46 }
```

```
47
48 [HttpPost]
49 [reference]
50 public IActionResult Edit(Technician technician)
51 {
52     if (ModelState.IsValid)
53     {
54         string message;
55         if (technician.TechnicianID == 0)
56         {
57             data.Insert(technician);
58             message = technician.Name + " was added";
59         }
60         else
61         {
62             data.Update(technician);
63             message = technician.Name + " was updated";
64         }
65         data.Save();
66         TempData["message"] = message;
67         return RedirectToAction("List", "Technician");
68     }
69     else
70     {
71         ViewBag.Action = (technician.TechnicianID == 0) ? "Add" : "AddEdit";
72         return View(technician);
73     }
74 }
75
76 [HttpGet]
77 [reference]
78 public IActionResult Delete(int id)
79 {
80     var technician = data.Get(id);
81     return View(technician);
82 }
83
84 [HttpPost]
85 [reference]
86 public IActionResult Delete(Technician technician)
87 {
88     var t = data.Get(technician.TechnicianID);
89     data.Delete(t);
90     data.Save();
91     TempData["message"] = $"{t.Name} was deleted";
92     return RedirectToAction("List", "Technician");
93 }
94
95 }
```

CustomerController.cs:

```
1 using System.Collections.Generic;
2 using Microsoft.AspNetCore.Mvc;
3 using SportsPro.Models;
4 namespace SportsPro.Controllers
5 {
6     [reference]
7     public class CustomerController : Controller
8     {
9         [reference]
10         private IRepository<Customer> data { get; set; }
11
12         [reference]
13         public CustomerController(IRepository<Customer> rep)
14         {
15             data = rep;
16         }
17
18         [Route("[controller]s")]
19         [reference]
20         public IActionResult List()
21         {
22             var customers = data.List(new QueryOptions<Customer>
23             {
24                 OrderBy = c => c.LastName
25             });
26             return View(customers);
27         }
28
29         [HttpGet]
30         [reference]
31         public IActionResult Add()
32         {
33             ViewBag.Action = "Add";
34             return View("AddEdit", new Customer());
35         }
36
37         [HttpGet]
38         [reference]
39         public IActionResult Edit(int id)
40         {
41             ViewBag.Action = "Edit";
42             var customer = data.Get(id);
43             return View("AddEdit", customer);
44         }
45     }
46 }
```

Richie Wertz Sports Pro Back-End

```
CustomerController.cs
SportsPro.Controllers.CustomerController
data

[HttpPost]
public IActionResult Edit(Customer customer)
{
    string message;
    if (ModelState.IsValid)
    {
        if (customer.CustomerID == 0)
        {
            data.Insert(customer);
            message = customer.FullName + " was added";
        }
        else
        {
            data.Update(customer);
            message = customer.FullName + " was updated";
        }
        data.Save();
        TempData["message"] = message;
        return RedirectToAction("List", "Customer");
    }
    else
    {
        ViewBag.Action = (customer.CustomerID == 0) ? "Add" : "Edit";
        return View(customer);
    }
}

[HttpGet]
public IActionResult Delete(int id)
{
    var customer = data.Get(id);
    return View(customer);
}

[HttpPost]
public IActionResult Delete(Customer customer)
{
    var c = data.Get(customer.CustomerID);
    data.Delete(c);
    TempData["message"] = $"{c.FullName} was deleted";
    return RedirectToAction("List", "Customer");
}
```

IncidentController.cs

```
IncidentController.cs
SportsPro.Controllers.IncidentController
data

using System.Collections.Generic;
using System.Linq;
using Microsoft.AspNetCore.Mvc;
using SportsPro.Models;

namespace SportsPro.Controllers
{
    public class IncidentController : Controller
    {
        private SportsProUnit data { get; set; }

        public IncidentController(SportsProContext ctx)
        {
            data = new SportsProUnit(ctx);
        }

        [Route("[controller]")]
        public IActionResult List(string filter = "all")
        {
            IncidentListViewModel model = new IncidentListViewModel
            {
                Filter = filter
            };

            IEnumerable<Incident> query = data.Incidents.List(new QueryOptions<Incident>
            {
                Includes = "Customer, Product",
                OrderBy = i => i.DateOpened,
            });

            if (filter == "unassigned")
            {
                query = query.Where(i => i.TechnicianID == null);
            }

            if (filter == "open")
            {
                query = query.Where(i => i.DateClosed == null);
            }

            List<Incident> incidents = query.ToList();
            model.Incidents = incidents;

            return View(model);
        }
    }
}
```

Richie Wertz

Sports Pro Back-End

```
IncidentController.cs - SportsPro.Controllers.IncidentController - data

51 private IncidentViewModel GetViewModel()
52 {
53     IncidentViewModel model = new IncidentViewModel()
54     {
55         Customers = data.Customers.List(new QueryOptions<Customer>
56         {
57             OrderBy = c => c.FirstName
58         }),
59         Products = data.Products.List(new QueryOptions<Product>
60         {
61             OrderBy = p => p.Name
62         }),
63         Technicians = data.Technicians.List(new QueryOptions<Technician>
64         {
65             OrderBy = t => t.Name
66         })
67     };
68     return model;
69 }
70
71 @routes
72 public ActionResult Filter(string id)
73 {
74     return RedirectToAction("List", new { Filter = id });
75 }
76
77 [HttpGet]
78 @routes
79 public ActionResult Add()
80 {
81     IncidentViewModel model = GetViewModel();
82     model.Incident = new Incident();
83     model.Action = "Add";
84     return View("AddEdit", model);
85 }
86
87 [HttpGet]
88 @routes
89 public ActionResult Edit(int id)
90 {
91     IncidentViewModel model = GetViewModel();
92     var incident = data.Incidents.Get(id);
93     model.Incident = incident;
94     model.Action = "Edit";
95     return View("AddEdit", model);
96 }
97
98
99
100
101
102
```

```
IncidentController.cs - SportsPro.Controllers.IncidentController - data

103 [HttpPost]
104 @routes
105 public ActionResult Edit(Incident incident)
106 {
107     IncidentViewModel model = GetViewModel();
108     if (incident.IncidentID == 0)
109     {
110         model.Action = "Add";
111     }
112     else
113     {
114         model.Action = "Edit";
115     }
116     if (ModelState.IsValid)
117     {
118         string message;
119         if (model.Action == "Add")
120         {
121             data.Incidents.Insert(incident);
122             message = incident.Title + " was added";
123         }
124         else
125         {
126             data.Incidents.Update(incident);
127             message = incident.Title + " was updated";
128         }
129         data.Save();
130         TempData["message"] = message;
131         return RedirectToAction("List");
132     }
133     else
134     {
135         model.Incident = incident;
136         return View("AddEdit", model);
137     }
138 }
139
140
141
142
```

