

Tipo : Guía de Laboratorio
Capítulo : Creando aplicaciones ASP.NET MVC
Duración : 180 minutos

I. OBJETIVO

Crear el Controller para Customer

II. REQUISITOS

Los siguientes elementos de software son necesarios para la realización del laboratorio:

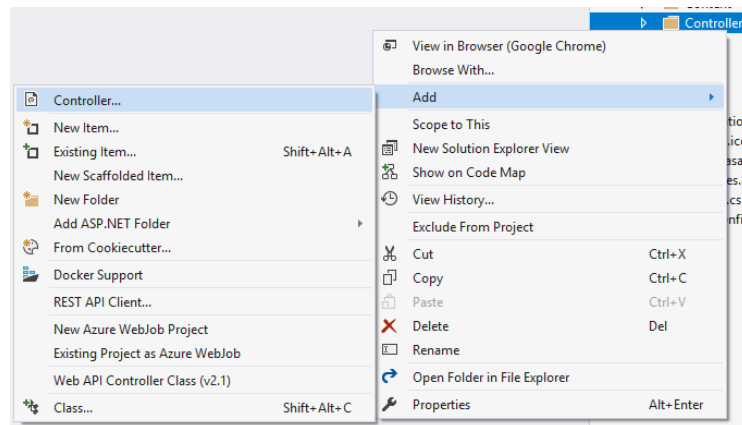
- Windows 10 (como mínimo Windows 8)
- Visual Studio 2017 (como mínimo Visual Studio 2015)

III. EJECUCIÓN DEL LABORATORIO

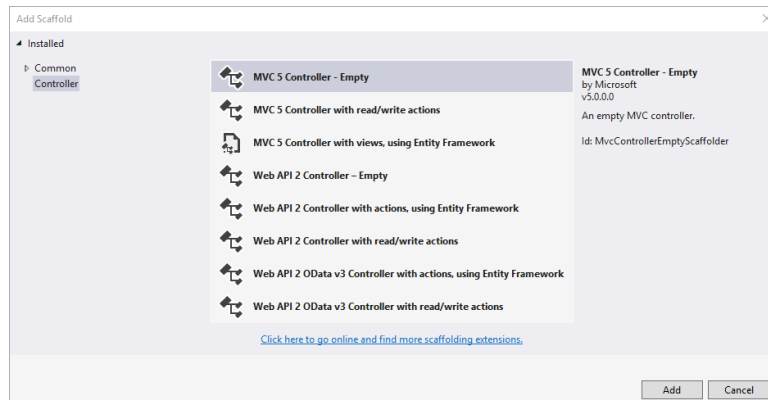
- Laboratorio 4.3: Crear el Controller para Customer.

Crear un proyecto simple en MVC

1. Abrir la solución Cibertec.
2. En el proyecto "**Cibertec.Mvc**" hacer clic derecho en la carpeta "**Controllers**" y bicar el cursor en Add y hacer clic en "**Controller**":

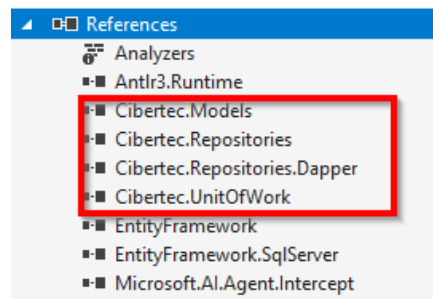


Seleccionar MVC 5 Controller – Empty



El nombre del controlador es “**CustomerController**”.

- Para poder hacer uso de nuestra implementación del patrón repositorio y unit of work, debemos de referenciar en nuestro proyecto MVC a los proyectos creados: **Cibertec.Models**, **Cibertec.Repositories**, **Cibertec.UnitOfWork** y **Cibertec.Repositories.Dapper**



- Agregamos una nueva cadena de conexión en nuestro “web.config”:

```
<add name="NorthwindConnection"
connectionString="Server=.;Database=Northwind_Lite;
Trusted_Connection=True;MultipleActiveResultSets=True"
providerName="System.Data.SqlClient" />
```

Teniendo como resultado en el web.config:

```
<connectionStrings>
  <add name="DefaultConnection" connectionString="Data Source=(LocalDb)\MSSQLLocalDB;Att
  <add name="NorthwindConnection" connectionString="Server=.;Database=Northwind_Lite; Tr
</connectionStrings>
```

- En el “**CustomerController**”, agregamos el siguiente código:

```
using Cibertec.Repositories.Dapper.Northwind;
using Cibertec.UnitOfWork;
using System.Configuration;
using System.Web.Mvc;

namespace Cibertec.Mvc.Controllers
{
    public class CustomerController : Controller
    {
        private readonly IUnitOfWork _unit;
        public CustomerController()
        {
            _unit = new
NorthwindUnitOfWork(ConfigurationManager.ConnectionStrings["NorthwindCo
nnection"].ToString());
        }
    }
}
```

```

    }
    public ActionResult Index()
    {
        return View(_unit.Customers.GetList());
    }
}

```

6. Compilamos la solución para validar que no tengamos ningún error.

IV. EVALUACIÓN

1. ¿Cuáles son las carpetas principales en una aplicación MVC?

Las carpetas principales son: Controllers, Views, Models y wwwroot

2. ¿Qué pasa si al nombre del controlador no se le coloca el prefijo Controller?

Por convención todos los archivos que van a ser controladores deben tener el prefijo Controller al final del nombre para que puedan ser reconocidos por Asp.NET como tales.

3. ¿Cuál es el tipo de datos de los métodos del controller?

El tipo base de todos los controladores es IActionResult.

4. ¿Qué es una vista Layout?

Es la vista maestra, en ella se establece la estructura base (html) de todas las vistas que tendrá la aplicación web. Algunas vistas pueden no utilizar la vista base.

5. Los modelos, ¿pueden ser librería de clases?

Sí, los modelos son un concepto que involucra entidades, acceso a datos, reglas de negocio y pueden ser implementadas en librería de clases (dll).

6. ¿Qué son las vistas parciales?

Al igual que las vistas normales, estas permiten ser reutilizadas en ciertas partes de la aplicación.