```
IncidentController.cs - SportsPro.Controllers.IncidentController - data

143 [HttpPost]
144 @routes
145 public ActionResult Save(Incident incident)
146 {
147     if (ModelState.IsValid)
148     {
149         string message;
150         if (incident.IncidentID == 0)
151         {
152             data.Incidents.Insert(incident);
153             message = incident.Title + " was added";
154         }
155         else
156         {
157             data.Incidents.Update(incident);
158             message = incident.Title + " was updated";
159         }
160         data.Save();
161         TempData["message"] = message;
162         return RedirectToAction("List");
163     }
164     else
165     {
166         IncidentViewModel model = GetViewModel();
167         model.Incident = incident;
168         if (incident.IncidentID == 0)
169         {
170             model.Action = "Add";
171         }
172         else
173         {
174             model.Action = "Edit";
175         }
176         return View("AddEdit", model);
177     }
178 }
179
180 [HttpGet]
181 @routes
182 public ActionResult Delete(int id)
183 {
184     var incident = data.Incidents.Get(id);
185     return View(incident);
186 }
187
```

```
IncidentController.cs
[HttpPost]
public IActionResult Delete(incident incident)
{
    var i = data.Incidents.Get(incident.IncidentID);
    data.Incidents.Delete(i);
    data.Save();
    TempData["message"] = $"{i.Title} was deleted";
    return RedirectToAction("List");
}
```

RegistrationController.cs:

```
RegistrationController.cs
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Http;
using Microsoft.EntityFrameworkCore;
using SportsPro.Models;

namespace SportsPro.Controllers
{
    public class RegistrationController : Controller
    {
        private SportsProInit data { get; set; }

        public RegistrationController(SportsProContext ctx)
        {
            data = new SportsProInit(ctx);
        }

        [HttpGet]
        public IActionResult GetCustomer()
        {
            ViewBag.Customers = data.Customers.List(new QueryOptions<Customer>
            {
                OrderBy = c => c.LastName
            });

            int custID = HttpContext.Session.GetInt32("custID") ?? 0;
            Customer customer;
            if (custID == 0)
            {
                customer = new Customer();
            }
            else
            {
                customer = data.Customers.Get(custID);
            }

            return View(customer);
        }
    }
}
```

```
RegistrationController.cs
[HttpPost]
[Route("[controller]")]
public IActionResult List(Customer customer)
{
    HttpContext.Session.SetInt32("custID", customer.CustomerID);
    if (customer.CustomerID == 0)
    {
        TempData["message"] = "You must select a customer.";
        return RedirectToAction("GetCustomer");
    }
    else
    {
        return RedirectToAction("List", new { id = customer.CustomerID });
    }
}

[HttpGet]
[Route("[controller]")]
public IActionResult List(int id)
{
    RegistrationViewModel model = new RegistrationViewModel
    {
        CustomerID = id,
        Customer = data.Customers.Get(id),
        Products = data.Products.List(new QueryOptions<Product>
        {
            OrderBy = p => p.Name
        }),
        Registrations = data.Registrations.List(new QueryOptions<Registration>
        {
            Includes = "Customer, Product",
            Where = r => r.CustomerID == id
        })
    };
    return View(model);
}

[HttpPost]
public IActionResult Filter(int customerID = 0)
{
    return RedirectToAction("List", new { ID = customerID });
}
```

```
RegistrationController.cs
[HttpPost]
public IActionResult Register(RegistrationViewModel model)
{
    if (model.ProductID == 0)
    {
        TempData["message"] = "You must select a product.";
    }
    else
    {
        Registration registration = new Registration
        {
            CustomerID = model.CustomerID,
            ProductID = model.ProductID
        };
        data.Registrations.Insert(registration);
        var customer = data.Customers.Get(model.CustomerID);
        var product = data.Products.Get(model.ProductID);
        try
        {
            data.Save();
            TempData["message"] = $"{product.Name} was registered to {customer.FullName}";
        }
        catch (DbUpdateException ex)
        {
            string msg = (ex.InnerException == null) ? ex.Message : ex.InnerException.Message;
            if (msg.Contains("duplicate key"))
            {
                TempData["message"] = "This product is already registered to this customer";
            }
            else
            {
                TempData["message"] = "Error accessing database: " + msg;
            }
        }
        return RedirectToAction("List", new { ID = model.CustomerID });
    }
}
```

```
RegistrationController.cs
118 [HttpPost]
119 public IActionResult Delete(int customerId, int productId)
120 {
121     Registration registration = new Registration
122     {
123         CustomerID = customerId,
124         ProductID = productId
125     };
126     var customer = data.Customers.Get(customerID);
127     var product = data.Products.Get(productId);
128     data.Registrations.Delete(registration);
129     data.Save();
130     TempData["message"] = $"Registration {product.Name} was unregistered from {registration.Customer.FullName}";
131     return RedirectToAction("List", new { id = customerID });
132 }
133
134 }
135
136 }
```

TechIncidentController.cs:

```
TechIncidentController.cs
1 using System.Linq;
2 using Microsoft.AspNetCore.Mvc;
3 using Microsoft.EntityFrameworkCore;
4 using SportsPro.Models;
5 using SportsPro.Controllers;
6 namespace SportsPro.Controllers
7 {
8     [Route("api/techincidents")]
9     public class TechIncidentController : Controller
10     {
11         private SportsProInit data { get; set; }
12         public TechIncidentController(SportsProContext ctx)
13         {
14             data = new SportsProInit(ctx);
15         }
16         [HttpGet]
17         public IActionResult Get()
18         {
19             ViewBag.Technicians = data.Technicians.ToList(new QueryOptions<Technician> {
20                 OrderBy = c => c.Name
21             });
22             int? techID = HttpContext.Session.GetInt32("techID");
23             Technician technician;
24             if (techID == null)
25             {
26                 technician = new Technician();
27             }
28             else
29             {
30                 technician = data.Technicians.Get((int)techID);
31             }
32             return View(technician);
33         }
34     }
35 }
36
37 }
```

```
TechIncidentController.cs
38 [HttpPost]
39 public IActionResult List(Technician technician)
40 {
41     HttpContext.Session.SetInt32("techID", technician.TechnicianID);
42     if (technician.TechnicianID == 0)
43     {
44         TempData["message"] = "You must select a technician.";
45         return RedirectToAction("Get");
46     }
47     else
48     {
49         return RedirectToAction("List", new { id = technician.TechnicianID });
50     }
51 }
52
53 [HttpGet]
54 public IActionResult List(int id)
55 {
56     var model = new TechIncidentViewModel
57     {
58         Technician = data.Technicians.Get(id),
59         Incidents = data.Incidents.ToList(new QueryOptions<Incident> {
60             Includes = "Customer, Product",
61             OrderBy = i => i.DateOpened,
62             WhereClauses = new WhereClauses<Incident> {
63                 { i => i.TechnicianID == id },
64                 { i => i.DateClosed == null }
65             }
66         });
67     };
68     return View(model);
69 }
70
71 }
72
73 }
```

```
TechIncidentController.cs
74 // Methods for the Technician Edit page
75 [HttpPost]
76 public IActionResult Edit(int id)
77 {
78     int? techID = HttpContext.Session.GetInt32("techID");
79     var model = new TechIncidentViewModel
80     {
81         Technician = data.Technicians.Get((int)techID),
82         Incident = data.Incidents.Get(new QueryOptions<Incident> {
83             Includes = "Customer, Product",
84             Where = i => i.IncidentID == id
85         });
86     };
87     return View(model);
88 }
89
90 [HttpPost]
91 public IActionResult Edit(TechIncidentViewModel model)
92 {
93     Incident i = data.Incidents.Get(model.IncidentID);
94     i.Description = model.Incident.Description;
95     i.DateClosed = model.Incident.DateClosed;
96     data.Incidents.Update(i);
97     TempData["message"] = $"Incident {i.Description} was updated";
98     data.Save();
99     int? techID = HttpContext.Session.GetInt32("techID");
100     return RedirectToAction("List", new { id = techID });
101 }
102
103 }
104
105 }
106
107 }
108
109 }
```

ValidationController.cs:

```
1 using Microsoft.AspNetCore.Mvc;
2 using SportsPro.Models;
3
4 namespace SportsPro.Controllers
5 {
6     public class ValidationController : Controller
7     {
8         private IRepository<Customer> cdata { get; set; }
9         public ValidationController(SportsProContext ctx) => cdata = new Repository<Customer>(ctx);
10
11         public JsonResult CheckEmail(string emailAddress, int customerID)
12         {
13             if (customerID == 0) // only check for new customers - don't check on edit
14             {
15                 string msg = Check.EmailExists(cdata, emailAddress);
16                 if (!string.IsNullOrEmpty(msg))
17                 {
18                     return Json(msg);
19                 }
20
21                 TempData["okEmail"] = true;
22                 return Json(true);
23             }
24         }
25     }
26 }
```

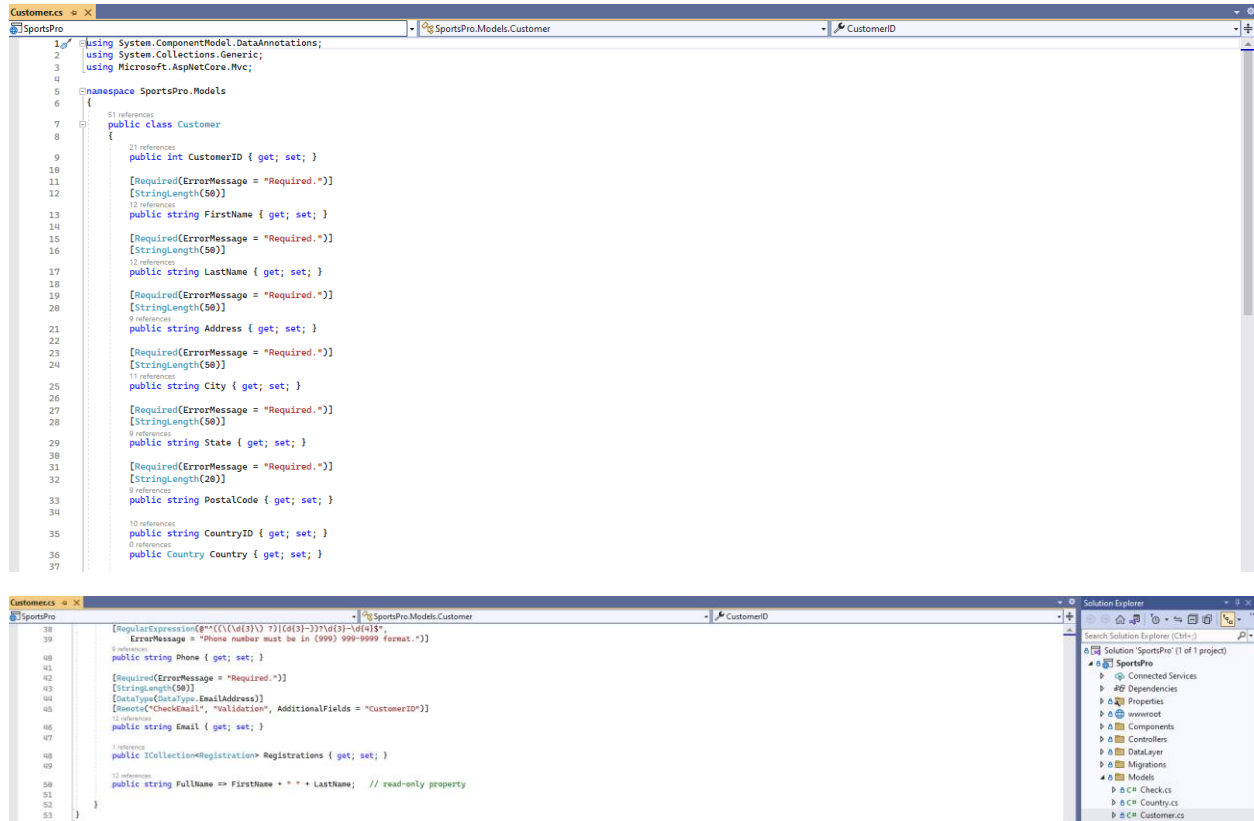
Check.cs Model:

```
1 using System.Linq;
2
3 namespace SportsPro.Models
4 {
5     public static class Check
6     {
7         public static string EmailExists(IRepository<Customer> data, string email)
8         {
9             string msg = "";
10             if (!string.IsNullOrEmpty(email))
11             {
12                 var customer = data.Get(new QueryOptions<Customer>
13                 {
14                     Where = c => c.Email.ToLower() == email.ToLower()
15                 });
16                 if (customer != null)
17                     msg = "Email address already in use.";
18             }
19             return msg;
20         }
21     }
22 }
```

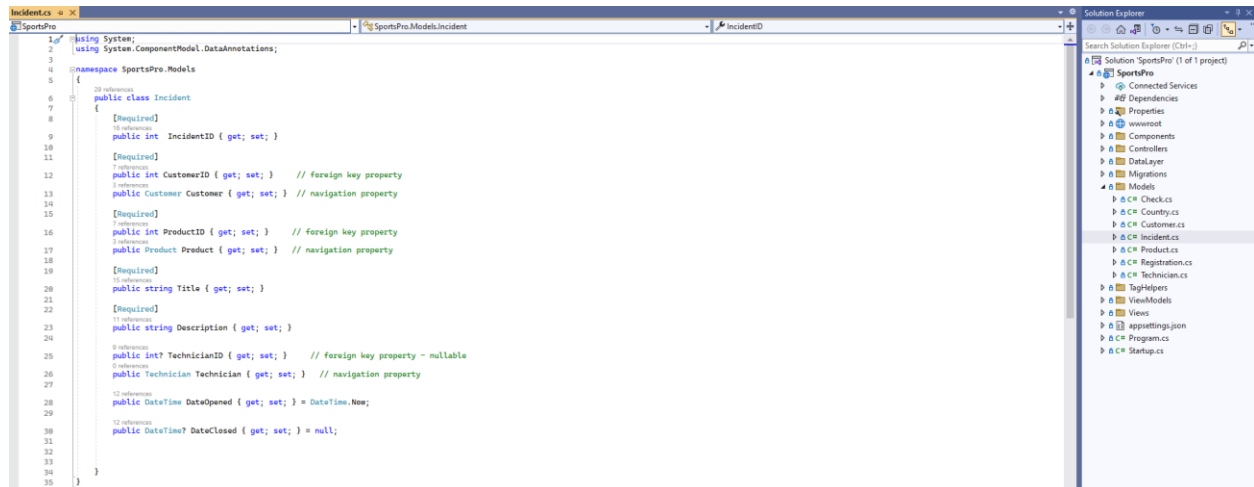
Country.cs Model:

```
1 using System;
2 using System.Collections.Generic;
3 using System.ComponentModel.DataAnnotations;
4
5 namespace SportsPro.Models
6 {
7     public class Country
8     {
9         public string CountryID { get; set; }
10
11         public string Name { get; set; }
12     }
13 }
14 }
```

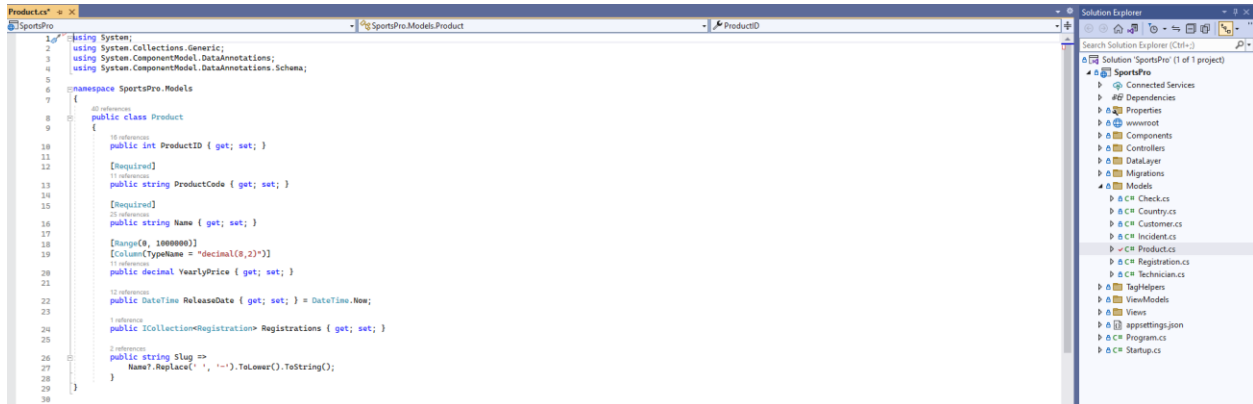
Customer.cs Model:



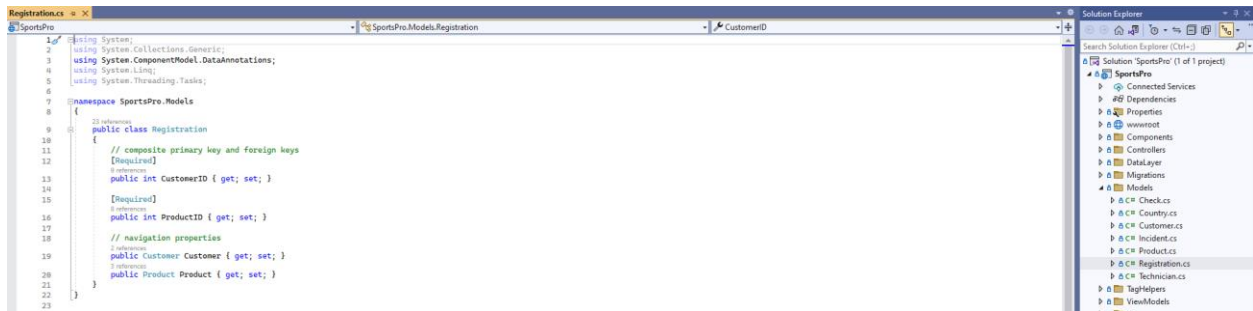
Incident.cs Model:



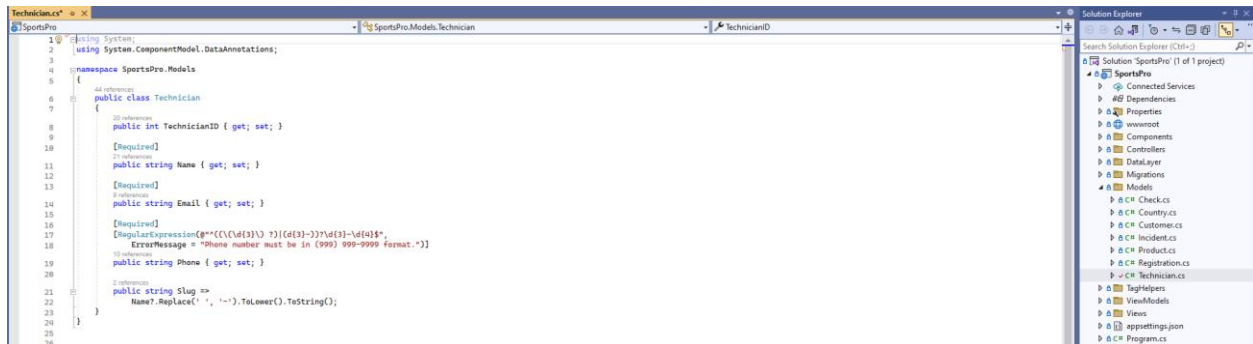
Product.cs Model:



Registration.cs Model:



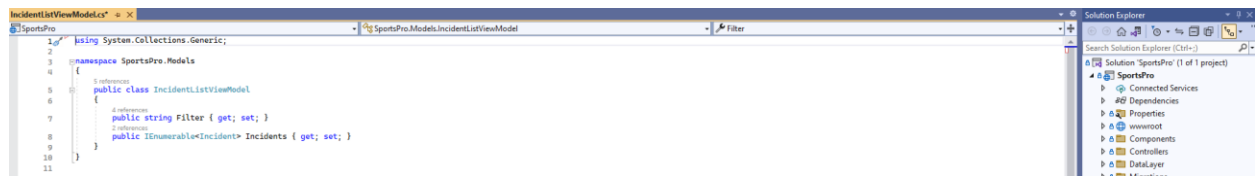
Technicians.cs Model:



DropDownOptionsViewModel.cs:



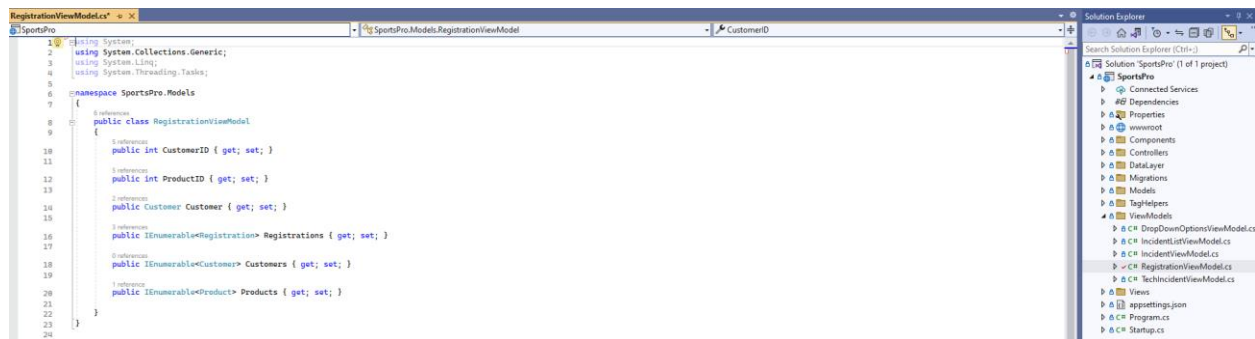
IncidentListViewModel.cs:



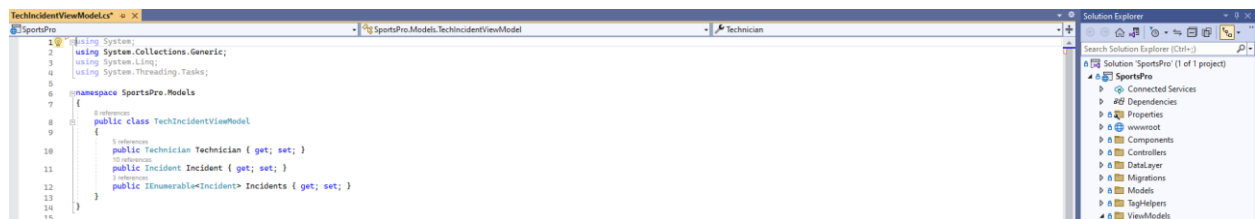
IncidentViewModel.cs:



RegistrationViewModel.cs:



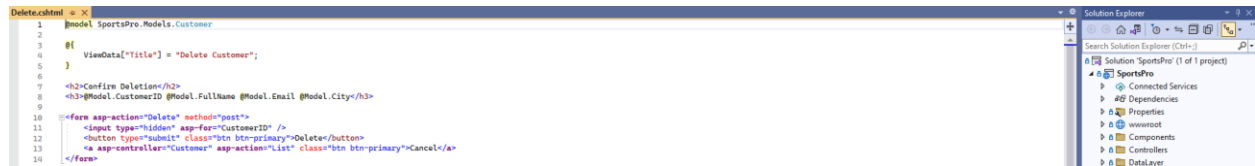
TechIncidentViewModel.cs:



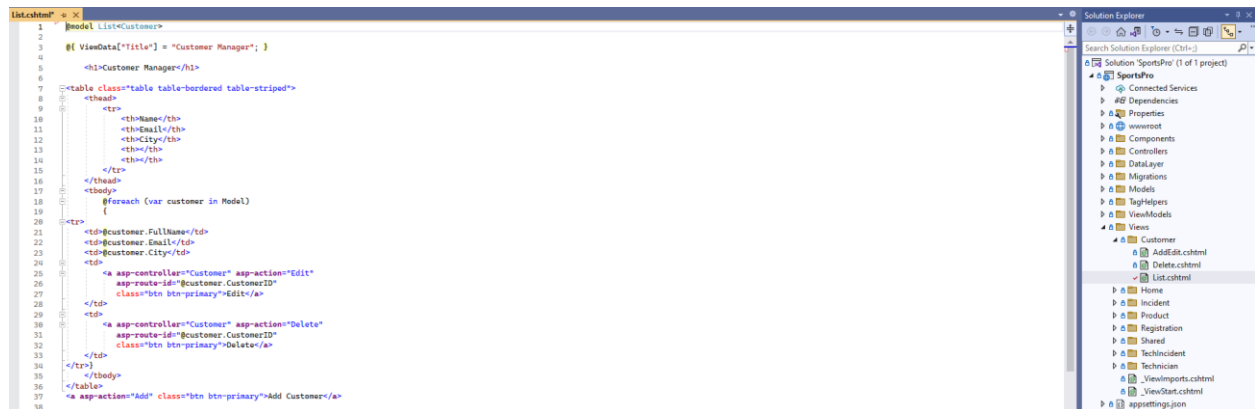
Customer Views AddEdit.cshtml:



Customer Views Delete.cshtml:



Customer Views List.cshtml:



Home Views About.cshtml:

```
1 @model SportsPro.Models.About
2
3 @{
4     ViewData["Title"] = "Admin";
5 }
6
7 <div class="jumbotron bg-secondary text-white">
8     <h1 class="text-light">About</h1>
9     <p class="text-light">SportsPro Technical Support constantly strives to provide the highest level of technical support for all products in the SportsPro family.</p>
10 </div>
```

Home Views Index.cshtml:

```
1 @model SportsPro.Models.Index
2
3 @{
4     ViewData["Title"] = "Technicians";
5 }
6
7 <h1>SportsPro Technical Support</h1>
8 <p>Sports management software for the sports enthusiast</p>
9
10 <div class="list-group">
11     <a asp-controller="Product" asp-action="List" class="list-group-item">Manage Products</a>
12     <a asp-controller="Technician" asp-action="List" class="list-group-item">Manage Technicians</a>
13     <a asp-controller="Customer" asp-action="List" class="list-group-item">Manage Customers</a>
14     <a asp-controller="Incident" asp-action="List" class="list-group-item">Manage Incidents</a>
15     <a asp-controller="Registration" asp-action="List" class="list-group-item">Manage Registrations</a>
16 </div>
17
18 <h1>Technicians</h1>
19
20 <div class="list-group">
21     <a asp-controller="Incident" asp-action="Get" class="list-group-item">Get</a>
22     <a asp-controller="Incident" asp-action="Update" class="list-group-item">Update</a>
23 </div>
```

Incident Views AddEdit.cshtml:

```
1 @model SportsPro.Models.Incident
2
3 @{
4     string title = Model.Action + " Incident";
5     ViewBag.Title = title;
6 }
7
8 @section scripts {
9     <partial name="ValidationScriptsPartial" />
10 }
11
12 <h1>@ViewBag.Title</h1>
13
14 <form asp-action="Save" method="post">
15     <div asp-validation-summary="All" class="text-danger"></div>
16     <div class="form-group">
17         <label asp-for="Incident.CustomerID">Customer</label>
18         <select asp-for="Incident.CustomerID" class="form-control">
19             <option value="">Select a customer...</option>
20         </select>
21     </div>
22     <div class="form-group">
23         <label asp-for="Incident.ProductID">Product</label>
24         <select asp-for="Incident.ProductID" class="form-control">
25             <option value="">Select a product...</option>
26         </select>
27     </div>
28     <div class="form-group">
29         <label asp-for="Incident.Title">Title</label>
30         <input asp-for="Incident.Title" class="form-control" />
31     </div>
32     <div class="form-group">
33         <label asp-for="Incident.Description">Description</label>
34         <input asp-for="Incident.Description" class="form-control" />
35     </div>
36     <div class="form-group">
37         <label asp-for="Incident.TechnicianID">Technician</label>
38         <select asp-for="Incident.TechnicianID" class="form-control">
39             <option value="">Select a technician...</option>
40         </select>
41     </div>
42     <div class="form-group">
43         <label asp-for="Incident.DateOpened">Date Opened</label>
44         <input asp-for="Incident.DateOpened" class="form-control" />
45     </div>
46     <div class="form-group">
47         <label asp-for="Incident.DateClosed">Date Closed</label>
48         <input asp-for="Incident.DateClosed" class="form-control" />
49     </div>
50     <div class="form-group">
51         <input type="hidden" asp-for="Incident.IncidentID" />
52         <button type="submit" class="btn btn-primary">Save</button>
53         <a asp-controller="Incident" asp-action="List" class="btn btn-primary">Cancel</a>
54     </div>
55 </form>
```

Incident Views Delete.cshtml:

```
1 @model SportsPro.Models.Incident
2
3 @{
4     ViewData["Title"] = "Delete Incident";
5 }
6
7 <h1>Confirm Deletion</h1>
8
9 <div class="form-group">
10     <input type="hidden" asp-for="IncidentID" />
11     <button type="submit" class="btn btn-primary">Delete</button>
12     <a asp-controller="Incident" asp-action="List" class="btn btn-primary">Cancel</a>
13 </div>
```

Incident Views List.cshtml:

```
1 @model IncidentListViewModel
2
3 @ ViewData["Title"] = "Incident Manager";
4
5 <h1>Incident Manager</h1>
6
7 <!--Filtering for the page-->
8 <nav class="nav nav-pills mb-3">
9     <a asp-action="Filter" asp-route-id="all" my-filter="@Model.Filter"
10        class="nav-item nav-link">all incidents</a>
11     <a asp-action="Filter" asp-route-id="unassigned" my-filter="@Model.Filter"
12        class="nav-item nav-link">unassigned incidents</a>
13     <a asp-action="Filter" asp-route-id="open" my-filter="@Model.Filter"
14        class="nav-item nav-link">open incidents</a>
15 </nav>
16
17 <table class="table table-bordered table-striped">
18     <thead>
19         <tr>
20             <th>Title</th>
21             <th>Customer</th>
22             <th>Product</th>
23             <th>Data Opened</th>
24             <th></th>
25         </tr>
26     </thead>
27     <tbody>
28         @foreach (var incident in Model.Incidents)
29         {
30             <tr>
31                 <td>@incident.Title</td>
32                 <td>@incident.Customer.FullName</td>
33                 <td>@incident.Product.Name</td>
34                 <td>@incident.DataOpened.ToShortDateString()</td>
35                 <td>
36                     <a asp-controller="Incident" asp-action="Edit"
37                       asp-route-id="@incident.Id" class="btn btn-primary">Edit</a>
38                     <a asp-controller="Incident" asp-action="Delete"
39                       asp-route-id="@incident.Id" class="btn btn-primary">Delete</a>
40                 </td>
41             </tr>
42         }
43     </tbody>
44 </table>
45
46 <a asp-action="Add" class="btn btn-primary">Add Incident</a>
```

Product Views AddEdit.cshtml:

```
1 @model SportsPro.Models.Product
2
3 @ ViewData["Title"] = "Product";
4
5 @section scripts
6 {
7     <partial name="ValidationScriptsPartial" />
8 }
9
10 <h1>@ViewData["Title"]</h1>
11
12 <form asp-action="Edit" method="post">
13     <div asp-validation-summary="All" class="text-danger"></div>
14     <div class="form-group">
15         <label asp-for="ProductCode">Code</label>
16         <input asp-for="ProductCode" class="form-control">
17     </div>
18     <div class="form-group">
19         <label asp-for="Name">Name</label>
20         <input asp-for="Name" class="form-control">
21     </div>
22     <div class="form-group">
23         <label asp-for="YearlyPrice">Yearly Price</label>
24         <input asp-for="YearlyPrice" class="form-control">
25     </div>
26     <div class="form-group">
27         <label asp-for="ReleaseDate">Release Date</label>
28         <input asp-for="ReleaseDate" class="form-control">
29     </div>
30     <input type="hidden" asp-for="ProductID" />
31     <button type="submit" class="btn btn-primary">Save</button>
32     <a asp-controller="Product" asp-action="List" class="btn btn-primary">Cancel</a>
33 </form>
```

Product Views Delete.cshtml:

```
1 @model SportsPro.Models.Product
2
3 @ ViewData["Title"] = "Delete Product";
4
5 <h1>Confirm Deletion</h1>
6
7 <h2>@Model.ProductID @Model.ProductCode @Model.Name @Model.YearlyPrice @Model.ReleaseDate</h2>
8
9 <form asp-action="Delete" method="post">
10     <input type="hidden" asp-for="ProductID" />
11     <button type="submit" class="btn btn-primary">Delete</button>
12     <a asp-controller="Product" asp-action="List" class="btn btn-primary">Cancel</a>
13 </form>
```

Product Views List.cshtml:

```
1 @model List<Product>
2
3 @{
4     ViewData["Title"] = "Product Manager";
5 }
6
7 <h1>Product Manager</h1>
8
9 <table class="table table-bordered table-striped">
10     <thead>
11         <tr>
12             <th>Code</th>
13             <th>Name</th>
14             <th>Price</th>
15             <th>Release Date</th>
16             <th></th>
17         </tr>
18     </thead>
19     <tbody>
20         @foreach (var product in Model)
21         {
22             <tr>
23                 <td>@product.ProductCode</td>
24                 <td>@product.Name</td>
25                 <td>@product.YearlyPrice</td>
26                 <td>@product.ReleaseDate</td>
27                 <td>
28                     <a asp-controller="Product" asp-action="Edit"
29                       asp-route-id="@product.ProductID"
30                       asp-route-slug="@product.Slug"
31                       class="btn btn-primary">Edit</a>
32                 </td>
33             </tr>
34             <tr>
35                 <td>
36                     <a asp-controller="Product" asp-action="Delete"
37                       asp-route-id="@product.ProductID"
38                       asp-route-slug="@product.Slug"
39                       class="btn btn-primary">Delete</a>
40                 </td>
41             </tr>
42         }
43     </tbody>
44 </table>
45
46 <a asp-action="Add" class="btn btn-primary">Add Product</a>
```

Registration Views GetCustomer.cshtml:

```
1 @model Customer
2
3 @{
4     ViewBag.Title = "Registrations";
5 }
6
7 <h1>Get Customer</h1>
8
9 <form asp-action="List" method="post">
10     <select asp-for="CustomerID" class="form-control">
11         @foreach (var customer in Model.Customers)
12         {
13             <option value="@customer.ID">@customer.Name</option>
14         }
15     </select>
16     <button type="submit" class="btn btn-primary">Select</button>
17 </form>
```

Registration Views List.cshtml:

```
1 @model RegistrationViewModel
2
3 @{
4     ViewBag.Title = "Registrations";
5 }
6
7 <h1>Customer: @Model.Customer.FullName</h1>
8
9 <h1>Registrations</h1>
10
11 @if (Model.Registrations.Count() > 0)
12 {
13     <table class="table table-bordered table-striped">
14         <thead>
15             <tr>
16                 <th>Product</th>
17                 <th></th>
18             </tr>
19         </thead>
20         <tbody>
21             @foreach (Registration r in Model.Registrations)
22             {
23                 <tr>
24                     <td>@r.Product.Name</td>
25                     <td>
26                         <form asp-action="Delete" method="post">
27                             <input type="hidden" name="CustomerID" value="@r.CustomerID" />
28                             <input type="hidden" name="ProductID" value="@r.ProductID" />
29                             <button type="submit" class="btn btn-primary">Delete</button>
30                         </form>
31                     </td>
32                 </tr>
33             }
34         </tbody>
35     </table>
36 }
37
38 <p>The selected customer has no registered products.</p>
39
40 <h1>Product</h1>
41
42 <form asp-action="Register" method="post">
43     <div class="form-group">
44         <select asp-for="ProductID" class="form-control">
45             @foreach (var product in Model.Products)
46             {
47                 <option value="@product.ID">@product.Name</option>
48             }
49         </select>
50         <input type="hidden" asp-for="CustomerID" />
51         <button type="submit" class="btn btn-primary">Register</button>
52     </div>
53 </form>
```

Shared Views _Layout.cshtml:

The screenshot shows the Visual Studio IDE with the 'Layout.cshtml' file open in the left pane. The code defines a layout with a navigation bar and a footer. The navigation bar includes links for Home, Product, Technicians, Customers, Incidents, and Registrations. The footer contains a copyright notice. The right pane shows the Solution Explorer with the project structure for 'SportsPro', including folders for Controllers, Views, and Models. The 'Views' folder is expanded, showing sub-folders for 'Home', 'Product', 'Registration', and 'Shared'.

Shared Views _ValidationScriptsPartial.cshtml:

![Screenshot of a web browser showing a validation script error. The console displays a message: 'ValidationScriptPartial:chtml' followed by two lines of script tags. The first line is <script src=\](\"/lib/jquery-validation-unobtrusive/jquery.validate.unobtrusive.min.js\"/>.)

TechIncident Views AddEdit.cshtml:

```

1 AddEdit.cshtml @ x
2
3 @model TechIncidentViewModel
4
5 @{ ViewData["Title"] = "Edit Incident"; }
6
7 @section scripts
8 {
9     @partial name="ValidationScriptsPartial" />
10 }
11
12 <h1>@ViewData["Title"]</h1>
13
14 <form asp-action="Edit" method="post">
15     <div class="form-group">
16         <label class="w-25">Technician: </label>
17         <label>@Model.Technician.Name</label>
18     </div>
19
20     <div class="form-group">
21         <label class="w-25">Customer: </label>
22         <label>@Model.Incident.Customer.FullName</label>
23     </div>
24
25     <div class="form-group">
26         <label class="w-25">Product: </label>
27         <label>@Model.Incident.Product.Name</label>
28     </div>
29
30     <div class="form-group">
31         <label class="w-25">Title: </label>
32         <label>@Model.Incident.Title</label>
33     </div>
34
35     <div class="form-group">
36         <label class="w-25">Date Opened: </label>
37         <label>@Model.Incident.DateOpened</label>
38     </div>
39
40     <div class="form-group">
41         <label asp-for="Incident.Description">Description</label>
42         <textarea asp-for="Incident.Description" class="form-control"></textarea>
43     </div>
44
45     <div class="form-group">
46         <label asp-for="Incident.DateClosed">Date Closed</label>
47         <input type="text" asp-for="Incident.DateClosed" class="form-control" />
48     </div>
49
50     <input type="hidden" asp-for="Incident.IncidentID" />
51
52     <button type="submit" class="btn btn-primary">Save</button>
53     <a asp-action="List" asp-route=id="@Model.Technician.TechnicianID"
54         class="btn btn-primary">Cancel</a>
55 </form>

```

TechIncident Views Get.cshtml:

```
1 @model Technician
2
3 @if (ViewData["Title"] == "Get Technician"; )
4
5 <h1>Technician</h1>
6
7 <form asp-action="List" method="post">
8   <select asp-for="TechnicianID" class="form-control">
9     <option value="">Select a Technician...</option>
10    @foreach (Technician t in ViewBag.Technicians)
11    {
12      <option value="@t.TechnicianID">@t.Name</option>
13    }
14   </select>
15   <button type="submit" class="btn btn-primary">Select</button>
16 </form>
17
```

TechIncident Views List.cshtml:

```
1 @model TechIncidentViewModel
2
3 @if (ViewData["Title"] == "List Incidents for Technician"; )
4
5 <h1>Technician: @Model.Technician.Name</h1>
6 <h2>Assigned/Open Incidents</h2>
7 @if (Model.Incidents.Count() > 0)
8 {
9   <table class="table table-bordered table-striped">
10     <thead>
11       <tr>
12         <th>Title</th>
13         <th>Customer</th>
14         <th>Product</th>
15         <th>Data Opened</th>
16       </tr>
17     </thead>
18     <tbody>
19       @foreach (var Incident in Model.Incidents)
20       {
21         <tr>
22           <td>@Incident.Title</td>
23           <td>@Incident.Customer.FullName</td>
24           <td>@Incident.Product.Name</td>
25           <td>@Incident.DataOpened.ToShortDateString()</td>
26         </tr>
27         <tr>
28           <td colspan="4">
29             <a asp-action="Edit"
30               asp-route-id="@Incident.IncidentID"
31               class="btn btn-primary">Edit</a>
32           </td>
33         </tr>
34       </tbody>
35     </table>
36   }
37   <else>
38     <p>The selected technician has no open incidents.</p>
39   </if>
40   <a asp-action="Get" class="btn btn-primary">Switch Technician</a>
41
```

Technician Views AddEdit.cshtml:

```
1 @model SportsPro.Models.Technician
2
3 @if (string.IsNullOrEmpty(ViewBag.Action) || ViewBag.Action == "Add")
4   ViewData["Title"] = "Add Technician";
5 else
6   ViewData["Title"] = "Edit Technician";
7
8 @section scripts {
9   <partial name="ValidationScriptsPartial" />
10 }
11
12 <h1>@ViewData["Title"]</h1>
13
14 <div class="form-group">
15   <div class="text-danger" style="display:none"></div>
16   <label asp-for="Name" class="form-label">Name</label>
17   <input asp-for="Name" class="form-control">
18 </div>
19
20 <div class="form-group">
21   <label asp-for="Email" class="form-label">Email</label>
22   <input asp-for="Email" class="form-control">
23 </div>
24
25 <div class="form-group">
26   <label asp-for="Phone" class="form-label">Phone</label>
27   <input asp-for="Phone" class="form-control">
28 </div>
29
30 <input type="hidden" asp-for="TechnicianID" />
31
32 <button type="submit" class="btn btn-primary">Save</button>
33 <a asp-controller="Technician" asp-action="List" class="btn btn-primary">Cancel</a>
34 </form>
35
```

Technician Views Delete.cshtml:

```
1 @model SportsPro.Models.Technician
2
3 @if (ViewData["Title"] == "Delete Technician"; )
4
5 <h1>Delete Technician</h1>
6
7 <div class="text-danger" style="display:none"></div>
8
9 <div class="form-group">
10   <label asp-for="Name" class="form-label">Name</label>
11   <input type="text" value="@Model.Name" class="form-control">
12 </div>
13
14 <div class="form-group">
15   <label asp-for="Email" class="form-label">Email</label>
16   <input type="text" value="@Model.Email" class="form-control">
17 </div>
18
19 <div class="form-group">
20   <label asp-for="Phone" class="form-label">Phone</label>
21   <input type="text" value="@Model.Phone" class="form-control">
22 </div>
23
24 <input type="hidden" asp-for="TechnicianID" />
25
26 <button type="submit" class="btn btn-primary">Delete</button>
27 <a asp-controller="Technician" asp-action="List" class="btn btn-primary">Cancel</a>
28 </form>
29
```

Technician Views List.cshtml:

```
1 @model List<Technician>
2 <!--Created by: Chais Strand-->
3 @{ ViewData["Title"] = "Technician Manager"; }
4
5 <h1>Technician Manager</h1>
6
7 <table class="table table-bordered table-striped">
8     <thead>
9         <tr>
10             <th>Name</th>
11             <th>Email</th>
12             <th>Phone</th>
13             <th></th>
14             <th></th>
15         </tr>
16     </thead>
17     <tbody>
18         @foreach (var technician in Model)
19         {
20             <tr>
21                 <td>@technician.Name</td>
22                 <td>@technician.Email</td>
23                 <td>@technician.Phone</td>
24                 <td>
25                     <a asp-controller="Technician" asp-action="Edit"
26                       asp-route-id="@technician.TechnicianID"
27                       asp-route-slug="@technician.Slug"
28                       class="btn btn-primary">Edit</a>
29                 </td>
30                 <td>
31                     <a asp-controller="Technician" asp-action="Delete"
32                       asp-route-id="@technician.TechnicianID"
33                       asp-route-slug="@technician.Slug"
34                       class="btn btn-primary align-middle">Delete</a>
35                 </td>
36             </tr>
37         }
38     </tbody>
39 </table>
40 <a asp-action="Add" class="btn btn-primary">Add Technician</a>
```

_ViewImports.cshtml:

```
1 @using SportsPro
2 @using SportsPro.Models
3 @addTagHelper *, Microsoft.AspNetCore.Mvc.TagHelpers
4 @addTagHelper *, SportsPro
```

_ViewStart.cshtml:

```
1 @{
2     Layout = "_Layout";
3 }
4
```